АНОТАЦІЯ

Обсяг пояснювальної записки становить **105** сторінок, **53** ілюстрації, **27** таблиць та **47** джерел за переліком посилань.

Метою дослідження є розробка комп’ютерної системи для визначення стану кровообігу з додатковими модулями для дослідження подібності патернів та проведення глобальної кластеризації.

Завдання:

* Проведення логістичної регресії та дискримінантного аналізу для підтвердження ефективності алгоритму евклідової відстані
* Розробка модулю дослідження для оцінки подібності кластерів
* Пошук оптимальної кількості кластерів шляхом зменшення базової кількості
* Побудова нових результуючих таблиць;
* Проектування та реалізація програмного продукту для визначення стану системи кровообігу.

Для розробки програмного продукту було застосовано середовище програмування Microsoft Visual Studio 2017 Community, зокрема використано мова програмування C# та фреймворк .Net Framework 4.5 з використанням елементів Windows Form Application.

Результати магістерської дисертації описані в статтях:

«Estimation of Algorithms Efficiency in the Task of Biological Objects Clustering» та опубліковані в журналі Innovative biosystems and bioengineering, vol. 2 · no. 2;

«Automated Assessment of a Students Circulatory System Functional State Using Martine's Test» та опубліковані в журналі Innovative biosystems and bioengineering, vol. 2 · no. 3;

«Застосування алгоритму знаходження мінімальної відстані для визначення групи ризику студента» та опубліковані в журналі The scientific heritage № 23 (23), 2018;

«Порівняння систем прогнозування та алгоритму знаходження мінімальної відстані для визначення групи ризику студента» та опубліковані в журналі The scientific heritage № 23 (23), 2018.

Проект розроблено на замовлення кафедри «Фізичного виховання» НТУУ «КПІ ім. Ігоря Сікорського» та буде впроваджений в навчальну програму в 2018-2019 н.р. (акт впровадження від 10 грудня 2018 р.)

Ключові слова: кластер, дисперсійний аналіз, регресія, дискримінантний аналіз, програмний додаток, результуюча таблиця

ABSTRACT

Обсяг пояснювальної записки становить **105** сторінок, **53** ілюстрації, **27** таблиць та **47** джерел за переліком посилань.

Метою дослідження є розробка комп’ютерної системи для визначення стану кровообігу з додатковими модулями для дослідження подібності патернів та проведення глобальної кластеризації.

Завдання:

* Проведення логістичної регресії та дискримінантного аналізу для підтвердження ефективності алгоритму евклідової відстані
* Розробка модулю дослідження для оцінки подібності кластерів
* Пошук оптимальної кількості кластерів шляхом зменшення базової кількості
* Побудова нових результуючих таблиць;
* Проектування та реалізація програмного продукту для визначення стану системи кровообігу.

Для розробки програмного продукту було застосовано середовище програмування Microsoft Visual Studio 2017 Community, зокрема використано мова програмування C# та фреймворк .Net Framework 4.5 з використанням елементів Windows Form Application.

Результати магістерської дисертації описані в статтях:

«Estimation of Algorithms Efficiency in the Task of Biological Objects Clustering» та опубліковані в журналі Innovative biosystems and bioengineering, vol. 2 · no. 2;

«Automated Assessment of a Students Circulatory System Functional State Using Martine's Test» та опубліковані в журналі Innovative biosystems and bioengineering, vol. 2 · no. 3;

«Застосування алгоритму знаходження мінімальної відстані для визначення групи ризику студента» та опубліковані в журналі The scientific heritage № 23 (23), 2018;

«Порівняння систем прогнозування та алгоритму знаходження мінімальної відстані для визначення групи ризику студента» та опубліковані в журналі The scientific heritage № 23 (23), 2018.

Проект розроблено на замовлення кафедри «Фізичного виховання» НТУУ «КПІ ім. Ігоря Сікорського» та буде впроваджений в навчальну програму в 2018-2019 н.р. (акт впровадження від 10 грудня 2018 р.)

Ключові слова: кластер, дисперсійний аналіз, регресія, дискримінантний аналіз, програмний додаток, результуюча таблиця

ЗМІСТ

[ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ 7](#_Toc529958185)

[ВСТУП 8](#_Toc529958186)

[РОЗДІЛ 1 ЗАДАЧА З МЕДИЧНОЇ ТОЧКИ ЗОРУ 9](#_Toc529958187)

[1.1 Алгоритм квадрату евклідової відстані 9](#_Toc529958188)

[1.2 Опис бази даних студентів 9](#_Toc529958189)

[Висновки до розділу 1 10](#_Toc529958190)

[РОЗДІЛ 2 МАТЕРІАЛИ ТА МЕТОДИ ДОСЛІДЖЕННЯ 11](#_Toc529958191)

[2.1 Дисперсійний аналіз 11](#_Toc529958192)

[2.2 Регресійний аналіз 11](#_Toc529958193)

[2.2.1 Бінарна логістична регресія 11](#_Toc529958194)

[2.2.2 Лінійна регресія і метод найменших квадратів 11](#_Toc529958195)

[РОЗДІЛ 3 ПРОВЕДЕННЯ АНАЛІЗІВ ДЛЯ РОЗРОБКИ КОМП’ЮТЕРНОЇ СИСТЕМИ 12](#_Toc529958196)

[Висновки до розділу 3 21](#_Toc529958197)

[РОЗДІЛ 4 ПРОГРАМНИЙ ПРОДУКТ ДЛЯ ВИЗНАЧЕННЯ ФУНКЦІОНАЛЬНОГО СТАНУЧИСТЕМИ КРОВООБІГУ 21](#_Toc529958198)

[4.1 Проектування програмного продукту 21](#_Toc529958199)

[4.2 Розробка програмного продукту 30](#_Toc529958200)

[4.3 Робота з програмним продуктом 30](#_Toc529958201)

[Висновок до розділу 4 41](#_Toc529958202)

[ЗАГАЛЬНІ ВИСНОВКИ 42](#_Toc529958203)

[ПЕРЕЛІК ПОСИЛАНЬ 43](#_Toc529958204)

[ДОДАТОК А 46](#_Toc529958205)

[ДОДАТОК Б 49](#_Toc529958206)

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

АТС – систолічний артеріальний тиск

АТД – діастолічний артеріальний тиск

ЧСС – частота сердечний скорочень (пульс)

БД – база даних

ПП – програмний продукт

# ВСТУП

Систематичний самоконтроль стану здоров’я студента дозволяє оптимізувати роботу факультету фізичного виховання НТУУ «КПІ ім. Ігоря Сікорського» та дозволяє підібрати найбільш оптимальний вид спорту, що сприятиме поліпшенню функціонуванню системи організму. Одним із найпоширеніших методів спостереження за фізичним станом організму є функціональна проба Мартіне, яка дає можливість відслідковувати динаміку зміни артеріального тиску та частоти серцевих скорочень між станом спокою та на кожній хвилині після навантаження, включно до п’ятої. У більшості випадків характеристика зміни пульсу і тиску під час функціональної проби досить точно відображає стан серцево-судинної системи у студента. Визначення характерних закономірностей станів системи кровообігу дозволить розробити механізм порівняння цих станів а також вдосконалити алгоритм для оцінки функціонального стану системи кровообігу.

Актуальним є моніторинг фізіологічного стану протягом усього періоду фізичних і спортивних тренувань з періодичним визначення регуляторних реакцій на тестове навантаження. Це потребує розробки нових модулів для досліджень та вдосконалення програмного продукту для визначенню функціонального стану. Результатом використання програми є пришвидшення роботи лікарів та визначення діагнозу без довготривалої затримки.

Мета роботи: вдосконалення системи реєстрації змін функціонального стану системи кровообігу шляхом порівняння параметрів тесту із заздалегідь відомими функціональними схемами кровообігу з подальшим розробленням системи порівняння функціональних патернів.

У відповідності з метою ставлять такі завдання:

* Дослідження існуючої проблеми.
* Проведення глобальної кластеризації.
* Проведення дискримінантного аналізу та логістичної регресії для визначення коректності алгоритму евклідової відстані.
* Поєднання та автоматизація обраних методів.
* Реалізація програмного додатку.
* Зменшення кількості кластерів.
* Побудова нових результуючих таблиць з виведенням графіків.
* Визначення внутрішньогрупової дисперсії.
* Дослідження та вибір оптимальної кількості патернів.
* Вдосконалення програмного продукту.

Об’єкт дослідження. База даних студентів та викладачів Національного Університету України «Київського політехнічного університету ім. Ігоря Сікорського».

Предмет дослідження. Алгоритм розрахунку коефіцієнтів рівняння регресії з виведення графічного матеріалу на екран користувача.

Методи дослідження. Програмний продукт «Clusterbox» та відповідні результуючі таблиці. Для реалізації програмного продукту буде застосовано середовище програмування Microsoft Visual Studio 2017 Community, зокрема використано мову програмування C# та фреймворк .Net Framework 4.5 з використанням елементів Windows Form Application.

Наукова новизна одержаних результатів. Реалізація програмного комплексу для визначення подібності функціональних груп. Розробка методу автоматичної глобальної кластеризації на базі квадрату евклідової відстані з вибором кількості кластерів. Розробка модулю універсальної кластеризації, що передбачає задати будь-яку кількість змінних та кластерів для дослідження.

Практичне значення одержаних результатів. Програмний продукт може бути використаний в медичних клініках та учбових закладах для оцінки функціональних реакцій організму. Насамперед використання програми передбачено на факультеті фізичного виховання Національного технічного університету України «Київського політехнічного університету ім. Ігоря Сікорського».

Апробація результатів дисертації. Результати досліджень були оприлюднені у доповіді «Оцінка функціональних реакцій на тестове навантаження у студентів 1-2 курсу. Жінки. Чоловіки» на конференції «Презентація наукових розробок студентів і аспірантів» за участі корейської делегації від 28 вересня 2017, що проходила в корпусі №1, кімнаті № 155 Національного технічного університету України «Київського політехнічного університету ім. Ігоря Сікорського».

Публікації. Результати магістерської дисертації описані в 4 статтях: «Estimation of Algorithms Efficiency in the Task of Biological Objects Clustering» та опубліковані в журналі Innovative biosystems and bioengineering, vol. 2 · no. 2; «Automated Assessment of a Students Circulatory System Functional State Using Martine's Test» та опубліковані в журналі Innovative biosystems and bioengineering, vol. 2 · no. 3; «Застосування алгоритму знаходження мінімальної відстані для визначення групи ризику студента» та опубліковані в журналі The scientific heritage № 23 (23), 2018; «Порівняння систем прогнозування та алгоритму знаходження мінімальної відстані для визначення групи ризику студента» та опубліковані в журналі The scientific heritage № 23 (23), 2018.

Структура дисертації. Дисертація побудована за класичним типом та викладена на **208** сторінках машинописного тексту. Складається з вступу, **4** розділів, висновків, списку використаних літературних джерел, який містить **187** найменувань, **6** – на кирилиці, **181** – на латиниці. У роботі представлено **15** рисунків і **45** таблиць.

# РОЗДІЛ 1 ОГЛЯД ЛІТЕРАТУРИ ЗА ТЕМОЮ І ВИБІР НАПРЯМІВ ДОСЛІДЖЕНЬ

## Алгоритм квадрату евклідової відстані

Квадрат евклідової відстані є однією із мір відстані, що використовуються для встановлення подібності або відмінності об’єктів класифікації. Зазвичай використовують просту евклідову відстань, що в багатовимірному просторі є геометричною. Але, якщо ознаки досліджуваних об’єктів були виміряні в різних одиницях, то евклідова відстань може втратити сенс. Тому для ефективного використання даного алгоритму доцільно проводити нормування ознак кожного об’єкту дослідження.

Використання квадрату евклідової відстані виправдане в тих випадках, коли надання більшого значення більш віддаленим об’єктам один від одного підвищує якість класифікації об’єктів. У тому випадку, коли слід дослідити відстань між об’єктами, які за однаковим набором змінних є різними, то доцільно використовувати квадрат евклідової відстані.

Прикладом використання квадрату евклідової відстані є різноманітні алгоритми кластеризації. Зазвичай комп’ютерні програми використовують евклідову відстань за замовчуванням в стратегіях об’єднання або в методі деревоподібної кластеризації. Невід’ємною частиною використання даного алгоритму є статистика, зокрема пакет для обробки даних – IBM SPSS. Також важливу роль квадра евклідової відстані грає в алгоритмах оптимізації та машинному навчанні.

Ще одним прикладом використання алгоритму є реалізація експертних систем прийняття рішень, серед яких є навіть система для вибору дипломного керівника для студента. Критерій вибору будується по характеристикам студентів з вказанням їх значень, але самі характеристики студенти надають індивідуально та самостійно. За наданими характеристиками будується об’єкт класу «Образ». Задача пошуку наукового керівника полягає у визначенні найбільш близького екземпляру класу «Викладач», що був побудований за наданими характеристиками викладача, для сформованого об’єкта «Образ». Якщо кожен екземпляр класу «Викладач» розглядати як окремий кластер, тоді задача зведеться до кластеризації наступного типу: заданий об’єкт класу «Образ» необхідно класифікувати в один із кластерів виходячи з деякої міри близькості. Подібність між об’єктом класифікації та кластерами визначається у залежності від метричної відстані між ними. Таким чином задача класифікації зводиться до задачі визначення функції близькості між об’єктами даних класів – вибір міри відстані між об’єктами. Оскільки відстань Махаланобіса доцільно використовувати у випадку, коли кореляція між змінними є ненульовою, то ефективним для даної задачі є використання квадрату евклідової відстані.

Відстань між двома об’єктами складається із суми різниці значення ознак двох об’єктів. Так як одна ознака може характеризуватися декількома значеннями, то кожний доданок може бути представлений у виді суми різниці декількох значень однієї ознаки або кожне значення (кожна характеристика) однієї ознаки може бути розглянуто як значення окремої ознаки.

Таким чином для реалізації критерію вибору наукового керівника було побудовано експертну систему, що в своїй сутності використовує квадрат евклідової відстані. База даних експертної системи включає характеристики викладачів. Дерево рішень представляє собою набір правил для обрахунку функції близькості між об’єктами, наведеними в базі даних, і об’єктом класу «Образ», характеристики якого визначає студент (користувач експертної системи). Найдені значення функції близькості задають рейтинг для кожного об’єкта класу «Викладач».

Кожне рішення експертної системи складається зі списку об’єктів класу «Викладач» із вказанням їх характеристик, а також рейтингу даного об’єкта. Рейтинг об’єкта визначається за формулою евклідової відстані між цим об’єктом і об’єктом класу «Образ».

Також яскравим прикладом використання алгоритму квадрату евклідової відстані є задачі з області біології та медицини. Наприклад, розроблена система «Clbsterbox» дозволяє дослідити функціональний резерв організму людини та визначити якість його відновлення після фізичних навантажень. Зокрема програма дозволяє оцінити стан системи кровообігу за пробою Мартіне. Сама проба полягає у наступному: необхідно виміряти значення артеріального тиску до навантаження і записати дані до окремої таблиці. Після цього зробити 20 присідань за 30 секунд – стандартне фізичне навантаження. Наступним кроком є замір значень артеріального тиску та пульсу на кожній хвилині після навантаження. Дані заміри необхідно робити протягом п’яти хвилин та заносити значення до таблиці.

Таким чином оцінюється пристосування організму до різних видів фізичних навантажень з різною інтенсивністю. Серед всіх реакцій серцево-судинної системи виділяють 5 основних типів, які представлені на рисунку 1.1.



Рисунок 1.1 – Основні типи реакції серцево-судинної системи на фізичні навантаження

Зазвичай проба Мартіне використовується у спортивних сферах, для визнання фізичних можливостей осіб, швидкості відновлення організму після та підчас фізичних навантажень різних видів. В клінічній практиці цей метод дозволяє вивчати функціональні можливості серцево-судинної системи враховуючи вікову категорію піддослідного. Виходячи з практичного досвіду, були встановлені оптимальні норми необхідної фізичної активності для певних вікових категорій при використанні проби Мартіне. Таким чином:   
для осіб до 40 років без виражених відхилень у стані здоров’я – 20 присідань на 30 секунд;   
для осіб до 50 років – 15 присідань на 22 секунд;   
для осіб більше 50 років без виражених відхилень – 10 присідань на 15 секунд.

Якщо результати проби вкладаються в нормологічний тип реакції то й функціональний стан серцево-судинної системи вважається задовільним.

Існує можливість використання проби Мартіне і в діагностичних цілях. Наприклад – якщо показники проби потрапляють до несприятливого типу реакції, це є основою вважати що тахікардія в стані спокою є показником захворювань серцево-судинної системи. У випадку, коли до навантажень пульс є стабільним, а відновлення відбувається хвилеподібно, може виникнути негативна фаза пульсу. Нерідко виникає ситуація коли пульс нормалізується на показниках нижчих. ніж ті, що були до навантаження – це є підставою вважати що тахікардія підчас стану спокою зумовлена порушеннями нервової системи. Якщо до навантаження ЧСС вище норми, а після проби всі показники входять до нормологічного типу реакції, але, при цьому, пульс є підвищеним чи таким же як і до навантаження - припускають, що тахікардія в спокої зумовлена гіперфункцією щитовидної залози. Подальші обстеження можуть уточнити, а частіше підтвердити результати функціональних проб.

Після проведення проби Мартіне для нашого дослідження буде отримано 18 значень, що характеризують динаміку зміни артеріального тиску. За даними значеннями можна побудувати графіки ЧСС-АТС та ЧСС-АТД і побачити як відновлюється організм після фізичних вправ. Якщо значення до навантаження не близькі до значень після навантаження, тоді це свідчить про погану відновлюваність організму та потребує внесення певних змін до способу життя пацієнта.

Для дослідження групи ризику студента доцільно використовувати програмний продукт «Clusterbox». В його наявності є 7 груп, що характеризують стан відновлення кровоносної системи для чоловіків та 8 – для жінок. Кожна група містить свої значення артеріального тиску до навантаження та значення на кожній хвилині після навантаження, включно до п’ятої хвилини. Всі вони занесені до спеціальної таблиці, що називається результуючою таблицею, яка поставляється разом з програмним продуктом. Також кожна група містить свої рекомендації щодо покращення стану системи кровообігу та відповідні характеристики.

Оскільки у наявності програмного продукту є 7 груп чоловічої статні та 8 жіночої, тоді задача полягає у визначення до якої з груп відноситься пацієнт зі своїми значеннями тесту. З цього випливає звичайна задача класифікації об’єкта (розглянуто з боку чоловіків): у наявності 7 класифікаційних груп та один об’єкт, що необхідно класифікувати. Для цього треба визначити на якій відстані знаходиться об’єкт до кожної класифікаційної групи та знайти мінімальну відстань. Алгоритм знаходження мінімальної відстані побудований на основі алгоритму квадрату евклідової відстані і працює наступним чином: від показника студента, що досліджується віднімаємо середнє значення в групі, а результат підносимо до квадрату*.* Дану процедуру повторюємо 18 разів, оскільки в нас 18 показників. Після проведення цієї процедури всі результати додаються і отримане значення стає відстанню до центру групи (квадрат евклідової відстані). Такі обчислення проводяться для кожної групи окремо, а відстань, яка буде мінімальною, характеризуватиме групу, до якої відноситься студент.

Результатом роботи алгоритму та програми в цілому є виведення знайденої групи і її характеристик на екран користувача. Таким чином можна буде дослідити стан фізичного здоров’я окремого пацієнта.

Проблемою даного програмного продукту є неможливість визначити групу ризику для декількох студентів або цілої бази даних і збереження результатів до окремої таблиці. Також проблематика полягає у неможливості порівняти групи між собою. Ми можемо побудувати графіки зміни артеріального тиску і побачити розташування даних груп у просторі, але це не дає змоги оцінити їх подібність або відмінність у повному обсязі. Для цього достатньо було б реалізувати алгоритм лінійної регресії для додаткової інтерпретації класифікаційних груп. Ще одним недоліком програми є те, що вона працює лише з однією таблицею, яка йде в комплекті з нею, а також немає можливості вибору кількості змінних для дослідження. Наприклад, ми б хотіли дослідити відновлення організму на шести хвилинах, але програмним продуктом просто не передбачено такого функціоналу.

Таким чином ми бачимо, що програмний продукт є досить корисним, але його функціонал дуже обмежений. Він не є універсальним та не може бути використаний для інших досліджень, тому необхідним є розроблення додаткових модулів для дослідження, розширення функціоналу та можливостей продукту.

## 1.2 Дисперсійний аналіз

Дисперсійний аналіз - аналіз мінливості ознаки під впливом будь-яких контрольованих змінних факторів (ANOVA - «Analisis of Variance»). Основною метою дисперсійного аналізу є дослідження значущості відмінності між середніми.

Мета дисперсійного аналізу - дослідження наявності або відсутності істотного впливу будь-якого якісного або кількісного фактору на зміни досліджуваної результативної ознаки. Для цього фактор, який імовірно має або не має істотного впливу, поділяють на класи (інакше кажучи, групи) і з'ясовують, чи однаковий вплив фактору шляхом дослідження значущості між середніми в наборах даних, у відповідних градаціях фактору.  
При дисперсійному аналізі визначають питому вагу сумарного впливу одного або декількох факторів. Істотність впливу фактору визначається шляхом перевірки гіпотез:

, де a – число класів градації – всі класи градації мають одне значення середніх, : не всі  рівні – не всі класи градації мають одне значення середніх.

Якщо вплив фактору не суттєвий, то несуттєва і різниця між класами градації цього фактору і в ході дисперсійного аналізу нульова гіпотеза H0 не відкидається. Якщо вплив фактору істотний, то нульова гіпотеза H0 відхиляється: не всі класи градації мають одне і теж середнє значення, тобто серед можливих різниць між класами градації одна або кілька є суттєвими.

Схематично дисперсійний аналіз поділяється на певні категорії, які визначаються у залежності від кількості факторів, які беруть участь у дослідженні, кількості змінних, на які може впливати фактор та від співвідношення вибірок значень між собою. Якщо при аналізі наявний лише один фактор, вплив якого досліджується, тоді такий аналіз називається однофакторним та поділяється на два види:

* Аналіз вибірок, що не пов’язані між собою (аналіз різних, незв’язних вибірок)
* Аналіз вибірок, що пов’язані між собою. Наприклад, коли на одній і тій самій групі проводять декілька вимірів, але в різних умовах

Якщо при аналізі необхідно дослідити одночасний вплив двох і більше факторів, тоді такий аналіз називається фагатофакторним, який в свою чергу ділиться на категорії за типом вибірки.

**Однофакторний дисперсійний аналіз**

У разі однофакторного дисперсійного аналізу мається на увазі, що середні генеральних сукупностей, з яких були вилучені вибірки, - рівні, іншими словами, всі вони відносяться до однієї генеральної сукупності і відмінності носять випадковий характер. Для перевірки теорій в разі дисперсійного аналізу використовується F-розподіл. F-статистика приймає тільки позитивні або нульові значення.

Процедура дисперсійного аналізу полягає у визначенні співвідношення систематичної (групової) дисперсії до випадкової (внутрішньогрупової) дисперсії в вимірюваних даних. Як показник мінливості використовується сума квадратів відхилення значень параметра від середнього: SS (Sum of Squares). Загальна сума квадратів SSTotal розкладається на міжгрупову суму квадратів SSBG і внутрішньогрупову суму квадратів SSWG :

SSTotal = SSBG + SSWG

У разі якщо вірна , то як внутрішньогрупова, так і міжгрупова дисперсії є оцінками однієї і тієї ж дисперсії і повинні бути приблизно рівні.

|  |  |
| --- | --- |
| , де | (1.1) |



Виходячи з цього значення F має бути близько до 1 в разі, якщо статистично значущих відмінностей все-таки немає. Критичне значення F визначається рівнем значущості (зазвичай 0,05 або 0,01) і внутрішньогруповим і міжгруповим числом ступенів свободи (ν). Воно досить складне для обчислення, тому частіше використовуються табличні значення із зазначенням α, νBG, νWG.

Міжгруповое число ступенів свободи:

νBG = m - 1.

m – число груп.

Внутрішньогрупове число ступенів свободи:

νWG = n – m.

n – кількість спостережень в кожній з груп.

## 1.3 Регресійний аналіз

Основною метою регресійного аналізу є прогнозування значення залежної змінної використовуючи змінні, які є незалежними. Також метод регресійного аналізу використовують для побудови моделі, яка описує залежність між змінними, з подальшою оцінкою значимості отриманого рівняння.

До використання методів регресійного аналізу зазвичай переходять, якщо зв’язки між досліджуваними параметрами мають статистичну значимість. Для цього спеціально шукають такий набір функцій, який пов’язує аргументи з результативним показником. Після цього будується рівняння регресії, розраховується коефіцієнт детермінації і аналізується його точність.

Метод регресійного аналізу потребує, щоб аргументи були незалежними та мали нормальний розподіл з константними значеннями дисперсій. Аналіз побудови регресіних рівнянь має на меті пошук залежності між ознаками, які досліджуються. Знайдене рівняння буде показувати, як у середньому змінюється *у* при зміні будь-якого з і матиме вигляд:

|  |  |
| --- | --- |
|  | (1.2) |

де - залежна змінна (вона завжди одна);

- незалежні змінні (фактори) (їх може бути декілька).

Аналогічно до дисперсійного аналізу, регресійний називається простим (однофакторним), якщо у дослідженні є одна незалежна змінна. В іншому випадку він є багатофакторним.

У порівнянні з кореляційним аналізом, який показує наскільки істотним є зв’язок, регресійний передбачає, що зв’язок є і шукає модель цього зв’язку, яка описується рівнянням регресії.

У випадку, якщо між змінними можна виразити кількісне відношення у вигляді комбінації обраних змінних, тоді доцільним є використання методу регресії. У такому випадку комбінація змінних буде використовуватися для прогнозування значення, що може приймати залежна змінна, яка обчислюється на значеннях незалежних змінних у певному наборі, який був заданий дослідником. Для найпростішої побудови моделі на базі отриманого рівняння регресії використовують метод лінійної регресії, але, на жаль, він дає значну похибку і тому більшість моделей не можна якісно побудувати за допомогою цього методу.

Для детальнішого опису рівняння, отримане за допомогою регресійного аналізу, необхідно знати умовний закон розподілу результативного показника *у.* На практиці не завжди вдається отримати дану інформації, тому у більшості випадків використовують пошук апроксимацій для функції заснованих на вихідних статистичних даних. У рамках окремих модельних припущень про тип розподілу вектора показників  може бути отриманий загальний вигляд рівняння регресії

|  |  |
| --- | --- |
|  | (1.3) |

Для відновлення за вихідними даними, що дасть найкращий результат, найчастіше використовують наступні критерії адекватності:[29].

* Метод найменших квадратів, згідно з яким мінімізується квадрат відхилення спостережуваних значень результативного показника  , від модельних значень , де коефіцієнти рівняння регресії;  значення вектора аргументів на "-М спостереженні:

|  |  |
| --- | --- |
|  | (1.4) |

* Метод найменших модулів , згідно з яким мінімізується сума абсолютних відхилень спостережуваних значень результативного показника від модульних значень , тобто

|  |  |
| --- | --- |
|  | (1.5) |

* Метод мінмакса зводиться до мінімізації максимуму модуля відхилення спостережуваного значення результативного показника у, від модельного значення , тобто

|  |  |
| --- | --- |
|  | (1.6) |

Найчастіше використовується метод найменших квадратів для побудови моделі регресії, що передбачає мінімізацію суми квадратів відхилень відхилень фактичних значень результатного ознаки від його розрахункових значень, тобто:

|  |  |
| --- | --- |
|  | (1.7) |

де *y* - число спостережень;

розрахункове значення результатного фактору.

Задачу МНК розв’язують за допомогою оцінки функції регресії, побудованої параметрично, шляхом аналізу заледності однієї величини*Y*, значення якої () від класу невипадкових величин .

Припустимо, що нам відомо, що досліджуваний процес у лінійно залежить від параметрів входу х.

Нехай відомо, що вихідний параметр процесу, який вивчається, позначимо його *y*, лінійно залежить від вхідного параметра *x*

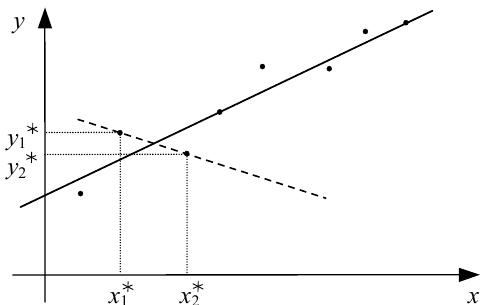


Рисунок 1.2 — Графічна інтерпретація причин, які обумовлюють необхідність використання МНК

Таким чином можемо сказати, що даний процес може бути описаний у вигляді наступного рівняння:

|  |  |
| --- | --- |
|  | (1.8) |

де *a* і *b* — коефіцієнти, значення яких обчислюються за допомогою відомих значень, що передаються величині х для визначення вихідної величини у.

Таким чином можна скласти систему з двох алгебраїчних рівнянь:

|  |  |
| --- | --- |
|  | (1.9) |

Оскільки будь-які експериментальні вимірювання дають похибку, що обумовлена різними факторами, то для оцінки і зменшення її можна використовувати МНК. Таким чином через наявність похибки в експериментальних даних, розв’язок системи рівнянь буде також складати похибку.

Наприклад, якщо використати лише значення  (рис. 1.2) для розв’язання системи рівнянь (2.2), то похибка буде вже не у відсотках, а у характері функціональної залежності (пунктирна лінія на рис. 1.2).

Таким чином німецький математик Фрідріх Гаусс запропонував розв’язок системи рівнянь, що направлений на пошук коефіцієнтів *a*, *b* моделі (1.9), де необхідним було сформувати суму квадратів різниць  між теоретично заданими за допомогою рівняння (1.8) значеннями вихідної координати *y* при значеннях аргументу   та її експериментальними значеннями :

|  |  |
| --- | --- |
|  | (1.10) |

Наступним кроком був пошук таких значень коефіцієнтів, які мінімізували б рівняння (1.10).

Від цієї процедури і назва методу — метод найменших квадратів.

З курсу вищої математики нам відомо, що мінімум функції знаходиться за допомогою взяття похідної від цієї функції та прирівняти її до нуля.

Згідно з цією ідеєю, підставимо у вираз (1.10) замість  його значення з (1.8) і візьмемо від отриманого виразу частинні похідні за *b* та *a*, які прирівняємо до нуля, тобто

|  |  |
| --- | --- |
|  | (1.11) |
|  | (1.12) |

За допомогою математичних перетворень можемо отримати систему нормальних рівнянь Гаусса (1.13). Розв’язок даної системи дозволить знайти такі значення параметрів, які будуть мінімізувати систему рівнянь.

|  |  |
| --- | --- |
|  | (1.13 |

**1.4 Дискримінантний аналіз**

За своєю природою дискримінантний аналіз схожий на метод регресійного аналізу. Його часто використовують у тому випадку, коли є набір класифікованих даних і необхідно знайти хоча б одну функцію, яка виступатиме у ролі моделі, здатну віднести певне дослідження до однієї з визначених груп. Таким чином основною метою дискримінантного аналізу є класифікації даних - визначення класу, до якого належить новий об'єкт.

Дискримінантний аналіз передбачає наявність навчальної вибірки, де заздалегідь відомо який об’єкт до якого класу відноситься. На навчальній вибірці будується модель, яка в майбутньому дозволяє класифікувати нові об’єкти.

У ролі дискримінантного аналізу найчастіше береться лінійна функція записана у вигляді формули (1.14):

|  |  |
| --- | --- |
| , | (1.14) |

де *Х1,Х2,…,Хm*– значення ознак у даного об'єкта;

*С1,С2,…,Сm* – дискримінанті множники.

За допомогою дискримінантних множників виконуємо перехід від m-мірного простору первинних показників до одновимірного простору.

Лінійну функцію можна розглядати як проекцію даного об'єкта на деяку (одновимірну) дискримінантну вісь.

У процедурі дискримінантного аналізу дискримінантні множники визначаються таким чином, щоб забезпечити найбільшу відмінність між проекціями першої та другої вибірок на дискримінантну вісь.

Дискримінантний аналіз потрібно проводити з використанням мінімальної кількості функцій. Їхня кількість залежить від конфігурації класів в багатовимірному просторі дискримінантних змінних. Щоб визначити, скільки функцій необхідно, використовують перевірку функцій на значимість. Для оцінки значущості використовують або А-статистику Уілкса або ксі – квадрат [5].

Критерій значення Уілкса обчислюють за формулою (1.15):

|  |  |
| --- | --- |
| , | (1.15) |

де *К –* кількість значень;

*k –* число вже обчислених дискримінаційних функцій.

Чим ближче значення критерію *К*, тим краща відмінності класів, а чим ближче до 1, тим відмінність гірша.

Значення ксі-квадрат розраховують за формулою (1.16):

|  |  |
| --- | --- |
| , | (1.16) |

де *р –* кількість членів у дискримінатної функції, виключаючи вільний член функції.

Якщо це значення більше критичного із заданим рівнем значущості і числом ступенів свободи *(р-к) (К-k-1)*, то значимість підтверджується.

Канонічна дискримінантна функція для загального випадку k класів записана у формулі (1.17):

|  |  |
| --- | --- |
| , | (1.17) |

де *fki* — значення канонічної дискримінантної функції для 1-го об'єкта в *k*-му класі;

uj — шукані коефіцієнти дискримінантної функції;

*Хjki* — значення дискримінантної змінної *Хj* для i-го об'єкта в класі *k*.

Функцію будують таким чином, щоб її середні значення для різних класів якомога більше розрізнялися. При цьому сукупність функцій повинна утворювати ортогональний простір, тобто функції - незалежні один від одного. З цього випливає, що кількість функцій нe може бути більше кількості класів мінус 1 або числа дискримінантних змінних (в залежності від того, яка з цих величин менше).

Розраховані значення канонічної дискримінантної функції fki, розглядають як точки в деякому просторі. Для кожної групи можна розрахувати центр групування. Тому в цій новій системі координат для нового об'єкта розраховують відстань від нього до кожної точки групування. Зазвичай для цього використовують квадрат відстані Махаланобіса.

## Висновки до розділу 1

Таким чином в даному розділі було розглянуто застосування алгоритму квадрату евклідової відстані в різних областях науки. Наведено приклад застосування в задачах біології та медицини, а також надано оцінку та характеристику програмному продукту для визначення групи ризику студента. Окрім даної системи аналогів знайдено не було. Ми визначили вектор направлення нашої наукової діяльності, розглянули теоретичні відомості щодо досліджень, які будуть проведені для покращення та підтвердження істинності попередніх досліджень Також нами було вияснено необхідні проблеми, які треба розв'язати шляхом розробки нових модулів для програмного продукту «Clusterbox».

# РОЗДІЛ 2 МАТЕРІАЛИ ТА МЕТОДИ ДОСЛІДЖЕННЯ

## 2.1 Результуюча таблиця програмного продукту «Cluserbox».

Результуюча таблиця, що використовується за замовчуванням програмним продуктом «Clusterbox», побудована на базі даних студентів молодших курсів Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського», що містить 1495 досліджень. У її наявності є 669 обстежень жіночої статі та 826 спостережень чоловіків, що показують стан системи кровообігу. Кількість груп в результуючій таблиці для кожної статі окремо було визначено за допомогою кластерного аналізу, що був проведений Настенко Є.А. та Носовець О.К.

Аналіз кластеризації – це багатомірна процедура с статистики, яка направлена на збір даних, що несуть інформацію про об’єкти, а точніше, про їх вибірку, з подальшим впорядкуванням об’єктів в порівняно-однакові класи, групи чи кластери. Кластеризація є одним із методів, що відносять до класу навчання без учителя [12].

Зазвичай до аналізу кластеризації до нього неможливо застосувати алгоритми перевірки статистичної значимості, проте його результат надає найбільш можливі вагомі значення. Не маючи апріорних гіпотез щодо класу даних. Дослідник може використати цю особливість та застосувати алгоритм кластеризації.

В свою чергу, кластеризація ставить за мету структуризації отриманих даних. Тобто їх організації в певну структуру. Сам кластерній аналіз включає в себе набур декількох різних алгоритмів , які направлені на виконання задачі класифікації.

При класифікації великих масивів даних на групи кластерний аналіз є інструментом, який неможливо замінити, оскільки він дає змогу зробити це без особливих проблем [13].

Алгоритм кластеризації, що був застосований для визначення кількості груп носить назву – метод кластеризації к-середніх, і являє собою версію ЕМ-алгоритму, котрий також застосовується для розділення суміші гаусової функції. Він розбиває множину елементів векторного простору на завчасно відоме число кластерів k. Робота алгоритму зводиться до мінімізації середньоквадратичного відхилення на точках кожного кластеру. Основна ідея – це те, що на кожній ітерації перераховується центр кластерів для кожного кластеру, котрий був отриманий на попередньому кроці.

|  |  |
| --- | --- |
|  | (2.1) |

Після цього вектори розбиваються на кластери знову, згідно з тим, який з обраних центрів виявився ближче до метрики [14].

В результаті отримано стовпчик з номерами кластерів, що характеризують групу ризику. База даних, що була використана для побудови результуючої таблиці для використання алгоритму визначення оптимальних характеристик студента, вже містила відповідний стовпчик з групами та представляла собою файл формату \*.xls (рис.2.1)

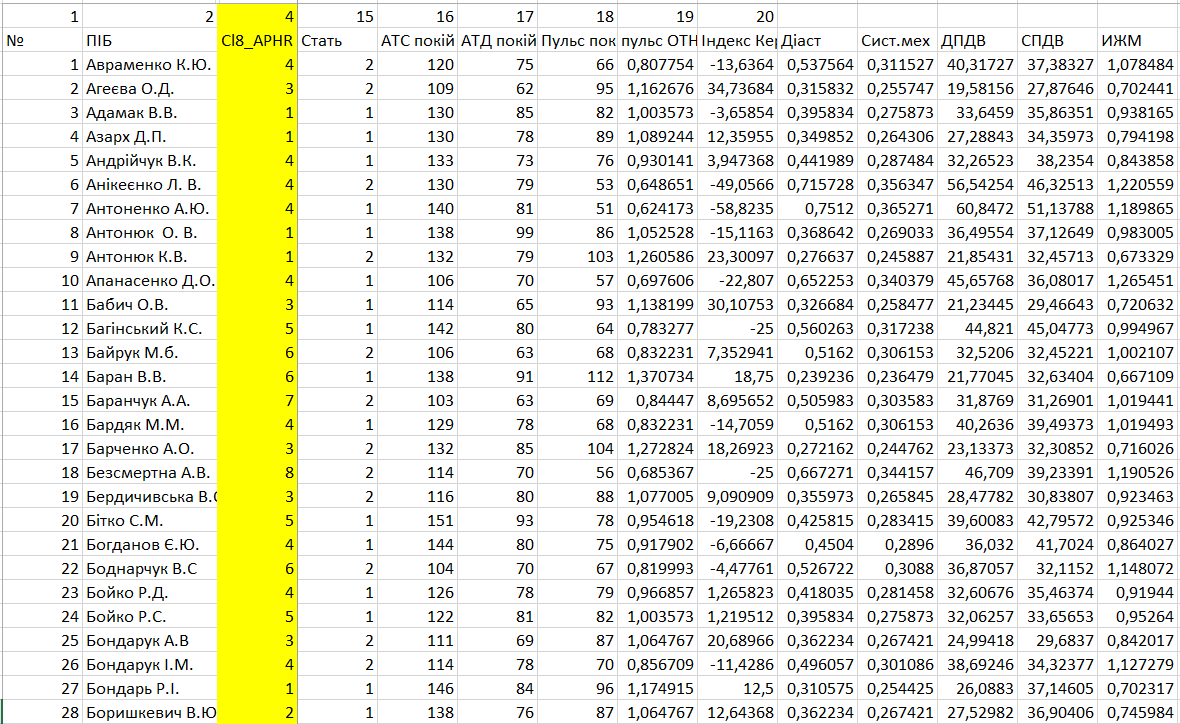


Рисунок 2.1 – Приклад бази даних студентів

База даних також містила показники дихальної системи (ємність легень, проба Генче, проба Штанге, частота дихання), нервової системи (проста зорова-моторна реакція, складна зорово-моторна реакція, режим нав’язаного ритму та ін.), кровоносної системи (артеріальний тиск, пульс, індекс Кердо та ін.) та показники фізичного стану (вік, зріст, вага, індекс маси тіла та ін.). Але для дослідження було обрано показники кровоносної системи, зокрема, артеріальний тиск та пульс у стані спокою, а також на 1-5 хвилинах після навантаження.

Вхідну таблицю було перетворено до прийнятного вигляду, який без проблем сприймається програмою для обробки даних в SPSS. Після цього був проведени дисперсійний аналіз для визначення середьного значення для кожної змінної в кожному кластері.

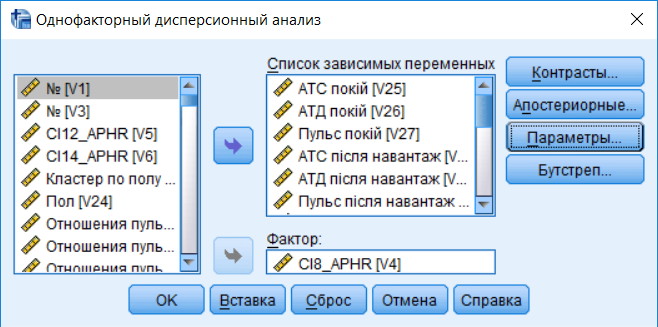


Рисунок 2.2 – Вибір необхідних параметрів для аналізу

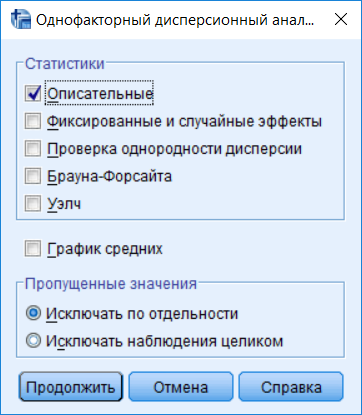


Рисунок 2.3 – Вибір потрібних статистик

Результатом проведення дисперсійного є наступна таблиця:

Таблиця 2.1 – Описові статистики по кластерам

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | N | Среднее | Стд. отклонение | Стд. Ошибка |
|
| АТС покій | 1 | 113 | 135,84 | 9,647 | ,908 |
| 2 | 130 | 119,08 | 9,518 | ,835 |
| 3 | 106 | 124,02 | 9,499 | ,923 |
| 4 | 156 | 131,24 | 9,641 | ,772 |
| 5 | 44 | 137,68 | 12,649 | 1,907 |
| 6 | 46 | 142,39 | 13,807 | 2,036 |
| 7 | 199 | 117,49 | 9,740 | ,690 |
| Итого | 794 | 126,50 | 13,033 | ,463 |
| АТД покій | 1 | 113 | 80,23 | 6,291 | ,592 |
| 2 | 130 | 70,73 | 6,193 | ,543 |
| 3 | 106 | 76,43 | 6,243 | ,606 |
| 4 | 156 | 74,44 | 5,686 | ,455 |
| 5 | 44 | 83,52 | 6,048 | ,912 |
| 6 | 46 | 86,59 | 8,676 | 1,279 |
| 7 | 199 | 67,72 | 6,237 | ,442 |
| Итого | 794 | 74,45 | 8,445 | ,300 |
| Пульс покій | 1 | 113 | 84,44 | 6,610 | ,622 |
| 2 | 130 | 88,80 | 8,443 | ,740 |
| 3 | 106 | 98,94 | 11,080 | 1,076 |
| 4 | 156 | 69,69 | 9,037 | ,724 |
| 5 | 44 | 71,75 | 9,684 | 1,460 |
| 6 | 46 | 99,46 | 10,899 | 1,607 |
| 7 | 199 | 72,82 | 8,493 | ,602 |
| Итого | 794 | 81,45 | 14,171 | ,503 |

Після цього середнє значення та стандартне відхилення додаються до окремої компактної таблиці.

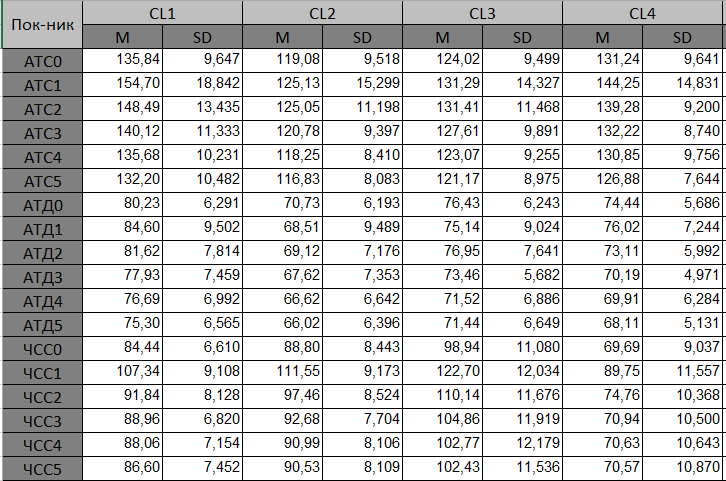


Рисунок 2.4 – Згрупована таблиця результатів

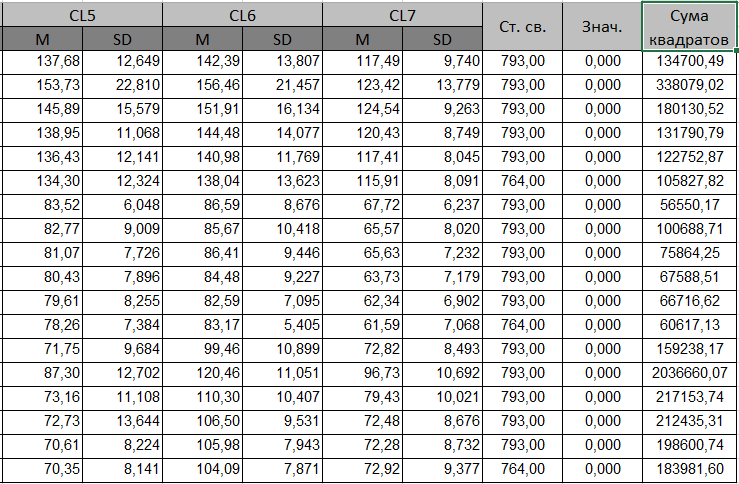


Рисунок 2.5 – Згрупована таблиця результатів

Побудовані таблиці є результуючими та поставляються разом з програмним продуктом.

Для наступних досліджень нами було використано базу студентів та викладачів НТУУ «КПІ ім. Ігоря Сікорського», що пройшли пробу Мартіне декілька разів. Вона містить 599 досліджень, серед яких 323 чоловічої та 276 жіночої. База даних є аналогічною до тієї, за даними якої вже побудовано результуючу таблицю, тому аналогічно її було зведено до прийнятного вигляду для роботи в статистичному пакеті SPSS.

На відміну від попередньої бази, наявна не містить значень групи ризику в повному обсязі, тому ми не можемо охарактеризувати всіх студентів. Також невідомо чи відповідає колонка з кластерами дійсності, оскільки на цей раз не наявні розшифрування та методи отримання кластерів.

Але можливе застосування програмного продукту для визначення групи ризику і заодно є можливість переконатись в справній роботі алгоритму квадрата евклідової відстані. Задача перевірки ефективності алгоритму випливає з того, що попередні дослідження показали високий відсоток збіжності (80%) між алгоритмом кластеризації методом k-середніх та реалізованим у програмі алгоритмом знаходження мінімальної відстані.

Висновки до розділу 2. У даному розділі нами було розглянуто опис бази даних для попередніх досліджень, на базі якої побудовано результуючу таблицю. Наведено характеристику даної таблиці, метод її побудови та використання. Також було описано вхідну таблицю для майбутніх досліджень та доведено необхідність проведення тестів для підтвердження ефективності алгоритму евклідової відстані.

# РОЗДІЛ 3 ПРОВЕДЕННЯ АНАЛІЗІВ ДЛЯ РОЗРОБКИ КОМП’ЮТЕРНОЇ СИСТЕМИ ДЛЯ ВИЗНАЧЕННЯ РЕАКЦІЙ НА ТЕСТОВЕ НАВАНТАЖЕННЯ

Для перевірки коректності алгоритму квадрата евклідової відстані необхідно розщепити базу даних по статі та провести кластеризацію для кожної половини окремо. Проведення кластеризації для кожного пацієнта окремо передбачає вдосконалення програмного продукту та розробку нового модулю кластеризації, який би міг класифікувати одразу всіх пацієнтів в базі даних.

Наступним кроком для дослідження ефективності алгоритму є проведення бінарної логістичної регресії та дискримінантного аналізу. Якщо результат даних тестів буде складати 80% і більше, тоді можна буде стверджувати, що алгоритм квадрату евклідової відстані працює стабільно та ефективно. Це дасть змогу провести наступні дослідження, які будуть слугувати для порівняння наявних кластерів між собою. Також доцільно буде дослідити і графіки середніх значень, отримані на базі результуючих таблиць з попередніх досліджень. У разі збіжності результату алгоритму порівняння кластерів на новій базі даних з графіками, отриманими на навчальній (попередній) базі даних можна буде приступити до побудови нових результуючих таблиць, що в майбутньому будуть використані для дослідження.

Таким чином можемо встановити порядок дій для досягнення поставленої мети:

1. Приведення бази даних до прийнятного для SPSS виду;
2. Розщеплення бази даних по статі;
3. Проведення глобальної кластеризації;
4. Проведення логістичної регресії;
5. Проведення дискримінантного аналізу;
6. Дослідження графіків, побудованих на базі результуючих таблиць;
7. Розробка модулю для дослідження подібності кластерів (реалізація побудови ліній тренду);
8. Побудова нових результуючих таблиць;
9. Розробка модулю універсальної кластеризації;
10. Проведення дисперсійного аналізу;
11. Вдосконалення автоматизації програмного продукту «Clusterbox».

У ході проведення досліджень вихідну базу даних було приведено до вигляду, наведеному на рисунку:

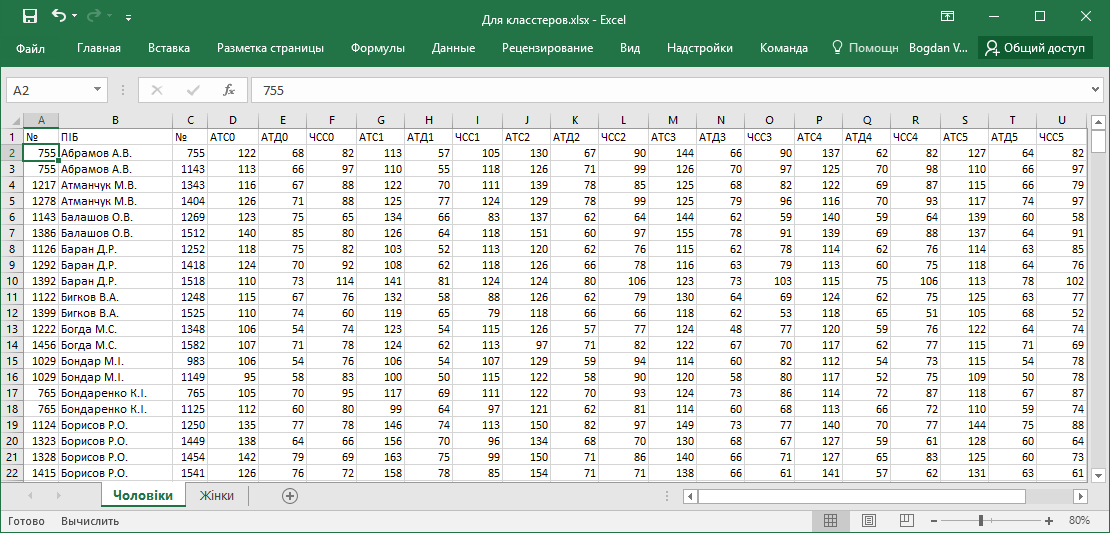


Рисунок 3.1 – База даних для дослідження

На рисунку ми можемо побачити, що для дослідження були обрані дані артеріального систолічного тиску, артеріального діастолічного тиску та значення частоти серцевих скорочень. Вони розташовані у порядку зростання: від стану спокою до п’ятої хвилини включно після навантаження. Також ми можемо побачити, що база даних вже розщеплена по статі, тому можемо перейти до визначення групи ризику для чоловіків (наступні дослідження будуть проведені для чоловічої частини бази даних).

Для визначення групи ризику скористаємося програмним продуктом «Clusterbox», попередньо модифікувавши його, додавши функцію глобального режиму, здатну розставити кластери та мітки (субгрупи ризику) не для одного студента, а для всіх одразу, що присутні в базі даних. На даному етапі дослідження дані зберігаються до текстових файлів. Після цього вони переносяться до таблиці з дослідженнями. Таким чином наша база набуває наступного вигляду:

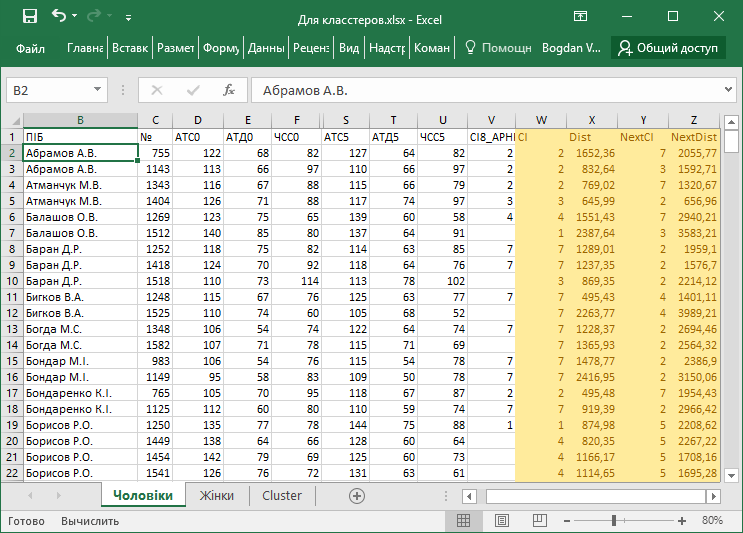


Рисунок 3.2 – База даних для дослідження

Вона містить чотири додаткові поля: Cl – характеризує кластер (група ризику), до якого відноситься студент, Dist – мінімальна відстань до цього кластеру, NextCl – субоптимальний кластер (група ризику, що йде наступною після визначеної), NextDist – відстань до субоптимального кластеру. Маючи всі необхідні дані для перевірки ефективності алгоритму квадрату евклідової відстані можемо провести наступні тести: бінарна логістична регресія та дискримінантний аналіз.

Оскільки класифікатор студентів (група ризику) є не бінарною змінною, було прийнято рішення розбити пацієнтів на групи методом «один проти всіх» та вирівняти дані в групах, де були отримані занадто асиметричні показники класифікації. Результати дослідження представлені для третього кластеру, але аналогічна процедура проводилася для кожного кластеру окремо. Результуючу таблицю для всіх кластерів зображено у таблиці 1.

На рисунку 4 зображено встановлення параметрів, які вводилися для отримання класифікації даних.

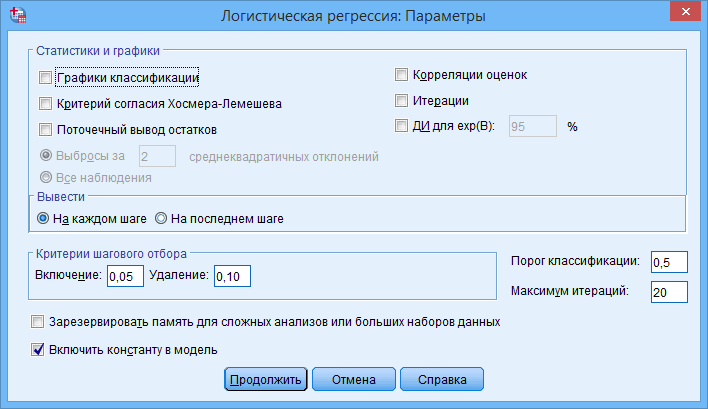


Рисунок 3.3 – Зображення параметрів

**Розбиття на класи «один проти всіх»**

На рисунку 5 наведені результати класифікації при об’єднанні у групи: перша група – студенти, що знаходяться в 3 кластері, друга – студенти в інших кластерах (3 проти 1, 2, 4, 5, 6, 7).

Для побудови логістичної регресії для великої кількості предикторів використано метод умовного включення.

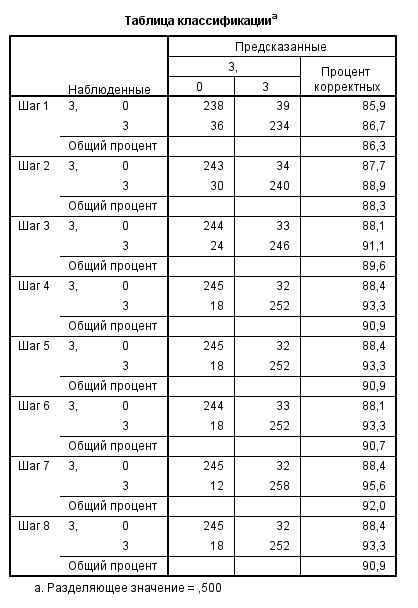


Рисунок 3.4 – Результати класифікації ЛР

За результатами дослідження ми бачимо, що загальний відсоток коректно спрогнозованих даних складає 90,9%. При цьому з таблиці можна зробити висновок про те, що із загального числа студентів, які знаходяться в 3 кластері, рівного 270, тестом вірно були визнані 252. Інші 18 є хибно негативними. Таким чином, відсоток коректності склав 88,4%. Із загальної кількості спостережень (277), що відносяться до інших кластерів, коректно були визнані тестом 245. В цьому випадку відсоток класифікації склав 88,4%.

Для побудови рівняння регресії було використано наступну таблицю:

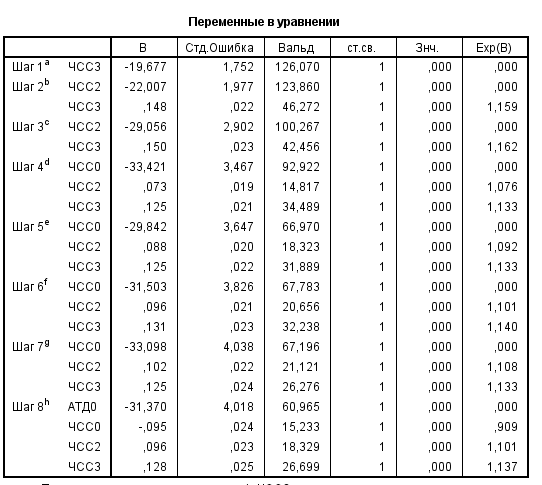


Рисунок 3.5 – Змінні для рівняння регресії

Таким чином, рівняння регресії набуває вигляду:

Для поліпшення якості класифікаторів було вирішено розширити матрицю змінних за допомогою нелінійних перетворень. Тому необхідним кроком було встановлення в параметрах аналізу покрокового режиму для відбору змінних

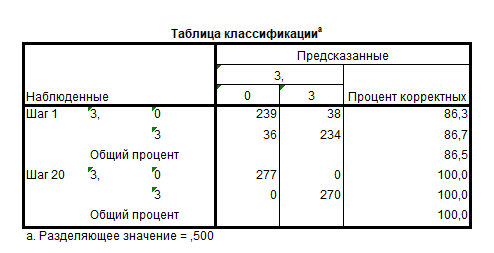


Рисунок 3.6 – Результати класифікації ЛР з нелінійними перетвореннями

За результатами дослідження ми бачимо, що відсоток коректно спрогнозованих даних складає 100,0%. Але слід зазначити, що це при умові неповної моделі, оскільки при обробці даних SPSS видав наступне попередження:

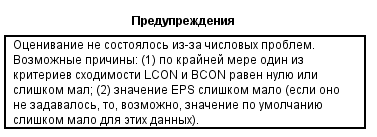


Рисунок 3.7 – Попередження

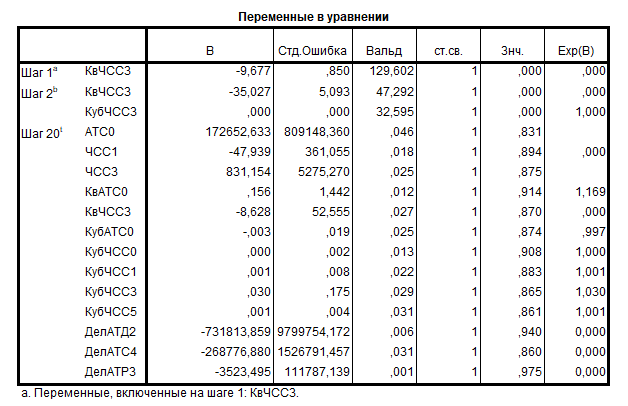


Рисунок 3.8 – Змінні для рівняння регресії з нелінійними перетвореннями

Таким чином, рівняння регресії має вигляд:

Порівнюючи дві моделі ми можемо дійти висновку, що модель, яка побудована на базі даних з додатковими змінними нелінійних перетворень є більш складною, хоча й не повною, але водночас дає вищий результат в порівнянні з моделлю, що включають лише істинні змінні.

Таблиця 3.1 – Порівняльна характеристика

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Класифікатор** | **Позначення** | **ЛР на істинних даних** | **ЛР з нелінійними перетвореннями** | **Результат** | |
| 1-Всіх | Заг. Кор., % | 90,4 | 100 | 2 |  |
| Ск-ть | 3/21 | 13/84 |  |  |
| Ск,% | 14,29 | 15,48 |  | 1 |
| 2-Всіх | Заг. Кор., % | 77,4 | 91 | 2 |  |
| Ск-ть | 4/21 | 7/84 |  |  |
| Ск,% | 19,05 | 8,33 |  | 2 |
| 3-Всіх | Заг. Кор., % | 90,9 | 100\* | 2 |  |
| Ск-ть | 4/21 | 13/84 |  |  |
| Ск,% | 19,05 | 15,48 |  | 2 |
| 4-Всіх | Заг. Кор., % | 97,5 | 95,8 | 1 |  |
| Ск-ть | 5/21 | 13/84 |  |  |
| Ск,% | 23,81 | 15,48 |  | 2 |
| 5-Всіх | Заг. Кор., % | 99,7\* | 100 | 1 |  |
| Ск-ть | 3/21 | 6/84 |  |  |
| Ск,% | 14,29 | 7,14 |  | 2 |
| 6-Всіх | Заг. Кор., % | 100\* | 100 | 1 |  |
| Ск-ть | 4/21 | 5/84 |  |  |
| Ск,% | 19,05 | 5,95 |  | 2 |
| 7-Всіх | Заг. Кор., % | 88 | 100 | 2 |  |
| Ск-ть | 4/21 | 15/84 |  |  |
| Ск,% | 19,05 | 17,86 |  | 2 |

Розшифрування міток в таблиці 3.1:

1. Заг.Кор,% - загальний відсоток коректно-спрогнозованих даних
2. Ск-ть – складність моделі
3. Ск,% - відсоток складності моделі (чим нижче, тим краще)
4. 21 – кількість істинних змінних
5. 84 – кількість істинних змінних разом з нелійнійними перетвореннями
6. 1- виграш ЛР на істинних даних
7. 2 – виграш ЛР з нелінійними перетвореннями
8. \* - неповність моделі

З таблиці 1 видно, що аналіз, проведений на базі нелінійних перетворень дає складніші рівняння моделі, але їх складність, беручи до уваги всі змінні, в переважній кількості менша. Слід також зауважити, що у 4 тестах із 7 відсоток коректно спрогнозованих даних збільшився.

Беручи до уваги таблицю 3.1 ми можемо сказати, що проведення логістичної регресії, включаючи нелінійні перетворення, а саме: операція взяття квадрату та кубу, і операція взяття оберненої змінної, будує складніші рівняння регресії, але самі моделі є простішими за складністю і дають більш високий відсоток коректності.

Наступним кроком дослідження є проведення дискримінантного аналізу, оскільки метод бінарної регресії дає похибку при дослідженні 3 кластеру на даних з нелінійними перетвореннями та при дослідженні 5,6 кластерів на істинних даних. Аналогічно до розділу 3 результати дослідження представлені для третього кластеру, але дана процедура проводилася для кожного кластеру окремо.

Всі дані було класифіковано методом дискримінантного аналізу, використовуючи покроковий відбір. На рис. 3.9 показаний вибір відстані та критерій розпізнавання.

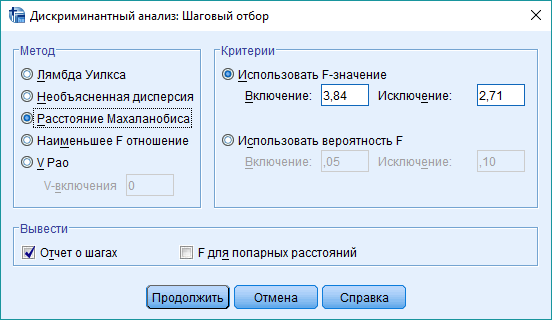


Рисунок 3.9 – Вибір методів

У вікні вибору класифікації, що зображено на рис 4. показаний вибір розрахунку класифікації.

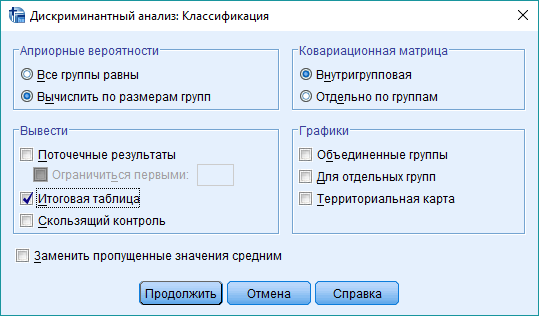


Рисунок 3.10 – Вибір класифікації

**Результати аналізу для груп «3 проти 1,2,4,5,6,7»**

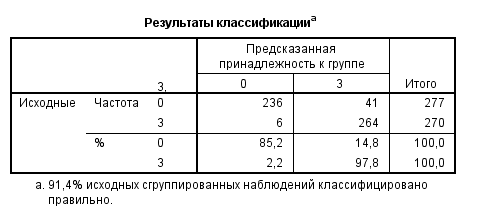


Рисунок 3.11 – Результати класифікації

За результатами дискримінантного аналізу ми бачимо, що 91,4% вихідних згрупованих спостережень класифіковано правильно.

Для побудови дискримінантної функції застосовуються дані з наступної таблиці:



Рисунок 3.12 – Коефіцієнти для дискримінантної функції

Таким чином, дискримінантна функція набуває вигляду:

Для підвищення якості моделі, як і для логістичної регресії до аналізу штучно були введені нелінійні змінні , та для кожної змінної.

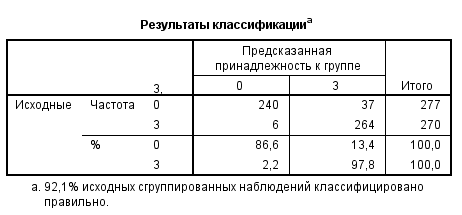


Рисунок 3.13 – Результати класифікації з нелінійними перетвореннями

З рисунку 3.13 ми бачимо, що відсоток правильно класифікованих спостережень складає 92,1%



Рисунок 3.14 – Коефіцієнти для дискримінантної функції з нелінійними перетвореннями [6,7].

Таким чином, дискримінантна функція має вигляд:

Результати дослідження показують, що модель, побудована з додатковими змінними дає більш високий результат класифікації даних. Слід зазначити, що приріст є незначним і складає 0,7%, а модель при цьому стала складнішою в 2,25 рази.

Аналогічно до пунку регресійного аналізу було побудовано таблицю порівняльної характеристики для дискримінантного аналізу щоб оцінити результати дослідження.

Таблиця 3.2 – Порівняльна характеристика

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Класифікатор** | **Позначення** | **ДА на істинних даних** | **ДА з нелінійними перетвореннями** | **Результат** | |
| 1-Всіх | Заг. Кор., % | 87,2 | 94 | 2 |  |
| Ск-ть | 3/21 | 9/84 |  |  |
| Ск,% | 14,29 | 10,71 |  | 2 |
| 2-Всіх | Заг. Кор., % | 77,2 | 84,6 | 2 |  |
| Ск-ть | 4/21 | 4/84 |  |  |
| Ск,% | 19,05 | 4,76 |  | 2 |
| 3-Всіх | Заг. Кор., % | 91,4 | 92,1 | 2 |  |
| Ск-ть | 4/21 | 9/84 |  |  |
| Ск,% | 19,05 | 10,71 |  | 2 |
| 4-Всіх | Заг. Кор., % | 85,4 | 88,9 | 2 |  |
| Ск-ть | 4/21 | 7/84 |  |  |
| Ск,% | 19,05 | 8,33 |  | 2 |
| 5-Всіх | Заг. Кор., % | 94,3 | 97,6 | 2 |  |
| Ск-ть | 5/21 | 12/84 |  |  |
| Ск,% | 23,81 | 14,29 |  | 2 |
| 6-Всіх | Заг. Кор., % | 93,9 | 97,1 | 2 |  |
| Ск-ть | 4/21 | 12/84 |  |  |
| Ск,% | 19,05 | 14,29 |  | 2 |
| 7-Всіх | Заг. Кор., % | 88 | 88,5 | 2 |  |
| Ск-ть | 3/21 | 3/84 |  |  |
| Ск,% | 14,29 | 3,57 |  | 2 |

З таблиці 3.2 видно, що аналіз, проведений на базі нелінійних перетворень дає складніші рівняння моделі, але їх складність, беручи до уваги всі змінні, менша. Слід також зауважити, що у 7 тестах із 7 відсоток коректно спрогнозованих даних збільшився. За допомогою дискримінантного аналізу було побудовано функції прогнозування для кожного кластера, при чому результати дослідження показують, що введення нелінійних змінних (для кожної вхідної змінної) до аналізу покращую результат класифікації.

Виходячи з результатів досліджень, ми можемо сказати, що при порівнянні алгоритмів, у 2 випадках із 7 (виключаючи неповну модель при логістичній регресії) найбільш точну і просту модель будує дискримінантний аналіз, ще в 2 із 7 випадках – метод логістичної регресії. Ще у одному випадку логістична регресія та дискримінантний аналіз дають ідентичні результати. При цьому дискримінантний аналіз є більш простим та універсальним у використанні, оскільки при його застосуванні ми завжди маємо справу тільки з однією статистичною процедурою, в якій беруть участь одна категоріальна залежна змінна і кілька незалежних змінних з будь-яким типом шкали. Також з досліджень видно, що логістична регресія не завжди може побудувати повну модель. Тому ми ввели нелінійні змінні для методів логістичної регресії та дискримінантного аналізу.

Дослідження результатів проведення логістичної регресії та дискримінантного аналізу показують, що в середньому відсоток класифікації становить близько 92, але це з урахуванням аналізу на істинних даних. Таким чином, високий відстоток класифікації методом логістичної регресії показує, що наш алгоритм розставляє кластери досить ефективно, оскільки колонка з еталонними значеннями легко сприймається і розшифровується аналізом логістичної регресії в SPSS. Відсоток класифікації дискримінантним аналізом на істинних даних складає в середньому 88.2, що також є досить високим показником.

Отже, можемо з впевненістю сказати, що алгоритм знаходження мінімальної відстані з подальшим визначенням групи ризику, робота якого зображена на риснуку 3.15 ,працює коректно та ефективно.

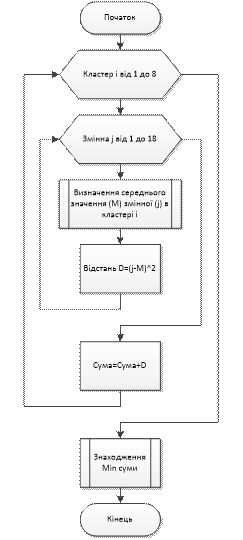


Рисунок 3.15 – Блок-схема алгоритму знаходження мінімальної відстані до кластеру [32].

Тому ми можемо перейти до дослідження графіків, отриманих на базі результуючих таблиць, що побудовані на навчальній базі даних (база студентів молодших курсів НТУУ «КПІ ім. Ігоря Сікорського», що містить 1495спостережень). Таким чином середні значення АТС і АТД (рисунок 2.4 та рисунок 2.5) дають змогу побудувати відповідні графіки по кластерам.

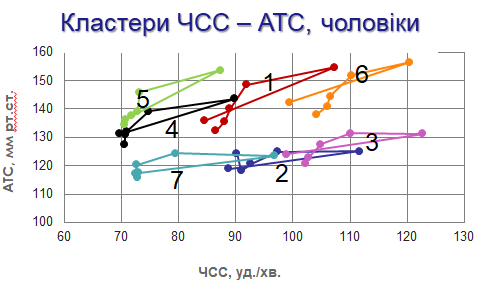


Рисунок 3.16 – Графіки ЧСС-АТС по кластерам

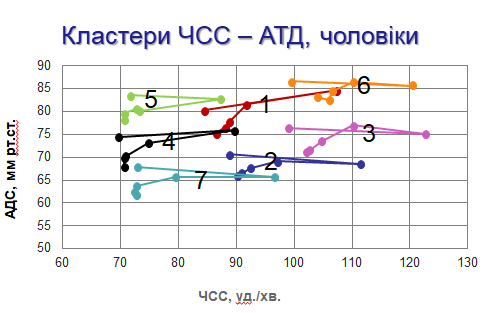


Рисунок 3.17 – Графік ЧСС-АТД по кластерам

Графіки зображають динаміку систолічного та діастолічного тисків та зміни пульсу підчас проведення проби Мартіне. Виходячи з побудованих графіків можна сказати, що деякі кластери перетинаються, але, при цьому, їх властивості різні.

За даними попередніх досліджень були сформовані таблиці з характеристиками та рекомендаціями для кожного кластеру.

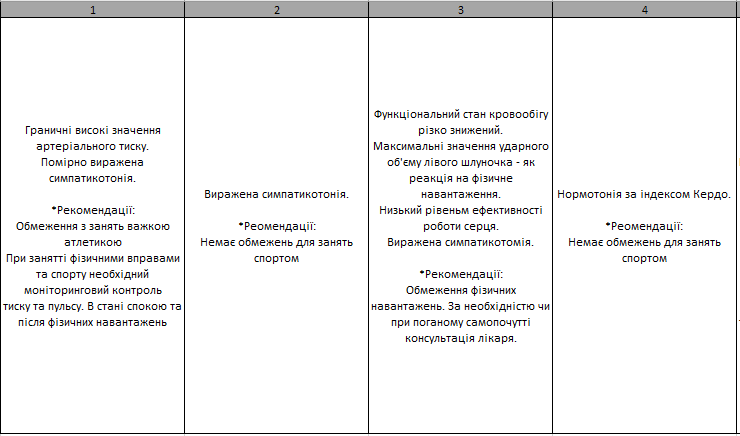


Рисунок 3.18 – Опис кластерів 1-4 та рекомендації до них

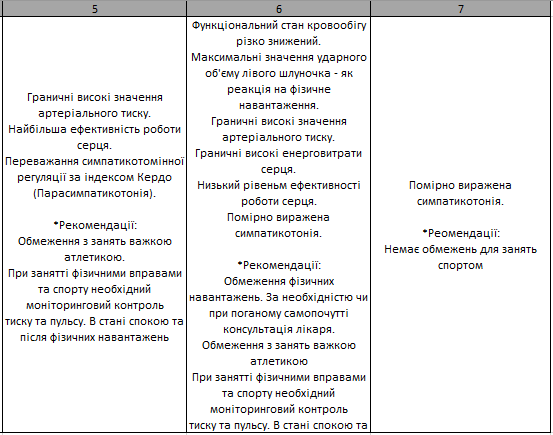


Рисунок 3.19 – Опис кластерів 5-7 та рекомендації до них

# Як ми бачимо, за показником АТС кластери 2 і 3, 7 і 2 перетинаються, а за показником АТД перетинаються кластери під номерами 1 і 4, 2 і 7, 1 і 6. Таким чином висуваємо гіпотезу про те, що вони мають спільну підгрупу. Також ми можемо побачити, що за АТС кластери 4 і 5 знаходяться приблизно в одному діапазоні як за значеннями тиску, так і за значеннями пульсу.

# Оскільки ефективність алгоритму була підтверджена, а деякі кластери знаходяться поруч, то можемо розробити модуль програми, який буде порівнювати кластери між собою на подібність. Для цього ми використовуємо метод побудови ліній трендів.

Висновки до розділу 4. В даному розділі було проведено глобальну кластеризацію за допомогою програмного продукту «Clusterbox». Результат кластеризації вхідної бази даних використано для методів логістичної регресії та дискримінантного аналізу. Проведені тести даними методами підтверджують ефективність алгоритму квадрата евклідової відстані, що дає шлях до дослідження графіків результуючих таблиць. Дослідження показує, що кластери 4 та 5 в графічному вигляді схожі між собою, тому ми можемо приступити до розробки та реалізації програмного додатку для оцінки подібності кластерів.

# РОЗДІЛ 4 ПРОГРАМНИЙ ПРОДУКТ ДЛЯ ВИЗНАЧЕННЯ ФУНКЦІОНАЛЬНОГО СТАНУ СИСТЕМИ КРОВООБІГУ

## 4.1 Проектування програмного продукту

Контекстна діаграма

Дана діаграма показує процес використання додатку для побудови ліній регресій, що дає змогу оцінити подібність вибраних кластерів. Якщо обрані кластери є подібними, тоді можна буде зменшити початкову кількість кластерів для підвищення ефективності оцінки стану системи кровообігу.

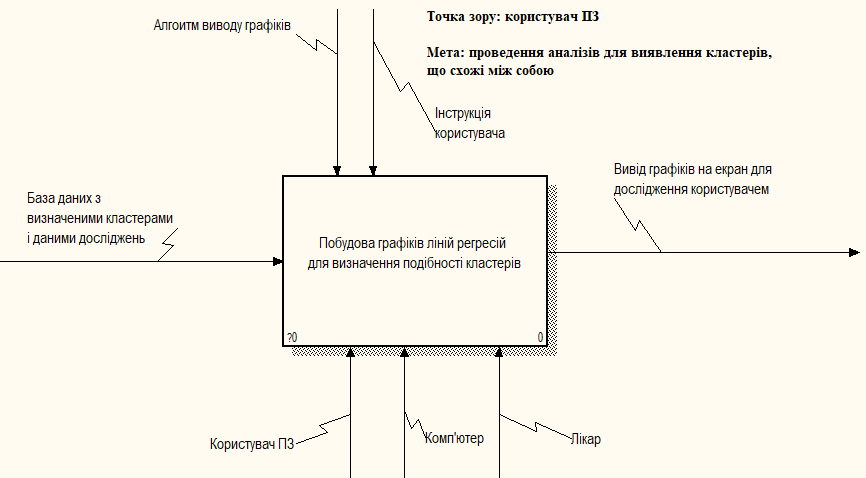


Рисунок 4.1 – Контекстна діаграма [33].

Вхідними даними для роботи програмного додатку є розщеплена по статі база даних зі значеннями артеріального тиску та пульсу у стані спокої та на 1-5 хвилинах після проведення тесту на навантаження, значення різниці систолічного та діастоличного тиску на 3-5 хвилинах після навантаження та визначені кластер, субкластер (мітка), мінімальна та субмінімальна відстані. Вихідними даними є вивід ліній трендів (регресій) за обраними кластерами на екран користувача.

Діаграма декомпозиції першого рівня

Діаграма декомпозиції першого рівня для додатку, розробленого для побудови ліній регресій, показує нам процес роботи програмного продукту на окремих етапах застосування.

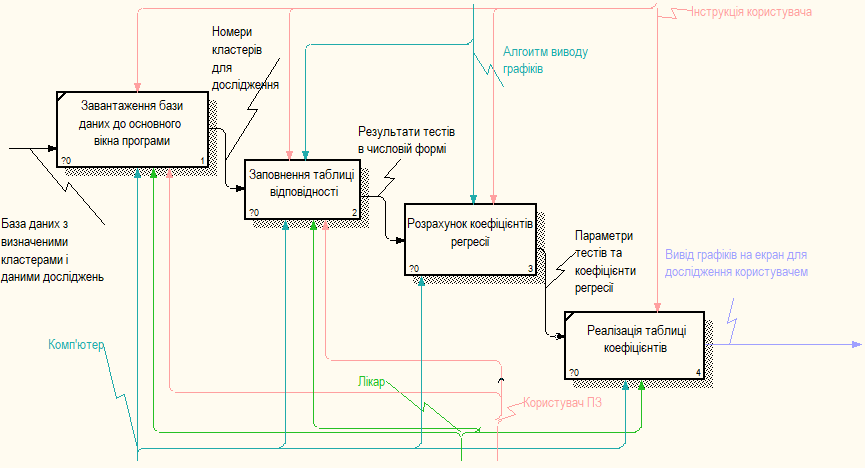


Рисунок 4.2 – Діаграма декомпозиції першого рівня [34].

З діаграми видно, що весь процес роботи програми складаєтсья з чотирьох етапів: завантаження бази даних до основного вікна програми, заповнення таблиці відповідності, розрахунок коефіцієнтів регресії, реалізація таблиці коефіцієнтів. Всі ці етапи необхідні для побудови та виводу графіків на екран

Діаграма декомпозиції другого рівня

Для більш детального опису процесу розрахунку коефіцієнтів регресії було побудовано діаграму декомпозиції другого рівня.

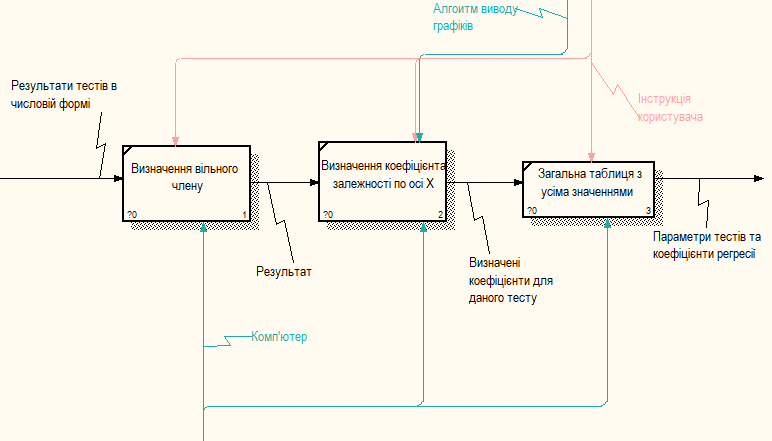


Рисунок 4.3 – Діаграма декомпозиції другого рівня [33, 34].

Даний процес включає в себе три блоки: визначення вільного члену, визначення коефіцієнта залежності по осі Х та загальну таблицю з усіма значеннями, яка має бути заповнена . Вхідними даними для розрахункового блоку є оезультати тестів в числовій формі, а вихідними - параметри тестів та коефіцієнти регресії.

Діаграма дерева вузлів

Процеси та підпроцеси, що були розглянуті вище зручно скомпонувані в діаграму дерева вузлів. Вона дозволяє оцінити послідовність основних процесів в цілому.

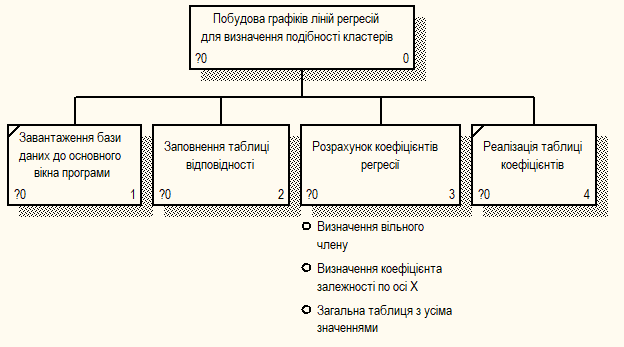


Рисунок 4.4 – Діаграма дерева вузлів [35].

Use Case діаграма

Дана діаграма спрямована на демонстрацію процесів та їх виконання в ході алгоритму виходячи з точки зору самих учасників, що безпосередньо беруть учать у використанні програмного продукту. Учасники цієї системи: лікар-адміністратор та користувач, які на діаграмі позначені фігурою людини. Можливі дії представлені еліпсами та зображають доступні комбінації розвитку сценарію використання і роботи програмного продукту. Одні варіанти використання можуть бути частиною інших варіантів, розширювати їх або бути більш узагальненою версією.

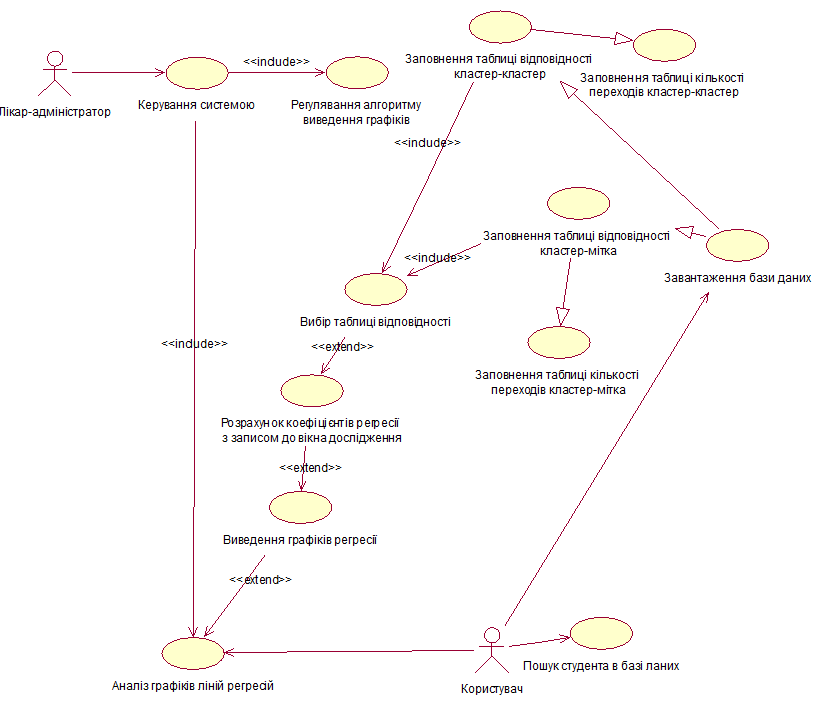


Рисунок 4.5 – Діаграма Use Case [36].

Схема демонстує нам, що лікар займається тільки адмініструванням системи і аналізом результатів програми, тоді як користувач сам вносить свої результати в програмний продукт.

Діаграма послідовності

Діаграма послідовності дає змогу спостерігати за виконанням актором визначеного варіанту використання, враховуючи їх послідовність в часі, а також показує взаємодію логічних елементів між собою.



Рисунок 4.6 – Діаграма послідовності [37]

Діаграма відображає послідовність в якій використовує систему користувач. Починається робота з підготовки таблиці для дослідження. Наступним кроком є завантаження її до програми, де дані з таблиці будуть перевірені на коректність. Далі у відповідних комірках вибираються кластери для дослідження, після чого автоматично заповнюється таблиця відповідності кластер-кластер. Заповнена таблиця слугує базою для побудови графіків. На кожному етапі виведення графіків розраховуються відповідні коефіцієнти регресії. Дані коефіцієнти реєструються у відповідній таблиці. Після побудови всіх графіків (АТС-ЧСС, АТД-ЧСС, АТР-ЧСС) даний результат необхідно проаналізувати, чим на останньому кроці і займається користувач. Також на діаграмі ми можемо побачити, що функціонал додатку передбачає пошук студента в завантаженій базі даних після заповнення таблиці відповідності. Пошук студента дає можливість подивитись чи потрапляють значення параметрів тесту студента у радіус кластера, до якого відноситься студент. Якщо значення виходять за радіус, тоді на екран користувача виводяться графіки регресій для подальшого дослідження[37].

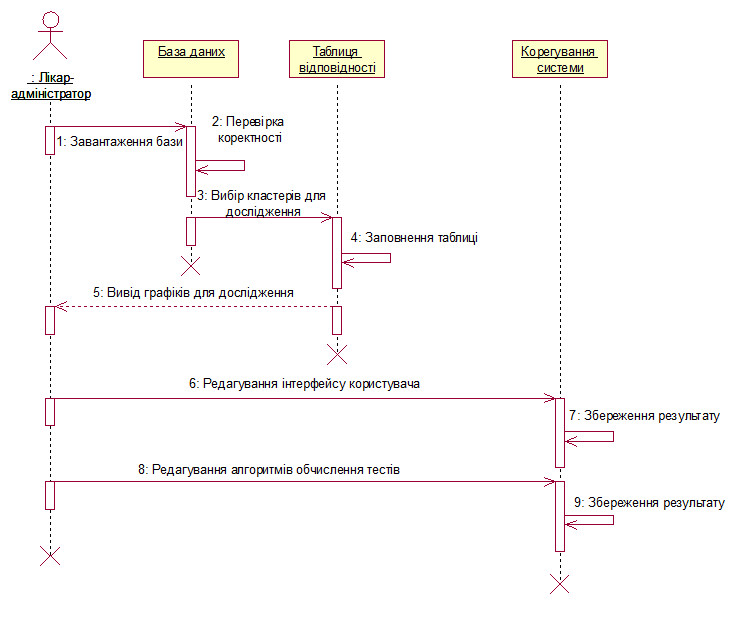


Рисунок 4.7 – Діаграма послідовності роботи зі сторони лікаря-адміністратора

Діаграма послідовності роботи зі сторони лікаря-адміністратора аналогічна за роботою до діаграми користувача. Але на відміну від неї лікар-адміністратор має право вносити зміни до інтерфейсу програми та редагувати алгоритми обчислення тестів [37].

Діаграма кооперації

Діаграма кооперації створена для візуалізації взаємодії користувача з основними елементами програмного продукту. Основні етапи позначені прямокутником. Проміжні дії розташовуються між основними блоками. Стрілками вказаний напрямок руху, тобто до чого призведе та чи інша дія.



Рисунок 4.8 – Діаграма кооперації

Діаграма кооперації для користувача програмного продукту дає змогу відслідкувати його взаємодію з елементами інтерфейсу та побачити побудову і виведення графіків на екран.



Рисунок 4.9 – Діаграма кооперації роботи програмного додатку лікарем-адміністратором

Зображена діаграма кооперації роботи лікаря-адміністратора з прогрмним додатком демонструє процес побудови графіків ліній трендів з виведенням результату для аналізу. Також ми бачимо, що лікар може регулювати алгоритми обчислення тим самим вдосконялюючи роботу програмного додатку [38].

Діаграма діяльності

Дана діаграма передбачає всі варіанти можливого викоритання програмного продукту. Тобто кожен блок відображає дію, а стрілки, що їх поєднують – вказуть на послідовність виконання. Таким чином вибудовується декілька варіантів використання або функціонування системи вцілому. Діаграма діяльності (так само, як і діаграми станів та переходів) зображаються у вигляді орієнтовного графу, де вершини – дії, а ребра – переходи між діями.

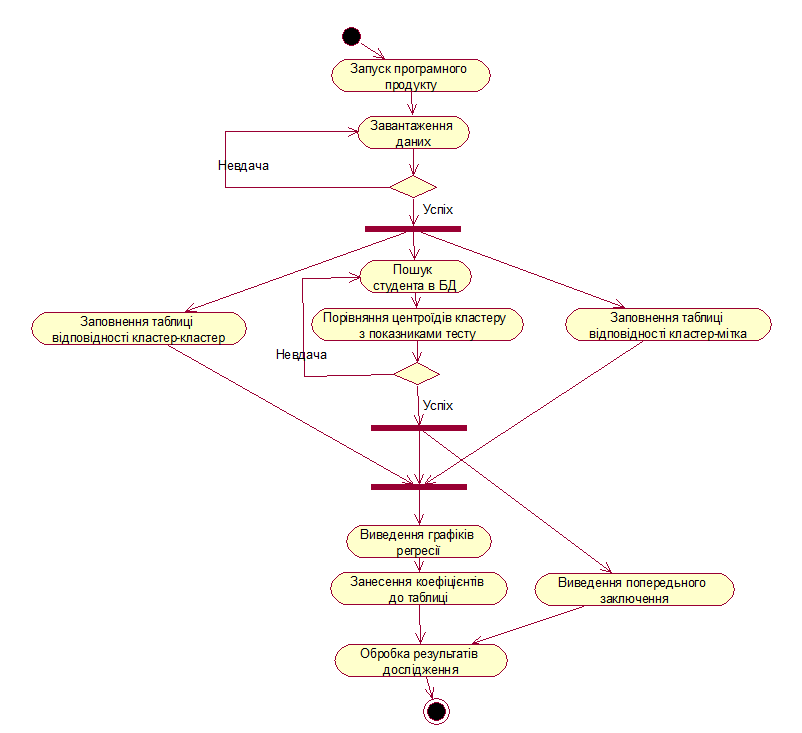


Рисунок 4.10 – Діаграма діяльності [39].

Діаграма станів

Діаграма станів є безпосереднім схематичним зображенням стану системи на тому чи іншому етапі. Таким чином вона демонструє всі можливі послідовності і звязки роботи алгоритму починаючи від початку і до його кінця, при цьому враховані можливі успіх, невдача та пропуски на певних кроках при проходжені алгоритму.

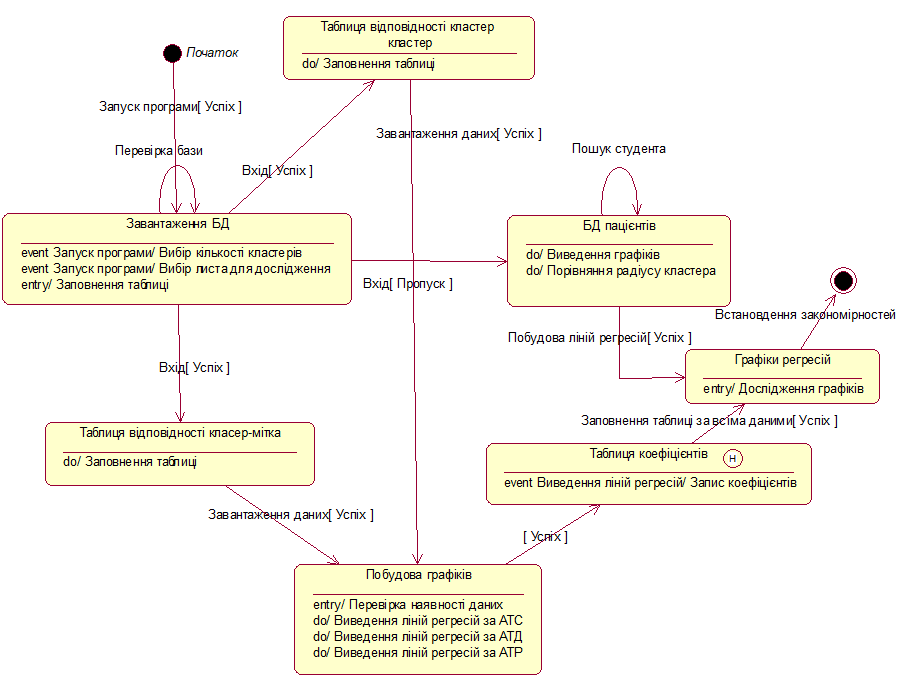


Рисунок 4.11 – Діаграма станів [40].

## 4.2 Розробка програмного продукту

Програмний продукт розроблено в середовищі Microsoft Visual Studio 2017 Community за допомогою мови програмування С# з використанням елементів WinForms та фреймворку .Net Framework. Для демонстрації результатів кластеризації був розроблений багатофункціональний інтерфейс користувача. Призначення даного доданку - це дослідження подібності кластерів, шляхом проведення лінійної регресії, та виводу ліній трендів. У разі, якщо лінії трендів схожі між собою та їх коефіцієнти близькі за значенням, можна зменшити кількість кластерів та побудувати нові результуючі таблиці.

Оскільки, при кластеризації, ми визначаємо субоптимальну групу (мітку), то даний додаток дає змогу дослідити переходи між кластерами, будуючи у відповідному вікні графіки ліній трендів для кластер-мітка та мітка-кластер.

Додаток працює у сукупності з програмним продуктом «Cluserbox» та має на меті покращення роботи програми для дослідження стану кровоносної системи.

Інтерфейс користувача програми є мінімалістичним та інтуїтивно зрозумілим, що полегшує роботу з програмним продуктом.

Також програмним додатком передбачено пошук студента в завантаженій базі даних для того, щоб подивитись чи перевищує радіус студента радіус кластера. Додатково розраховується загальна кількість зміни кластерів в базі даних за одним студентом. Дані дослідження показують динаміку роботи алгоритму та можуть бути використані для моделювання системи виводу характеристик кластеру, до якого відноситься студент, та рекомендацій мітки, якщо радіус студента перевищує радіус кластера.

Вхідними даними для роботи програми є діастолічний та артеріальний тиски, серцевий тиск і частота серцевих скорочень на першій, другій та третій хвилинах після навантаження. Саме ці показники найкраще характеризують поведінку організму при проведенні тесту Мартіне. Також дослідження передбачає включення до вхідних даних значень АТР1, АТР2, АТР3 (різниця артеріального тиску між діастолічним та систолічним тиском на першій, другій та третій хвилинах після навантаження).

Вихідними даними є виведення графіків в залежності від методу дослідження: при порівнянні кластерів – графіки ліній трендів кластер-кластер, при дослідженні зміни кластеру – кластер-мітка (мітка-кластер).

Основні можливості системи, що розроблені:

* + Перевірка коректності завантаженої бази даних;
  + Заповнення таблиць спостереженнями, що відносяться до вибраного кластеру;
  + Побудова графіків ліній тренду кластер-кластер;
  + Можливість вибору таблиці для побудови графіків зміни кластеру (графіки ліній трендів кластер-мітка та мітка-кластер) ;
  + Розрахунок кількості переходів кластер-кластер;
  + Розрахунок кількості переходів кластер-мітка;
  + Пошук студента у завантаженій базі з визначенням радіусу кластеру та вектору направлення студента з подальшим порівнянням та виведенням відповідного графіку за необхідністю.

## 4.3 Робота з програмним додатком

При запуску програмного додатку ми бачимо відповідне вікно для досліджень, що передбачає заповнення відповідних полів.

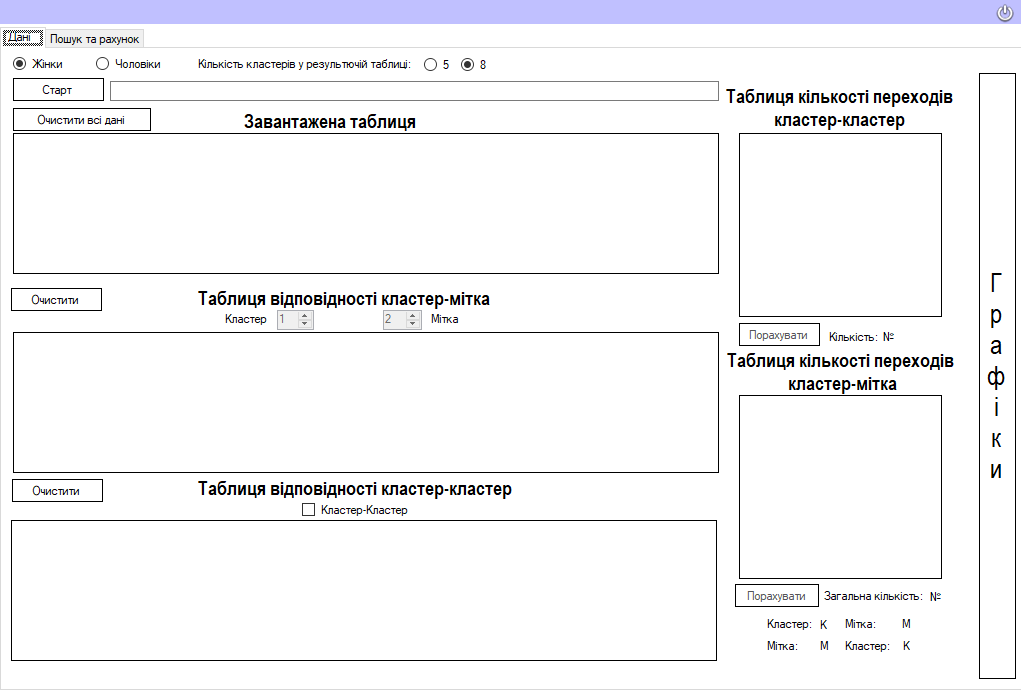


Рисунок 4.12 – Основне вікно програмного додатку

Для цього необхідно вибрати, для якої частини бази даних буде проводитися дослідження, та натиснути кнопку «Старт». Наступним кроком є вибір бази даних, що мітить прізвище студента, значення артеріального тиску та пульсу в стані спокою та на кожній хвилині після навантаження, включно до п’ятої хвилини, а також значення різниці діастолічного та систолічного тиску на першій, другій і третій хвилині після навантаження.

Якщо вибраного листа не буде знайдено у базі даних буде видано наступне попередження:

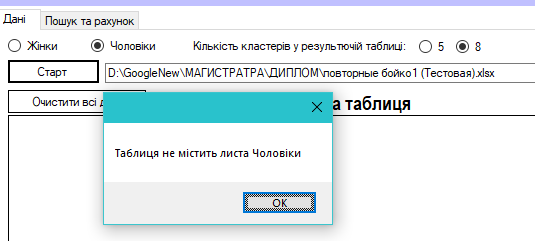


Рисунок 4.13 – Попередження про відсутність листа для дослідження

Для того, щоб завантажити іншу базу для дослідження, необхідно щоб поле з шляхом до бази було пустим. В іншому випадку за цим шляхом буде йти пошук бази даних.

Якщо у таблиці є даний лист, але на ньому відсутнє хоча б одне з основних полів, які мають бути завантажені, програмою буде оброблено виключення з виведенням інформації на екран користувача.

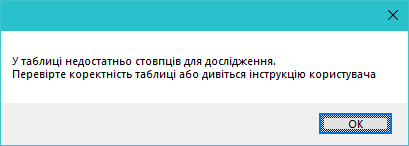


Рисунок 4.14 – Попередження про відсутність стовпців для дослідження

Слід також зазначити, що база даних для завантаження повинна бути у форматі таблиці Misrosoft Excel та бути у розширенні \*.xls або \*.xlsx. Якщо спробувати завантажити базу іншого формату або взагалі інший файл, програмою буде видано попередження:

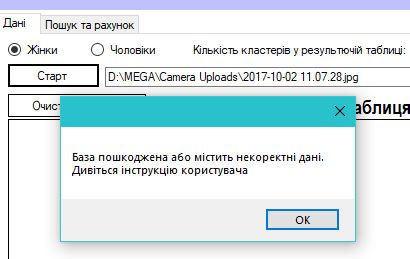


Рисунок 4.15 – Попередження про пошкодження бази даних

При успішному завантаженні бази даних до програми вікно «Завантажена таблиця» буде заповненим і користувачу буде доступний пункт з вибором відповідного кластеру та мітки для дослідження.

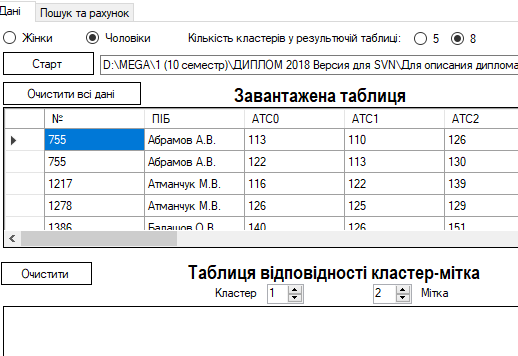


Рисунок 4.16 – Завантажена база даних

Оскільки ми хочемо дослідити подібність кластерів, необхідно встановити галочку навпроти напису «Кластер-Кластер» у таблиці відповідності кластер-кластер. Якщо галочка буде ввімкненою, тоді ми зможемо обрати два кластери, які будуть порівнюватися між собою. При цьому, вікно таблиці «кластер-кластер» буде автоматично заповнюватися. Дані у таблиці будуть показувати всі дані в обраних кластерах. Оскільки на графіках (рисунок 3.16 та рисунок 3.17) кластери 4 та 5 знаходяться поруч, тому для дослідження було обрано саме їх.

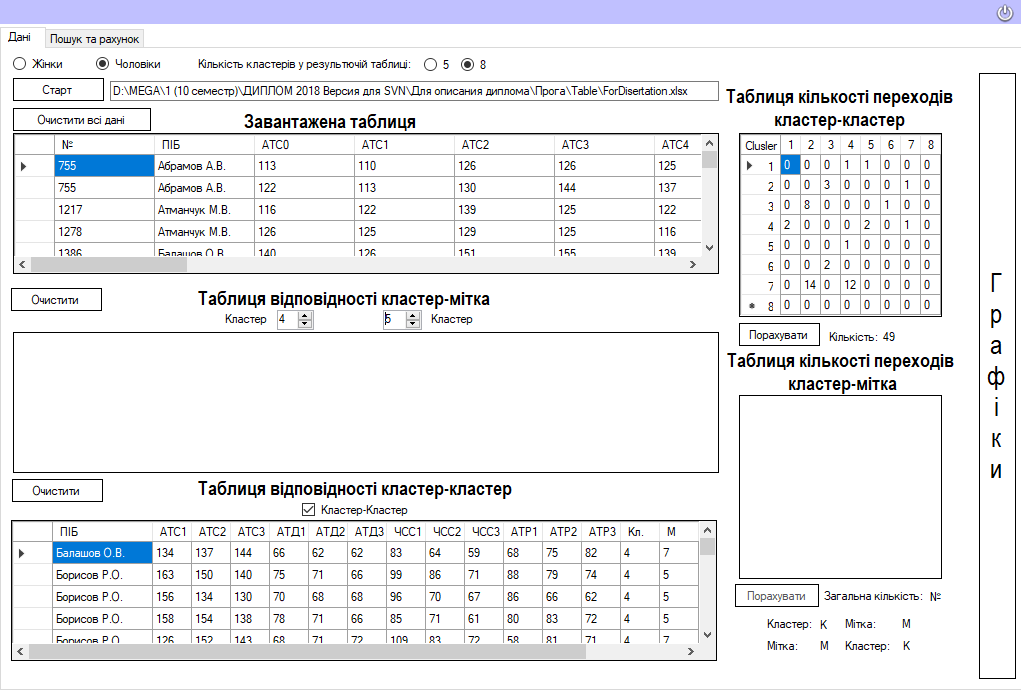


Рисунок 4.17 – Заповнена таблиця відповідності кластер-кластер

Також на рисунку 4.17 ми можемо побачити заповнену таблицю кількості переходів кластер-кластер. Вона показує скільки змін кластерів є у базі даних. Наприклад, студент Іванов проходив дослідження тричі. Перший раз він був у дургому кластері, а мітка показувала на сьомий. При повторному обстеженні – кластер змінився на сьомий. Студентів, які змінили свій кластер з другого на сьомий всього 14. Загальна кількість зміни кластеру у відповідності з визначеною міткою – 49.

Оскільки таблиця відповідності кластер-кластер заповнена, можемо побудувати відповідні графіки. Для цього необхідно натиснути на клавішу «Графіки».

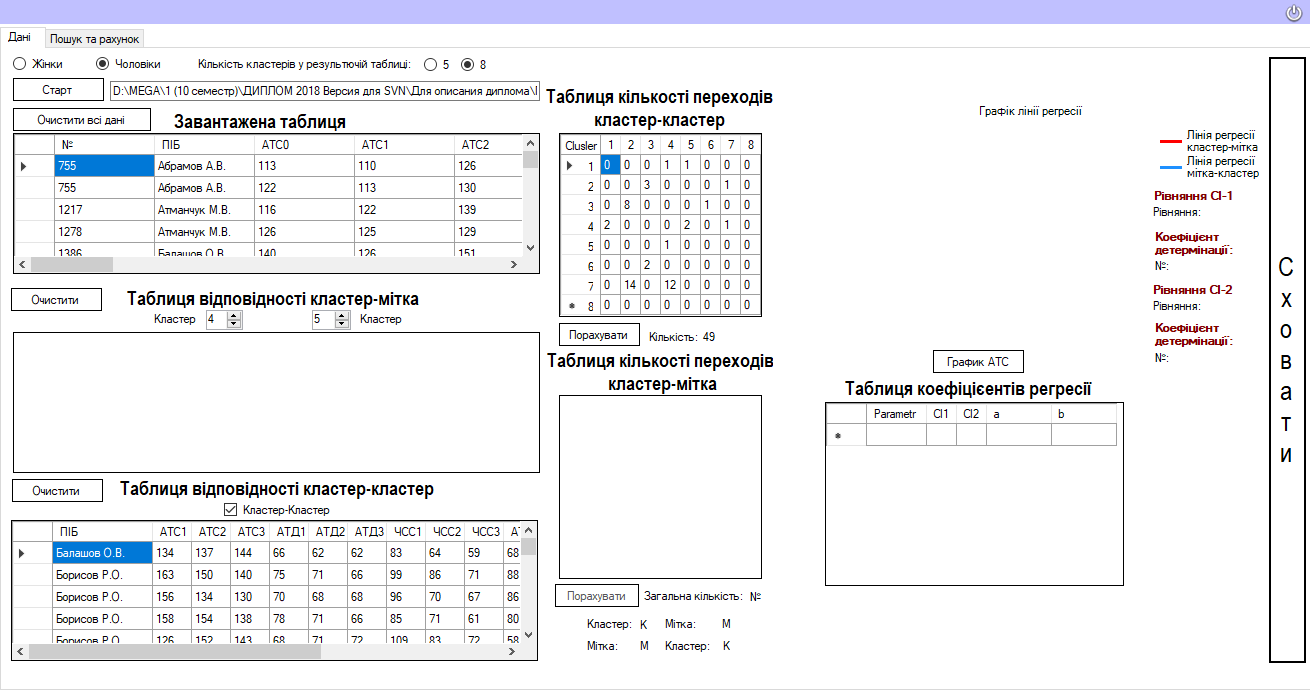


Рисунок 4.18 – Вікно з виведенням графіків

Ми бачимо, що у нас з’явилася додаткова область для виводу графіків. При натисненні на «Графік АТС» побудуються лінії трендів «кластер 4» та «кластер 5», дані яких завантажені до таблиці відповідності кластер-кластер. Слід зазначити, що спочатку на екран виведуться лінії трендів за значеннями АТС, про що свідчить заголовок над графіком, а напис на клавіші зміниться на «Графік АТД».

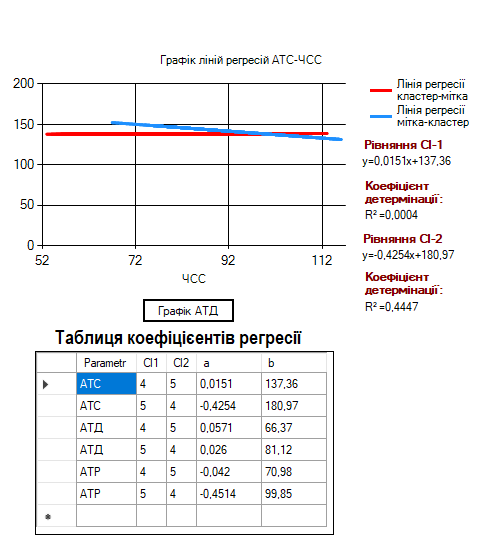


Рисунок 4.19 – Графік ліній регресій АТС-ЧСС із таблицею коефіцієнтів

Також ми можемо побачити, що коефіцієнти ліній регресій записуються до таблиці коефіцієнтів. Таким чином користувач може оцінити не тільки візуальну складову графіків, але й аналітичну.

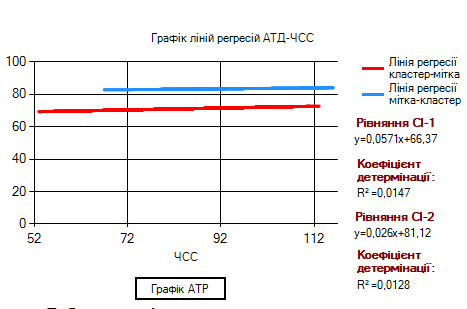


Рисунок 4.20 – Графік ліній регресій АТД-ЧСС

За проведеними дослідженнями за кластерами 4 і 5 ми бачимо, що за АТС лінії регресії досить схожі між собою. Аналогічні висновки ми можемо зробити з графіків, побудованих на значеннях АТД. З нього чітко видно, що лінії трендів ідуть паралельно одна одній , а значення АТД відрізняються приблизно на 8-10 позицій. Аналогічну картину ми можемо побачити і на графіках, отриманих на базі результуючих таблиць (рисунок 3.16 та рисунок 3.17). Таким чином ми можемо сказати, що кластери 4 та 5 мають схожі властивості, що підтверджують наші дослідження, тому кількість кластерів може бути зменшеною.

Також модуль дослідження передбачає пошук студента в базі даних з визначенням розташування показників тесту відносно центроїдів кластеру. Якщо показники тесту більші, ніж радіус кластеру, до якого відноситься студент, то виводиться відповідний графік для порівняння результатів. Дані дослідження показують динаміку роботи алгоритму та можуть бути використані для моделювання системи виводу характеристик кластеру, до якого відноситься студент, та рекомендацій мітки, якщо радіус студента перевищує радіус кластера. Робота алгоритму наведена на рисунку 4.21.

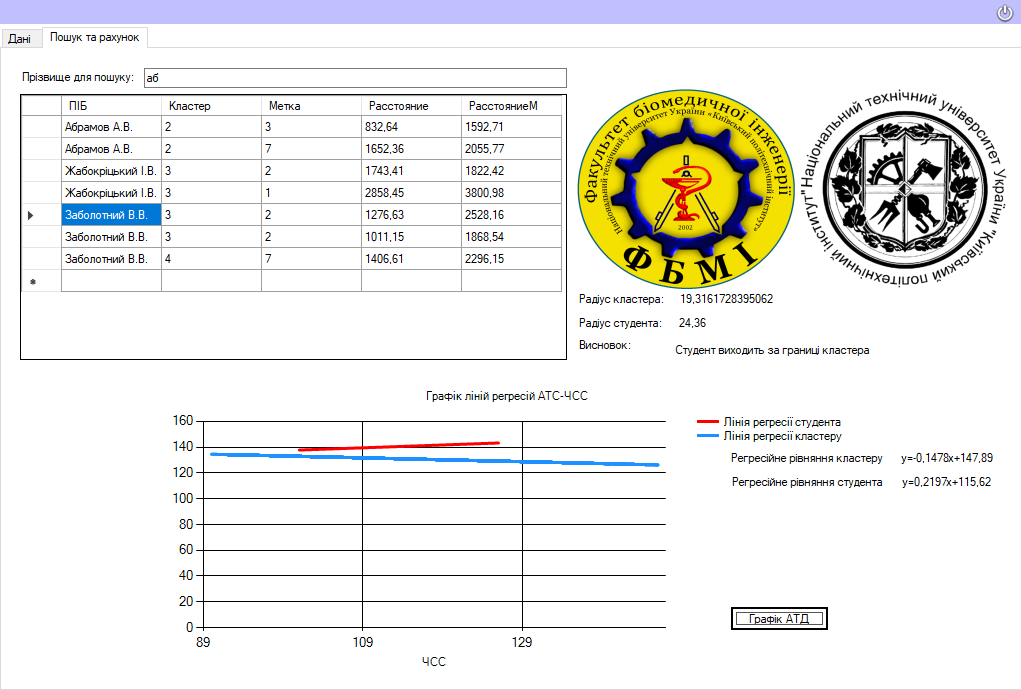


Рисунок 4.21 – Пошук студента в БД з виводом графіків

## 

## Висновок до розділу 4

В цьому розділі проведено проектування програмного продукту яке представлено у діаграмах та схемах декомпозиції, послідовності, станів та кооперації. Така візуалізація надає змогу чітко відобразити алгоритм виконання, його послідовність, структурованість, подати представлення дій користувачів програмного продукту в залежності від їх функціональної ролі, надає повну картину можливостей програмного продукту та сценаріїв його використання.

Далі детально розглянутій інтерфейс програмного продукту, надані приклади його використання та результати роботи. Також були розглянуть вся можливі випадки та умови невиконання програмою певних задач в залежності від конкретних умов роботи з додатком.

**Розділ 5. Аналіз зменшення кількості кластерів з вдосконаленням програмного продукту**

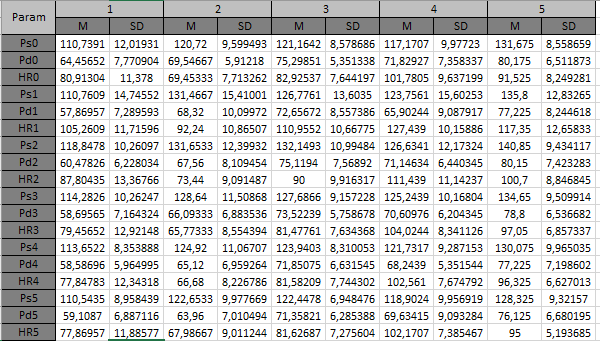
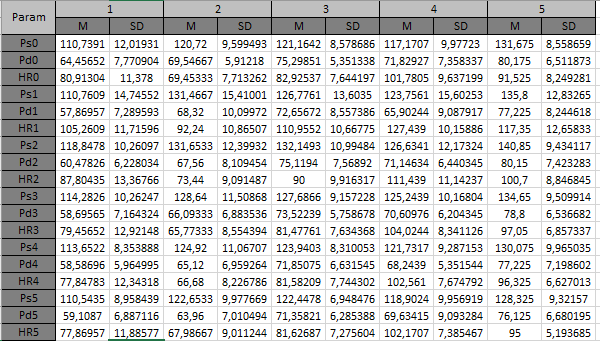
Наші дослідження показали, що зменшення кількості кластерів є необхідним кроком для поліпшення роботи алгоритму класифікації на основі квадрату евклідової відстані.

Таким чином ми повторно провели кластеризацію алгоритмом k-середніх. Спочатку було вирішено зменшити кількість кластерів на дві позиції, тому кількість кластерів становить 5.

Після кластеризації ми занесли результати до окремої таблиці Excel. Після чого завантажили її до SPSS та провели дисперсійний аналіз з виведенням описових статистик. Приклад проведення аналізу наведено для значення систолічного тиску до навантаження, але аналогічні таблиці було отримано і для всіх інших значень тиску та пульсу.

Рисунок 5.1 – Описові статистики

За результатами дисперсійного аналізу було побудовано результуючу таблицю, яка містить показники стандартного відхилення та середніх значень за кожною змінною для кожного кластера.

Рисунок 5.2 - Результуюча таблиця

Результуюча таблиця виступає базою даних для побудови нових графіків.

Рисунок 5.3 - Графіки АТД, ЧСС для 5 кластеру

Рисунок 5.4 - Графіки АТС, ЧСС для 5 кластеру

Аналогічні аналізи були проведені для чотирьох і трьох кластерів, в ході чого було отримано відповідні результуючі таблиці та побудовано наступні графіки:

Рисунок 5.5 - Графіки АТД, ЧСС для 4 кластеру

Рисунок 5.6 - Графіки АТС, ЧСС для 4 кластеру

Рисунок 5.7 - Графіки АТД, ЧСС для 3 кластеру

Рисунок 5.8 - Графіки АТС, ЧСС для 3 кластеру

Графіки, на базі трьох кластерів, є найменш інформативними, оскільки втрачається група з даними, що показують високі показники систолічного тиску. Графіки на базі чотирьох кластерів показують, що група з низькими показниками систолічного тиску зникає, але в цілому інші графіки показують, що кластери досить різні. Тому виходячи з графіків можемо зробити висновок, що найбільш інформативними є графіки на базі п’яти кластерів. Також можемо спостерігати як змінюється дисперсія зі зменшенням кількості кластерів

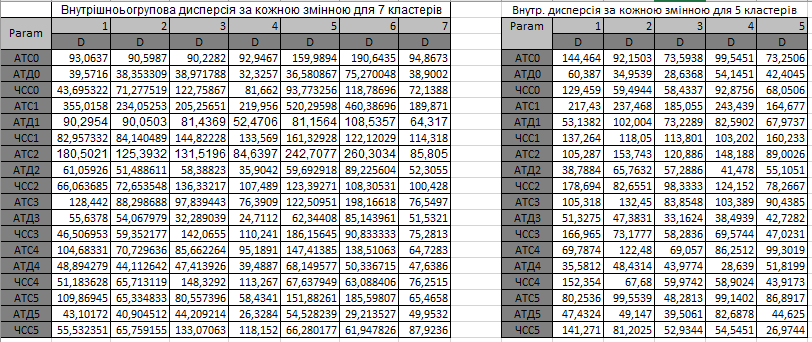


Рисунок 5.9 – Зміна дисперсій для 7 і 5 кластерів

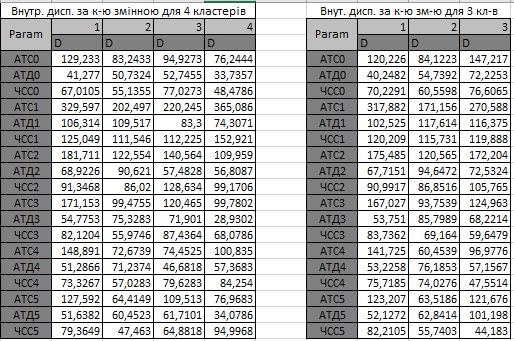


Рисунок 5.10 – Зміна дисперсій для 4 і 3 кластерів

З порівняльної таблиці бачимо, що дисперсія досить сильно змінюється при зменшенні кластерів на 2 і 3 позиціях, але при зменшенні на 4, дисперсія майже не змінилася, тому вибирати найоптимальнішу кількість виходячи з аналітичної частини треба серед таблиць на 5 і 4 кластери.

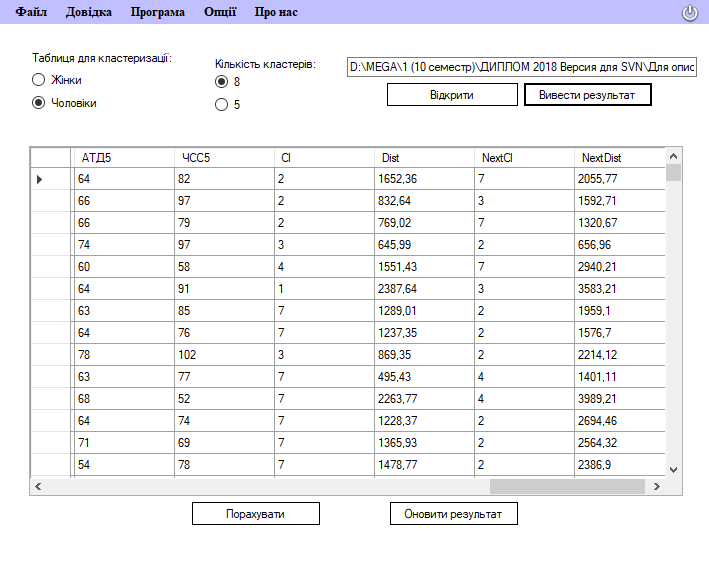
Виходячи з графіків та дисперсійного аналізу чітко видно, що таблиці на 5 кластерів дають найоптимальніший результат, тому було вирішено розширити функціонал програмного продукту Clusterbox і додати до нього можливість вибору результуючої таблиці на 5 кластерів.

Таким чином, при запуску основного програмного продукту, виводиться вікно з вибором режиму кластеризації. Воно було розроблене спеціально, щоб у користувача була можливість класифікувати в автоматичному режимі за наявними в програмі результуючими таблицями всю базу даних.



Рисунок 5.11 – Вибір режиму кластеризації

При натисненні на клавішу «Глобальний режим» відкривається вікно для проведення глобальної кластеризації.

Рисунок 5.12 – Глобальна кластеризація

Для проведення дослідження необхідно обрати з пункту «Кількість кластерів», яку результуючу таблицю слід використовувати для кластеризації, а також визначити для якої частини бази буде проведена кластеризація. Наступним кроком є завантаження таблиці для дослідження. Якщо вона вже містить значення кластеру, мітки (субкластеру), мінімальної та субмінімальної відстані, то можна вивести дані на екран. В іншому випадку необхідно буде провести дослідження, а потім оновити результат.

Слід зазначити, що якщо база даних містить значення кластеру, то при натисненні клавіші «Порахувати» випливе попередження: якщо натиснути так, тоді дані будуть перезаписані. Для прикладу проведемо кластеризацію з використанням результуючої таблиці на 5 кластерів та перезаписом результатів попередніх досліджень.

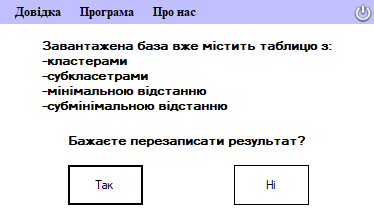


Рисунок 5.13 – Вікно з попередженням

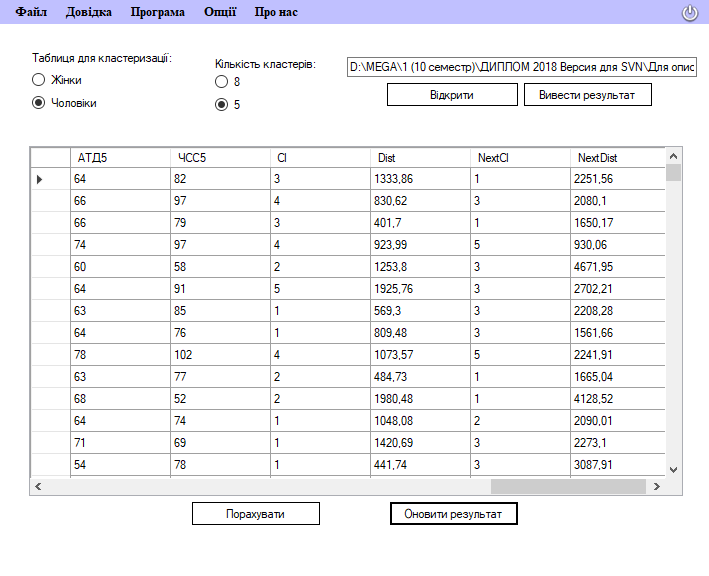


Рисунок 5.14 – Оновлене вікно глобальної кластеризації

При оновленні результату ми бачимо, що дані були успішно перезаписані.

З вікна «Глобальна кластеризація» тож можемо легко перейти до вікна «Одинична кластеризація» за допомогою контекстного меню.

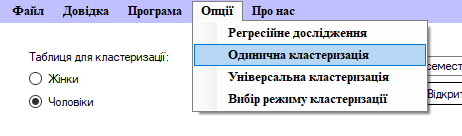


Рисунок 5.15 – Вибір одиничної кластеризації

Також можна перейти до модулю проведення регресійного аналізу, що було розглянуто у розділі 4.

Вікно для одиничної кластеризації було модифіковано таким чином, що ми можемо вибрати результуючу таблицю, яка буде використана для розрахунку мінімальної відстані до кластеру.

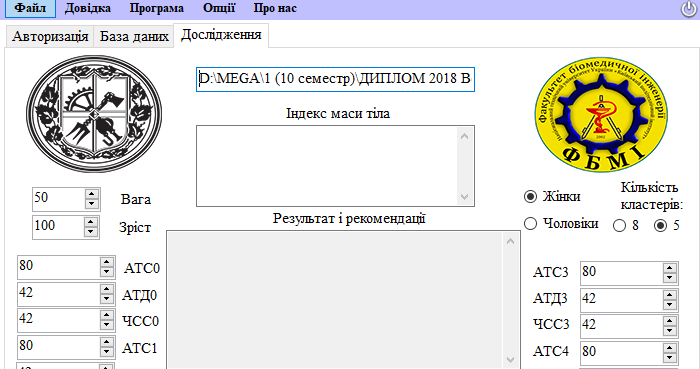


Рисунок 5.16 – Вікно одиничної кластеризація

Висновок до розділу 5

Даний розділ розкриває сутність роботи. В ньому розкрито і обґрунтовано зменшення кількості кластерів. Розглянуто умови та процес зміни кількості кластері в залежності від їх порівняння на побудованих графіках. Нами було отримано декілька нових результуючих таблиць і визначено оптимальну кількість кластерів для чоловіків, що становить 5 інформативних груп. Наведено додаткові модулі для дослідження, серед яких є глобальна кластеризація. Також проведено дисперсійний аналіз для порівняння зміни дисперсій із зменшенням кількості кластерів.

Поданий детальний опис алгоритму роботи з програмним додатком у розрізі проведення певного дослідження в базі даних. Надана інструкція користування програмним додатком для отримання результатів.

# ЗАГАЛЬНІ ВИСНОВКИ

У результаті роботи над магістерською дисертацією було опрацьовано багато наукових джерел інформації стосовно теми дисертації, проаналізовано основні методи реалізації алгоритму квадрату евклідової відстані та побудови регресійних рівнянь, набуто професійних вмінь та навичок у роботі з сучасними інформаційними технологіями.

У ході виконання проекту нами було проаналізовано результуючі таблиці та графіки, отримані на основі бази даних молодших курсів НТУУ «КПІ ім. Сікорського», що містить 1495 спостережень, проаналізовано нову базу даних, що містить 599 спостережень, розщеплено її по статі та проведено глобальну кластеризацію чоловічої групи. Нами було проведено логістичну регресію та дискримінантний аналіз та підтверджено ефективність алгоритму квадрата евклідової відстані. На основі цього нами було розроблено програмний додаток для визначення подібності функціональних патернів, визначено кількість оптимальних кластерів та побудовано нові результуючі таблиці для автоматизації процесу визначення регуляторних реакцій на тестове навантаження.

Результатом магістерської дисертації стала комп’ютерна система для визначення стану системи кровообігу з додатковими модулями глобальної кластеризації, універсальної кластеризації та модулем знаходження подібних кластерів. Програмним продуктом також передбачено збереження результатів кластеризації до завантаженої бази даних шляхом створення нової таблиці з відповідними стовпчиками.

На даному етапі програмний продукт повністю готовий до використання. Програма може бути вдосконалена та доповнена новими функціями.

# ПЕРЕЛІК ПОСИЛАНЬ

1. Ильинич В.И. Физическая культура студента и жизнь: учебник /

В.И. Ильинич. – М.: Гардарики, 2010. – 366 с.

1. Ланда Б.Х. Методика комплексной оценки физического развития и физической подготовленности: учеб. пособие / Б.Х. Ланда. – М.: Советский спорт, 2011. – 348 с.
2. Оценка функционального состояния и адаптивных возможностей организма у студентов вуза в процессе занятий физической культурой: методические указания для преподавателей физической культуры и студентов / сост. С.Е. Бебинов. – Омск: Изд-во СибАДИ, 2004. – 16 c.
3. Пономарёв В.В. Интегративный контроль физкультурного образования школьников на Крайнем Севере / В.В. Пономарёв В.Е. Лыков. – М.: Теория и практика физической культуры, 2007. – 130 с.
4. Холодов Ж.К. Теория и методика физической культуры и спорта / Ж.К. Холодов. – 10-е изд. – М., 2012. – 480 с.
5. Брехман И.И. Валеология – наука о здоровье. – М.: ФиС, 1990. – 208с.
6. Войтенко В.П. Здоровье здоровых. Введение в санологию. – К.: Здоров’я, 1991. – 248с.
7. Дембо А.Г. Врачебный контроль в спорте. – М.: Медицина, 1988.–288 с.
8. Казначеев В.П., Баевский Р.М., Берсенева А.П. Донозологическая диагностика в практике массовых обследований населения. – Л.: Медицина, 1980. – 210с.
9. Карпман В.Л., Белоцерковский Э.Б., Гудков И.А. Тестирование в спортивной медицине. – М.: ФиС, 1988. – 208с.
10. Оценка функциональных способностей организма человека с помощью функциональных проб [Електронний ресурс] – Режим доступу до ресурсу: http://intranet.tdmu.edu.ua
11. Кластерний аналіз [Електронний ресурс] – Режим доступу до ресурсу: http://www.nickart.spb.ru/analysis/cluster.php
12. Задачі кластерного аналізу [Електронний ресурс] – Режим доступу до ресурсу: https://sites.google.com/site/ne4itkalogika/necitka-klasterizacia/zadaci-klasternogo-analizu
13. Кластерний аналіз [Електронний ресурс] – Режим доступу до ресурсу: http://studopedia.su/7\_34114\_klasterniy-analiz.html
14. Індекс маси тіла [Електронний ресурс] – Режим доступу до ресурсу: http://meduk.net.ua/archives/13904
15. Розрахунок ідеальної ваги [Електронний ресурс] – Режим доступу до ресурсу: http://malibu-sport.com.ua/uk/kak-rasschitat-svoj-idealnyj-ves/
16. Індекс маси тіла [Електронний ресурс] – Режим доступу до ресурсу: http://a-yak.com/indeks-masi-tila-chi-vse-u-vas-v-poryadku-z-vagoyu/
17. Основні завдання та методи математичної статистики [Електронний ресурс] – Режим доступу до ресурсу: http://pidruchniki.com/14170120/statistika/predmet\_matematichnoyi\_statistiki
18. Завдання описової статистики [Електронний ресурс] – Режим доступу до ресурсу: http://biblio.royalwebhosting.net/zadachi-opisatelnoy-statistiki.html
19. Статистика і обробка даних [Електронний ресурс] – Режим доступу до ресурсу: http://psyfactor.org/lib/stat1.htm
20. Дисперсійний аналіз [Електронний ресурс] – Режим доступу до ресурсу: http://um.co.ua/4/4-7/4-70248.html
21. Дисперсійний аналіз [Електронний ресурс] – Режим доступу до ресурсу: http://ukrefs.com.ua/print:page,1,133443-Dispersionnyiy-analiz.html
22. Дисперсійний аналіз даних [Електронний ресурс] – Режим доступу до ресурсу: http://wiki.tntu.edu.ua
23. Основні заходи відстаней [Електронний ресурс] – Режим доступу до ресурсу: http://stud.com.ua/75021/statistika/osnovni\_zahodi\_vidstaney
24. Відстані між об’єктами і міра близькості [Електронний ресурс] – Режим доступу до ресурсу: http://bibliograph.com.ua/economicheskaya-statistika-2/11.htm
25. Кластерний аналіз. Загальне поняття, його математичні основи і завдання [Електронний ресурс] – Режим доступу до ресурсу: http://westudents.com.ua/glavy/88806--27-klasterniy-analz-zagalne-ponyattya-yogo-matematichn-osnovi-ta-zavdannya.html
26. Ієрархічна кластеризація [Електронний ресурс] – Режим доступу до ресурсу: https://studme.com.ua/1435012010869/marketing/ierarhicheskaya\_klasterizatsiya.htm
27. IBM SPSS Statistic Base [Електронний ресурс] – Режим доступу до ресурсу: http://www-03.ibm.com/software/products/ru/spss-stats-base
28. Программное обеспечение IBM SPSS [Електронний ресурс] – Режим доступу до ресурсу: http://www.predictivesolutions.ru/software/
29. Однофакторний дисперсійний аналіз [Електронний ресурс] – Режим доступу до ресурсу: https://www.ibm.com/support/knowledgecenter/ru/SSLVMB\_22.0.0/com.ibm.spss.statistics.help/spss/base/idh\_onew.htm
30. Описательные статистики [Електронний ресурс] – Режим доступу до ресурсу: https://www.ibm.com/support/knowledgecenter/ru/SSLVMB\_22.0.0/com.ibm.spss.statistics.help/spss/base/idh\_desc.htm
31. Создание простой блок-схемы [Електронний ресурс] – Режим доступу до ресурсу: https://support.office.com/en-US/article/Basic-tasks-in-Visio-B58073C2-12C8-4981-AD7E-235066FA910D
32. IDEF0 методология. Нотация, принципы моделирования [Електронний ресурс] – Режим доступу до ресурсу: http://www.nazametku.com/dlia-raboty/idef0-metodologiya-notaciya-principy-model/
33. Моделирование деловых процессов [Електронний ресурсу] – Режим доступу до ресурсу: http://www.info-system.ru/designing/methodology/sadt/sadt\_for\_bp.html
34. Діаграма дерева вузлів [Електронний ресурсу] – Режим доступу до ресурсу: http://studopedia.com.ua/1\_162876\_diagrami-dereva-vuzliv-i-FEO.html
35. Диаграмма сценариев использования в процессе разработки ПО [Електронний ресурс] – Режим доступу до ресурсу: https://habrahabr.ru/company/luxoft/blog/312188/
36. UML-схемы последовательностей: справочные материалы [Електронний ресурс] – Режим доступу до ресурсу: https://msdn.microsoft.com/ru-ru/library/dd409377.aspx
37. Диаграмма кооперации [Електронний ресурс] – Режим доступу до ресурсу: http://itteach.ru/rational-rose/diagrammi-vzaimodeystviya
38. Диаграмма деятельности UML [Електронний ресурс] – Режим доступу до ресурсу: http://www.planerka.info/item/Diagrammy-deyatelnosti-UML
39. Автоматическая генерация тестов для графического пользовательского интерфейса по UML диаграммам действий [Електронний ресурс] – Режим доступу до ресурсу: http://citforum.ru/SE/testing/generation\_uml/
40. Інструкція з охорони праці №97. Для працівників, які використовують персональні комп'ютери. – Львів, 2010, 5 с.
41. ГОСТ 12.0.003-74. Система стандартов безопасности труда. Опасные и вредные производственные факторы. Классификация. М.: Изд-во стандартов, 1988. - 79 с.
42. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99 [Електронний ресурс]. – Режим доступу до ресурсу: http://mozdocs.kiev.ua/view.php?id=1972
43. ДСанПіН 3.3.2.007-98 Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ЕОМ [Електронний ресурс]. – Режим доступу до ресурсу: http://document.ua/derzhavni-sanitarni-pravila-i-norm.. – Document.ua
44. ДСН 3.3.6.037-99. Санітарні норми виробничого шуму, ультразвуку та інфразвуку [Електронний ресурс]. – Режим доступу до ресурсу: http://mozdocs.kiev.ua/view.php?id=1789
45. СНиП 2.09.02—85. Строительные нормы и правила. Производственные здания. М: ГОССТРОЙ СССР, 1991. – 35 с.
46. Методичні вказівки до виконання розділу «Охорона праці» в дипломних роботах для студентів Факультету біомедичної інженерії / Уклад.: Демчук Г.В., Качинська Н.Ф., Чикунова Н.П. К.: НТУУ «КПІ», -15 с

# ДОДАТОК А

**Технічне завдання**

1. Найменування програмного продукту

Найменування ПП: «Програма для оцінки функціональних реакцій на тестове навантаження у студентів»

1. Призначення і область застосування

ПП призначений для:

− Визначення групи студента

− Виведення характеристик групи

− Виведення рекомендацій та застережень

Програмний продукт має власний, приємний GUI-інтерфейс і може використовуватися в:

− Медичних закладах для оцінки серцево-судинної системи

− Учбових закладах для оцінки функціональних реакцій організму

1. Вимоги до ПП
   1. Вимоги до функціональних характеристик

ПП повинен забезпечувати можливість виконання перерахованих нижче функцій:

− Завантаження результуючої таблиці до програми

− Розрахунок мінімальної відстані

− Розрахунок субмінімальної відстані

− Визначення найближчого кластеру

− Перевірка на попадання об’єкта в радіус кластеру

− Виведення характеристик та рекомендацій для кластеру

− Збереження результатів до локальної бази даних

* 1. Вимоги до забезпечення надійного функціонування ПП

Надійне (стійке) функціонування ПП має бути забезпечене виконанням Замовником сукупності організаційно-технічних заходів, перелік який наведено нижче:

− Організацією безперебійного живлення технічних засобів

− Використанням ліцензійного програмного забезпечення

− Відсутністю сторонніх або шкідливих програм, що можуть привезти до непрацездатності даної програми

* 1. Час відновлення після відмови

Оскільки в роботі використовується відлагоджений продукт, то час відновлення після відмови, викликаного збоєм електроживлення технічних засобів (іншими зовнішніми чинниками), не фатальним збоєм (не крахом) операційної системи, не повинно перевищувати 5-ми хвилин за умови дотримання умов експлуатації технічних і програмних засобів.

Аналогічно час відновлення після відмови, викликаного несправністю технічних засобів, фатальним збоєм ОС, не повинно перевищувати часу, необхідного на усунення несправностей технічних засобів і переустановлення програмних засобів.

* 1. Відмови через некоректні дії користувача системи

Відмови програмного продукту внаслідок некоректних дій користувача при взаємодії з ПП через GUI-інтерфейс недопустимі.

1. Умови експлуатації
   1. Кліматичні умови експлуатації

Кліматичні умови експлуатації, при яких повинні забезпечуватися задані характеристики, повинні задовольняти вимогам, що пред’являються до технічних засобів в частині умов їх експлуатації.

* 1. Вимоги до складу і параметрів технічних засобів

До складу технічних засобів повинен входити IBM-сумісний персональний комп’ютер (ПЕОМ), що включає в себе

− Процесор Intel ® Pentium G3240 ™, не менше

− ОЗУ об’ємом, 2 Гігабайти, не менше

− HDD, 32 Гігабайти, не менше

− Операційну систему Windows XP або більш пізню версію

* 1. Вимоги до програмних засобів, які використовуються ПП

Системні програмні засоби, що використовуються ПП, повинні бути представлені ліцензійною локалізованою версією операційної системи Windows XP (або пізнішої версії). Програмним забезпеченням: Microsoft Vusial Studio 2013 та Microsoft Access 2010.

* 1. Спеціальні вимоги

ПП повинен забезпечувати роботу користувача за допомогою

GUI-інтерфейсу. При необхідності встановити додаток AccessDatabaseEngine.

1. Стадії та етапи розробки
   1. Стадії розробки

* Аналіз вимог
* Проектування
* Реалізація
* Тестування і налагодження роботи ПП
* Розробка документації
* Впровадження та супровід

1. Порядок прийому та контролю
   1. Види випробувань

* Перевірка алгоритмів знаходження мінімальної та субмінімальної відстані за допомогою програми Microsoft Excel 2010.
* Перевірка роботи ПП на різних комп’ютерах
* Перевірка роботи ПП в різних версіях ОС Windows
  1. Загальні вимоги до приймання роботи

Робота виконана згідно вимог методичних вказівок.

# ДОДАТОК Б

**Лістинг методу розрахунку мінімальної відстані з подальшим визначення кластеру**

private void btnCulculate\_Click(object sender, EventArgs e)

{

if (tbPathToFile.Visible == true)

{

try

{

double[] DataEnter = new double[18];

double dist = 0.0;

Data();

DataEnter = Data();

btnStart\_Click();

double dataExcelFormat;

List<double> SummClasters = new List<double>();

for (int i = 1; i < dataGridViewExcel.Columns.Count; ++i)

{

double summ = 0;

for (int j = 1; j < dataGridViewExcel.Rows.Count; ++j)

{

double.TryParse(dataGridViewExcel[i, j].Value.ToString(), out dataExcelFormat);

summ += Math.Pow((DataEnter[j - 1] - dataExcelFormat), 2);

}

SummClasters.Add(summ);

}

List<double> SummClastersCopy = new List<double>();

for (int i = 0; i < SummClasters.Count; i++)

{

SummClastersCopy.Add(SummClasters[i]);

}

SummClastersCopy.Sort();

for (int i = 0; i < DataEnter.Length;i++ )

{

dist += DataEnter[i];

}

double middle = (dist / 18);

double min = SummClastersCopy[0];

double nextMin = SummClastersCopy[1];

string read = "";

double middlePoint=0.0;

for (int i = 0; i < DataEnter.Length;i++ )

{

middlePoint += Math.Abs(DataEnter[i] - middle);

}

tbResult.Clear();

tbRecomend.Clear();

tbIndexMass.Clear();

for (int i = 0; i < SummClasters.Count; i++)

{

if (min == SummClasters[i])

{

switch (i)

{

case 0: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " +Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

case 1: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " + Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

case 2: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " + Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

case 3: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " + Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

case 4: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " + Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

case 5: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " + Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

case 6: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " + Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

case 7: tbResult.Text += "Студент відноситься до кластеру \*" + (i + 1) + "\*" + Environment.NewLine +

"Мінімальна відстань до кластеру становить: " + Math.Round(SummClasters[i], 2) + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); cluster = i + 1; break;

}

}

}

if ((Math.Round((middlePoint/18),2)> radiusCl[cluster - 1]) || (nextMin < ((min \* 0.16) + min))) {

for (int i = 0; i < SummClasters.Count; i++)

{

if (nextMin == SummClasters[i])

{

switch (i){

case 0: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

case 1: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

case 2: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

case 3: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

case 4: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

case 5: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

case 6: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

case 7: tbRecomend.Text += "Наступна відстань після мінімальної становить \*" + Math.Round(SummClasters[i], 2) + "\*"

+ Environment.NewLine + "Це кластер \*" + (i + 1) + "\*" + Environment.NewLine + Environment.NewLine +

"Кластер має наступні характристики: " + ReadFile(read, i + 1); break;

}

}

}

}

BodyMassIndex();

}

catch (FormatException)

{

MessageBox.Show("Перевірте корректність введених даних\nДив.інструкцію користувача\"Програма->Інструкція користувача\"");

}

catch (InvalidOperationException)

{

MessageBox.Show("Встановіть додаток \"AccessDatabaseEngine\"\n з папки \"Додаток\"");

}

catch (OleDbException)

{

if (File.Exists(tbPathToFile.Text)){

MessageBox.Show("Невірні поля у завантаженому файлі\nДив.інструкцію користувача\"Програма->Інструкція користувача\"");

}

else if (!File.Exists(tbPathToFile.Text)){

tbPathToFile.Visible = true;

MessageBox.Show("Файл не знайдено\nВиберіть інший файл!");

}

}

}

else {

MessageBox.Show("Виберіть файл\nНатисніть Файл->Відкрити");

}

}