МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет радіоелектроніки
Факультет Центр післядипломної освіти
(повна назва)
Кафедра Програмної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

другий (магістерський)
(рівень вищої освіти)

Дослідження методів прогнозування для оцінки успішності студентів університету за підсумками поточного навчання.

Виконала:
студентка 2 курсу, групи ІПЗздм-22-1
Коровайна В.С.
(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного забезпечення
Тип програми освітньо-наукова
Керівник доц. Колесніков Д.О.
(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри _____ З.В. Дудар
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Центр післядипломної освіти

ЗАТВЕРДЖУЮ
зав. каф. ПІ, к.т.н., проф.
_____ Дудар З. В.
(підпис)
«___» _____ 2024р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

Студенці групи ІПЗздм-22-1 Коровайної Владиславі Сергіївні

1. Тема роботи: “Дослідження методів прогнозування для оцінки успішності студентів університету за підсумками поточного навчання”.

2. Наказ на практику №27Стз від «12» лютого 2024 р.

3. Вихідні дані до роботи: опис основних задач проекту, його мета та очікувані результати, вимоги до документації та стандарти, стандарти тестування та розробки програмного забезпечення, календарний графік проекту та терміни завершення кожного етапу, вирішення задачі багатокритеріального вибору, побудова математичної моделі.

4. Перелік питань, що потрібно опрацювати в роботі: аналіз предметної області (алгоритм K-найближчих сусідів, метод опорних векторів, нейронні мережі), попередня обробка даних, модель машинного навчання.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз предметної галузі	01.04.2024	виконано
2	Виявлення проблематики галузі	08.04.2024	виконано
3	Здійснення огляду математичних моделей	15.04.2024	виконано
4	Розробка алгоритму передобробки даних	22.04.2024	виконано
5	Дослідження можливості прискорення	29.04.2024	виконано
6	Побудова плану експерименту	06.05.2024	виконано
7	Імплементація прототипів обраних моделей	13.05.2024	виконано
8	Аналіз результатів експерименту	20.05.2024	виконано
9	Написання пояснювальної записки	27.05.2024	виконано
10	Підготовка доповіді та презентації	27.05.2024	виконано
11	Підготовка роботи для перевірки на антиплагіат та проходження нормоконтролю	31.05.2024	виконано
12	Оцінка роботи рецензентами	08.06.2024	виконано
13	Отримання відзиву від керівника роботи	10.06.2024	виконано
12	Попередній захист кваліфікаційної роботи	10.06.2024	виконано
13	Здача роботи у електронний архів	10.06.2024	виконано
14	Отримання допуску до захисту	12.06.2024	виконано
15	Захист кваліфікаційної роботи	14.06.2024	виконано

Дата видачі завдання 01 квітня 2024 р.

Студент _____ Коровайна В. С.
(підпис)

Керівник роботи доц. Колесніков Д.О.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить 63 сторінки, 26 рисунків, 3 таблиць, 7 джерел, 5 додатків.

ПРОГНОЗУВАННЯ УСПІШНОСТІ СТУДЕНТІВ, МОДЕЛЬ МАШИННОГО НАВЧАННЯ, ПІДГОТОВКА ДАНИХ, НАБІР ДАНИХ, КЛАСИФІКАЦІЯ.

Об’єктом дослідження є процес розробки прогнозної моделі для оцінки успішності студентів університету за підсумками поточного навчання.

Мета роботи – побудова прогнозної моделі підсумків сесії студентів залежно від оціночних параметрів поточної успішності.

У процесі дослідження було проведено аналіз основних проблем у цій галузі, поставлено цілі для їх безпосереднього виконання. Також було проведено попередню обробку даних для побудови моделі машинного навчання. Після цього було побудовано різні моделі машинного навчання, а також оцінено якісні показники кожної моделі. Після вибору оптимальної моделі було створено графічний інтерфейс користувача прогнозної моделі.

У результаті дослідження було створено прогнозну модель успішності студентів університету, а також графічний інтерфейс для її використання. Визначено найбільш значущі фактори під час прогнозування успішності у студентів.

PREDICTING STUDENT PERFORMANCE, MACHINE LEARNING MODEL, DATA PREPARATION, DATA SET, CLASSIFICATION.

The object of research is the process of developing a predictive model for assessing the performance of university students based on the results of current studies.

The purpose of the study is to build a predictive model of students’ session results depending on the estimated parameters of current performance.

In the course of the study, the main problems in this area were analyzed, and goals were set for their direct implementation. We also conducted preliminary data processing to build a machine learning model. After that, various machine learning models were built, and the qualitative indicators of each model were evaluated. After selecting the optimal model, a graphical user interface for the predictive model was created.

As a result of the study, a predictive model of university students’ academic performance was created, as well as a graphical interface for its use. The most significant factors in predicting student performance have been identified.

ЗМІСТ

ВСТУП 6
1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ 7
1.1 Алгоритм K-найближчих сусідів 10
1.2 Метод опорних векторів 12
1.3 Нейронні мережі 14
1.4 Постановка задачі 14
2. ПОПЕРЕДНЯ ОБРОБКА ДАНИХ 18
3. МОДЕЛЬ МАШИННОГО НАВЧАННЯ 30
4. ГРАФІЧНИЙ ІНТЕРФЕЙС КОРИСТУВАЧА ПРОГНОЗНОЇ МОДЕЛІ 35
ВИСНОВКИ 40
ПЕРЕЛІК ПОСИЛАНЬ 41
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ЗА НАУКОВИМИ НАПРЯМАМИ КЕРІВНИКА ТА НАУКОВЦІВ КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ 43
Додаток А. Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ 43
Додаток Б. Лістинг коду з попередньої підготовки даних 43
Додаток В. Апробація результатів роботи 51
Додаток Г. Слайди презентації 58
Додаток Д. Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення вимогам ДСТУ 3008: 2015 63

ВСТУП

Рівень успішності студента у ЗВО є своєрідною формою діагностики і прогнозування міри віддачі майбутнього фахівця. У свою чергу успіхи студентів - це показник діяльності ЗВО в рішенні учбово-виховних завдань. Для того, щоб вирішувати данні задачі максимально ефективно потрібно постійну об'єктивну оцінку, коригування і управління. Проте, без прогнозування управління неможливе. Тому виникає необхідність прогнозування успішності студентів на усіх етапах навчання.

Маючи відомості про тих студентів, які найімовірніше до кінця семестру матимуть академічні заборгованості, якщо не змінять поточну тенденцію, ми можемо вплинути на студентів, тим самим підвищити рівень їх успішності.

Метою цієї роботи є створення прогнозної моделі успішності студентів ХНУРЕ. Наявність такої моделі дозволить приділяти пильнішу увагу студентам, які потрапляють в групу ризику великої кількості боргів по учбових дисциплінах, а як слідство, будуть претендентами на відрахування. Визначення таких студентів на ранніх етапах дозволить детальніше і персонально працювати з ними для того, щоб вони успішніше справлялися з учбовим навантаженням.

Виходячи з мети, ця робота включає наступні завдання: огляд літератури присвяченої цьому питанню, вивчення використовуваних методів і алгоритмів, очищення і підготовка початкових даних, розробка прогнозної моделі, апробація результатів, створення графічного інтерфейсу користувача для модуля прогнозування успішності студентів.

1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Освіта грає одну з найважливіших ролей у будь-якій державі. Від якості освіти, існуючої в конкретному суспільстві, багато в чому залежать темпи його економічного і політичного розвитку, його моральний стан. Стрімкий розвиток інформаційних технологій дозволяє автоматизувати багато сфер діяльності людей підвищуючи їх ефективність, і освіта не є виключенням. У цій роботі мова піде про створення прогнозної моделі успішності студентів за поточними оцінками за допомогою технологій аналізу даних.

Мірою виміру якості освіти, що здобувається, для конкретного студента служать його оцінки по пройдених предметах. Якщо говорити про учбовий заклад, то одним із заходів якості освіти, що надається їм, служить сукупність оцінок його учнів. Своєчасні заходи по наданню допомоги студентам, які не справляються з учбовим навантаженням, є однією з основних частин по учбово-виховній роботі у ЗВО, які впливають на показники якості освіти в учбовому закладі.

Останнім часом, було зроблено багато різних змін для поліпшення якості освіти у ЗВО. Наприклад, перехід від традиційної системи оцінок до бальних виключає можливість того, що студент, який не ходив на зайняття увесь семестр просто прийде і здасть іспит. Для допуску до іспиту йому треба набрати певну кількість балів, а для цього у свою чергу необхідно відвідувати зайняття і виконувати поточні завдання.

Такий підхід ефективно впливає на розуміння учбового матеріалу. Згідно з багатьма дослідженнями, інформація, яка була вивчена впродовж тривалого проміжку часу, збережеться на довгий час, тоді як, "зубріння" в ніч перед іспитом може дати лише добрий результат на іспиті, який у результаті буде зданий, але залишкових знань від предмета у студента зовсім не залишиться. Окрім цього, існує контроль рубежу, призначена дата в середині семестру, коли певна частина матеріалу має бути здана. Проте, знаючи специфіку студентського життя, багато студентів все одно залишають все на останній момент. Деяким студентам вдається

здати предмет, інші ж залишаються з боргом на інший семестр. Проблема полягає в тому, що дія, що управляє, з боку єдиного деканату ХНУРЕ починається тільки після того факту, що у студента вже з'явилася заборгованість.

Таким чином, ці заходи не можна назвати превентивними. Головним завданням прогнозної моделі є виділення таких студентів і проведення з ними певних бесід до того, коли з'явиться проблема у вигляді академічної заборгованості. Нині ця міра реалізована частково тим, що у кожної групи першокурсників є куратор, і цей куратор супроводжує кожну групу аж до 2го курсу, в розкладі відводяться такий захід як "година куратора", де він аналізує успішність студентів. Це є відмінною практикою, і відмовлятися від неї не можна, проте, тут грає роль людський чинник. Не завжди куратор може донести до студентів важливість відвідування зайняття. Впровадження нової системи прогнозування кількості заборгованостей у кінці семестру по поточній успішності є важливим етапом автоматизації процесу учбово-виховної роботи. Таким чином, єдиний деканат зможе чинити дію, що управляє, на студентів, що потрапили в групу ризику, набагато раніше за виникнення реальної проблеми. Тим самим, з'явиться можливість допомогти студентам закінчувати семестр успішно і формувати професійні компетенції, а тих, хто не зацікавлений в навчанні, відраховувати раніше і звільняти місця для тих, хто дійсно хоче і готовий отримувати знання.

Основна увага приділена саме системі прогнозування, оскільки йдеться не просто про контроль відвідуваності зайняття студентом. Аналіз даних показує, що на кінцевий результат семестру впливає не лише один чинник відвідуваності, а цілий комплекс різних даних. Наприклад, існує ціла категорія студентів, які вже працюють за фахом, в основному це магістранти і студенти останніх курсів. У таких студентів не завжди є можливість відвідування зайняття, і проте вони показують відмінні результати на кінець сесії, внаслідок того, що розуміють багато аспектів професії на більш високому рівні чим їх одногрупники.

Насправді на успішність студента впливає величезна кількість чинників, і одними з найважливіших є мотивація на навчання, моральний стан, відношення з

одногрупниками і так далі, але завдяки тому, що система видаватиме проблемних студентів і з кожним буде можливість працювати детально, виявляти які проблеми існують саме у цього студента, який ризикує закінчив семестр з великий кількість борг. Виявлення таких параметрів як мотивація, цілеспрямованість, психологічні дані можливо, але це вимагає великого числа тестів, які повинні проводиться на регулярній основі. Ці методи не мають високої ефективності за рахунок складності їх проведення, а також перевірки і інтерпретації результатів.

Після з'ясування того, що ця проблема дійсно є актуальною, варто розглянути існуючі методи її рішень.

На даний момент у відкритому доступі немає матеріалів, присвячених впровадженню і використанню системи прогнозування успішності студента у ЗВО. Хоча ця ідея не є інноваційною, приблизно з 2010 року виходять статті про прогнозування успішності абітурієнтів, прогнозуванні успішності учнів по певному курсу, де за початкові дані як правило беруться наступні параметри: рівень поточних знань по предмету, кількість пропусків, проміжний контроль, оцінки по курсах, що забезпечують, до дисципліни.

Далі розглянемо алгоритми машинного навчання, які застосовуються для вирішення схожих завдань у сфері прогнозування успішності студентів. Будуть приведені методи, що найчастіше зустрічаються, і алгоритми, а також опис існуючих робіт з конкретними завданнями, для яких ці методи застосовуються.

Деякі джерела застосовують метод кластеризації, на наш погляд це не зовсім вірно, оскільки у нас вже влучні класів, а саме кількість заборгованостей, так що в даному випадку слід застосовувати методи класифікації. Кластеризація відноситься до методів машинного навчання без учителя, а класифікація і регресія у свою чергу до навчання з учителем, оскільки мають в якості початкових даних маркери для кожного класу. Методи кластерного аналізу для оцінки підсумкової оцінки по предмету були застосовані в статті [1].

1.1 Алгоритм K-найближчих сусідів

Один з методів, що найчастіше зустрічаються, для вирішення аналогічного завдання - це алгоритм K-найближчих сусідів :

Алгоритм K-найближчих сусідів (KNN, k nearest neighbors) - це тип керованого алгоритму машинного навчання, який може використовуватися як для класифікації, так і для завдань прогнозування регресії. Цей алгоритм є одним з найпростіших для розуміння, але проте він довів свою ефективність у ряді завдань і використовується не лише в учбових цілях. Алгоритм найближчих сусідів також називають "ледачим" класифікатором, тому що в процесі навчання він не будує яку-небудь модель, а просто зберігає дані. Усі обчислення починаються тільки тоді, коли необхідно класифікувати нові дані.

Суть цього алгоритму полягає в тому, що прогнозування значень нових даних ґрунтоване на їх близькості до вже маркірованих даних в повчальному наборі. Іншими словами, якщо нова точка даних має серед своїх найближчих сусідів 4 точки класу A і 1 точку класу B, то ця нова точка буде визначена як клас A. Таким чином алгоритм k- найближчих сусідів має 2 найважливіші параметри, а саме метрика відстані (Евклідове, Манхеттенське або Хеммінговське) і кількість сусідів, яку ми розглядатимемо. Візуальне представлення роботи алгоритму представлене на рисунку 1.1.

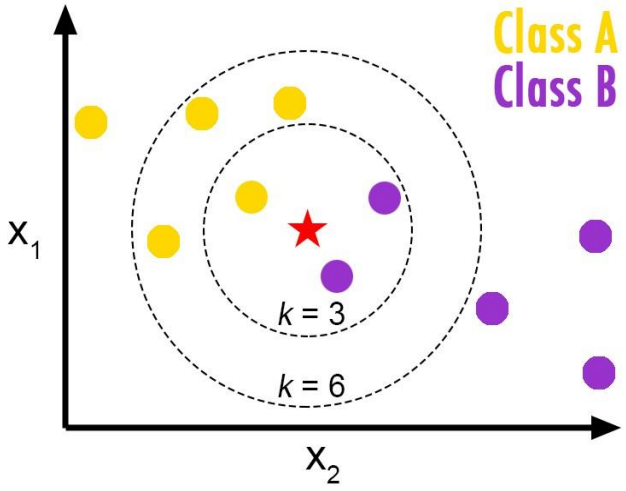


Рисунок 1.1 – Робота алгоритму KNN

На цьому рисунку зображений масив вихідних точок, які маркіровані кольором залежно від приналежності до класу. Зірочкою помічена точка, яку необхідно класифікувати. Усі точки представлені в двовимірному виді, по осях X1 і X2. Якщо встановлене число найближчих сусідів дорівнює 3, то немаркірований об'єкт відноситиметься до класу B, якщо 6, то вже до класу A.

Цей алгоритм має багато нюансів, наприклад, ми можемо додати ваги даним при голосуванні залежно від близькості до нашого немаркірованого об'єкту. Саме ці нюанси роблять алгоритм KNN актуальним для вирішення цілого спектру завдань.

Переваги цього алгоритму :

- Простота розуміння і інтерпретації.

- Добре працює на нелінійних даних.

- Універсальний алгоритм, підходить як для завдань класифікації, так і регресії.

- Має відносно високу точність.

Недоліки алгоритму :

- Велика витрата пам'яті на зберігання усіх даних у відмінності від алгоритмів, які використовують побудову моделей.

- Чутливість до масштабу даних.

- Повільне прогнозування у разі великої кількості даних.

- Висока чутливість до "шуму" в даних.

У статті [2] цей алгоритм використовується для класифікації оцінки окремого студента за кожним предметом на підставі його минулих оцінок і оцінок студентів минулих курсів, з максимально аналогічними параметрами по цих предметах. У цій статті приведена досить висока точність алгоритму для цього завдання, максимальна помилка прогнозу оцінки складала 0,55 балу. Проте враховувалося лише 307 студентів однієї спеціальності і ні про яке впровадження цієї методики в систему університету мови не йшло.

Також цей алгоритм використовується в статті [3], де прогнозується оцінка студента за екзамен з певного предмета, на підставі його оцінок по попередніх предметах, його відвідуваності і оцінок проміжного контролю.

1.2 Метод опорних векторів

Метод опорних векторів (SVM) – це сімейство схожих алгоритмів навчання з навчанням для розв'язання задач класифікації та регресії. Це один з найпоширеніших методів навчання, який належить до сімейства лінійних класифікаторів. Однією з характеристик методу опорних векторів є те, що він послідовно зменшує емпіричну помилку класифікації та збільшує дисперсію. Тому цей метод також називають методом класифікації на максимальній відстані.

Основну ідею методу опорних векторів можна проілюструвати на прикладі: на площині є точки, маркіровані на 2 класи, які лінійно роз’єднані. У такому разі результуючою функцією буде площина, яка розділяє ці класи. Проте, можливо провести безліч гіперплощин, які розділять ці класи. Для того, щоб знайти оптимальну гіперплощину, необхідно знайти максимальну суму векторів нормалі від класу A і класу B. Візуальне відображення цього методу можна побачити на рисунку 1.2. На цьому рисунку опорними векторами будуть перпендикулярні нормалям, які зображені на рисунку 1.2.

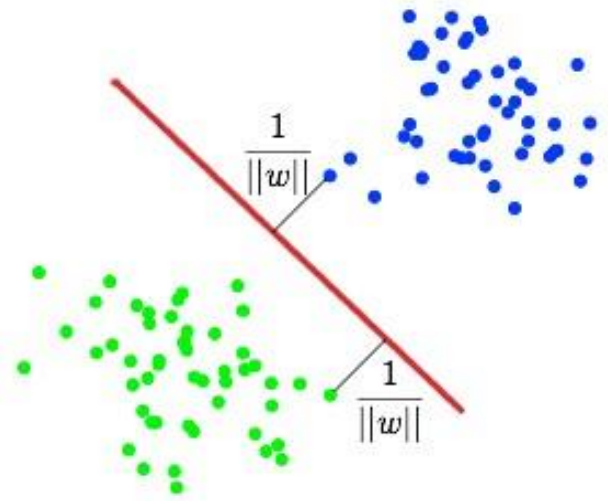


Рисунок 1.2 – Метод опорних векторів

Формальний опис цього методу наступний – нехай ми маємо повчальну вибірку:

$$\{(X_1, C_1), (X_2, C_2), \ldots, (X_i, C_i)\},$$

де

X_i – це p-мірний речовий вектор;

C_i – значення 1 або -1, яке приймає клас.

Метод опорних векторів будує класифікуючу функцію у виді:

$$F(x)=sign([w,x]+b), \tag{1.1}$$

де

[,] – скалярне множення;

w – нормальний вектор до розділяючої гіперплощини;

b – допоміжний параметр.

Таким чином, можна записати це усе у вигляді завдання оптимізації, яка має рішення і при цьому єдине:

$$\begin{cases} \|\mathbf{w}\|^2 \to \min \\ c_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1, \quad 1 \leq i \leq n. \end{cases} \tag{1.2}$$

Це завдання вирішується методом квадратичного програмування і за допомогою множників Лагранжа.

Був розглянутий випадок, коли існують 2 роздільні класи. На практиці майже завжди класи лінійно нерозділені, і стоїть завдання класифікації більш ніж 2х класів. Для вирішення завдання з лінійно нероздільними класами ми дозволяємо класифікаторові припускатися помилки на повчальній вибірці. Запишемо рівняння для такого допущення.

$$\begin{cases} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \to \min_{w,b,\xi_i} \\ c_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1 - \xi_i, \quad 1 \leq i \leq n \\ \xi_i \geq 0, \quad 1 \leq i \leq n \end{cases} \tag{1.3}$$

де C – параметр налаштування методу,

$\mathcal{E}_i$ – величина допустимої помилки.

Для вирішення мультикласових завдань використовується узагальнений метод опорних векторів за рахунок того, що перехід до класифікації на безліч класів здійснюється розбиттям на 2 класи, таких як відповідний клас і не відповідний. Ця стратегія також називається "Один проти усіх" і використовується для застосування бінарних класифікаторів для мультикласових завдань.

Переваги алгоритму :

- Завдання добре вивчене і має єдине рішення.

- Принцип оптимальної розділяючої гіперплощини призводить до упевненої класифікації.

- Еквівалентний двошаровій нейронній мережі, де число нейронів на прихованому шарі визначається автоматично як число опорних векторів.

Недоліки:

- Нестійкість до шуму, викиди в початкових даних безпосередньо впливають на побудову розділяючої гіперплощини.

- Немає відбору ознак.

- Необхідно підбирати методи побудови ядер і випрямляючих просторів окремо для кожного завдання.

Цей алгоритм використовується в статті [3], де прогнозується оцінка студента за екзамен з певного предмета, на підставі його оцінок по попередніх предметах, його відвідуваності і оцінок проміжного контролю.

1.3 Нейронні мережі

Окрім вищевикладених методів для прогнозування успішності студентів використовують ще і нейронні мережі. Зустрічається досить багато згадок про можливість використання нейронних мереж для вирішення цього завдання, проте інформації про реальне впровадження таких прогнозних моделей не зустрічається.

Нейронні мережі – це математичні моделі, засновані на організаційних і функціональних принципах біологічних нейронних мереж. Нейронні мережі

навчаються, а не програмуються в звичайному розумінні. Під час навчання нейронні мережі здатні ідентифікувати та узагальнювати складні взаємозв'язки між вхідними та вихідними даними. Після навчання мережа здатна передбачати майбутні значення для заданої послідовності на основі серії минулих значень.

Ілюстрація принципу роботи нейронної мережі представлена на рисунку 1.3. Нейронна мережа складається з нейронів, шарів і синапсів. Нейрони зображені у вигляді вузлів різного кольору. Усі вузли одного кольору відносяться до одного шару нейронної мережі. Синапси - це лінії, які зв'язують нейрони одного шару з нейронами іншого шару. Синапси мають всього один параметр, і це вага.

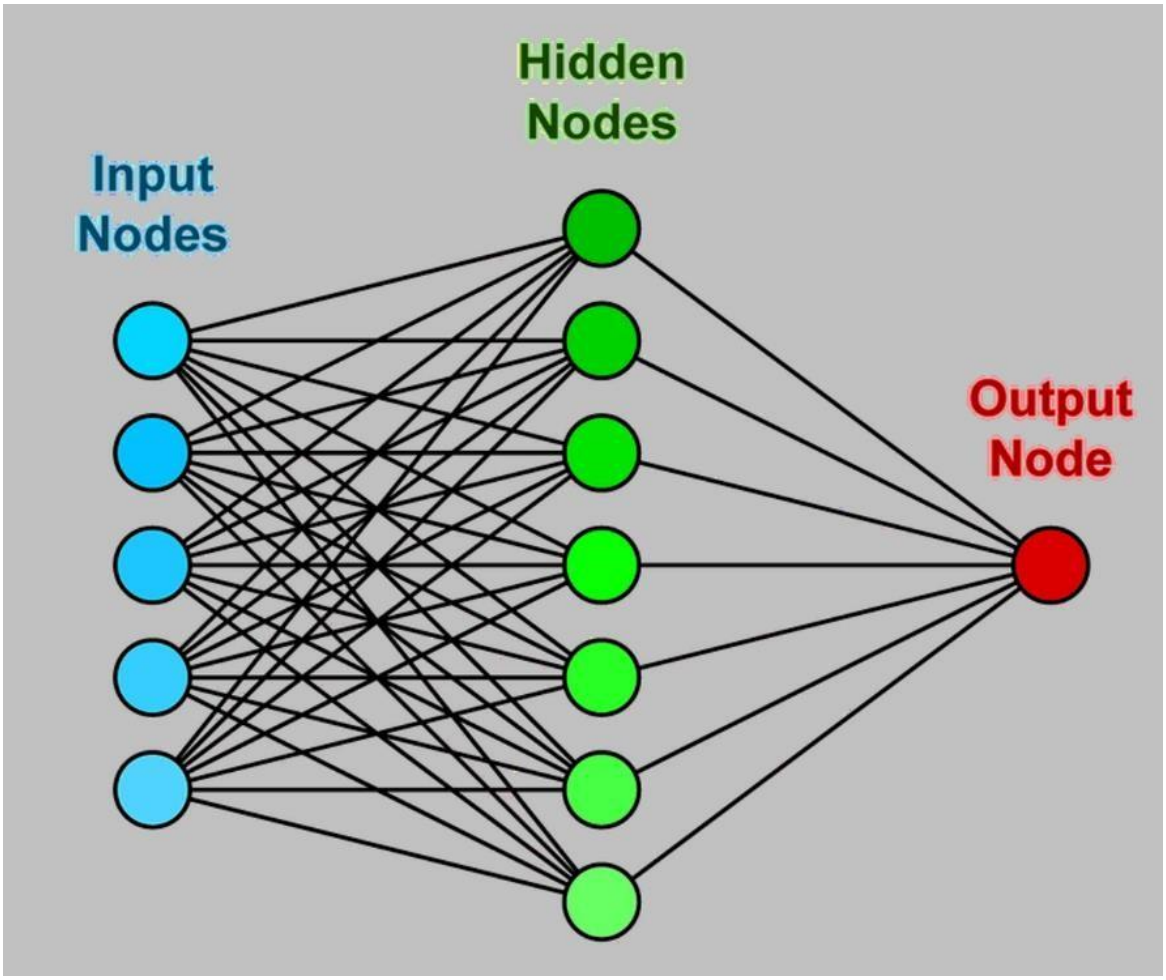


Рисунок 1.3 – Нейронна мережа

Кожен нейрон виконує певну математичну функцію, таким чином на вхід йому потрапляє множина значення, а на виході одно. Таким чином, на виході виходить певне значення, яке видала вже навчена нейронна мережа.

Переваги:

- Стійкість до шумів вхідних даних.

- Самонавчання і "креативність". Можливість рішення таких завдань, які не вирішуються іншими алгоритмами.

- Адаптація до змін, перенавчання.

Недоліки:

- Для великих мереж неможливість заздалегідь навіть приблизно оцінити час навчання мережі.

- Складність інтерпретації результату.

- Приблизність отриманої відповіді.

Розглянемо детальніше застосування нейронної мережі для вирішення завдання прогнозування успішності студентів. У статті [4] нейромережева модель навчена, щоб передбачати чи буде студент перспективним. Вирішується завдання бінарної класифікації абітурієнтів за такими вхідними даними як номер школи, оцінки по фізиці і математиці і професії батьків. У статті [5] нейронна мережа застосовується для прогнозування оцінки по курсу інформатики. У статті [6] розглянуто завдання класифікації студентів за результатами НМТ.

Отже, були розглянуті алгоритми, які найчастіше використовуються для вирішення проблеми прогнозування успішності учнів на основі різних початкових даних і вирішують різні завдання, будь то успішність по конкретному предмету або загальна картина успішності в усіх дисциплінах. Розглянуті приклади прогнозування не носять системного характеру, а є лише спробами наблизитися до вирішення цієї проблеми. Очевидно, що для успішного вирішення поставленого завдання необхідно застосувати декілька методів і порівняти їх результати.

1.4 Постановка задачі

Об'єктом дослідження є процес розробки прогнозної моделі для оцінки успішності студентів університету за підсумками поточного навчання.

Мета роботи – побудова прогнозної моделі підсумків сесії студентів залежно від оціночних параметрів поточної успішності.

У процесі дослідження необхідно провести аналіз основних проблем у цій галузі, поставити цілі для їх безпосереднього виконання. Також буде проведено попередню обробку даних для побудови моделі машинного навчання. Після цього буде побудовані різні моделі машинного навчання, а також оцінені якісні показники кожної моделі. Після вибору оптимальної моделі буде створено графічний інтерфейс користувача прогнозної моделі.

У результаті дослідження буде створено прогнозну модель успішності студентів університету, а також графічний інтерфейс для її використання. Крім того, будуть визначені найбільш значущі фактори під час прогнозування успішності у студентів.

2. ПОПЕРЕДНЯ ОБРОБКА ДАНИХ

Якість моделей машинного навчання дуже сильно залежить від якості початкових даних. Проте, реальні дані дуже часто погано структуровані, мають пропущені значення, шуми, а також помилкові значення. За умови, що дані не були якісно підготовлені ніякі налаштування алгоритмів машинного навчання не зможуть забезпечити високу прогнозну точність цих моделей. Підготовка даних перед їх аналізом займає близько 80 відсотків робочого часу фахівця з машинного навчання, але ця робота є необхідною.

У цій роботі для підготовки даних буде використана методика, описана в статті [7].

Для ознайомлення з даними і їх підготовкою для подальшого аналізу буде використана мова програмування Python та її бібліотеки. Вибір мови програмування Python обумовлений високою продуктивністю при обробці даних, простотою і великою кількістю бібліотек для машинного навчання. Python є однією з кращих мов програмування для роботи з даними. У цій роботі використовуватимуться наступні бібліотеки Python:

- NumPy. Ця бібліотека додає підтримку великих багатовимірних масивів і матриць, а також високорівневі команди математичних функцій з дуже високою продуктивністю над цими масивами [7].

- Pandas. Програмна бібліотека для обробки і аналізу даних.

- Seaborn. Бібліотека для візуалізації даних, грунтована на іншій бібліотеці програмної мови Python Matplotlib.

- Scikit - learn. Бібліотека машинного навчання на з відкритим початковим кодом.

- PyQt5 - набір розширень графічного фреймворка Qt для мови програмування Python, виконаний у вигляді розширення Python. Ця бібліотека практично повністю

реалізує можливості Qt і дозволяє створювати графічний інтерфейс для програм написаних на Python.

Насамперед, для того, щоб працювати з даними за допомогою Python необхідно їх перетворити у формат, з яким працює ця мова і його бібліотеки. В даному випадку таким форматом є DataFrame з бібліотеки Pandas.

Після перетворення є можливість ознайомитися з основними характеристиками початкового датасета (див. рисунок 2.1).

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8551 entries, 0 to 8550
Data columns (total 24 columns):
Форма навчання                                              8551 non-null object
Кваліфікація                                                8551 non-null object
Курс                                                        8551 non-null int64
Спеціальність                                               8551 non-null object
Профіль                                                     6676 non-null object
Випуск. відділ.                                             8551 non-null object
Випуск. школа                                               7988 non-null object
Група                                                       8551 non-null object
Навч. підрозділ.                                            8551 non-null object
ID                                                          8551 non-null int64
Форма фінансування                                          8551 non-null object
Країна                                                      8533 non-null object
Громадянство                                                8551 non-null object
Стать                                                       8551 non-null object
Дата народження                                             8551 non-null object
Академвідпустка (чинна) так / ні                            8551 non-null object
Усього                                                      8551 non-null int64
Позитивних                                                  8551 non-null int64
Незадовільних                                               8551 non-null int64
Дисципліни з яких отримано незадовільні оцінки              6370 non-null object
Пропусків з дисциплін за якими отримано незадовільні оцінки 8551 non-null int64
Усього годин з дисциплін з яких отримано незадовільні оцінки 8539 non-null float64
dtypes: float64(2), int64(7), object(15)
```

Рисунок 2.1 – Основні характеристики початкового датасета

Можна спостерігати, що в кортежі деяких атрибутів мають нульові значення, для більшої наочності слід подивитися в яких саме атрибутах вони є (див. рисунок 2.2).

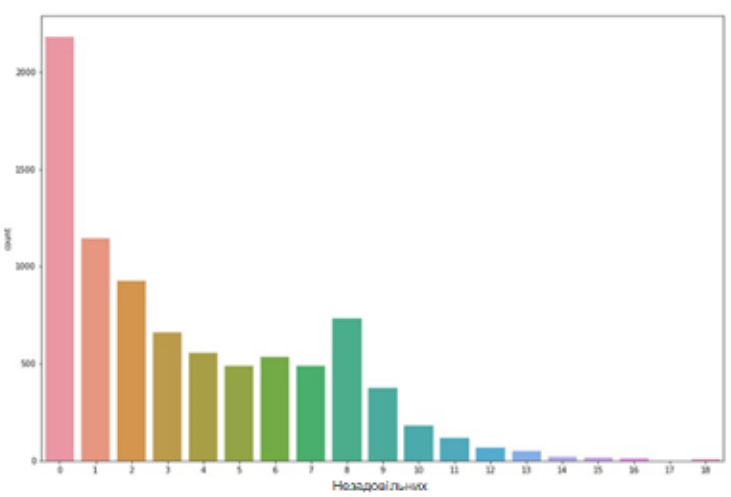
Можна бачити, що атрибут "Дисципліни по яких отримані незадовільні оцінки" сам по собі припускає наявність "миссингов", тоді як "профіль" і "країна" швидше за все пропущене значення в результаті заповнення бази даних.

Далі будуть розглянуті дані в найпростіших розрізах, для того, щоб зрозуміти початковий вектор підготовки даних.

В першу чергу розглянемо кількість (а разом і з ним якість, яку оцінимо по кількості боргів) боржників (див. рисунок 2.3).

Форма навчання	False
Кваліфікація	False
Курс	False
Спеціальність	False
Профіль	True
Випуск. відділ.	False
Випуск. школа	True
Група	False
Навч. підрозділ.	False
ID	False
Форма фінансування	False
Країна	True
Громадянство	False
Стать	False
Дата народження	False
Академвідпустка (чинна) так / ні	False
Усього	False
Позитивних	False
Незадовільних	False
Дисципліни з яких отримано незадовільні оцінки	True
Пропусків з дисциплін з яких отримано незадовільні оцінки	False
Усього годин з дисциплін з яких отримано незадовільні оцінки	True
Усього годин пропусків у семестрі	False
Усього годин аудиторних занять у семестрі	True

Рисунок 2.2 – Наявність нульових значень у параметрів



Незадовільних	ID
0	2181
1	1146
2	926
3	660
4	557
5	489
6	535
7	487
8	733
9	372
10	179
11	117
12	66
13	50
14	20
15	14
16	11
17	1
18	7

Name: ID, dtype: int64

Рисунок 2.3 – Кількість незадовільних оцінок

Щоб враховувати цей атрибут (кількість незадовільних оцінок) надалі необхідно розділити цей домен значень атрибуту на групи, оскільки різниця між

студентом без боргів і з одним боргом більше, ніж між студентами з 18 і 17 боргами, де по суті це вже не грає принципового значення.

Було вирішено розподілити студентів на 3 групи залежно від кількості боргів:

- 1 група "успішно" - 0 боргів
- 2 група "борги" - від 1 до 6 боргів
- 3 група "багато боргів" - від 7 і більше (максимальні 18 боргів)

Для того, щоб було зручніше аналізувати цей критерій, був створений додатковий стовпець в датасете, де вказано до якої групи належить студент.

Розглянемо загальну кількість студентів в цих групах (див. рисунок 2.4):

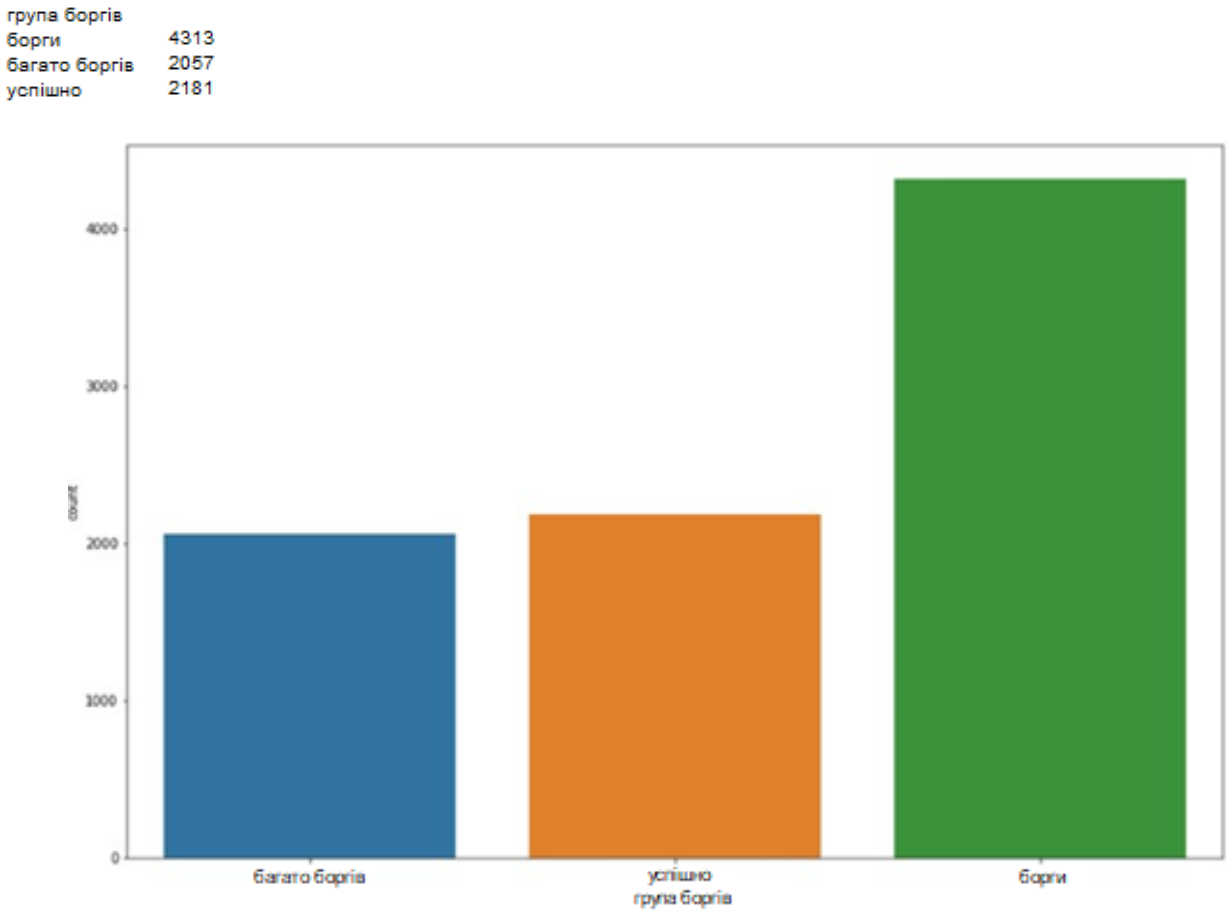


Рисунок 2.4 – Розподіл студентів по "групах боргів"

Розглянемо склад студентів по курсах (і кваліфікації) (див. рисунки 2.5 та 2.6).

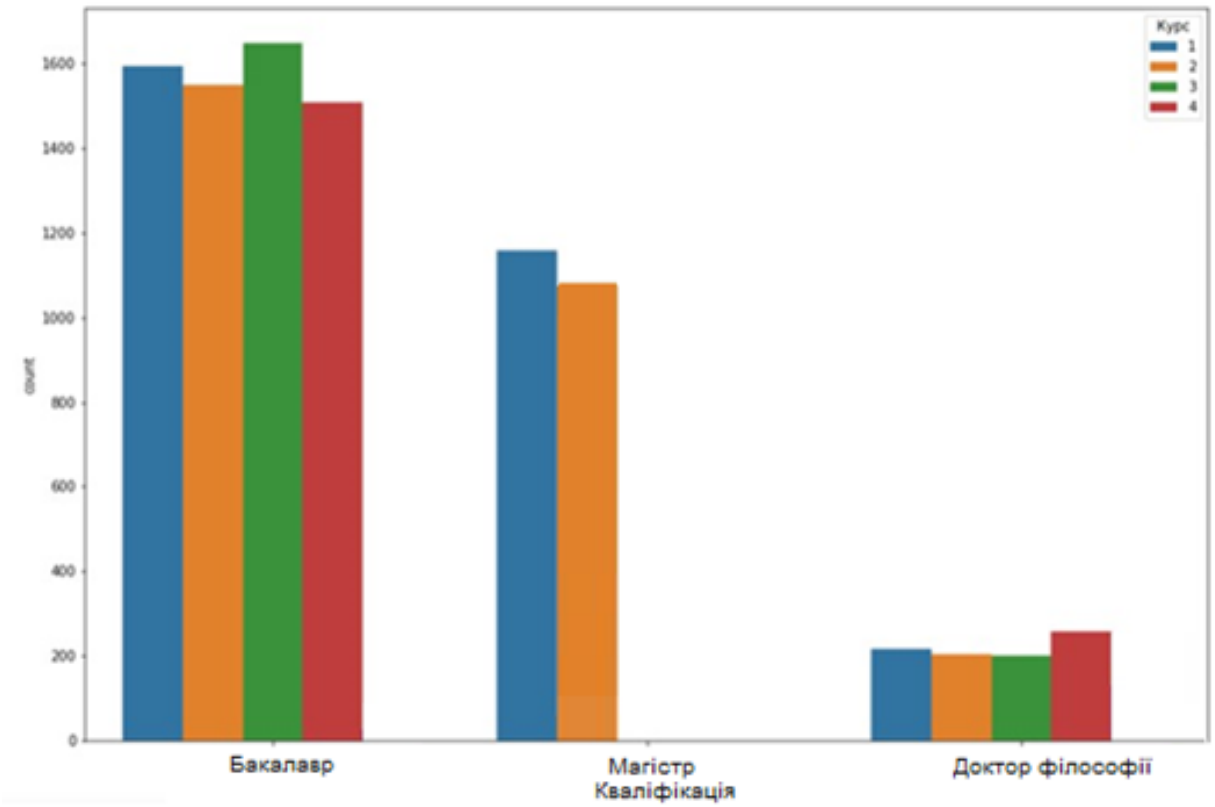


Рисунок 2.5 – Розподіл по кваліфікації і курсу навчання

Кваліфікація	Курс	
Бакалавр	1	1595
	2	1552
	3	1649
	4	1508
Магістр	1	1159
	2	1075
Доктор філософії	1	215
	2	203
	3	201
	4	257

Name: ID, dtype: int64

Рисунок 2.6 – Розподіл по кваліфікації і курсу навчання

Розглянемо групи залежно від форми навчання і кваліфікації (див. рисунок 2.7).

Форма навчання	Кваліфікація	
Заочна	Бакалавр	2090
	Доктор філософії	270
Денна	Бакалавр	4238
	Магістр	1145
	Доктор філософії	736
Заочно-дистанційна	Магістр	72

Name: ID, dtype: int64

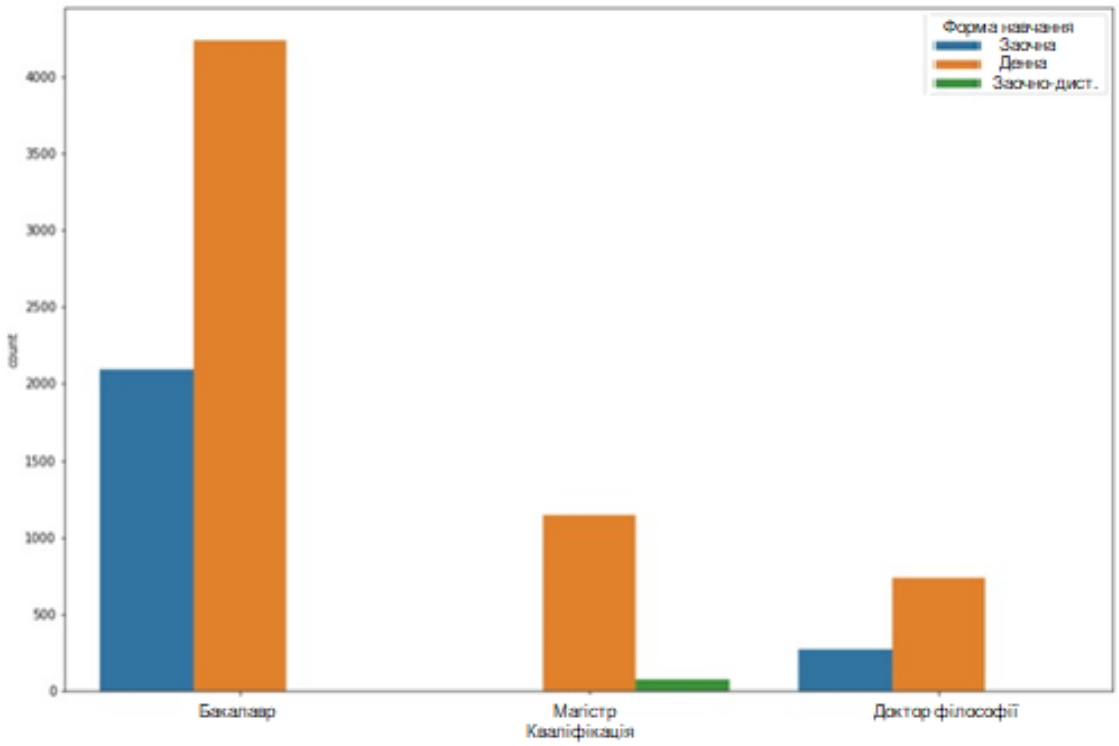


Рисунок 2.7 – Розподіл студентів по "групах боргів"

Як видно з графіку, порівняти форми навчання на предмет кореляції з кількістю незадовільних оцінок ми може тільки у бакалавраті і зрозуміти чи є різницю між очною і заочною формою навчання у відсотках.

Визначимо процентне співвідношення студентів серед цих 6 груп за параметрами успішності (див. рисунки 2.8-2.14).

Форма навчання	Кваліфікація	група боргів	
Заочна	Бакалавр	борги	853
		багато боргів	1004
		успішно	233
	Доктор філософії	борги	170
		багато боргів	38
		успішно	62
Денна	Бакалавр	борги	2117
		багато боргів	642
		успішно	1479
	Магістр	борги	691
		багато боргів	228
		успішно	226
	Доктор філософії	борги	434
		багато боргів	121
Заочно-дистанційна	Магістр	успішно	181
		борги	48
		багато боргів	24

Діаграма відсоткового співвідношення студентів з боргами і без для групи: Бакалавр, Очна форма навчання
група боргів
борги 2117
багато боргів 642
успішно 1479
Name: ID, dtype: int64

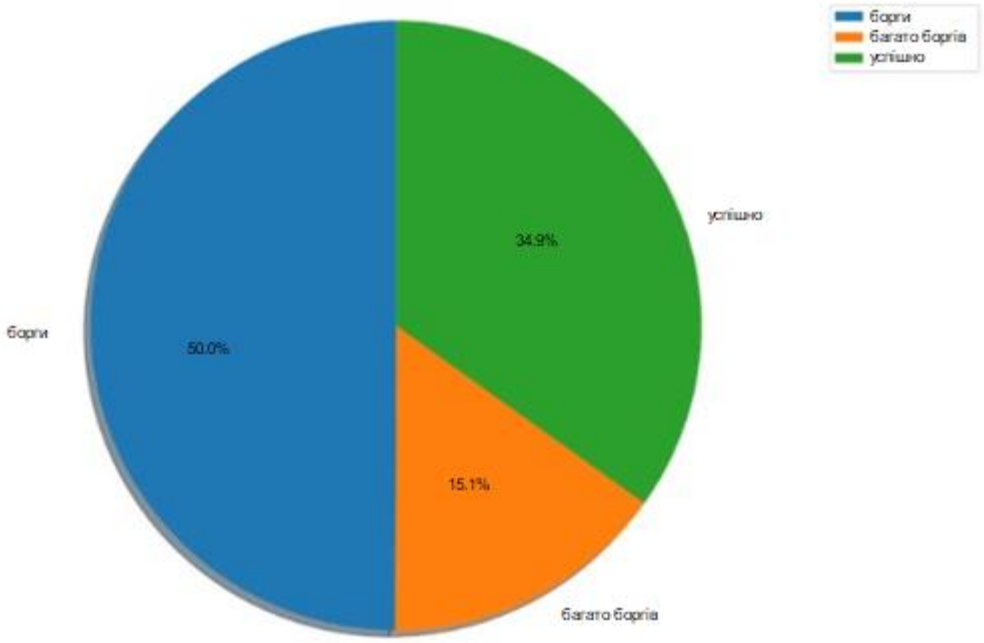


Рисунок 2.8 – Розподіл студентів по "групах боргів"

Діаграма відсоткового співвідношення студентів з боргами і без для групи: Бакалавр, Заочна форма навчання
група боргів
борги 853
багато боргів 1004
успішно 233
Name: ID, dtype: int64

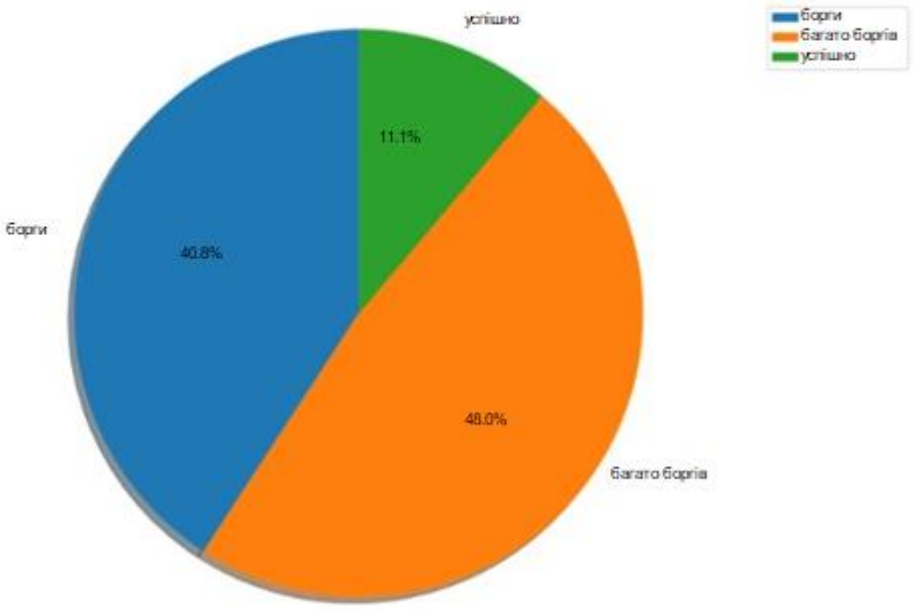


Рисунок 2.9 – Розподіл студентів по "групах боргів"

Діаграма відсоткового співвідношення студентів з боргами і без для групи: Доктор філософії, Очна форма навчання

група боргів
борги 434
багато боргів 121
успішно 181
Name: ID, dtype: int64

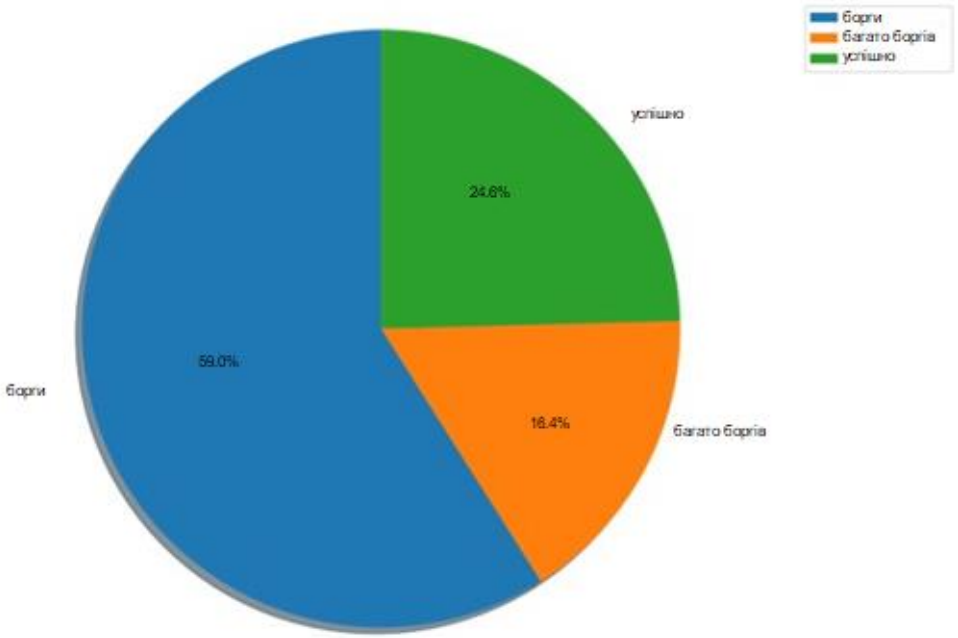


Рисунок 2.10 – Розподіл студентів по "групах боргів"

Діаграма відсоткового співвідношення студентів з боргами і без для групи: Доктор філософії, Заочна форма навчання

група боргів
борги 170
багато боргів 38
успішно 62
Name: ID, dtype: int64

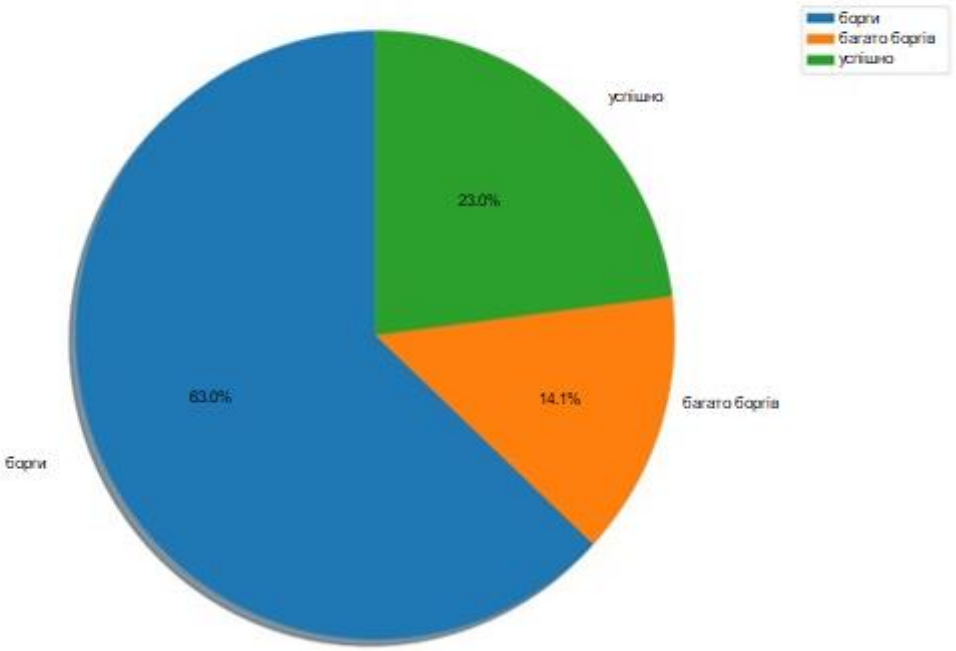


Рисунок 2.11 – Розподіл студентів по "групах боргів"

Діаграма відсоткового співвідношення студентів з боргами і без для групи: Магістр, Очна форма навчання
група боргів
борги 691
багато боргів 228
успішно 226
Name: ID, dtype: int64

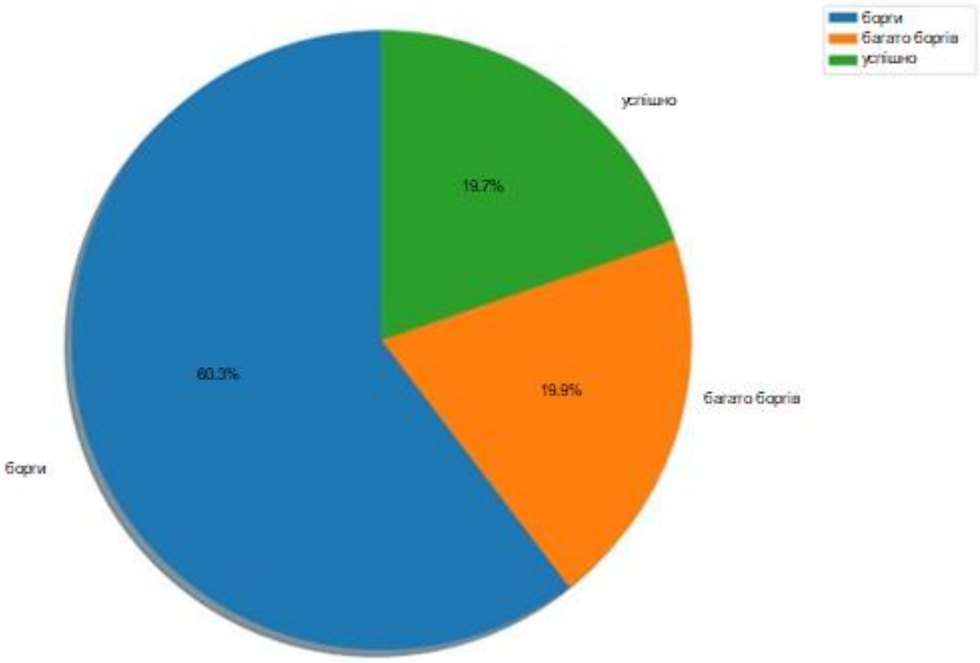


Рисунок 2.12 – Розподіл студентів по "групах боргів"

Діаграма відсоткового співвідношення студентів з боргами і без для групи: Магістр, Заочно-дистанційна форма навчання
група боргів
борги 48
багато боргів 24
Name: ID, dtype: int64

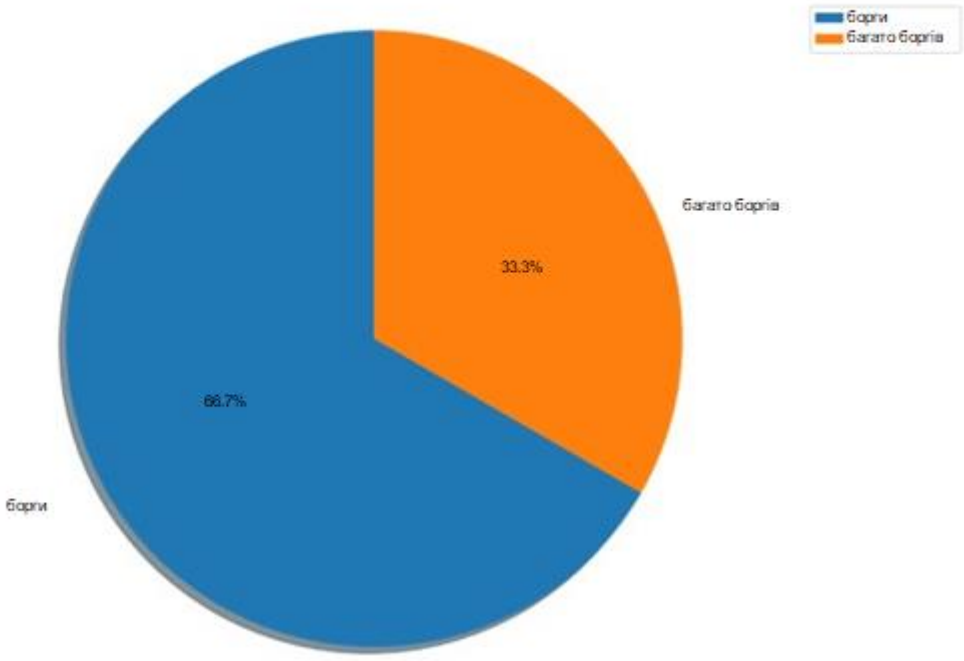


Рисунок 2.13 – Розподіл студентів по "групах боргів"

Стать	
Жіноча	2501
Чоловіча	6050

Name: ID, dtype: int64

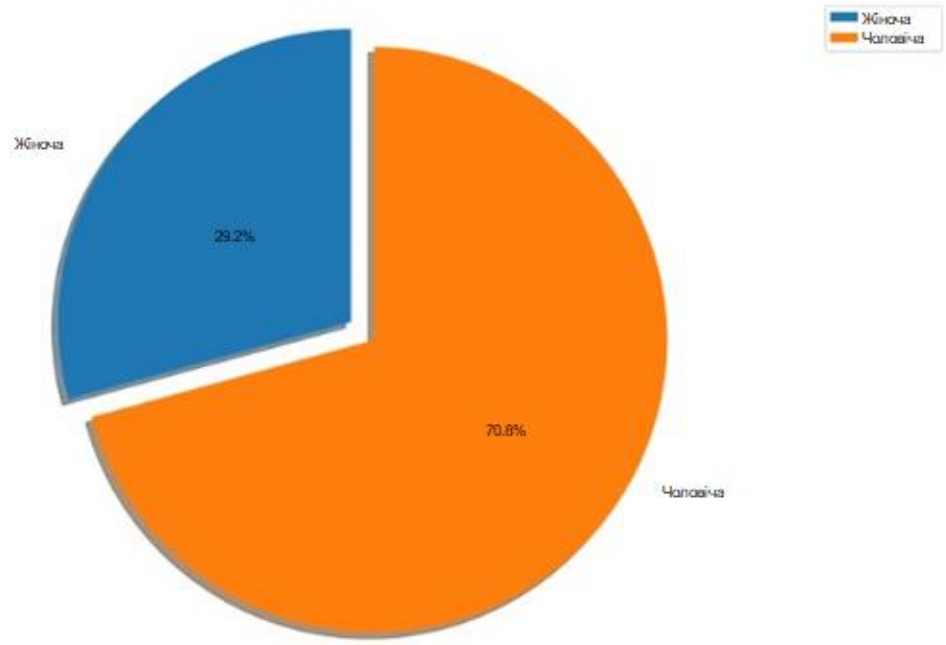


Рисунок 2.14 – Розподіл студентів за статевою ознакою

Випуск. факультет	група боргів	
Факультет автоматики і комп'ютеризованих технологій	борги	543
	багато боргів	309
	успішно	266
Факультет інформаційних радіотехнологій та технічного захисту інформації	борги	421
	багато боргів	181
	успішно	197
Факультет інформаційно-аналітичних технологій та менеджменту	борги	451
	багато боргів	201
	успішно	254
Факультет комп'ютерних наук	борги	1170
	багато боргів	463
	успішно	717
Факультет комп'ютерної інженерії та управління	борги	1029
	багато боргів	408
	успішно	499
Факультет інфокомунікацій	борги	507
	багато боргів	160
	успішно	210

Курс	група боргів	
1	борги	1645
	багато боргів	739
	успішно	585
2	борги	1166
	багато боргів	590
	успішно	57
3	борги	978
	багато боргів	513
	успішно	359
4	борги	464
	багато боргів	195
	успішно	1106
5	борги	60
	багато боргів	20
	успішно	74

Середня кількість незадовільних оцінок на факультетах

Випуск. факультет	
Факультет автоматики і комп'ютеризованих технологій	3.923214
Факультет інформаційних радіотехнологій та технічного захисту інформації	3.197747
Факультет інформаційно-аналітичних технологій та менеджменту	3.512141
Факультет комп'ютерних наук	3.407234
Факультет комп'ютерної інженерії та управління	3.264463
Факультет інфокомунікацій	3.319270

Name: Незадовільних, dtype: float64

Медіана незадовільних оцінок на факультетах

Випуск. факультет	
Факультет автоматики і комп'ютеризованих технологій	3
Факультет інформаційних радіотехнологій та технічного захисту інформації	2
Факультет інформаційно-аналітичних технологій та менеджменту	2
Факультет комп'ютерних наук	3
Факультет комп'ютерної інженерії та управління	2
Факультет інфокомунікацій	2

Name: Незадовільних, dtype: int64

Топ по середній кількості незадовільних оцінок на студента по різних спеціальностях.

Спеціальність	
121 – Інженерія програмного забезпечення	7.509009
122 – Комп'ютерні науки	7.111888
123 – Комп'ютерна інженерія	6.222222
125 – Кібербезпека та захист інформації	5.529412
174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка	5.323077
126 – Інформаційні системи та технології	5.000000
124 – Системний аналіз	4.812500
113 – Прикладна математика	4.769231
172 – Електронні комунікації та радіотехніка	4.755352
175 – Інформаційно-вимірювальні технології	4.649770
176 – Мікро- та наносистемна техніка	4.588235
186 – Видавництво та поліграфія	4.583333
171 – Електроніка	4.541667
173 – Авіоніка	4.377778
176 – Мікро- та наносистемна техніка	4.333333

Топ спеціальностей по медіані незадовільних оцінок.

Спеціальність	
121 – Інженерія програмного забезпечення	8.0
122 – Комп'ютерні науки	8.0
123 – Комп'ютерна інженерія	6.5
125 – Кібербезпека та захист інформації	6.0
174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка	5.5
126 – Інформаційні системи та технології	5.0
124 – Системний аналіз	5.0
113 – Прикладна математика	4.0
172 – Електронні комунікації та радіотехніка	4.0
175 – Інформаційно-вимірювальні технології	4.0
176 – Мікро- та наносистемна техніка	4.0
186 – Видавництво та поліграфія	4.0
171 – Електроніка	3.5
173 – Авіоніка	3.5
176 – Мікро- та наносистемна техніка	3.5

Також з поточного файлу були отримані дані про кількість незадовільних оцінок за кожною дисципліною по усьому університету.

Назва дисципліни	Кількість боржників
Українське фахове мовлення (залік)	1860
Іноземна мова (залік)	953
Філософія (іспит)	718
Основи права (залік)	685
Фізичне виховання (залік)	668
Вища математика (іспит)	643
Фізика (іспит)	591
Економіка та бізнес (іспит)	560
Комп'ютерна дискретна математика (іспит)	524
Основи програмування (іспит)	515
Об'єктно-орієнтоване програмування (іспит)	488
Алгоритми та структури даних (іспит)	436
Архітектура комп'ютера та комп'ютерних мереж (іспит)	423
Операційні системи (іспит)	395
Архітектура програмного забезпечення (іспит)	388

Кількість боржників по предметах за фахом

121 – Інженерія програмного забезпечення:

Назва дисципліни	Кількість боржників
Українське фахове мовлення (залік)	125
Комп'ютерна дискретна математика (іспит)	62
Алгоритми та структури даних (іспит)	58
Архітектура комп'ютера та комп'ютерних мереж (іспит)	57
Операційні системи (іспит)	57
Архітектура програмного забезпечення (іспит)	57
Комплексний курсовий проект (КП)	56
Теорія ймовірностей та математична статистика (іспит)	56
Менеджмент проектів програмного забезпечення (залік)	56
Високорівневі мови програмування та фреймворки (залік)	51
Безпека програм та даних (іспит)	51
Мультимедіа-системи (іспит)	49
Якість програмного забезпечення та тестування (іспит)	49
Введення до IT-бізнесу (залік)	49
Емпіричні методи програмної інженерії (іспит)	49

Як видно з таблиці, предмети не дуже показові, оскільки, наприклад, предмет Українське фахове мовлення зустрічається на всіх факультетах та за усіма спеціальностями, тому логічно що по ньому найбільше боргів.

3. МОДЕЛЬ МАШИННОГО НАВЧАННЯ

Таким чином, після аналізу усіх параметрів, були виявлені наступні, які можуть більшою мірою впливати на успішність студента, а саме:

1. Форма навчання
2. Кваліфікація
3. Курс
4. Спеціальність
5. Академічна відпустка (що діє) – так / ні
6. Всього годин пропусків в семестрі
7. Всього годин аудиторного зайняття в семестрі

Ці параметри будуть використані в машинному навчанні з учителем з метою класифікації студента на предмет кількості заборгованостей.

Розглянемо кореляції між вибраними параметрами.

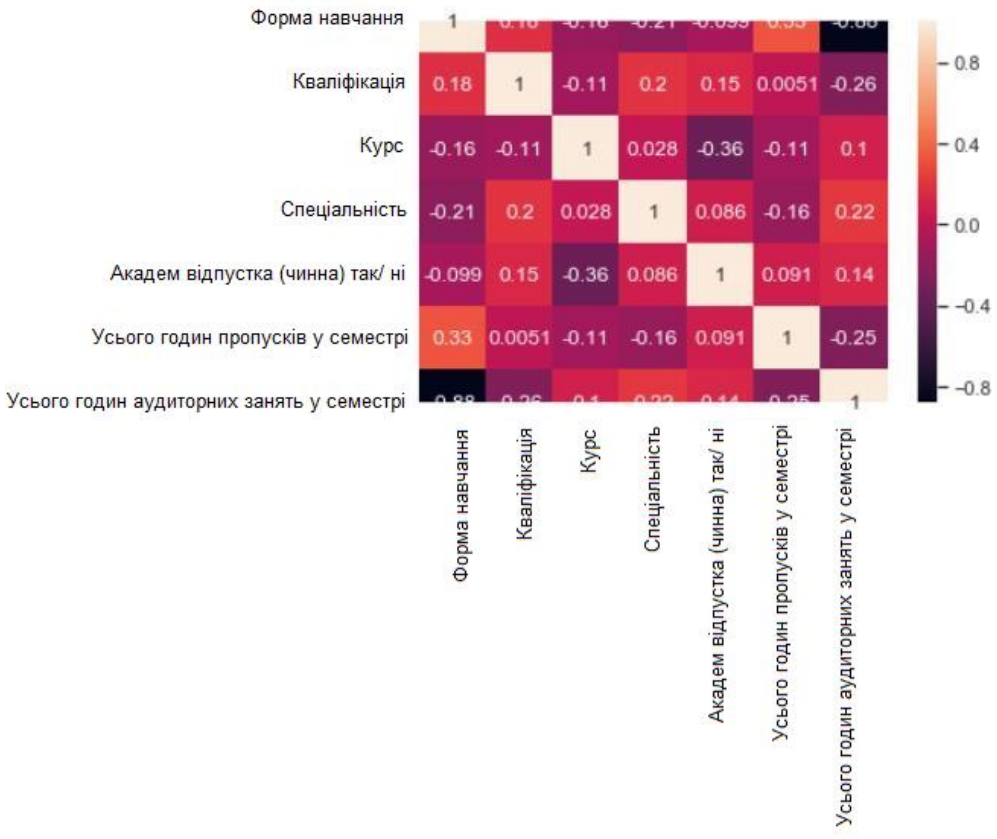


Рисунок 3.1 – Кореляція ознак

Для того, щоб приступити до навчання моделі, нам необхідно перетворити усі параметри в числові представлення. Для цього буде використана вже наявна в Python функція LabelEncoder з бібліотеки sklearn.preprocessing.

```
X = students_df[['Форма навчання', 'Кваліфікація', 'Курс', 'Спеціальність',
        'Академ відпустка (чинна) так / ні', 'Усього годин пропусків у семестрі',
       'Усього годин аудиторних занять у семестрі']]
Y = students_df['Група боргів']
```

```
from sklearn.preprocessing import LabelEncoder
le_f = LabelEncoder()
X['Форма навчання'] = le_f.fit_transform(X['Форма навчання'].values)
le_k = LabelEncoder()
X['Кваліфікація'] = le_k.fit_transform(X['Кваліфікація'].values)
le_s = LabelEncoder()
X['Спеціальність'] = le_s.fit_transform(X['Спеціальність'].values)
le_a = LabelEncoder()
X['Академ відпустка (чинна) так / ні'] = le_a.fit_transform(X['Академ відпустка (чинна) так / ні '].values)
```

Після перетворень ми отримуємо наступне: X – це датафрейм усіх параметрів, вже перетворений в числове значення, Y - це мітки.

```
X.head()
```

	Форма навчання	Кваліфікація	Курс	Спеціальність	Академ відпустка (чинна) так / ні	Усього годин пропусків у семестрі	Усього годин аудиторних занять у семестрі
0	0	0	4	64	1	0	1400.0
1	1	0	4	16	0	16	440.0
2	1	0	1	23	1	364	464.0
3	1	0	1	47	1	2	464.0
4	1	1	1	27	0	0	384.0

Для того, щоб приступити до створення моделі поділимо наші дані на повчальну і тестову вибірки.

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.3, random_state=0)
```

Перший класифікатор, який буде використаний це логістична регресія.

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression(C=1000.0, random_state=0 )
lr.fit(x_train,y_train)
```

```
LogisticRegression(C=1000.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=100,
                   multi_class='warn', n_jobs=None, penalty='l2',
                   random_state=0, solver='warn', tol=0.0001, verbose=0,
                   warm_start=False)
```

```
print(lr.score(x_train, y_train))
print(lr.score(x_test, y_test))
```

```
0.6668915500084331
0.6336088154269972
```

```
pred_y = lr.predict(x_test)
from sklearn import metrics
# Model Accuracy, how often is the classifier correct?
print(metrics.classification_report(pred_y, y_test))
```

```
              precision    recall  f1-score   support

       борги       0.84      0.60      0.70      1732
 багато боргів       0.48      0.74      0.58       433
      успішно       0.40      0.69      0.51       376

    accuracy                           0.63      2541
   macro avg       0.57      0.67      0.59      2541
weighted avg       0.71      0.63      0.65      2541
```

Дані метод показав, що певна залежність є і параметри вибрані вірно. Далі ми спробуємо інші види класифікації і порівняємо результати.

Застосуємо метод опорних векторів.

```
from sklearn import svm
clf = svm.SVC()
clf.fit(x_train,y_train)
```

```
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
    kernel='rbf', max_iter=-1, probability=False, random_state=None,
    shrinking=True, tol=0.001, verbose=False)
```

```
print(clf.score(x_train, y_train))
print(clf.score(x_test, y_test))
```

```
0.8794063079777366
0.7197953561589925
```

```
pred_y = clf.predict(x_test)
from sklearn import metrics
# Model Accuracy, how often is the classifier correct?
print(metrics.classification_report(pred_y, y_test))
```

```
              precision    recall  f1-score   support

       борги       0.86      0.68      0.76      1564
 багато боргів       0.54      0.85      0.66       421
      успішно       0.64      0.74      0.69       556

    accuracy                           0.72      2541
   macro avg       0.68      0.76      0.70      2541
weighted avg       0.76      0.72      0.73      2541
```

Тут ми можемо бачити, що метод опорних векторів схильний до перенавчання, оскільки досить великий розкид в 10 відсотків між тренувальним набором і тестовим у визначенні міток. Проте результат на тестовій вибірці майже на 10 відсотків вище, ніж у класифікатора на основі логістичної регресії.

Застосовуємо 3-й класифікатор "випадковий ліс"

```
from sklearn.ensemble import RandomForestClassifier
clas = RandomForestClassifier(max_depth=15, random_state=0)
clas.fit(x_train,y_train)
```

```
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                       max_depth=15, max_features='auto', max_leaf_nodes=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, n_estimators=10,
                       n_jobs=None, oob_score=False, random_state=0, verbose=0,
                       warm_start=False)
```

```
print(clas.feature_importances_)
```

```
[0.01858244 0.03035815 0.12210839 0.2179611  0.07945399 0.31382953
 0.21770641]
```

```
print(clas.score(x_train, y_train))
print(clas.score(x_test, y_test))
```

```
0.8775510204081632
0.7898465171192444
```

```
pred_y = clas.predict(x_test)
from sklearn import metrics
# Model Accuracy, how often is the classifier correct?
print(metrics.classification_report(pred_y, y_test))
```

```
              precision    recall  f1-score   support

       борги       0.82      0.78      0.80      1311
 багато боргів       0.74      0.85      0.79       579
     успішно       0.77      0.76      0.77       651

    accuracy                           0.79      2541
   macro avg       0.78      0.80      0.79      2541
weighted avg       0.79      0.79      0.79      2541
```

Випадковий ліс показав найкращий результат, хоча розкид між повчальною і тестовою вибіркою досить великий (10 точність прогнозу тестової вибірки складає 79 відсотків. Це досить високий показник точності. Так само були випробуваний випадковий ліс з різною максимальною глибиною і в результаті досвіду було виявлено, що глибина більше 15, не дає істотного результату визначення тестової вибірки, що впливає на точність.

Таким чином можна зробити висновок, що в першу чергу завдання за визначенням успішності студента на підставі таких параметрів як:

- Форма навчання
- Кваліфікація
- Курс
- Спеціальність
- Академ. відпустка (що діє) – та / ні
- Усього годин пропусків в семестрі
- Усього годин аудиторних зайнять в семестрі

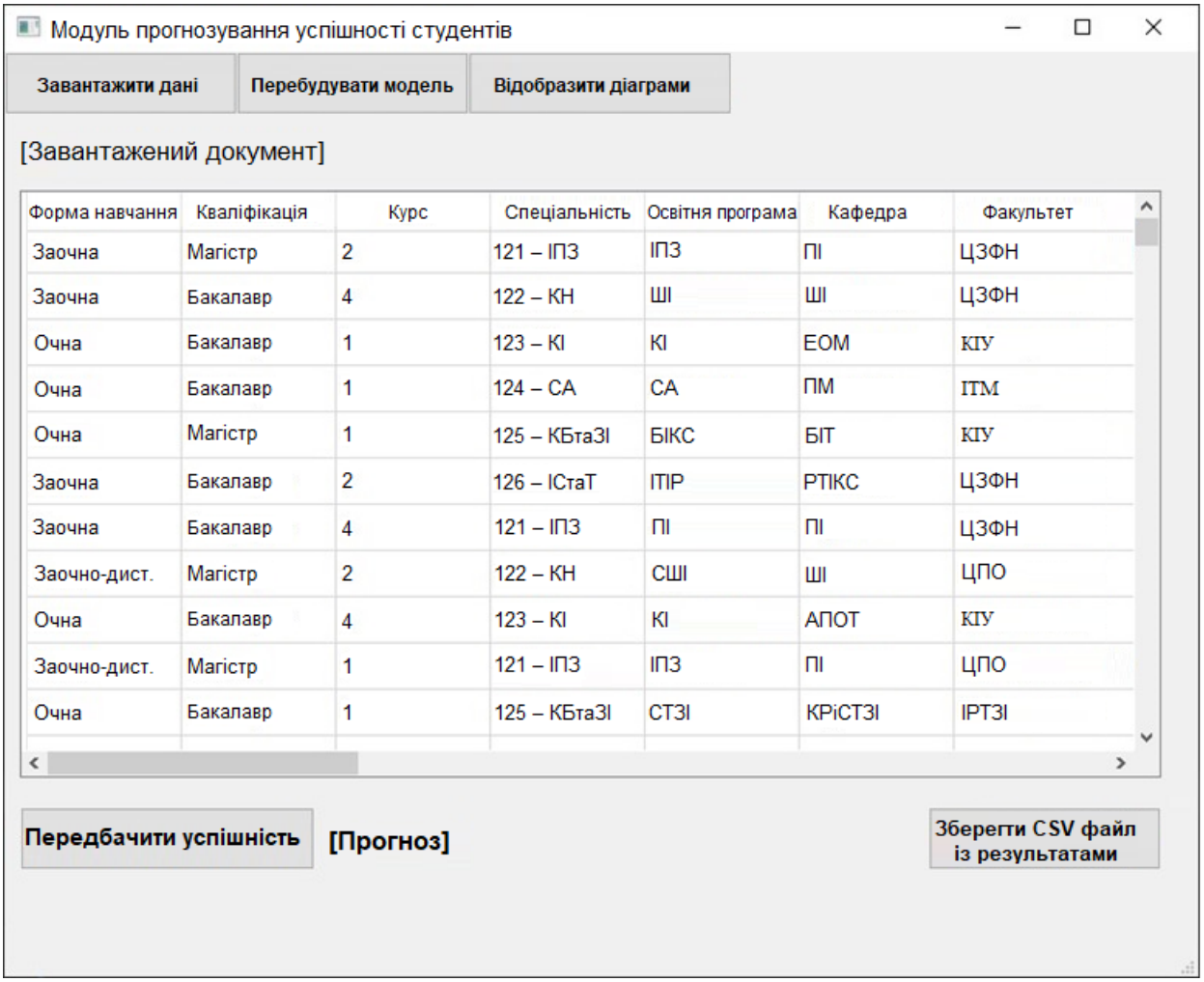
Також в результаті аналізу було виявлено, що найбільш значимий вклад в модель вносять 3 параметри, а саме:

- Усього годин пропусків в семестрі
- Усього годин аудиторних зайнять в семестрі
- Курс

Надалі точність моделі також можуть підвищити додаткові параметри, наприклад, хобі студента, участь в соціальному житті університету і так далі.

4. ГРАФІЧНИЙ ІНТЕРФЕЙС КОРИСТУВАЧА ПРОГНОЗНОЇ МОДЕЛІ

Був створений графічний інтерфейс для майбутнього модуля прогнозування в системі, який дозволяє завантажувати інформація про студентів у форматі файлу excel і відображати його на екрані. Вбудована модель прогнозування дозволяє зробити оцінку кількості академічних заборгованостей на кінець семестру для конкретного студента або для усіх відразу і вивести її на екран. Також модуль дозволяє додати дані до вже існуючої моделі, для того, щоб перебудувати її і підвищити її прогнозну точність. На рисунку 4.1 зображене головне меню модуля.



Форма навчання	Кваліфікація	Курс	Спеціальність	Освітня програма	Кафедра	Факультет
Заочна	Магістр	2	121 – ІПЗ	ІПЗ	ПІ	ЦЗФН
Заочна	Бакалавр	4	122 – КН	ШІ	ШІ	ЦЗФН
Очна	Бакалавр	1	123 – КІ	КІ	ЕОМ	КІУ
Очна	Бакалавр	1	124 – СА	СА	ПМ	ІТМ
Очна	Магістр	1	125 – КБтаЗІ	БІКС	БІТ	КІУ
Заочна	Бакалавр	2	126 – ІСтаТ	ІТІР	РТІКС	ЦЗФН
Заочна	Бакалавр	4	121 – ІПЗ	ПІ	ПІ	ЦЗФН
Заочно-дист.	Магістр	2	122 – КН	СШІ	ШІ	ЦПО
Очна	Бакалавр	4	123 – КІ	КІ	АПОТ	КІУ
Заочно-дист.	Магістр	1	121 – ІПЗ	ІПЗ	ПІ	ЦПО
Очна	Бакалавр	1	125 – КБтаЗІ	СТЗІ	КРіСТЗІ	ІРТЗІ

Рисунок 4.1 – Головне меню модуля

Після натиснення кнопки завантажити файл, відкривається провідник і через нього слід вибрати файл, який необхідно додати (див. рисунок 4.2).

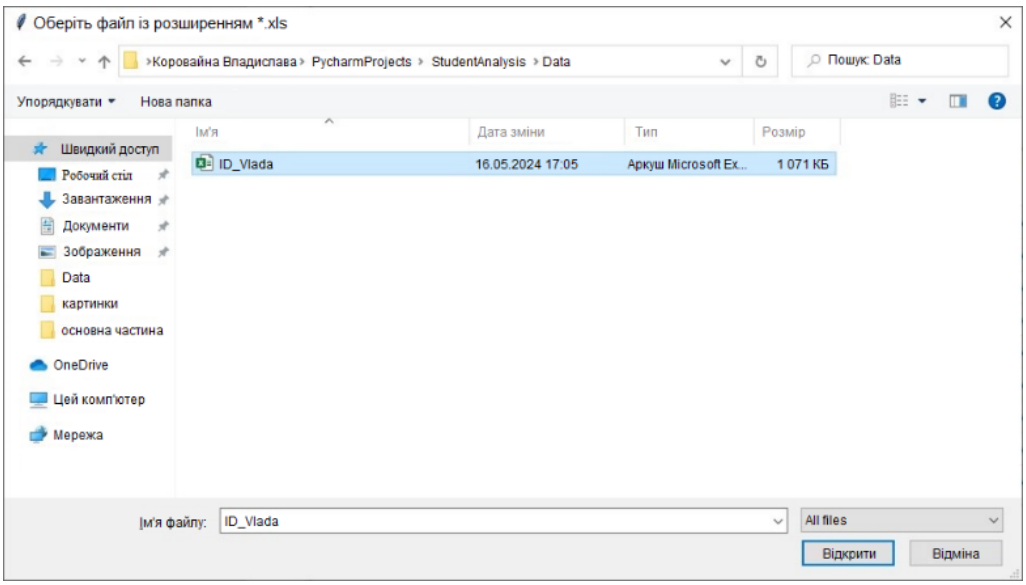


Рисунок 4.2 – Вікно вибору файлу для завантаження

Після цього розташування і назва файлу відобразиться вгорі. Після натиснення кнопки побудувати модель, модель машинного навчання перебудовується у відповідність з доданими даними (див. рисунок 4.3).

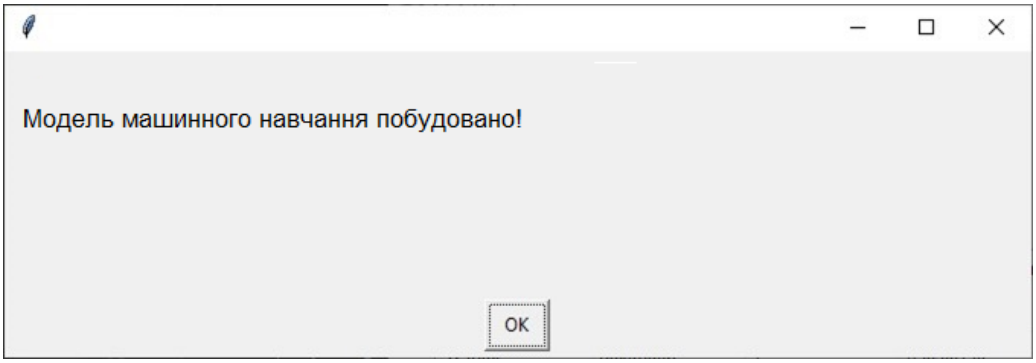


Рисунок 4.3 – Повідомлення про успішну побудову моделі

Також у вікні основного меню можна бачити кнопку "Робота з графіками". Натиснувши на неї, відкриється додаткове вікно. У цьому вікні існує можливість вибрати тип графіку і дані, по яких ми хочемо його побудувати (див. рисунок 4.4).

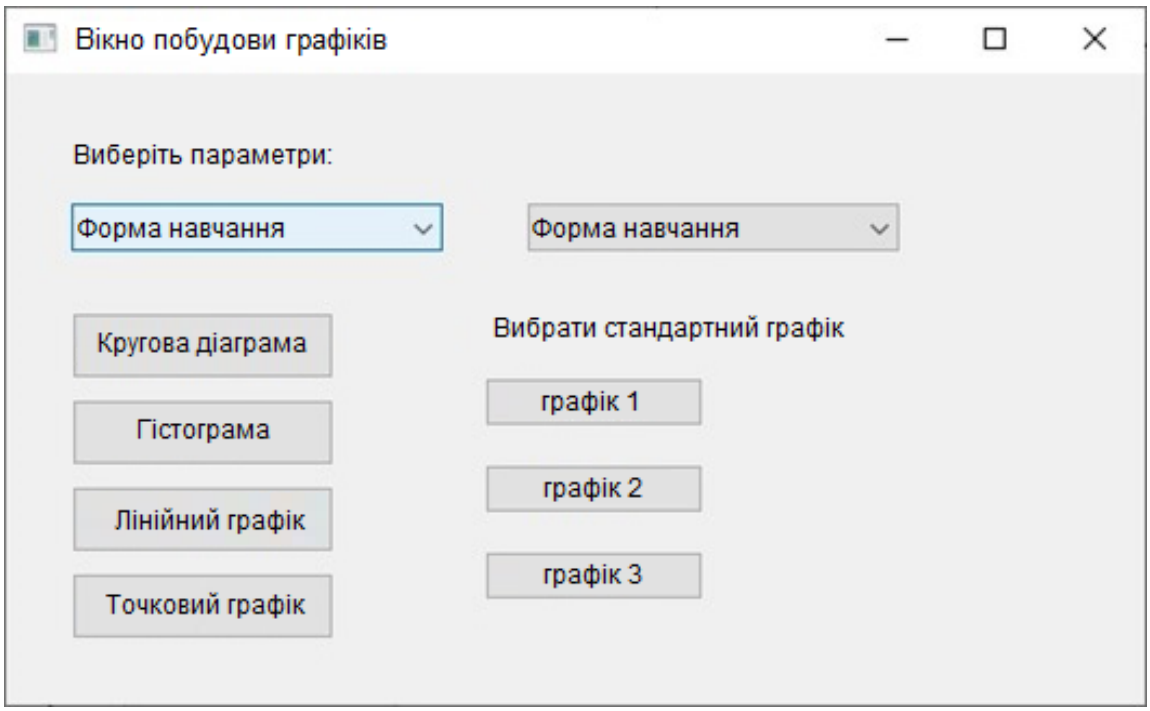


Рисунок 4.4 – Вікно побудови графіків

Параметри для побудови графіків можна вибрати з випадного списку (див. рисунок 4.5).

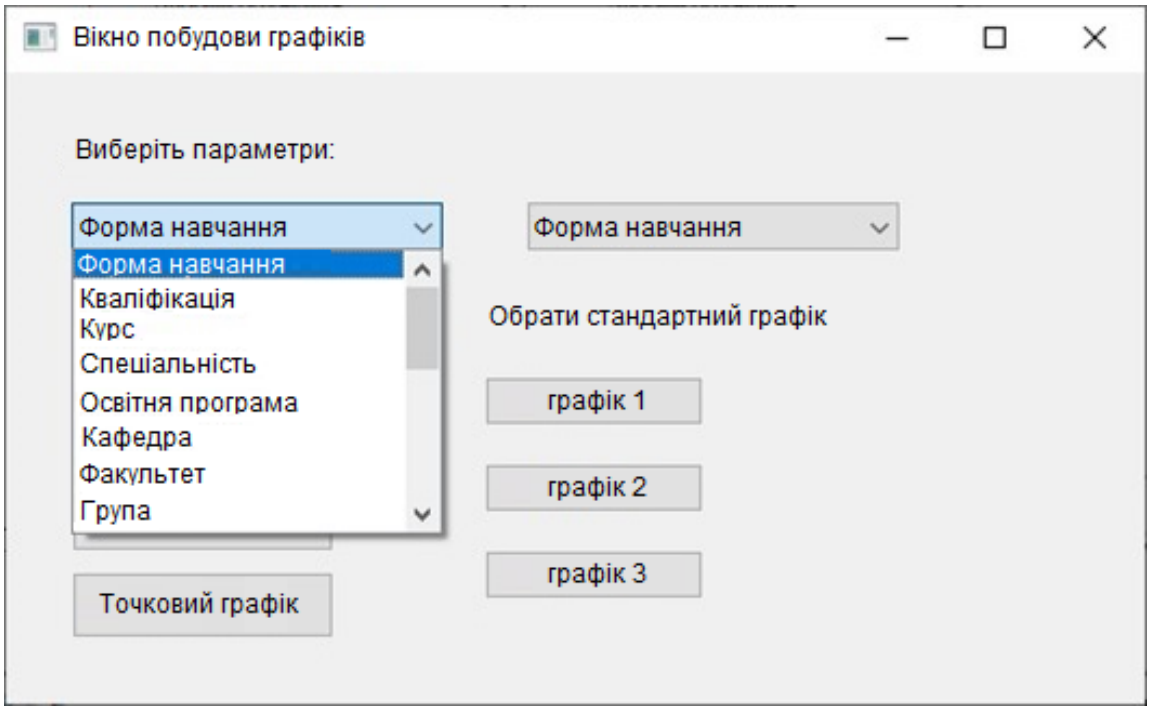


Рисунок 4.5 – Вікно побудови графіків

Наприклад, в першому випадку, була вибрана кругова діаграма з даними про форму навчання. У другому випадку був вибрана гістограма і якості даних це курс (див. рисунок 4.6).

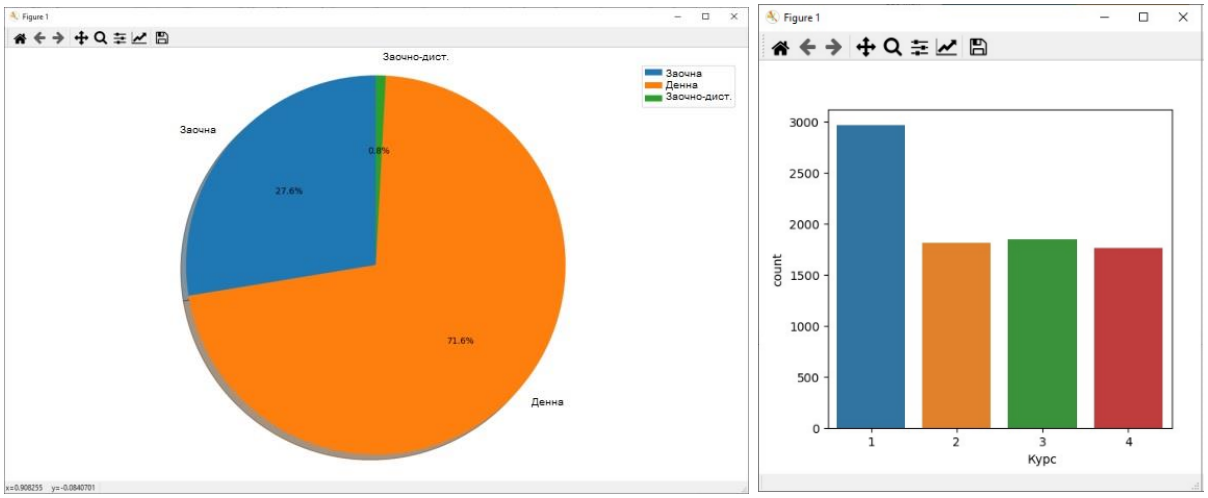


Рисунок 4.6 – Приклади графіків

Так само існує опція побудови складніших графіків, які є за умовчанням. Наприклад, перший графік - це гістограма розподілу студентів по кваліфікації і курсу (див. рисунок 4.7).

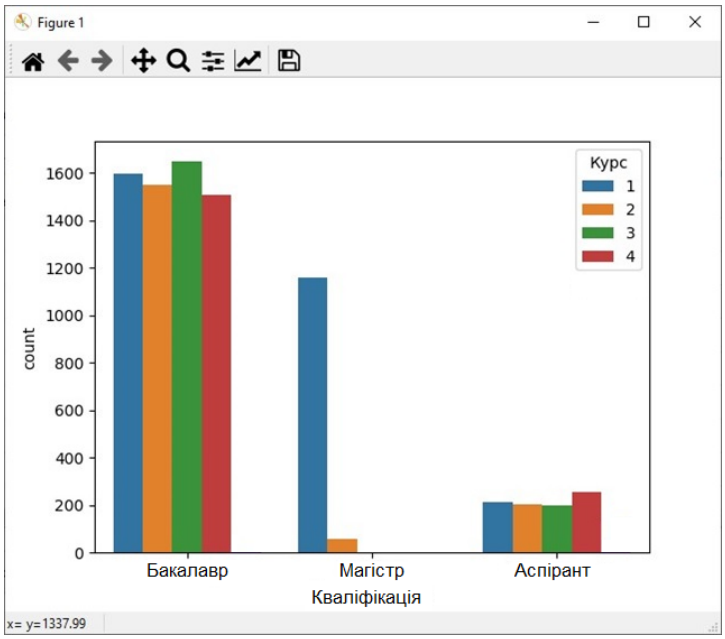


Рисунок 4.7 – Приклади графіків

Ця можливість побудови графіків дозволяє зробити первинну аналітику даних і вивчити деякі тренди.

Оскільки основна функція даної системи це прогноз, то ми можемо вибрати конкретного студента і подивитися пророцтво для нього (див. рисунки 4.8 та 4.9).

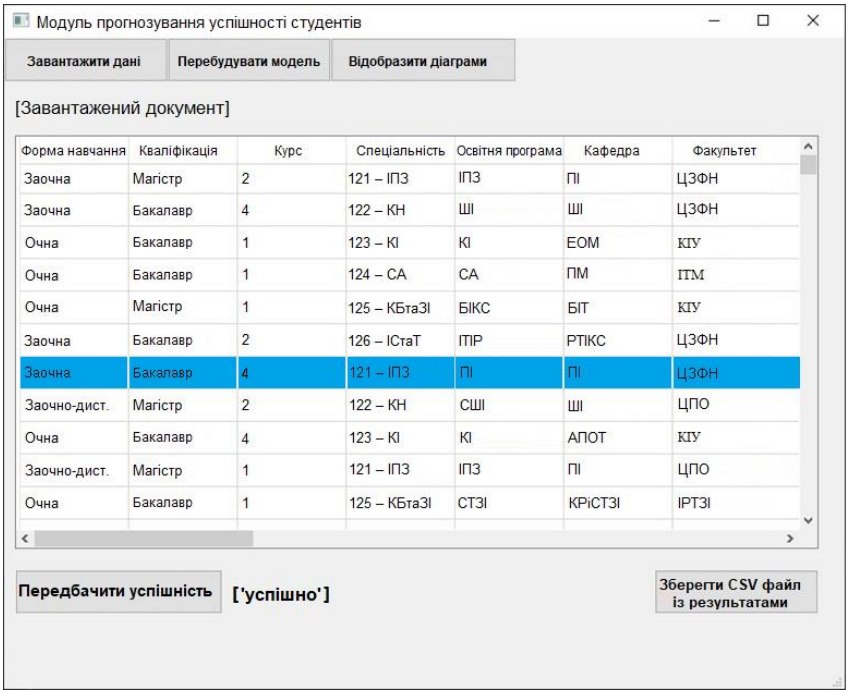


Рисунок 4.8 – Головне меню модуля (з прогнозом ‘успішно’)

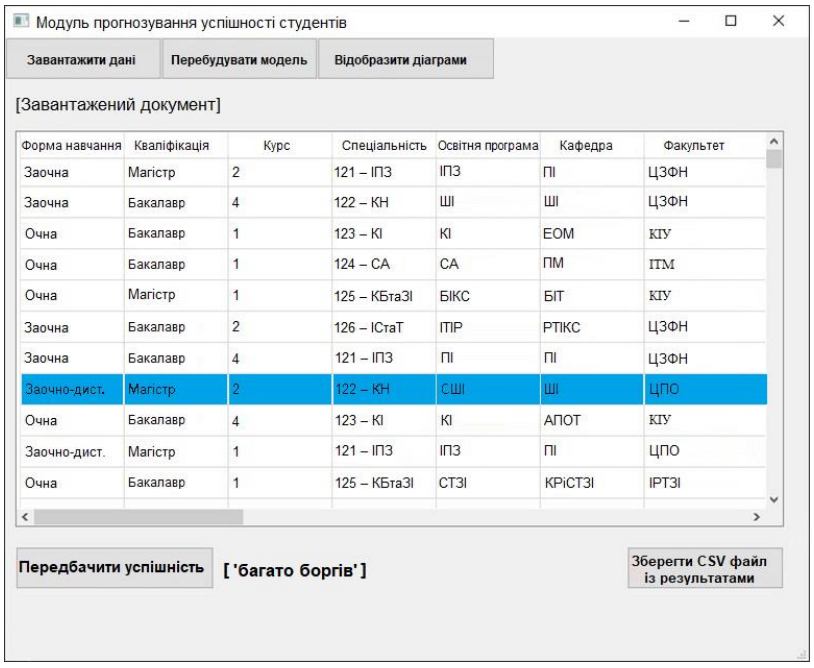


Рисунок 4.9 – Головне меню модуля (з прогнозом ‘багато боргів‘)

ВИСНОВКИ

В результаті виконання науково-дослідної практики були виконані наступні завдання:

1. Проаналізовані методи, які використовуються для вирішення аналогічних завдань у сфері освіти.

2. Проведена попередня обробка даних.

3. Побудована модель машинного навчання, здатна прогнозувати успішність студентів на кінець семестру.

4. Виявлені найбільш значимі ознаки у визначенні успішності студентів.

5. Створений графічний інтерфейс користувача прогнозної моделі успішності студентів.

Оскільки при попередній обробці даних були використані мітки для позначення "успішності студента", то тип машинного навчання був з учителем. Були апробовані 3 моделі машинного навчання з учителем, а саме: логістична регресія, метод опорних векторів і випадковий ліс. Найкращий результат на тестовій вибірці показав класифікатор "випадковий ліс". Цей алгоритм є оптимальним так, як має наступні достоїнства.

Окрім цього, був створений графічний інтерфейс для модуля по прогнозуванню успішності студентів на кінець семестру за поточними оцінками.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. В.О. Шевченко. Прогнозування успішності студентів на основі методів кластерного аналізу // Вісник ХНАДУ. 2015. №68.

2. В.О. Шевченко. Інформаційна технологія формування індивідуальних траєкторій самостійної роботи студентів // Вісник НТУ "ХПІ» 2015. №21(1130), 76. URL: https://repository.kpi.kharkov.ua/server/api/core/bitstreams/1a0c6ce0-fa4f-4ec4-a686-297774e2a33e/content (дата звернення: 15.05.2024).

3. В.О. Шевченко, А.І. Алгоритмізація процедури кластерного аналізу для прогнозування успішності студентів // Автомобіль і електроніка. Сучасні технології, 11/2017. URL: https://api.dspace.khadi.kharkov.ua/server/api/core/bitstreams/bed05a02-d9d0-438d-b50c-60106e8591c7/content (дата звернення: 15.05.2024).

4. O. Haitan, O. Nazarov. Hybrid approach to solving of the automated timetabling problem in higher educational institution // Системи управління, навігації та зв'язку, 2020, випуск 2(60). – С. 60-69. URL: http://journals.nupp.edu.ua/sunz/article/view/ 1833/1509 (дата звернення: 15.05.2024).

5. Al-Shehri H. et al. Student performance prediction using support vector machine and k-nearest neighbor // 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE). – IEEE, 2017. – С. 1-4. URL: https://ieeexplore.ieee.org/ document/7946847 (дата звернення: 15.05.2024).

6. Breazley, D. Python Cookbook, Third Edition / D. Breasley, B. K. Jones. – USA: O'Reilly Media, 2013. – 688 p.

7. McKinney, W. Python for Data Analysis. – USA: O'Reilly Media, 2013. – 453 p.

8. Y. Chen, T. Krishna, J. Emer, and V. Sze. Eyeriss: An energy-efficient recon-figurable accelerator for deep convolutional neural networks // IEEE Journal of Solid-State Circuits, 52(1), pp. 127-138, 2017. URL: https://ieeexplore.ieee.org/ document/7738524 (дата звернення: 15.05.2024).

9. D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick. Neuroscience-inspired artificial intelligence // Neuron, 95(2), pp. 245-258, 2017. URL: https://www.cell.com/ neuron/pdf/S0896-6273(17)30509-3.pdf (дата звернення: 15.05.2024).

10. I. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. Courville, and Y. Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues // AAAI, pp. 3295-3301, 2017. URL: https://arxiv.org/abs/1605.06069 (дата звернення: 15.05.2024).

11. C. Szegedy, S. loffe, V. Vanhoucke, and A. Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Leaming // AAAI Conference, pp. 4278-4284, 2017. URL: https://arxiv.org/abs/1602.07261 (дата звернення: 15.05.2024).

12. C. Wu, A. Ahrned, A. Beutel, A. Smola, and H. Jing. Recurrent recommender networks // ACM International Conference on Web Search and Data Mining, pp. 495-503, 2017. URL: https://dl.acm.org/doi/10.1145/3018661.3018689 (дата звернення: 15.05.2024).

13. L. Yu, W. Zhang, J. Wang, and Y. Yu. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient // AAAI Conference, pp. 2852-2858, 2017. URL: https://arxiv.org/abs/1609.05473 (дата звернення: 15.05.2024).

14. The Methods for the Prediction of Climate Control Indicators in the Internet of Things Systems / Zolotukhin, O., Filatov, V., Yerokhin, A., Kudryavtseva, M. // CEUR Workshop Proceedings, 2021, 3013, pp. 391–400. URL: https://ceur-ws.org/Vol-3013/20210391.pdf (дата звернення: 15.05.2024).

15. Study of Prediction and Classification Models in the Problems of Diabetes Among Patients with a Stroke in Different Living Conditions / Huliiev, N., Peretiaha, M., Khovrat, A., Teslenko, D., Nazarov, A. // Innovative Technologies and Scientific Solutions for Industries, (2 (24)), 54-61. URL: https://journals.uran.ua/itssi/article/view/285501/279567 (дата звернення: 15.05.2024).

16. Selection of Artificial Neural Networks for Disease Prediction / Iryna Kyrychenko, Oleksii Nazarov, Nural Huliiev and Oleksii Avdieiev // COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems, April 20–21, 2023, Kharkiv, Ukraine (CEUR Workshop Proceedings (CEUR-WS.org)). Volume I: Machine Learning Workshop, pp. 236–248. URL: https://ceur-ws.org/Vol-3387/paper18.pdf (дата звернення: 15.05.2024).

17. Building of Regression Models for Cryptocurrency Price Prediction / Smelyakov, K., Bizkrovnyi, O., Sharonova, N., Smelyakov, S., Chupryna, A. // Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Systems (COLINS), Volume I: Main Conference, 2022. In CEUR Workshop Proceedings, Vol-3171, 2022, pp. 1216-1232. URL: https://ceur-ws.org/Vol-3171/paper90.pdf (дата звернення: 15.05.2024).

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ЗА НАУКОВИМИ НАПРЯМАМИ КЕРІВНИКА ТА НАУКОВЦІВ КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ

4. O. Haitan, O. Nazarov. Hybrid approach to solving of the automated timetabling problem in higher educational institution // Системи управління, навігації та зв'язку, 2020, випуск 2(60). – С. 60-69. URL: http://journals.nupp.edu.ua/sunz/article/view/ 1833/1509 (дата звернення: 15.05.2024).

14. The Methods for the Prediction of Climate Control Indicators in the Internet of Things Systems / Zolotukhin, O., Filatov, V., Yerokhin, A., Kudryavtseva, M. // CEUR Workshop Proceedings, 2021, 3013, pp. 391–400. URL: https://ceur-ws.org/Vol-3013/20210391.pdf (дата звернення: 15.05.2024).

15. Study of Prediction and Classification Models in the Problems of Diabetes Among Patients with a Stroke in Different Living Conditions / Huliiev, N., Peretiaha, M., Khovrat, A., Teslenko, D., Nazarov, A. // Innovative Technologies and Scientific Solutions for Industries, (2 (24)), 54-61. URL: https://journals.uran.ua/itssi/article/view/ 285501/279567 (дата звернення: 15.05.2024).

16. Selection of Artificial Neural Networks for Disease Prediction / Iryna Kyrychenko, Oleksii Nazarov, Nural Huliiev and Oleksii Avdieiev // COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems, April 20–21, 2023, Kharkiv, Ukraine (CEUR Workshop Proceedings (CEUR-WS.org)). Volume I: Machine Learning Workshop, pp. 236–248. URL: https://ceur-ws.org/Vol-3387/paper18.pdf (дата звернення: 15.05.2024).

17. Building of Regression Models for Cryptocurrency Price Prediction / Smelyakov, K., Bizkrovnyi, O., Sharonova, N., Smelyakov, S., Chupryna, A. // Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Systems (COLINS), Volume I: Main Conference, 2022. In CEUR Workshop Proceedings, Vol-3171, 2022, pp. 1216-1232. URL: https://ceur-ws.org/Vol-3171/ paper90.pdf (дата звернення: 15.05.2024).