|  |
| --- |
| МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  “ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”  Кафедра “Системного аналізу та управління”  Відгук до переддипломної практики  Тема  „ Розробка інформаційної системи для кластеризації даних нечислової природи”  Керівник роботи:  ст. викладач каф. САіУ Коваленко С.В.  Виконавець:  студентка групи ІФ-59б Крутікова М.С.  Харків – 2013 |

Зміст

[Вступ 3](#_Toc358885065)

[1 Опис предметної області 4](#_Toc358885066)

[1.1 Рейтинг фільму 4](#_Toc358885067)

[1.2 Рекомендаційні системи 4](#_Toc358885068)

[1.3 Створення персональних рекомендацій 7](#_Toc358885069)

[1.4 Мета і завдання роботи 7](#_Toc358885070)

[2 Математична постановка задачі 10](#_Toc358885071)

[2.1 Задача колаборативної фільтрації 10](#_Toc358885072)

[2.2 Шкала і метрика 12](#_Toc358885073)

[2.3 Постановка задачі кластеризації 13](#_Toc358885074)

[2.3.1 Ієрархічний агломеративний алгоритм 15](#_Toc358885075)

[2.3.2 K-means модифікація алгоритму K-medoids 15](#_Toc358885076)

[2.4 Алгоритм DBSCAN 16](#_Toc358885077)

[3 Проектне рішення і архітектура системи 18](#_Toc358885078)

[3.1 База даних 18](#_Toc358885079)

[3.2 Архітектура системи 19](#_Toc358885080)

[3.3 Технологія підключення до бази даних 20](#_Toc358885081)

[3.4 Створення веб-сайту 20](#_Toc358885082)

[4 Тестовий приклад 22](#_Toc358885083)

[4.1 Приклади кластеризації точок двомірного простору 22](#_Toc358885084)

[4.2 Кластеризація даних на основі рейтингів 23](#_Toc358885085)

[4.3 Посібник користувача 26](#_Toc358885086)

[Висновки 29](#_Toc358885087)

[Список джерел інформації 30](#_Toc358885088)

Вступ

На сьогоднішній день для інтернет-індустрії великою затребуваністю користується розробка гнучких веб-сайтів для взаємодії з користувачем, які на основі даних, уведених користувачем, визначають його переваги в деякій сфері залежно від предметної області. Після цього користувачеві пропонується рекомендована інформація щодо даної області, яка з великою ймовірністю викличе у нього інтерес. Дані сайти мають відношення до переваг у виборі фільмів, книг, музичних композицій, найбільш цікавих тем для обговорення й мають назву «рекомендаційні системи».

У даній роботі досліджується завдання створення рекомендаційної системи по фільмам, яка використовує рейтинги всіх користувачів для визначення списку рекомендованих фільмів. Для виробництва рекомендацій використовуються алгоритми колаборативної фільтрації засновані на моделях, а саме – на моделях кластеризації даних. Кластеризація використовується для розбивки сукупності об'єктів на однорідні групи (кластери або класи), у даному конкретному завданні – для виділення груп однодумців – користувачів зі схожими перевагами у виборі фільмів.

Система представляє собою веб-портал для збору анкетної інформації й надання користувачам списку рекомендованих кінокартин.

Метою даної роботи є проведення наукового дослідження на реальних вибірках за допомогою тих людей, які погодилися взяти участь у цьому експерименті, дослідження різних алгоритмів і порівняння якості їх роботи для даного специфічного завдання з метою пошуку кращого розв'язку з реалізованих.

# Опис предметної області

## Рейтинг фільму

На сьогоднішній день в мережі Інтернет існує велика кількість веб-порталів, які містять контент пов'язаний з фільмами. Дані сайти дозволяють своїм зареєстрованим користувачам ділитися думкою про переглянуті фільми, дізнаватися про думки не тільки інших користувачів, але й відомих кінокритиків, переглядати різноманітну статистику по фільмам і стежити за кращими з них. Думка інших користувачів про новий фільм, що їх цікавить, допомагає відповістити на важливе питання – дивитися цей фільм чи ні.

Найзручніший спосіб виразити думку про фільм – поставити йому оцінку в межах певної шкали, тобто конвертувати свої враження від перегляду фільму у число, яке називають рейтингом фільму. Термін походить від англійського слова «rating» – оцінка, порядок, класифікація й означає суб'єктивну оцінку якого-небудь погано вимірного явища.

На підставі голосів користувачів з використанням спеціального математичного апарата відбувається обробка оцінок фільмів з метою одержання різних результатів, наприклад розрахунок середньої загальної оцінки фільма. На підставі загальної оцінки користувач може зробити висновок про те, варто дивитися кінокартину, що його цікавить, чи ні, щоб не бути розчарованим в своєму виборі.

## Рекомендаційні системи

Найбільший у світі сайт про кіно – Imdb (Internet Movie Database, imdb.com) представляє собою англомовний сайт, що має величезну базу даних про кінематограф: фільми, серіали, людей, які їх створюють: актори, сценаристи, режисери. Це один із самих відвідуваних ресурсів на планеті – 36 місце по відвідуваності у світі [[[1]](#endnote-2)]. Популярною можливістю Imdb є онлайн-голосування, де будь-який зареєстрований відвідувач сайту може голосувати за фільми, виставляючи їм рейтинг від 1 до 10 балів.

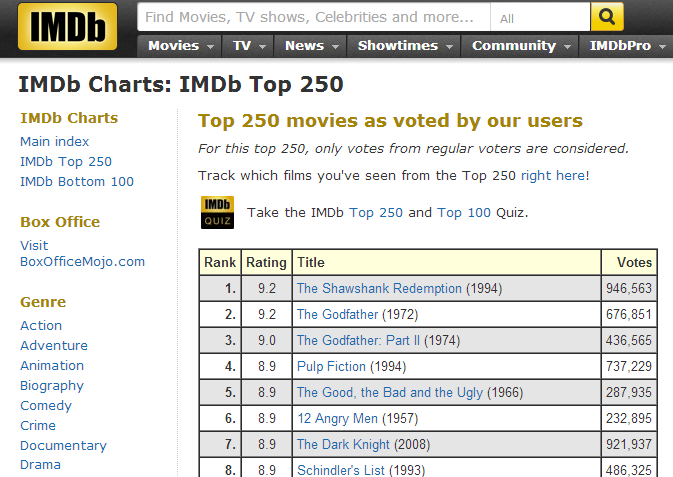


Рисунок 1.1 – Найвідоміший у світі сайт про кіно

Kinopoisk.ru – найбільший російськомовний сайт про кіно, один із самих відвідуваних і популярних ресурсів рунету. Тут можна знайти інформацію про будь-який фільм: подивитися кадри й трейлери, почитати рецензії як професійні, так і написані відвідувачами сайту, виставити фільму оцінку. Kinopoisk має свій список кращих фільмів за версією користувачів цього ресурсу, при цьому поруч із рейтингами кращих фільмів відображається рейтинг фільму з Imdb.

Сайт Imdb і Kinopoisk надають можливість користувачам отримувати список фільмів, які їм можуть сподобатись. Цей функціонал має індивідуальний характер, оскільки базується на аналізі рейтингів кожного окремого користувача і користувачів, які мають схожі смаки з даним. Для визначення списку фільмів, які можуть зацікавити користувача, необхідно передбачати оцінку, яку користувач поставив би цьому фільму, якщо б переглянув його. Для цього використовується спеціальний математичний апарат роботи зі специфічним даними, у вигляді списку оцінок для кожного користувача. Цей математичний апарат носить назву «колаборативна фільтрація», а системи, які його використовують називають рекомендаційними системами.

Найбільш відомими рекомендаційними системами вважаються відома компанія Amazon (для рекомендації різних товарів), Netflix (для рекомендації відеоматеріалів), Imhonet (рекомендації з багатьох типів контенту).

Netflix – американський онлайн-сервіс відеопрокату. Одним із ключових факторів успіху Netflix стало масштабне застосування персоналізованої рекомендаційної системи відео продукції, заснованої на рейтингах і оглядах своїх клієнтів [[[2]](#endnote-3)].



Рисунок 1.2 – Рекомендаційна система сайту відеопрокату Netflix

Імхонет – мультикультурний рекомендаційний сервіс (рекомендаційна система), експертну роботу в якому виконують самі користувачі. На відміну від рекомендаційних систем, що спеціалізуються тільки на одному типі контенту (наприклад, Last.fm – музика, Netflix – кінофільми), даний веб-портал працює одночасно з різним видом контенту. На основі виставлених оцінок користувачів програма будує його споживчий профіль, порівнює з іншими профілями різних користувачів і формує співтовариства однодумців – людей із близькими смаками [[[3]](#endnote-4)].

## Створення персональних рекомендацій

Основне завдання системи рекомендацій – маючи дані про об'єкти й користувачів, отримати список об'єктів, найбільш цікавих для конкретного користувача й сформувати список однодумців – людей близьких по смакам.

Створення якісної рекомендаційної системи, з одного боку, може привести до збільшення прибутків, а з іншого − економії часу споживача на пошуки необхідного об’єкту. Таким чином, подібні системи дуже важливі як для економіки, так і для інформаційного суспільства в цілому.

Для розрахунків рекомендацій найчастіше використовують методи колаборативної фільтрації. Ці методи припускають, що інтереси користувачів представлені оцінками, які вони дають об'єктам після перегляду, покупки і т.д. Основна ідея даних методів полягає в порівнянні між собою інтересів різних користувачів або об'єктів на основі цих оцінок. При цьому ніяка додаткова інформація про самих користувачів і об'єктах не використовується.

Незважаючи на успішне використання цих методів в системах електронної комерції, їх застосування пов'язане з декількома проблемами, такими як масштабування й розрідженість даних.

Проблема масштабування полягає в тому, що більшість алгоритмів, які використовуються при спільній фільтрації, мають нелінійну складність. Тому зі збільшенням кількості користувачів і об'єктів час, необхідний для пошуку рекомендацій, значно росте й часто стає неприйнятним. Популярним підходом до розв’язку цієї проблеми є використання алгоритмів інтелектуального аналізу даних і машинного навчання, а саме використання алгоритмів кластеризації даних.

У даній роботі проводиться дослідження підходу колаборативної фільтрації із застосуванням алгоритмів кластеризації даних.

## Мета і завдання роботи

Основна мета аналізу в даній роботі – виявити закономірності й спільності в статистичних даних, що представляють собою думки людей щодо фільмів, і сформувати групи людей із приблизно схожими інтересами у виборі кінокартин. Для формування груп по інтересам використовуються алгоритми кластеризації даних, після чого для кожної групи застосовуються алгоритми колаборативної фільтрації для створення рекомендацій.

Найбільш зручним і загальнодоступним засобом для збору статистичних даних є веб-сайт, який дозволить залучити до дослідження велику кількість людей, і швидко поширити посилання на розроблений сайт по глобальній мережі Інтернет. Тому в даній роботі було вирішено розробити веб-додаток з метою спрощення збору інформації, необхідної для аналізу, а також з можливістю надання кожному користувачеві індивідуальних результатів обробки даних по його персональних оцінках.

Frontend

Data Access

DB

Збір рейтингів

Пошук фільмів

Представлення статистики

Рекомендовані фільми

Web Scraper, Parser

Backend

Рисунок 1.3 – Зв'язок компонентів програми між собою

Для створення повноцінної програми необхідна розробка наступних компонентів, які представлені на рис. 1.3:

1. База даних (DB) – ядро всієї системи, розробка структури БД найбільш важлива й значима частина програми. База даних повинна містити інформацію про фільми, про зареєстрованих користувачів і їх рейтинги.
2. Web Scraper, Parser представляють собою програму, яка використовується для скачування інформації про фільми з популярних сайтів про кіноіндустрію, наприклад kinopoisk.ru, afisha.ru, filmz.ru. Отримана інформація заноситься в базу даних і формує базову статистичну інформацію.
3. Data Access – шар доступу до даних, автономна частина програми, що забезпечує зв'язок з базою даних і реалізує запити до даних.
4. Backend – частина програми, що реалізує аналіз і обробку даних, включає безпосередньо алгоритми кластеризації рейтингів.
5. Frontend – веб-інтерфейс, візуалізація/презентація результатів обробки/аналізу даних.

# Математична постановка задачі

## Задача колаборативної фільтрації

У типовому сценарії колаборативної фільтрації є список користувачів , і список об’єктів .

Кожен користувач характеризується списком об'єктів , про які він виразив свою думку. Вхідні дані для алгоритмів колаборативної фільтрації зручно представити у вигляді матриці рейтингів . Кожний елемент матриці представляє собою рівень переваги -того користувача відносно -того об'єкту.

Існує користувач , який характеризується списком оцінених об'єктів і називається активним користувачем. З використанням алгоритму колаборативної фільтрації для активного користувача повинен бути здійснений пошук об'єктів, який може бути виконаний у двох формах:

1. Передбачення – чисельне значення, що показує передбачене відношення до об'єкта для активного користувача . У термінах матриці оцінок передбачення представляє собою значення – передбачуваний рейтинг невідомого об'єкта для активного користувача, який характеризується набором оцінок ,
2. Рекомендація представляє собою список об'єктів, , які ймовірно найбільше сподобаються користувачеві. Список повинен складатися тільки з об'єктів, яким активний користувач не поставив оцінку раніше, тобто . Цей спосіб відомий як top-N рекомендація [[[4]](#endnote-5)].

Алгоритми колаборативної фільтрації можуть бути застосовані до користувачів або до фільмів. У випадку використання фільтрації по користувачах (User-based) невідомий рейтинг об'єкту виставляється на підставі рейтингів, які були проставлені тому ж об'єкту користувачами, схожими на даного. Алгоритм складається із трьох кроків:

1. Для активного користувача обчислити міри його близькості до всіх інших, активний користувач характеризується вектором , користувач – вектором .
2. Вибирається множина користувачів найбільш близьких до активного користувача . Множину сусідів формують користувачі, близькість між якими перевищила заданий поріг.
3. Невідомий рейтинг об'єкта передбачується на основі рейтингів обраних сусідів. Ваги користувачів представляють собою міру близькості з активним користувачем. Передбачення обчислюється наступним чином:

(2.1)

де – середній рейтинг активного користувача ; підсумовування проводиться по всіх користувачах, які оцінили об'єкт , тоді – рейтинг користувача з множини щодо об'єкта , – середній рейтинг користувача по всіх оцінених об'єктах [[[5]](#endnote-6)].

Якщо використовується фільтрація по об'єктах (Item-based) невідомий рейтинг об'єкту виставляється на підставі рейтингів інших схожих об'єктів, оцінених активним користувачем. Поточний об'єкт характеризується набором його оцінок у матриці . Алгоритм аналогічний попередньому підходу:

1. Для поточного об'єкта обчислити міри його близькості до всіх інших.
2. Сформувати множину сусідів для поточного об'єкта.
3. Невідомий рейтинг поточного об'єкта передбачується на основі рейтингів близьких до нього об'єктів. Ваги оцінок для об'єктів-сусідів представляють собою міру близькості з поточним об'єктом. Передбачення обчислюється наступним чином:

(2.2)

де підсумовування відбувається по всіх об'єктах , які оцінив активний користувач , – оцінка користувача щодо об'єкта [4].

## Шкала і метрика

Відповідно до теорії вимірів при математичному моделюванні реального явища або процесу необхідно насамперед установити типи шкал, у яких виміряні ті або інші змінні. З питанням про тип шкали безпосередньо зв'язана проблема адекватності методів математичної обробки результатів виміру. У загальному випадку адекватними є ті статистики, які інваріантні щодо допустимих перетворень даної шкали вимірів.

Оцінки експертів часто слід вважати виміряними в порядковій шкалі. Як показали численні досвіди, людина більш правильно (і з меншими складностями) відповідає на запитання якісного, наприклад, порівняльного, характеру, чим кількісного [[[6]](#endnote-7)].

У рекомендаційній системі кожна оцінка, поставлена користувачем, може бути представлена у вигляді значення, виміряного в порядковій шкалі. Однак не всі соціологи відносять даний тип вимірів саме до цієї шкали – деякі вважають за краще працювати з даними значеннями, як виміряними в інтервальній шкалі. Окремий випадок шкали інтервалів, який може бути використаний для представлення рейтингу – шкала різниць.

Коефіцієнт кореляції може бути визначений для даних шкали порядку, інтервалів або відносин, але не має значення для змінних виміряних у шкалі не більш номінальної. При обробці даних, виміряних у порядковій шкалі, можуть застосовуватися рангові статистичні методи. Для визначення тісноти зв'язку між ознаками для таких даних застосовуються методи рангової кореляції, тобто для визначення тісноти зв'язку допустимим є використання коефіцієнта кореляції Спірмена або Кендела. Для шкали інтервалів або відносин найчастіше використовується коефіцієнт кореляції Пірсона.

Таким чином, якщо вважати дані виміряними в шкалі різниць, яка є окремим випадком шкали інтервалів, то допустимим є також застосування лінійного коефіцієнта кореляції, або кореляції Пірсона. В даній роботі в якості міри схожості між об’єктами, представленими списком рейтингів, застосовується коефіцієнт кореляції Пірсона, коефіцієнт рангової кореляції Спірмена і косинусна міра.

## Постановка задачі кластеризації

Кластеризація дозволяє розбити вихідну множину об'єктів на задане або заздалегідь невідоме число кластерів (класів) однорідних об'єктів.

У даній задачі розбивка множини користувачів або об'єктів скорочує розмірність простору, перетворюючи тим самим простір великої розмірності в множину просторів меншої розмірності з меншою кількістю об'єктів, користувачів і рейтингів, що зменшує час, необхідний для формування онлайн рекомендацій. Також точність рекомендацій в порівнянні зі стандартним підходом колаборативної фільтрації може збільшитися, оскільки завдяки розбивці на кластери буде усунутий шум, який може негативно вплинути на результуючу рекомендацію.

Формально задача кластеризації описується наступним чином. Дана множина об'єктів даних , кожний з яких представлений набором атрибутів. Потрібно побудувати множину кластерів і відображення множини на множину тобто . Відображення задає модель даних, що є розв'язком задачі.

Визначається множина , де – досліджуваний об'єкт. Кожний з об'єктів характеризується набором деяких параметрів Кожна змінна може приймати значення з множини . Тоді, задача кластеризації полягає в побудові множини , де – кластер, що містить схожі один на одного об'єкти з множини . Кожен кластер характеризується наступною умовою:

(2.3)

де – коефіцієнт відмінності між об'єктами, – величина, що визначає міру відмінності для включення об'єктів в один кластер.

Вся трудність полягає в тому, що єдиної точної постановки задачі кластерного аналізу не існує. На рисунку 2.2 зображені різні форми кластерів, що створює великі складності формування єдиної теорії, яка б дозволяла чітко розділяти наведені комбінації класів. Перш ніж будувати теорію, що враховує подібні конфігурації точок, необхідно ясно усвідомлювати природу пред'явлених до розбивок вимог [[[7]](#endnote-8)].

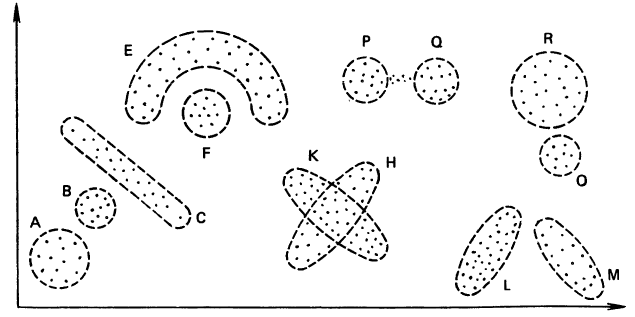


Рисунок 2.1 – Різні форми кластерів

У даній роботі реалізовано ієрархічний агломеративний алгоритм кластеризації, модифікація алгоритму K-medoid (PAM – Partitioning Around Medoids) і алгоритм DBSCAN, заснований на застосуванні поняття щільності (for density-based spatial clustering of applications with noise).

### Ієрархічний агломеративний алгоритм

На першому кроці вся множина представляється як множина кластерів На наступному кроці вибираються два найбільш близьких один до одного об’єкта (наприклад, і ) і об’єднуються в один загальний кластер. Нова множина, що складається вже з кластерів, буде мати вигляд:

(2.4)

Повторюючи аналогічні дії, сформується кластер, що полягає з об'єктів і співпадає з первісною вибіркою [11].

### K-means модифікація алгоритму K-medoids

Алгоритм K -medoids схожий на алгоритм K-means, однак відрізняється від нього вибором центрів кластерів, оскільки не використовує середніх точок. Кластеризація методом K-means дуже чутлива до шумів і викидів даних, з цієї причини, використовується алгоритм K-medoids, у якому в якості репрезентативних об'єктів використовуються медоіди на відміну від центрів мас, або центроідів. Алгоритм K-medoids використовує центрально-розташовані об'єкти як центри кластерів, тому він є менш чутливим до шумів у порівнянні з алгоритмом K-means.

В [[[8]](#endnote-9)] запропонована модифікація алгоритму K-medoids, перша фаза якого аналогічна першій фазі вибору початкових медоідів алгоритму PAM, запропонованого в [[[9]](#endnote-10)]. Друга фаза мінімізації цільової функції аналогічна алгоритму K-means і тому даний алгоритм має назву K-means-like Algorithm for K-medoids Clustering.

Фаза ініціалізації початкових медоідів полягає в послідовному виборі репрезентативних об'єктів, поки об'єктів не будуть обрані. Першим вибирається об'єкт, для якого сума відстаней до всіх інших об'єктів є мінімальною. Далі на кожному наступному кроці вибирається новий об'єкт, який мінімізує цільову функцію [9].

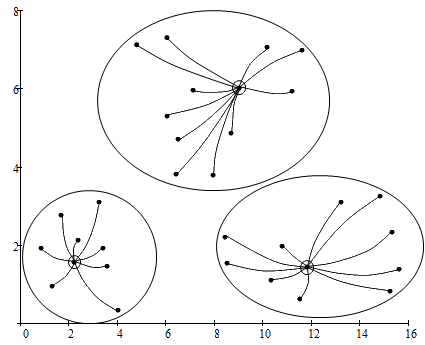


Рисунок 2.2 – Приклад роботи алгоритму K-medoids

Друга фаза алгоритму спрямована на покращення репрезентативних об'єктів і як наслідок кластеризації в цілому. Даний етап реалізує мінімізацію цільової функції суми відстаней між усіма об'єктами й найближчими медоідами [8].

## Алгоритм DBSCAN

На рис. 2.3 представлені приклади множини точок з різних вибірок. Дивлячись на графічне представлення цих даних людина легко й однозначно зможе виділити кластери будь-якої складної форми, а також визначити шуми – об’єкти, які не належать жодному із кластерів.

Головна причина, завдяки якій людина легко пізнає кластери, полягає в тому, що типова щільність точок усередині кластера значно вище, чим за його межами, щільність усередині простору шумів значно нижче щільності усередині кластера. На підставі строгої математичної формалізації цих принципів був сформований алгоритм кластеризації DBSCAN (Density Based Spatial Clustering of Application with Noise) [[[10]](#endnote-11)].

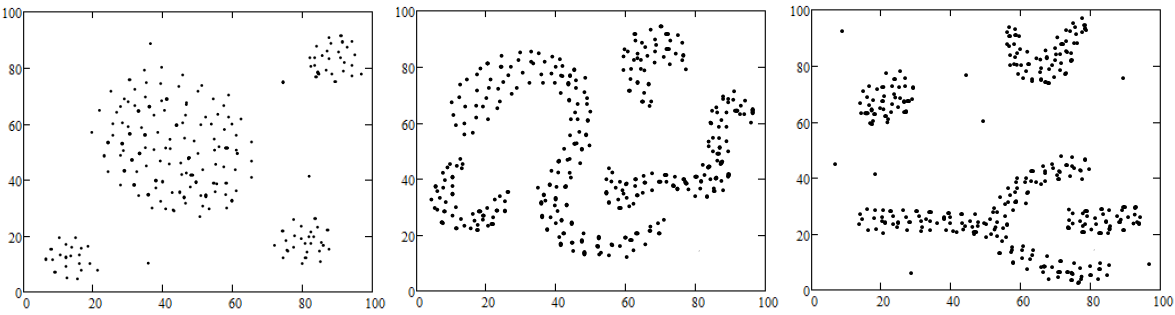


Рисунок 2.3 – Приклади кластерів

Щільність точок для даної точки визначається двома параметрами. Першим з них є – радіус «сусідства» точки . Тоді -сусідство точки , яке позначається як , визначається наступним чином:

(2.5)

Другим параметром визначення щільності точок є параметр – це мінімальна кількість точок, які розташовані ближче всього до даної точки згідно з певним радіусом . Згідно із заданими параметрами існує два типи точок – оточена й гранична. Точка називається оточеною (або ядром), якщо кількість точок у її сусідстві більше й дорівнює значенню параметра , тобто

Для пошуку кластерів, алгоритм починає обробку довільної точки і знаходить усі обєкти досяжні по щільності з даної точки згідно з параметрами алгоритму і . Якщо даний об'єкт – оточена точка, то всі об'єкти, досяжні по щільності від поточного об'єкта, з'єднуються в новий кластер. Інакше, якщо об'єкт не є оточеною точкою й не досяжний по щільності ні від якого об'єкта, то поточний об'єкт – шум.

# Проектне рішення і архітектура системи

## База даних

База даних створеної рекомендаційної системи містить 12 таблиць. Таблиця Movies містить основну інформацію про фільм – назва, тривалість і рік виходу фільму. Оскільки дані про фільми були зібрані з сайту kinopoisk, у базу даних також була занесена додаткова інформація про кожний фільм – рейтинг фільму, кількість людей, що голосували на сайті kinopoisk, унікальний ідентифікатор фільму в базі даних даного сайту. У таблиці Movies також знаходиться поле, що відповідає номеру кластера.

Таблиці Genres і Countries містять назву жанру й країни. Відношення таблиці Movies з таблицями Genres і Countries організоване по типу зв'язку «багато-до-багатьох» і розбивається через проміжні таблиці MovieGenres і MovieProductionCountries відповідно, і мають тип зв'язку « багато-до-одного» з таблицею Movies.

Таблиця ParticipantTypes містить назву ролі учасників фільму, таблиця MovieParticipants містить ім'я учасника. Зв'язок «багато-до-багатьох» таблиць розбивається через відношення «один-до-багатьох» таблиць ParticipantTypes і MovieParticipants з таблицею MovieParticipantTypes.

Відношення між таблицями Movies і MovieParticipantTypes представляє собою зв'язок «багато-до-багатьох», який розбивається через таблицю MovieProductionParticipants.

Таблиця Users містить облікові записи користувачів – дані про ім’я користувача і його електронну пошту. У таблиці також знаходиться поле, що відповідає номеру кластера, до якого належить даний користувач. У таблицю UserLoginHistory винесена інформація про історію логінів користувача

Рейтинги користувачів зберігаються в таблиці MovieRatings. Таблиці Movies і Users зв'язані відношенням «багато-до-багатьох» через таблицю MovieRatings, у якій, крім ідентифікаторів користувача й фільму, зберігається значення рейтингу й остання дата його поновлення.

## Архітектура системи

Рівень доступу до даних (Data Access Level) здійснює роботу з базою даних. Data Access Level містить класи, що реалізують добування даних із БД, додавання, зміну або видалення записів. Даний рівень не залежить від усіх інших рівнів, оскільки є основою всієї програми.

Рівень логіки програми (Business Logic) містить алгоритми кластеризації даних і створення рекомендацій. Даний рівень звертається до рівня доступу до даних, тобто безпосередньо залежний від нього, оскільки алгоритми застосовуються до даних, які необхідно витягти із БД за допомогою DAL.

Business Logic (Services)

Data Access Level (EF)

DB

Рисунок 3.1 – Компоненти рекомендаційної системи

Веб-додаток приймає запити користувачів і повертає відповідь у вигляді HTML-Сторінки. Даний рівень залежить від рівня доступу до даних, оскільки відображає на сторінках дані із БД, і від рівня бізнес-логіки, оскільки викликає алгоритми створення рекомендацій по запиту користувача.

Елемент Data Scraping, зображений на рис. 3.1, представляє собою окремий додаток, який використовується для заповнення бази даних вихідною інформацією про кінокартини за допомогою технологій Web scraping і Parsing веб-сторінок та виконання запитів до веб-ресурсу з інформацією про фільми. Дані про фільми з 1950 по 2012 року були витягнуті з веб-ресурсу Kinopoisk.ru.

## Технологія підключення до бази даних

У даній роботі використовується технологія ORM доступу до даних, а саме побудова шару DAL для веб-сайту на основі технології ADO.NET Entity Framework.

Entity Framework – це ORM система, що дозволяє створювати абстрагований від конкретної бази даних об'єктно-орієнтований data access layer. ORM (object relational mapping) – це технологія, яка дозволяє перетворювати дані між реляційними базами даних і об'єктно-орієнтованими мовами програмування. Використовуючи EF, можна взаємодіяти з реляційними базами даних застосовуючи об'єктно-орієнтовані конструкції.

В основі ADO.NET EF лежить модель сутностей – Entity Data Model. Основні об'єкти моделі – це сутності й зв'язки. Сутності (entities) представляють собою строго типізовані класи, з яких складається сутнісна модель даних (Entity Data Model), об'єкти, що базуються на цих сутностях, називаються сутнісними об'єктами (Entity objects).

## Створення веб-сайту

Веб-сайт, який реалізує рекомендаційну систему, розроблено з використанням фреймворку ASP.NET MVC 4, який представляє собою набір компонентів для створення сайтів, заснованих на підході використання шаблону MVC. При використанні у веб-розробці цей шаблон розділяє роботу веб-сайта на три окремі функціональні ролі: модель даних (model), користувацький інтерфейс (view) і керуючу логіку (controller). Таким чином, зміни, внесені в один з компонентів, оказують мінімально можливий вплив на інші компоненти.

Web Application

Model

Controller

Request

Response

View

DB

Рисунок 3.2 Взаємодія компонентів шаблону MVC

Модель представляє собою дані й логіку програми. Модель відповідає за зберігання, видалення й відновлення даних (роботу з базою даних), тобто містить методи роботи з базою даних.

Контролер забезпечує спільну роботу Моделі й В'ю. Він забезпечує зв'язок між користувачем і системою: контролює введені користувачем дані, використовує модель і представлення для реалізації необхідної реакції.

Рівень представлення одержує дані з Моделі у формі доступної для користувача. Дані можуть надходити в різних форматах від моделі: об'єктах, xml структурах і т.д. В'ю відповідає за відображення інформації (користувацький інтерфейс) [[[11]](#endnote-12)].

# Тестовий приклад

## Приклади кластеризації точок двомірного простору

Для перевірки роботи реалізованих методів кластеризації було проведено кластеризацію довільної вибірки, яка складається з точок двомірного простору. Для порівняння роботи методів було створено вибірку точок зі складною структурою кластерів, які мають різні форми, до того ж вибірка містить шуми – точки, які важко віднести до якогось кластеру. На рис. 4.1 представлено графік точок створеної вибірки у двомірному просторі.

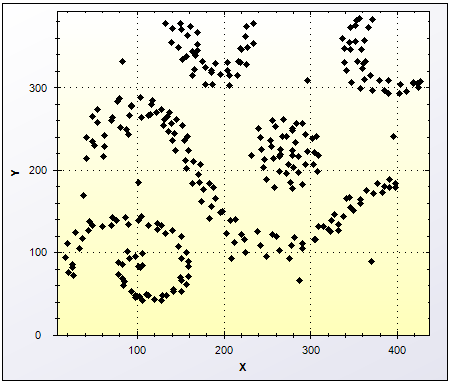


Рисунок 4.1 – Тестова вибірка

Для кількісного вираження якості отриманої кластеризації даних використовується коефіцієнт силует. Даний коефіцієнт також використовується для початкового завдання кількості кластерів, якщо алгоритм вимагає його як вхідного параметра.

За допомогою визначення якості кластеризації при різних вхідних параметрах були визначені оптимальні параметри для кожного з алгоритму, які максимізують якість кластеризації. Кількість кластерів для алгоритмів агломеративної кластеризації і K-medoids дорівнює 7, параметри для алгоритму DBSCAN: , .

З використанням алгоритму DBSCAN отримано природне розбиття точок на кластери з найвищим значенням якості розбиття і виявлені шуми, які не можуть бути віднесені ні до одного з кластерів.

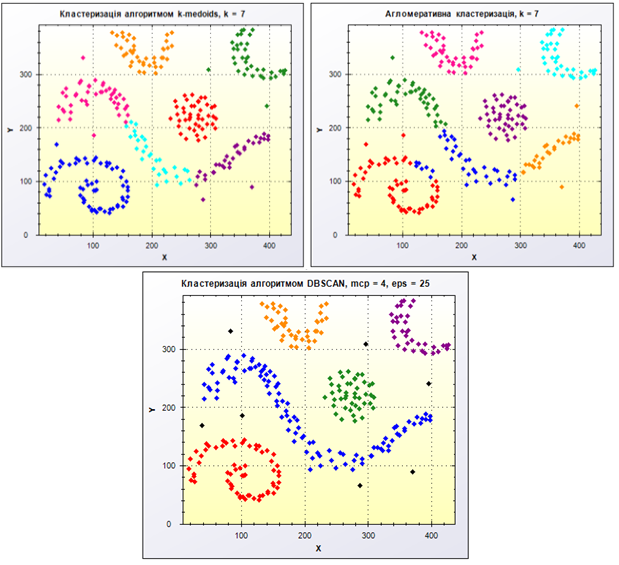


Рисунок 4.2 – Кластеризація вибірки методами агломеративної кластеризації, алгоритмом K-medoids і алгоритмом DBSCAN

## Кластеризація даних на основі рейтингів

Для того щоб одержати найбільш удалу розбивку об'єктів на кластери, необхідно вибирати таку кількість кластерів, яка максимізує коефіцієнт силует для даної вибірки. Однак велика кількість кластерів, навіть при значенні коефіцієнта якості, відповідного до вдалої розбивки, може негативно вплинути на створення рекомендацій, оскільки з кожним об'єктом у групі буде перебувати невелика кількість його сусідів, і як наслідок кількість загальних фільмів, для яких може бути передбачений рейтинг, різко скорочується. Крім того, на передбачений рейтинг будуть сильно впливати окремі користувачі, що також може погіршити якість рекомендацій.

Для кластеризації користувачів у вихідну вибірку включаються об’єкти, які мають більше двадцяти оцінок, що складає загалом 137 користувачів. Виберемо конкретного користувача системи з ідентифікатором 3, який поставив 63 оцінки і розглянемо кластери, в які потрапив даний користувач з використанням різних методів кластеризації і різними мірами відмінності.

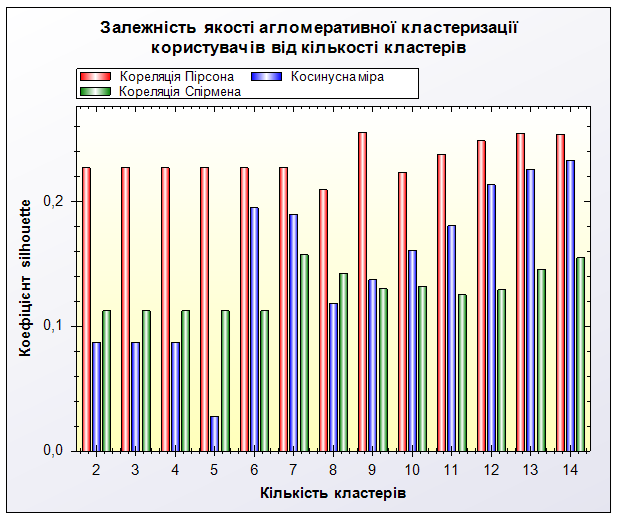


Рисунок 4.3 – Якість агломеративним кластеризації користувачів з використанням різних мір відмінності

Для визначення оптимального числа кластерів для конкретного алгоритму будується графік залежності числа кластерів від якості кластеризації з використанням різних мір відмінності. На рис. 4.3 представлено такий графік для агломеративної кластеризації.

Для вибраної кількості кластерів було отримано розбиття і на його основі сформовані рекомендації. З використанням отриманих передбачених рейтингів розраховано середню помилку (MAE) для кожного розбиття. В табл. 4.1 представлені результати якості кластеризації і якість рекомендацій для кожного розбиття.

Таблиця 4.1 – Характеристики розбиття користувачів методом агломеративної кластеризації для різних мір і кількості кластерів

|  |  |  |  |
| --- | --- | --- | --- |
| Міра відмінності | Кількість кластерів | Значення якості | MAE |
| Кореляція Пірсона | 9 | 0,2564 | 1,6812 |
| 7 | 0,2272 | 1,5122 |
| Косинусна міра | 14 | 0,2332 | 1,6444 |
| 6 | 0,1948 | 1,5549 |
| 3 | 0,0872 | 1,5750 |
| Кореляція Спірмена | 7 | 0,1572 | 1,5768 |
| 4 | 0,1125 | 1,5458 |

З таблиці 4.1. видно, що при виборі кількості кластерів, при яких досягається максимальне значення якості кластеризації досить великою є похибка створених на основі розбиття рекомендацій для кожного користувача. Дуже велика кількість кластерів для вибірки погіршує якість рекомендацій, оскільки розбиття містить багато кластерів з кількістю об’єктів менше 5. Таке розбиття є дуже неефективним при створенні рекомендацій. Таким чином, кількість кластерів була обрана з огляду на досягнення локального максимуму якості в більш вузькому проміжку кількості кластерів, а саме на проміжку . При виборі кластерів з цього проміжку отримані кращі значення якості рекомендацій.

Розбиття за допомогою агломеративної кластеризації з використанням коефіцієнта Пірсона при завданні значення 7 у якості кількості кластерів дає найменшу похибку при формуванні рекомендацій, а саме помилку в бали.

На рис. 4.4 зображено розбиття, отримане при використанні агломеративної кластеризації при різних мірах відмінності. Червоним кольором виділено кластер, в якому знаходиться обраний користувач.

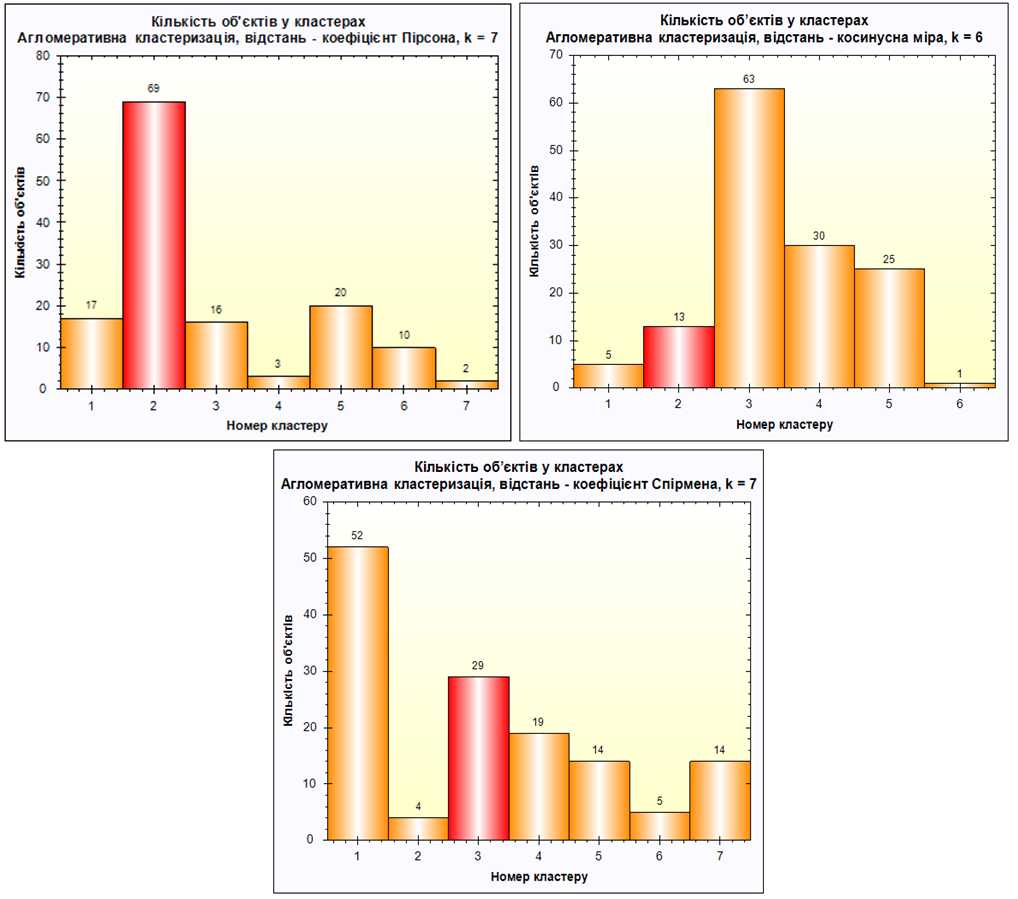


Рисунок 4.4 – Розбиття користувачів з використанням агломеративної кластеризації для різних мір відмінності

## Посібник користувача

Взаємодія користувача з системою відбувається через веб-інтерфейс. Для того, щоб користувач отримав персональні рекомендації від системи, йому необхідно виконати два кроки: пройти реєстрацію і оцінити набір більше ніж з двадцяти фільмів (для більш точної рекомендації бажано оцінити 50 фільмів). Після того як користувач зареєструвався і зайшов у систему під своїм ім’ям, він отримує можливість оцінювати фільми. Незареєстрований користувач має можливість лише проглядати інформацію о фільмах і виконувати пошук.

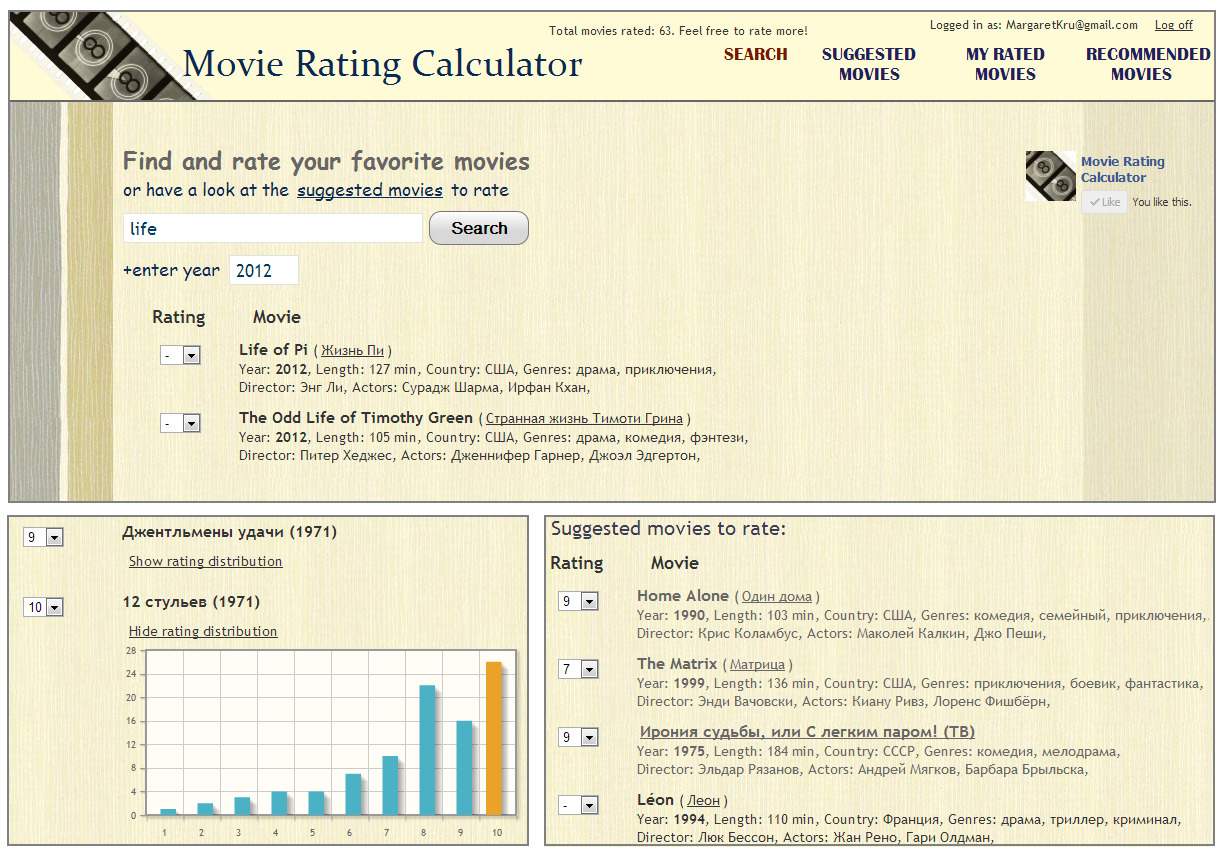


Рисунок 4.5 – Веб-інтерфейс: сторінки пошуку, статистики по оціненим фільмам і сторінки запропонованих для оцінки фільмів

На рис. 4.5 представлені сторінки розробленого веб-сайту. Сторінка пошуку дозволяє користувачу виконувати пошук фільмів по слову або його частині з назви фільму та по року виходу. Поруч з кожним фільмом знаходиться список для оцінки, який відображає рейтинг.

Однак, якщо користувач переглянув дуже велику кількість фільмів, задача пошуку кожного з них буде досить довгою і нецікавою. Для цього було розроблено сторінку з фільмами, які запропоновані користувачу для виставлення оцінки. Ця сторінка доступна з пункту меню «Suggested movies» і відображає всі фільми бази даних, які мають рейтинги. Фільми відсортовані в порядку зменшення кількості.

На сайті створено сторінку, яка містить список фільмів, які користувач оцінив. Ця сторінка містить статистичну інформацію щодо рейтингів до кожного оціненого фільму – кількість рейтингів та кількість користувачів, які оцінили фільм однаково з даним активним користувачем. Для кожного фільму з цього списку є можливість побудови гістограми, яка відображає розподілення користувачів на кожну можливу оцінку, яку отримав даний фільм. На рис. 4.5 представлено приклад гістограми для розподілення оцінок по фільму «12 стульев». Таким чином, користувач може дізнатися про кількість людей які поставили фільму конкретну оцінку та побачити в яку групу попав він сам.



Рисунок 4.6 – Сторінка з рекомендованими фільмами

Після того як зареєстрований користувач оцінив достатню кількість фільмів і проведено кластеризацію, він має змогу отримати список рекомендованих фільмів. Сторінка з рекомендаціями доступна через пункт меню «Recommended movies» і містить список Top-20 фільмів, які отримали найвищий рейтинг передбачений системою для цього користувача, які він не оцінив. На рис. 4.6 представлено фрагмент сторінки, яка містить рекомендовані фільми.

Висновки

В результаті виконання дипломної роботи була розроблена система для рекомендації фільмів. На основі рейтингів користувачів дана система формує для конкретного користувача передбачення у вигляді списку фільмів, які є найбільш близькими до його смаків.

Система для створення рекомендацій представляє собою веб-сайт як найзручніший спосіб зібрати велику кількість статистичних даних, з метою проведення аналізу. Розроблений інтерфейс веб-сайту максимально пристосований до потреб користувача, включає зручний пошук фільмів, систему, яка пропонує фільми для оцінювання і простий для розуміння і використання інтерфейс збору рейтингів.

Для аналізу зібраної статистичної інформації і подальшого створення списку рекомендованих фільмів застосовано математичний апарат колаборативної фільтрації на основі Model-based підходу, який передбачає створення описової моделі переваг або інтересів користувачів. Для створення моделі були використані алгоритми кластеризації даних і реалізовані наступні з них: агломеративна ієрархічна кластеризація, алгоритм K-medoids і алгоритм DBSCAN.

Для проведення аналізу і порівняння реалізованих алгоритмів створено окремий проект, в якому за допомогою міри якості кластеризації було порівняно роботу реалізованих алгоритмів розбиття даних при використанні різних мір відмінності, за допомогою міри якості рекомендацій порівняно методи створення рекомендацій на основі кластеризації користувачів та на основі фільмів при використанні різних алгоритмів кластеризації. На основі отриманих значень якості рекомендацій було обрано найбільш оптимальний алгоритм для кластеризації даних і відповідна міра відмінності та підібрані вхідні параметри для алгоритму.

Список джерел інформації

1. Інтернет видання: <http://www.imdb.com/stats> [↑](#endnote-ref-2)
2. Інтернет видання: <http://www.solarsquirrel.com/MIDS/System_Analysis.html> [↑](#endnote-ref-3)
3. Інтернет видання: <http://www.wikireality.ru/wiki/Имхонет> [↑](#endnote-ref-4)
4. Sarwar B. Item-based collaborative filtering recommendation algorithms. / B.Sarwar, G. Karypis, J. Konstan, J. Riedl // Proceedings of the 10th international conference on World Wide Web. – 2010. Vol. 5. – P. 285-295 [↑](#endnote-ref-5)
5. Manh Cuong Pham. A Clustering Approach for Collaborative Filtering Recommendation Using Social Network Analysis / Manh Cuong Pham, Yiwei Cao, Ralf Klamma, Matthias Jarke // Journal of Universal Computer Science. – 2011. Vol. 4. – P. 583-604 [↑](#endnote-ref-6)
6. Орлов А.И. Прикладная статистика. Учебник. / А.И. Орлов – М.: Экзамен, 2006. - 672 с. [↑](#endnote-ref-7)
7. Мандель И.Д. Кластерный анализ / И.Д. Мандель. – М.: Финансы и статистика, 1988. - 176 с. [↑](#endnote-ref-8)
8. Hae-Sang Park, Jong-Seok Lee and Chi-Hyuck Jun A K-means-like Algorithm for K-Medoids Clustering and Its Performance // Journal of Universal Computer Science. – 2009. Vol. 6 – P. 83-92. [↑](#endnote-ref-9)
9. Kaufman L. Finding Groups in Data: an introduction to Cluster Analysis. / L.Kaufman, P.J. Rousseeuw – John Wiley & Sons, 1990. – 342 p. [↑](#endnote-ref-10)
10. Martin Ester, Hans-Peter Kriegel, J&g Sander, Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, KDD’96. // Journal of Universal Computer Science. – 2008. Vol. 12 - Р. 226-231 [↑](#endnote-ref-11)
11. Гамма Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования. / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес – Питер, 2001. – 368 с. [↑](#endnote-ref-12)