Прикарпатський національний університет імені Василя Стефаника

Факультет математики та інформатики

Кафедра математичного та функціонального аналізу

Дипломна робота на тему:

“ Статистичний аналіз розподілу місць державного замовлення підготовки фахівців в університеті**”**

Студента 4 курсу, групи М - 41 спеціальності : 111 Математика

Вовк Б. М.

«___» _____ 2025 р.

Керівник професор кафедри мат. фун. аналізу доктор фіз-мат наук Осипчук М.М

Національна шкала: _____

Університетська шкала: _____ Оцінка ECTS: ____

Члени комісії:

_____

(прізвище та ініціали)

_____

(підпис)

_____

(прізвище та ініціали)

_____

(підпис)

_____

(прізвище та ініціали)

_____

(підпис)

м. Івано-Франківськ

2024

Зміст

Вступ

Актуальність теми зумовлена необхідністю об’єктивного та ефективного розподілу бюджетних місць у закладах вищої освіти. Правильне прогнозування того, які абітурієнти потраплять на місця державного замовлення (бюджет), має важливе значення для освітньої політики та справедливості. Сучасні методи класифікації, зокрема дискримінантний аналіз, широко застосовуються у різних галузях – від медицини та фінансів до маркетингу та соціології – де потрібно здійснювати точну і надійну класифікацію даних. Обсяг інформації постійно зростає, тому важливість методів, що дозволяють швидко й ефективно класифікувати дані, стає вирішальною. Дискримінантний аналіз (ДА) є одним із таких універсальних і ефективних методів, який може допомогти вирішувати прикладні задачі класифікації з високою точністю.

Об’єкт дослідження: процес розподілу місць державного замовлення (бюджетних місць) серед вступників університету.
Предмет дослідження: методи дискримінантного аналізу та їх застосування для прогнозування форми навчання (бюджет або контракт) на основі результатів вступних випробувань.

Мета роботи: статистичний аналіз розподілу бюджетних місць у університеті за допомогою методів дискримінантного аналізу з метою розробки моделі прогнозування форми навчання абітурієнта (бюджет/контракт) за його вступними показниками.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- Розкрити історію становлення методів дискримінантного аналізу та ключові етапи їх розвитку.
- Описати статистичні припущення дискримінантного аналізу
- Класифікувати базові методи дискримінантного аналізу (лінійний та квадратичний дискримінантні аналізи – LDA, QDA) та розглянути їх узагальнення.
- Проаналізувати переваги та обмеження застосування дискримінантного аналізу на практиці.
- Підготувати базу даних абітурієнтів (результати НМТ, додаткові бали, рішення про бюджет/контракт) для подальшого аналізу.
- Розробити модель лінійного дискримінантного аналізу (LDA) для прогнозування форми навчання.

- Інтерпретувати результати моделювання: визначити порогові значення балів для потрапляння на бюджет, оцінити ймовірності бюджетного місця, надати рекомендації абітурієнтам.
- Виконати порівняльний аналіз моделей LDA, QDA на тих самих даних і зробити висновки щодо найбільш доцільного підходу для даної задачі.

Методи дослідження включають математично-статистичний аналіз, комп'ютерне моделювання в середовищі R[5], а також аналіз літературних джерел з теми.

Структура роботи: Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

- **Розділ 1** подає теоретико-методологічні засади дискримінантного аналізу: історію розвитку методу, його статистичні припущення, класифікацію варіантів (LDA, QDA) й оцінку переваг та обмежень для подальшого застосування у вступній кампанії.
- **Розділ 2** описано підготовку реальних даних вступної кампанії 2024 р. Далі послідовно будуються й оцінюються моделі LDA та QDA. Наведено їхню точність, проаналізовано помилки класифікації та зроблено вибір на користь QDA як більш точного підходу на цих даних .
- **Розділ 3** містить практичне дослідження на матеріалі вступної кампанії: за допомогою розроблених моделей прогнозується форма навчання (бюджет / контракт) для абітурієнтів п'яти спеціальностей; визначаються порогові конкурсні бали та оптимальна стратегія розставлення пріоритетів заяв; порівнюються результати LDA, QDA і логістичної регресії на тестових даних .
- У **висновках** узагальнено результати дослідження, підтверджено ефективність QDA та сформульовано рекомендації для абітурієнтів і приймальних комісій .
- **Додатки** містять допоміжні матеріали: фрагменти вихідних даних, приклади R-коду зі скрипта Script.R, додаткові таблиці та графіки, що деталізують етапи аналізу .

Розділ 1. Теоретико-методологічні засади дискримінантного аналізу

1.1 Історія становлення методу та ключові етапи розвитку

Дискримінантний аналіз виник у середині XX століття як метод розв'язання задач класифікації в статистиці та розпізнаванні образів. Перші ідеї дискримінантного аналізу пов'язані з роботами Рональда Фішера – британського математика і статистика. У 1936 році Фішер опублікував класичну статтю **"The Use of Multiple Measurements in Taxonomic Problems"**, у якій представив метод лінійного дискримінантного аналізу (Linear Discriminant Analysis, LDA). Фішер продемонстрував, що можна знайти лінійну комбінацію змінних, яка максимально розділяє дві групи об'єктів (наприклад, різні види ірисів) при мінімальному розкиді всередині груп. Ця лінійна функція, відома як дискримінантна функція Фішера, забезпечує найбільш ефективну межу для класифікації об'єктів між двома класами. Таким чином, було закладено основи застосування дискримінантного підходу для вирішення таксономічних і класифікаційних задач [1].

Подальший розвиток теорії дискримінантного аналізу відбувався у 1940–1960-х роках. Дослідники того періоду розробили квадратичний дискримінантний аналіз (Quadratic Discriminant Analysis, QDA) – узагальнення LDA, яке дозволило враховувати нелінійні залежності між ознаками та різний рівень варіативності в різних класах. Застосування методу розширилося на різні галузі: соціальні науки, економіка, освіта, маркетингові дослідження та інші сфери, де виникала потреба класифікувати багатовимірні дані. У ці десятиліття було приділено особливу увагу адаптації дискримінантного аналізу для аналізу багатовимірних вибірок, пошуку ефективних алгоритмів оцінювання параметрів та перевірки статистичних гіпотез щодо відмінностей між групами.

Сучасний етап розвитку дискримінантного аналізу характеризується інтеграцією цього методу з іншими статистичними та машинними підходами. З появою потужних обчислювальних засобів зросла можливість застосування багатовимірних методів класифікації до великих масивів даних. Сьогодні дискримінантний аналіз часто використовується у комплексі з методами машинного навчання – наприклад, його поєднують з кластерним аналізом для попереднього виявлення груп, або порівнюють з логістичною регресією при вирішенні задач двокласової класифікації. В галузях на кшталт медицини, фінансової аналітики чи спортивної статистики, де обсяги даних великі, методи дискримінантного аналізу залишаються затребуваними завдяки своїй інтерпретованості та відносній простоті реалізації .

Ключові етапи розвитку методу можна підсумувати так:

- **1936 р.** – Р.Фішер запропонував лінійний дискримінантний аналіз, поклавши початок математичному апарату для задач класифікації

- **1940–60-ті рр.** – поява квадратичного дискримінантного аналізу та розширення застосувань ДА у різних науках.
- **1970–80-ті рр.** – розвиток алгоритмів реалізації ДА на комп'ютерах, поява програмних пакетів, перші порівняння з логістичною регресією.
- **1990–2000-ні** – інтеграція ДА з методами машинного навчання, поява регуляризованих версій методу (наприклад, регуляризований дискримінантний аналіз), застосування до великих баз даних.
- **Сучасність** – ДА залишається класичним інструментом статистики, що використовується поряд з сучасними алгоритмами ML; активно досліджуються способи підвищення стійкості ДА до порушення припущень та комбінування з іншими методами (наприклад, коваріаційний аналіз, байєсовські методи тощо).

- **Нормальність розподілу.** Припускається, що вектор ознак X для кожного класу має багатовимірний нормальний розподіл. Тобто, для K класів G_kсправедливе $X \mid G_k \sim \mathcal{N}(\mu_k, \Sigma_k)$, де μ_k – вектор середніх значень ознак у класі k, а Σ_k – коваріаційна матриця в класі. Нормальність є ключовим припущенням, оскільки аналітичні формули дискримінантних функцій імовірностей виводяться саме з нормального закону розподілу. На практиці це означає, що кожна кількісна ознака в межах класу приблизно підпорядковується гаусовому розподілу або може бути приведена до нього шляхом трансформацій (логарифмування, степеневі перетворення тощо). Порушення нормальності (наприклад, сильно асиметричні або багатомодальні розподіли ознак) може призводити до неточних оцінок параметрів і погіршення якості класифікації.
- **Рівність коваріаційних матриць .** Для лінійного дискримінантного аналізу передбачається, що коваріаційні матриці всіх класів рівні між собою: $\Sigma 1 = \Sigma 2 = \cdots = \Sigma K = \Sigma$. Ця гіпотеза означає, що варіабельність (розсіювання) ознак усередині кожної групи є однаковою. За таких умов межі, які розділяють класи, виходять лінійними. Якщо ж дисперсії/коваріації між класами різняться значною мірою, то оптимальні межі класифікації будуть нелінійними (квадратичними), і тоді ефективнішим є метод QDA, який не вимагає рівності Σ_k. Таким чином, перед застосуванням LDA важливо перевірити гомогенність дисперсій– чи можна вважати вибірки з різних груп такими, що мають спільну коваріаційну матрицю [2].
- **Незалежність спостережень.** Припускається, що об'єкти (вступники) вибрані незалежно один від одного. Це стандартне припущення статистичного моделювання: дані не повинні містити пар «пов'язаних» спостережень, які можуть спотворити оцінки (наприклад, повторні спостереження одного й того ж студента). В контексті вступної кампанії цю умову виконано, оскільки кожен абітурієнт подається в модель лише один раз і розглядається незалежно від інших.
- **Лінійність та адитивність ефектів.** Хоча дискримінантний аналіз дозволяє врахувати певну нелінійність (через QDA), класичний LDA фактично шукає лінійні комбінації ознак. Отже, вважається, що вплив

ознак на приналежність до класу є лінійно адитивним. Це означає, що відсутні складні взаємодії між предикторами або вони не суттєві. Якщо існують сильні нелінійні ефекти чи взаємодії, їх бажано явно включити в модель.

- **Метричні шкали ознак.** Дискримінантний аналіз вимагає кількісних (метричних) змінних або бінарних індикаторів. Номінальні категоріальні змінні не можна напряму використовувати в LDA/QDA, їх потрібно перекодувати чисельно (набір бінарних колонок). Порядкові змінні бажано розглядати як кількісні, якщо інтервали між рангами приблизно рівні, або теж перекодувати. Крім того, рекомендується уникати ознак, виміряних у сильно різних шкалах (наприклад, одна ознака в тисячах, інша в частках) без масштабування, хоча формули LDA містять коваріаційну матрицю, яка автоматично враховує масштаб через дисперсії. Для чисельної стабільності обчислень може бути доцільним стандартизувати змінні до порівнянних масштабів (нульового середнього та одиничного стандартного відхилення), особливо якщо величини дисперсій сильно різняться по ознаках.
- **Повнота даних та відсутність багатоколінеарності.** Вхідні дані не повинні мати значної частки пропущених значень – за потреби їх слід заповнити (імпутація) або видалити такі спостереження. Також припускається, що ознаки не є лінійними комбінаціями одна одної (відсутня повна колінеарність), інакше оцінка коваріаційної матриці Σ стає виродженою (необернеою). При наявності сильно корельованих предикторів (мультиколінеарності) модель LDA може страждати від нестабільності коефіцієнтів; у таких випадках або вилучають надлишкові змінні, або застосовують регуляризацію (наприклад, метод головних компонент перед ДА, або регуляризований ДА).

У випадку, якщо зазначені припущення виконуються хоча б апроксимаційно, дискримінантний аналіз дає оптимальні (за критеріїєм мінімізації помилок) результати класифікації. В протилежному разі, існує ризик, що результати будуть упередженими або модель недостовірною, і тоді слід розглянути використання робастних версій методу чи альтернативних підходів). На практиці перевірка припущень здійснюється шляхом аналізу даних: тест Шапіро-Уілка чи Q-Q графіки для нормальності, порівняння ковариаційних матриць – тестом Бокса, огляд кореляційних матриць для виявлення мультиколінеарності, тощо.

1.2 Класифікація базових методів (LDA, QDA) та їх узагальнення

Лінійний дискримінантний аналіз (LDA). Як зазначалося, LDA ґрунтується на припущенні спільної (однакової) коваріаційної матриці класів і нормальності розподілів. За цих умов функція правдоподібності для кожного класу є багатовимірною нормою з однаковим Σ, що приводить до лінійного дискримінантного правила. Основна ідея LDA – знайти таку лінійну комбінацію ознак

$$y = w_1x_1 + w_2x_2 + \cdots + w_px_p,$$

яка забезпечує максимальне розділення класів. Це досягається шляхом розв'язання оптимізаційної задачі: максимізувати відношення міжкласової варіації до внутрішньокласової. Коефіцієнти w_i визначаються з умови максимізації цього відношення (за критерієм Фішера) або, еквівалентно, через розв'язання узагальненого власного рівняння для матриць розсіювання.

Практично, для двох класів G_1 і G_2 лінійне дискримінантне правило зводиться до обчислення дискримінантного рахунку для кожного класу і порівняння їх:

$$\delta_k(x) = x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + ln\,\pi_k\,, \quad k\ =\ 1{,}2,$$

де π_k – апріорна ймовірність (частка) класу k у генеральній сукупності (як правило, оцінюється часткою в навчальній вибірці). У випадку рівних апріорних ймовірностей і відсутності зсуву, критерій спрощується до порівняння значень лінійної функції $f(x) = x^T\Sigma^{-1}(\mu_1 - \mu_2)$ з порогом. Геометрично межа класифікації – це гіперплощина, що рівновіддалена (з урахуванням Σ) від центрів мас двох класів. Новий об'єкт x належить до класу 1, якщо $\delta_1(x) > \delta_2(x)$ (інакше – до класу 2. У багатокласовому випадку ($K > 2$) будується $K - 1$ незалежних дискримінантних функцій, або використовується правило максимального $\delta_k(x)$ серед усіх класів k. Таким чином, LDA шукає **лінійні** межі між класами.

LDA вирізняється відносною простотою реалізації та інтерпретації. Його параметри можна оцінити аналітично: середні $\widehat{\mu_k}$ – як вибіркові середні по класах, спільна коваріація $\hat{\Sigma}$ – як зважена сума внутрішньокласових коваріацій. Ці оцінки ефективні при великому обсязі вибірки і нормальному розподілі даних. LDA також має зв'язок з статистичною теорією: за умови нормальності і рівних коваріацій дискримінантна функція Фішера еквівалентна т-статистиці для різниці середніх, а критерій значущості дискримінантної функції пов'язаний з мановою .

Квадратичний дискримінантний аналіз (QDA). QDA є узагальненням лінійного аналізу для випадків, коли дисперсійні матриці класів істотно різняться. У моделі QDA не робиться припущення $\Sigma_1 = \Sigma_2 = \cdots = \Sigma$, натомість для кожного класу оцінюється власна матриця Σ_k. Правило класифікації виводиться з тих самих припущень нормальності, але без спрощення спільної Σ. Як наслідок, *рівняння, що розділяє класи, набуває квадратичної форми*. Для двох класів гіперповерхня прийняття рішення описується рівнянням:

$$(x - \mu_1)^T\Sigma_1^{-1}(x - \mu_1) - (x - \mu_2)^T\Sigma_2^{-1}(x - \mu_2) = const,$$

що в загальному випадку є квадратичною поверхнею (наприклад, еліпсоїдом у 2D просторі). QDA класифікує x до класу 1, якщо його дискримінантна оцінка для класу 1 (вже з урахуванням Σ_1) більша, ніж для класу 2.

Таким чином, QDA моделює кожен клас окремо, дозволяючи різні форми розподілу. Це робить метод більш гнучким – він може ефективніше працювати, коли реальні границі між класами нелінійні. Проте за цю гнучкість сплачується ціною: значно більшою кількістю параметрів, які треба оцінити. Якщо в LDA потрібно оцінити Σ (приблизно $\frac{p(p+1)}{2}$ параметрів для p ознак), то в QDA – K таких матриць (у K разів більше параметрів). Для великих p та відносно малих обсягів даних QDA може переобучуватися. Отже, QDA доцільно застосовувати або при **суттєвій неоднаковості дисперсій** класів, або при великому розмірі вибірки, який дозволяє надійно оцінити коваріацію окремо для кожного класу. В інших випадках простіший LDA може давати більш стабільні результати [3].

Варто зазначити, що LDA і QDA легко узагальнюються на *багатокласові задачі*. Для $K > 2$ класів лінійний дискримінантний аналіз фактично зводиться до пошуку $K - 1$ незалежних лінійних дискримінантних функцій (через власні вектори матриці міжкласового розсіювання відносно внутрішньокласового). На практиці ж часто реалізується простіша схема «один проти всіх» або «один проти одного», де будуються двокласові правила і комбінуються для отримання багатокласового рішення. QDA при $K > 2$ просто обчислює для кожного класу свій квадратичний член і обирає клас з найбільшим апостеріорним ймовірністю. Усі ці підходи формально випливають з однієї моделі гауссових щільностей, тому математично коректні.

Отже, LDA та QDA є базовими представниками дискримінантного аналізу – перший робить більш жорсткі припущення і дає лінійну межу, другий – більш гнучкий за рахунок квадратичних меж. В залежності від характеру даних можливі їх узагальнення, які дозволяють покращити модель: введення регуляризації, врахування нелінійності через ядра, або використання робастних методів оцінювання. Ці розширення покликані зберегти переваги ДА (інтерпретованість, ефективність) у складніших умовах, де класичні припущення не виконуються повністю.

1.3 Переваги й обмеження дискримінантного аналізу у практичних застосуваннях

На завершення теоретичної частини підсумуємо загальні **переваги** та **обмеження** дискримінантного аналізу як методу класифікації, особливо з огляду на практичні задачі, подібні до аналізу вступних балів.

Переваги методу:

- **Висока ефективність при виконанні припущень.** Якщо розподіл даних близький до багатовимірної нормалі, а дисперсії класів подібні, дискримінантний аналіз дає статистично оптимальний критерій

класифікації (в сенсі мінімізації помилки). У таких умовах LDA часто перевершує альтернативи за точністю класифікації, використовуючи всю інформацію про структуру даних.

- **Інтерпретованість результатів.** Дискримінантні функції – це лінійні комбінації вихідних ознак, коефіцієнти яких вказують на внесок кожної ознаки у розрізнення класів. Це дозволяє проаналізувати, які змінні є найбільш впливовими для класифікації. В задачі з балами вступників, наприклад, можна безпосередньо побачити, наскільки вагомим є результат НМТ порівняно з додатковими балами. Інтерпретація ЛДА схожа до множинної регресії, що звично для аналітиків.
- **Облік апостеріорних ймовірностей та балансу класів.** На відміну від деяких методів (наприклад, жорстких правил дерев), LDA/QDA явно обчислюють апостеріорні ймовірності $P(G = k|x)$. Це дозволяє гнучко вибирати пороги вирішення в залежності від вимог (наприклад, можна налаштувати поріг вище 0.5, якщо важливіше виявити клас 1 з високою точністю). Крім того, через включення апріорних π_k метод автоматично враховує дисбаланс вибірки. Якщо, скажімо, відсоток бюджетних місць невеликий, LDA може вбудувати це у правило (змістить поріг) – не всі алгоритми класифікації роблять це настільки просто.
- **Швидкість і простота обчислень.** Оцінювання параметрів LDA виконується швидко: потрібно обчислити середні та коваріаційні матриці. Навіть QDA з великою кількістю параметрів при сучасних обчислювальних можливостях працює досить швидко для помірних розмірів даних. Це особливо важливо при багаторазовому застосуванні (наприклад, перекросс-валидації) або в реальному часі. LDA має аналітичне рішення, яке не потребує чисельної оптимізації, на відміну від деяких інших методів (логістична регресія потребує ітераційної оптимізації, але теж швидкої).
- **Застосовність до малих вибірок.** Завдяки вбудованій статистичній моделі, LDA може працювати навіть коли даних небагато, «переносячи» припущення про нормальність. Відомо, що при малих N дискримінантний аналіз іноді дає кращі результати, ніж більш гнучкі алгоритми, через ефект скорочення дисперсії моделі [1]. Для нашої задачі, якщо кількість минулих вступників обмежена, LDA все одно може побудувати зносну модель, тоді як методам як-от дерева може забракнути даних для навчання без переобучення.
- **Можливість статистичних перевірок.** На додачу до класифікації, дискримінантний аналіз надає інструменти для оцінки значущості моделі: критерій Вілкса-ламбда для перевірки відмінності між групами, F-тести для окремих дискримінантних функцій, тощо. Це дозволяє формально підтвердити, що групи дійсно відрізняються за заданими ознаками (наприклад, що сукупність балів статистично значуще різниться між бюджетниками і контрактниками).

Обмеження та недоліки методу:

- **Чутливість до порушення припущень.** Як вже згадувалось, головний мінус ДА – втрата оптимальності при невиконанні базових гіпотез. Якщо розподіл балів сильно відхиляється від нормального (наприклад, бімодальний) або дисперсії дуже різні, LDA може давати субоптимальні або упереджені результати класифікації. У практиці вступних кампаній розподіл сумарних балів може бути далеким від нормального (особливо якщо є мінімальний прохідний бал, який створює "урізання" знизу), тому треба перевіряти і за потреби застосувати інші методи чи трансформації ознак.
- **Наявність викидів у даних.** LDA оцінює середні та коваріації способом, чутливим до екстремальних значень (вибіркові середні/дисперсії сильно зміщуються під впливом викидів). Отже, поодинокі аномальні абітурієнти з дуже нетиповими балами можуть вплинути на положення межі класифікації. Перед застосуванням ДА варто очистити дані від викидів або використати робастні оцінки. Логістична регресія та дерева, до речі, теж чутливі до викидів у предикторах, але менш формально: LR – через вплив на градієнти оптимізації, дерева – можуть ізолювати викид окремим правилом, мінімізуючи його вплив.
- **Вимоги до співвідношення N та p.** Дискримінантний аналіз вимагає, щоб обсяг вибірки був помітно більшим за число параметрів, що оцінюються. Як правило, рекомендують $N > 3p$ чи навіть $N > 5p$ для надійності. Якщо ознак багато (більше десятків) при відносно невеликій кількості спостережень, оцінка коваріаційної матриці стає нестабільною. У нашій задачі кількість ознак невелика (результати кількох іспитів), тож це обмеження не критичне. Але в інших практичних застосуваннях (наприклад, десятки біомаркерів для класифікації пацієнтів) може виникнути потреба зменшувати розмірність (методом головних компонент) або застосовувати регуляризований ДА.
- **Лінійність границі (для LDA).** Якщо реальний розподіл вимагає дуже викривленої межі між класами, лінійна функція LDA може мати систематичну похибку. Квадратичний аналіз частково вирішує цю проблему, але теж обмежений квадратичною формою. Більш складні нелінійні кордони доведеться моделювати іншими методами (нейромережі, SVM з ядром, дерева). Хоча через ядровий підхід можна зробити і LDA нелінійним, на практиці частіше просто вибирають інший алгоритм.
- **Складність врахування взаємодій і нелінійних ефектів.** Дискримінантний аналіз "як є" не моделює взаємодію ознак, якщо явно не створити нові ознаки. Наприклад, якщо шанс на бюджет визначається високим балом з математики *або* укрмови (одна з двох достатня), то це логічне "АБО" не представляється жодною лінійною комбінацією – LDA його не зловить, а дерево – легко. Для врахування такого в ДА доведеться вводити додаткові змінні (макс{бал_мат, бал_укр}), що не завжди очевидно. Тобто метод менш гнучкий у побудові довільних правил, ніж, скажімо, дерева рішень.

- **Обмеженість для нефакторних класів.** Якщо цільова змінна не категоріальна, а, наприклад, ранжована або кількісна, дискримінантний аналіз прямо не застосовується (в таких випадках використовують регресійні методи або аналіз головних компонент). У нашому дослідженні клас (форма навчання) чітко бінарний, тож це не проблема. Але варто зазначити, що для більше ніж двох класів інтерпретація кількох дискримінантних функцій стає менш прозорою – треба розглядати багатовимірні графіки, що ускладнює пояснення результатів некваліфікованій аудиторії.

Підсумовуючи, дискримінантний аналіз є потужним інструментом класифікації, коли його припущення більш-менш виконуються. Він надає точні та інтерпретовані результати, дозволяє статистично обґрунтувати висновки. У практиці вступної кампанії умови методу загалом виконуються (результати тестів часто мають близький до нормального розподіл, різниця дисперсій між групами нерадикальна, ознаки незалежні). Це робить ДА привабливим для побудови моделі відбору на бюджетні місця. Разом з тим, слід пам’ятати про можливі відхилення (наприклад, якщо є особливі категорії вступників з іншими характеристиками – спортсмени, цільові напрямки тощо, – які можуть «випасти» з загальної моделі). У таких ситуаціях результати ДА потрібно інтерпретувати обережно і, за потреби, комбінувати з бізнес-правилами чи іншими алгоритмами. Надалі, у практичній частині роботи, ми побачимо, наскільки теоретичні переваги дискримінантного аналізу реалізуються на реальних даних та чи виникають зазначені обмеження.

РОЗДІЛ 2. Побудова моделей і оцінювання їхньої ефективності

2.1 Підготовка даних

Для статистичного моделювання використано дані вступної кампанії університету за 2024 рік.[4] Кожен запис містить інформацію про одну подану абітурієнтом заявку: конкурсний бал (сума балів зовнішніх екзаменів та інших критеріїв), спеціальність (освітню програму), пріоритет заявки (порядковий номер вибору спеціальності) та статус заявки за результатами конкурсу. Серед можливих статусів – «Допущено» (заявка допущена до конкурсу, але не отримала рекомендації на державне місце), «Рекомендовано (бюджет)» (заявник отримав рекомендацію до зарахування на місце державного замовлення), а також технічні статуси типу «Скасовано (втрата пріор.)» або «Деактивовано (зарах. на бюджет)». Для цілей класифікаційного аналізу обрано дві основні категорії статусу: **Допущено** (умовно відповідає зарахуванню на контракт) і **Рекомендовано (бюджет)** (зарахування на бюджет). Заявки з іншими статусами були виключені з вибірки.

Крім того, для моделювання вирішено сфокусуватися на фіксованому наборі із п’яти спеціальностей, щоб проаналізувати розподіл державних місць у межах споріднених програм. З повного переліку спеціальностей у даних (змінна «Спеціальність») було сформовано список із п’яти вибраних програм: *014 Середня освіта*, *111 Математика*, *113 Прикладна математика*, *121 Інженерія програмного забезпечення* та *122 Комп’ютерні науки*. Вибір саме цих спеціальностей зумовлений їхньою схожістю (математика та ІТ-напрям) та достатнім обсягом даних для аналізу. Загалом по цих п’яти програмах вибірка містить **1196** заяв, з яких **216** (≈18,1%) отримали рекомендацію на бюджет, а решта **980** – залишилися допущеними (контрактними).

Для побудови моделей дані були розділені на навчальну і тестову вибірки. Оскільки класи суттєво незбалансовані (бюджетні місця становлять лише 18% випадків), було застосовано підхід штучного балансування при формуванні навчальної вибірки. Зокрема, випадковим чином відібрано 100 заяв зі статусом «Рекомендовано (бюджет)» та 100 заяв зі статусом «Допущено» (усього 200 спостережень) із усіх доступних даних зазначених спеціальностей. Ця підвибірка використовується для навчання моделей. Решта **996** записів (833 допущених і 163 бюджетних) відкладена як тестова вибірка для оцінювання точності. Такий підхід дозволяє надати моделям збалансовані дані при навчанні, щоб вони краще виокремлювали рідкісний клас (бюджет), не будучи домінованої чисельною перевагою другого класу.

Для підготовки даних використовувався мова R. Спочатку дані завантажуються із CSV-файлу за допомогою функції read_delim() пакету **readr** із врахуванням потрібної кодування (Windows-1251 для українського тексту). Далі формується фактор змінної статусу заявки і визначаються рівні, що відповідають класам

для аналізу. Зокрема, вектор level містить дві категорії: "Допущено" та "Рекомендовано (бюджет)". На наступному кроці формується список усіх спеціальностей у наборі даних (special) і на його основі – вектор із п’яти вибраних спеціальностей sp (за їх позиціями або назвами). Потім кодом Data <- Data_03_08_24_proba[, c(5,9:11)] з повного датасету вибираються лише релевантні стовпці: спеціальність (5-й стовпець), статус заявки (9-й), конкурсний бал (10-й) і пріоритет (11-й). Залишаються тільки записи, статус яких входить до level, тобто або бюджет, або допущено. З використанням функції sample() здійснено вибір 100 індексів рядків для кожного класу статусу (рівні level[1] і level[2] відповідно), після чого формується навчальна вибірка DataTrain як об’єднання вибраних рядків обох класів. Решта спостережень потрапила до тестового набору DataTest. Таким чином, підготовлені дані містять змінні: **Конкурсний_бал** (числова), **Пріоритет** (ціле число 1–5) як предиктори, і **Статус заявки** (фактор з рівнями «Допущено»/«Рекомендовано (бюджет)») як цільову змінну для класифікації.

2.2 Лінійний дискримінантний аналіз (LDA)

Для розв’язання поставленого завдання класифікації спочатку було застосовано лінійний дискримінантний аналіз (Linear Discriminant Analysis, LDA). Метод LDA шукає таку лінійну комбінацію предикторних змінних, яка найкраще розділяє дві групи спостережень. Іншими словами, LDA проектує багатовимірні дані в простір низької розмірності (в даному випадку – на пряму), максимізуючи відстань між середніми значеннями класів та мінімізуючи розсіювання всередині класів. Класичний LDA базується на припущенні, що дані кожного класу розподілені нормально і мають однакові коваріаційні матриці для всіх класів. У випадку двокласової задачі це означає, що точки кожного з двох статусів (бюджет/контракт) приблизно підпорядковуються багатовимірному нормальному розподілу з різними середніми векторими, але спільною коваріаційною матрицею. За таких умов межа рішення між класами є лінійною. LDA оцінює параметри моделей розподілу – середні вектори μ_0, μ_1 та загальну коваріаційну матрицю Σ – на основі навчальної вибірки, а потім для нового спостереження x визначає приналежність до класу шляхом порівняння дискримінантних оцінок (або еквівалентно – шляхом підстановки в формулу байєсівського рішення) . Дискримінантна функція для класу (наприклад, «бюджет») має вигляд лінійного функціонала: $\delta_{\text{budget}}(x) = x^T\Sigma^{-1}\mu_1 - \frac{1}{2}\mu_1^T\Sigma^{-1}\mu_1 + \ln \pi_1$ де π_1 – апріорна ймовірність класу. Аналогічно обчислюється $\delta_{\text{contract}}(x)$ для іншого класу. Нове спостереження відноситься до того класу, для якого δ максимальна (еквівалентно, для двох класів – якщо різниця $\delta_1 - \delta_0$ перевищує певний поріг).

Використання LDA обґрунтоване тим, що він є досить стійким і ефективним методом для випадків, коли припущення нормальності та гомоскедастичності (рівності коваріацій) принаймні наближено виконуються. Навіть якщо розподіли відхиляються від нормальних, LDA часто дає непогані результати класифікації, особливо на великих вибірках. У нашій задачі предиктори –

конкурсний бал та пріоритет – можуть мати близький до нормального розподіл у межах кожного класу, хоча апріорі відомо, що дисперсія балів серед «бюджетників» може відрізнятися від «контрактників». Спочатку ми припустили спільну дисперсію, застосувавши LDA як базову модель.

LDA-модель була побудована на навчальній вибірці із 200 спостережень (порівну двох класів). При моделюванні було задано апріорні ймовірності класів $P(\text{бюджет}) = P(\text{контракт}) = 0.5$ для навчання – таким чином, модель намагається однаково добре розпізнавати обидва класи, не зважаючи на їхню реальну незбалансованість. У R це відповідає параметру prior=c(0.5,0.5) у функції lda(). Отримана лінійна дискримінантна функція має вигляд:

$$g(\text{бал}, \text{пріоритет}) = w_1 \cdot \text{бал} + w_2 \cdot \text{пріоритет} + c,$$

де w_1 і w_2 – оцінені коефіцієнти. Для нашої моделі $w_1 > 0$ та $w_2 < 0$, що свідчить про інтуїтивно очікуваний вплив: більший конкурсний бал збільшує схильність до класу «бюджет», тоді як більший номер пріоритету (тобто нижча пріоритетність спеціальності) зменшує шанси на бюджет. Це узгоджується з логікою відбору – абітурієнти з високим балом мають більше шансів отримати місце державного замовлення, а заяви, подані як пріоритет №1, мають більшу ймовірність бути рекомендованими на бюджет, ніж ті, що вказані як запасні варіанти.

Для оцінки ефективності LDA-моделі було проведено класифікацію спостережень тестової вибірки. Таблиця 2.1 представляє матрицю класифікації (confusion matrix) для результатів LDA на тестових даних (833 контрактних і 163 бюджетних заяви).

Таблиця 2.1 – Матриця класифікації для LDA-моделі (тестові дані)

Фактичний статус	**Прогнозовано: Допущено**	**Прогнозовано: Бюджет**	**Точність класу, %**
Допущено (контракт)	503	330	60,4%
Рекомендовано (бюджет)	46	117	71,8%
Загалом	–	–	**62,3%**

Як видно з таблиці, LDA правильно класифікував приблизно 62,3% випадків тестової вибірки. Модель виявилася відносно чутливою до класу «бюджет» (правильно ідентифіковано 71,8% бюджетних заявок), але водночас дала значну кількість хибних спрацьовувань – 39,6% заявок класу «Допущено» помилково віднесені до бюджету. Така поведінка є наслідком збалансованого навчання моделі: LDA налаштований не пропустити якнайбільше бюджетних випадків, через що «перестарається» і частину контрактних випадків класифікує як бюджетні. В реальних умовах це означає, що модель намагається виявити потенційних бюджетників навіть ціною нижчої точності для основної маси

абітурієнтів. При тому, якщо розглядати **загальну** точність 62,3%, вона нижча за тривіальну долю найбільшого класу (81,9% контрактних), що очікувано при збалансованому навчанні – модель не прагне максимальної кількості правильних відповідей загалом, а натомість збалансовує помилки двох типів. У випадку необхідності підвищити загальну точність можна було б задати LDA фактичні апріорні ймовірності класів (~0,82 і 0,18); тоді модель була б більш консервативною і майже всі заявки віднесла б до «контракту», досягаючи ~82% точності, але майже не виявляючи бюджетників. Таким чином, компроміс між повнотою (чутливістю) і точністю класифікації для меншого класу регулюється вибором порогу рішення або апріорів у моделі.

2.3 Квадратичний дискримінантний аналіз (QDA)

Наступним етапом дослідження стало застосування квадратичного дискримінантного аналізу (Quadratic Discriminant Analysis, QDA) для порівняння з LDA. QDA є узагальненням лінійного дискримінантного аналізу, яке знімає припущення про однаковість коваріаційних матриць класів. Іншими словами, QDA також припускає нормальний розподіл даних у кожному класі, але дозволяє кожному класу мати власну коваріаційну матрицю Σ_0 та Σ_1. Внаслідок цього межа між класами описується квадратичною поверхнею (квадратична функція ознак), а не прямою. Такий підхід є доцільним, коли варіація ознак у різних класах помітно відрізняється або коли існує взаємодія між предикторами, що по-різному проявляється у класах. За наявності достатньо великої вибірки для оцінки більшої кількості параметрів (коваріаційних матриць), QDA зазвичай забезпечує вищу гнучкість і може підвищити точність класифікації у випадках, коли припущення LDA порушені.

У наших даних є підстави очікувати, що припущення про однакову дисперсію не цілком виконується. Зокрема, можна припустити, що розкид конкурсаних балів та їх зв’язок із пріоритетом відрізняються для бюджетників і контрактників. Попередній аналіз це підтвердив: середній конкурсний бал бюджетних заявок в навчальній вибірці (~157,9) вищий, ніж у контрактних (~151,8), а середній пріоритет нижчий (2,13 проти 3,01). Окрім того, оцінки коваріації показали, що для бюджету бал і пріоритет мають від’ємну кореляцію (вищий бал – нижчий пріоритет), тоді як для контрактних – слабку додатну. За таких умов спільна лінійна модель може бути недостатньо точною, тому QDA має перевагу, дозволяючи окремо моделювати такі розходження.

Для побудови QDA-моделі використано ту ж навчальну вибірку. Модель оцінювалася за допомогою функції qda() пакету **MASS** (виклик аналогічний до lda(), лише змінено метод). Як і раніше, спочатку було використано рівні апріорні ймовірності класів 0,5/0,5 при навчанні. Отримані параметри включають оцінки середніх векторів балів і пріоритетів для кожного класу та дві коваріаційні матриці $\widehat{\Sigma_{\text{бюджет}}}$ і $\widehat{\Sigma_{\text{контракт}}}$. З модельних параметрів видно, що дисперсія конкурсний балу серед «Допущено» дещо більша, ніж серед «Рекомендовано (бюджет)», а кореляція між балом і пріоритетом має протилежний знак для двох класів (як і було зазначено). Таким чином, QDA

враховує цю різницю: для класу «бюджет» простір рішень витягнуто вздовж осі бал–пріоритет в інший бік, ніж для «контракту».

Результати класифікації тестових даних за QDA-моделлю подано в таблиці 2.2.

Таблиця 2.2 – Матриця класифікації для QDA-моделі (тестові дані)

Фактичний статус	**Прогнозовано: Допущено**	**Прогнозовано: Бюджет**	**Точність класу, %**
Допущено (контракт)	571	262	68,5%
Рекомендовано (бюджет)	58	105	64,4%
Загалом	–	–	**67,9%**

Як бачимо, квадратичний дискримінантний аналіз продемонстрував кращі показники на тестовій вибірці порівняно з лінійним. Загальна точність класифікації зросла до 67,9%. Помітно покращилася класифікація основного класу «контракт»: QDA правильно визначив ~68,5% допущених заяв (проти ~60,4% у LDA). Частка хибнопозитивних спрацьовувань (контракт, помилково віднесений до бюджету) зменшилась з 39,6% до 31,5%. Натомість точність розпізнавання бюджетних заяв дещо знизилася (64,4% проти 71,8% у LDA). Однак завдяки тому, що частка контрактних заяв значно більша, покращення по першому рядку матриці суттєво вплинуло на загальну метрику. Інтуїтивно це означає, що QDA-модель була більш консервативною у винесенні рішення «бюджет» – вона висуває до кандидата вищі вимоги, перш ніж віднести його до бюджету, ніж LDA-модель. Це узгоджується з очікуваннями: врахування різної варіації ознак дозволило моделі точніше окреслити область «бюджет», зробивши її більш вузькою і специфічною. В результаті менше контрактників помилково потрапили в цю область. Хоча при цьому деякі граничні бюджетні випадки модель не розпізнала (чутливість трохи знизилась), загальний баланс помилок став кращим.

Варто зазначити, що для реалізації QDA потрібен більший обсяг даних для оцінювання параметрів (коваріацій) з належною точністю. У нашому випадку кількість предикторів невелика (два), тож 200 об’єктів навчальної вибірки було достатньо, аби отримати стійкі оцінки. Якби число ознак було більшим, LDA міг би мати перевагу через меншу варіативність оцінок параметрів. Крім того, якщо розподіли сильно відхиляються від нормальних або існують значущі нелінійні залежності, обидва методи можуть втрачати точність. У таких випадках альтернативними підходами є, наприклад, логістична регресія або методи машинного навчання, що не покладаються на параметричні припущення (дерева рішень, $k - NN$ тощо).

2.4 Порівняння LDA і QDA

Порівнявши дві моделі дискримінантного аналізу, можемо зробити висновки щодо їх ефективності на наших даних. QDA перевершив LDA за загальною точністю класифікації (~68% проти ~62%) та особливо в частині зниження помилкової віднесеності контрактних заяв до бюджету. Це свідчить, що припущення LDA про однакову структуру коваріації було в нашому випадку порушене настільки, що врахування квадратичного члену поліпшило розділення. Фактично, межа між класами в просторі «бал–пріоритет» виявилась не зовсім лінійною. Метод QDA зміг це врахувати, побудувавши вигнуту межу, яка краще відсіяла «помилкові» бюджетні класифікації.

З іншого боку, LDA має переваги простоти та інтерпретації. Його лінійні коефіцієнти легко трактувати: як зазначалося, LDA чітко показав, що вищий бал сильно підвищує шанси на бюджет (коефіцієнт при балові додатний), а вищий номер пріоритету знижує шанси (від'ємний коефіцієнт при пріоритеті). У моделі QDA такої єдиної лінійної функції немає – замість цього є дві квадратичні форми для кожного класу, що складніше для економічної інтерпретації. Проте можна інтерпретувати побічно: наприклад, для класу «бюджет» модель очікує трохи меншу дисперсію балів, тобто бюджетники більш однорідні за балами, а для «контракту» – більший розкид. Такі тонкощі простежуються лише через аналіз параметрів моделі.

У плані обчислень LDA менш вимогливий, особливо на високій розмірності, тоді як QDA при додаванні нових предикторів потребує суттєво більше даних для надійної оцінки коваріаційних матриць (кількість параметрів зростає квадратично з числом ознак). В нашій задачі з двома ознаками це не було проблемою. Таким чином, з точки зору точності на незалежних даних, **квадратичний дискримінантний аналіз показав кращий результат, ніж лінійний**, що вказує на доцільність його використання для класифікації заявок за статусом у даному випадку.

2.5 Інтерпретація та вибір моделі

На основі проведених експериментів було обрано модель для подальшого використання в практичному аналізі. Оскільки QDA забезпечив вищу якість класифікації, надалі доцільно спиратися саме на квадратичний дискримінантний аналіз. Важливо підкреслити, що головні фактори, які визначають розподіл державного замовлення, – це **конкурсний бал** абітурієнта та **пріоритет вибраної спеціальності**. Обидва методи виявили істотний вплив цих змінних на результат. Високий бал є практично необхідною умовою для отримання рекомендації на бюджет (у вибірці середній бал бюджетників ~158 проти ~152 у контрактників), а пріоритет відображає конкурентність вибору: заявки, подані як перший пріоритет, частіше стають бюджетними, ніж ті самі заявки, подані нижчим пріоритетом, що підтвердилося в моделюванні.

За допомогою обраної QDA-моделі можна наближено визначити «поріг» конкурсного бала, вище якого шанси на бюджет суттєві. Для сукупності розглянутих спеціальностей цей пороговий рівень становить близько **155 балів** (з поправкою на пріоритет). Тобто при балові ~155 і пріоритеті №1 модель оцінює ймовірність потрапити на бюджет приблизно як 60%, а при нижчому пріоритеті – менше (наприклад, для пріоритету №2 при тому ж балові ~52%). При балові 170 шанс значно зростає (понад 80% для пріоритету №1). Ці результати відповідають реальним даним: у вибірці найнижчий бал серед рекомендацій на бюджет коливається від ~130 (для найменш конкурентної програми «Середня освіта») до ~166 (для найконкурентнішої «Інженерія ПЗ»).

Таким чином, побудовані моделі підтверджують, що для успішного отримання місця державного замовлення абітурієнту необхідно мати високий конкурсний бал та подати заявку на бажану спеціальність якнайвищим пріоритетом. Квадратичний дискримінантний аналіз краще враховує тонкі ефекти і тому обраний як робоча модель для подальшого практичного дослідження.

Загальні висновки до розділу 2: В ході другого розділу було розроблено та протестовано моделі дискримінантного аналізу для прогнозування розподілу місць державного замовлення. Підготовано вибірку реальних даних вступної кампанії, збалансовано класи для навчання. Теоретично і експериментально розглянуто лінійний та квадратичний дискримінантні аналізи. LDA продемонстрував базову здатність класифікувати заявки за статусом, виокремивши ключові фактори впливу (бал і пріоритет). QDA покращив точність класифікації, врахувавши неоднорідність даних, та зменшив кількість помилкових віднесень. На основі порівняння моделей зроблено вибір на користь QDA як більш ефективної в даній задачі. Моделі показали, що вирішальними чинниками для отримання бюджетного місця є високий конкурсний бал та високий пріоритет заявки, а критичний поріг балів для значної ймовірності бюджету становить приблизно 155–160 (залежно від спеціальності). Ці напрацювання слугують підґрунтям для подальшого практичного аналізу вступної кампанії, який буде проведено у наступному розділі.

РОЗДІЛ 3. Практичне дослідження вступної кампанії: прогноз форми навчання

3.1 Опис бази даних

У цьому розділі здійснюється практичне застосування розробленої моделі до задачі прогнозування форми навчання вступників, тобто визначення ймовірності отримання місця державного замовлення (бюджету) в залежності від їх конкурсного бала та пріоритетів заяв.[4] Для цього використано базу даних заяв абітурієнтів на п’ять вибраних спеціальностей університету за результатами вступної кампанії 2024 року (той самий набір даних, що описувався в розділі 2.1). Нагадаємо основні характеристики цієї бази:

- **Кількість записів:** 1196 заяв на обрані спеціальності (після фільтрації статусів).
- **Ознаки (поля):**
  - *Конкурсний_бал* – числовий показник (від ~130 до 200), який визначає рейтинг абітурієнта. Він обчислюється на основі результатів національного мультипредметного тесту (НМТ) або ЗНО та додаткових коефіцієнтів.
  - *Пріоритет* – цілочислове значення від 1 до 5, яке вказує пріоритетність даної спеціальності для абітурієнта (1 – найвищий пріоритет, 5 – найнижчий). Кожен вступник міг подати до 5 заяв на різні спеціальності, розставивши їх у порядку бажаності; алгоритм розподілу державного замовлення враховує цю пріоритетність .
  - *Статус заявки* – категоріальна ознака, що приймає значення «Рекомендовано (бюджет)» або «Допущено». Як зазначалося, «Рекомендовано (бюджет)» означає, що заявник отримав місце державного замовлення за цією заявою (його було рекомендовано до зарахування на бюджет), а «Допущено» означає, що заява залишилася без бюджету (але могла бути зарахована на контракт). Кожен абітурієнт міг отримати рекомендацію максимум за однією заявою – якщо це сталося, інші його заяви втрачали актуальність (в нашому наборі такі випадки або виключені, або відзначені технічними статусами на кшталт «деактивовано»).

Дані зібрано з внутрішньої системи приймальної комісії (ЄДЕБО) та підготовлено у вигляді таблиці. У розділі 2 ми вже виконали очищення та попередню обробку: залишили лише дві категорії статусів та п’ять спеціальностей, сформували навчальну і тестову вибірки. В цьому розділі ми використаємо підготовлені дані для побудови моделі в програмному середовищі R та проаналізуємо її висновки для практичних цілей.

3.2 Побудова лінійної дискримінантної моделі в R

Для побудови та аналізу моделей застосовується статистичне програмування мовою R. У файлі *Script.R* представлено код, що реалізує весь процес – від завантаження даних до тренування моделей і прогнозування. Розглянемо поетапно цей код, пояснюючи призначення кожного фрагмента.

1. Завантаження і підготовка даних. Першим кроком скрипту є зчитування CSV-файлу з даними:

```
# Завантаження даних

library(readr)

Data_03_08_24_proba <- read_delim("Data_03_08_24_proba.csv",

    delim = ";", locale = locale(encoding = "cp1251"), trim_ws = TRUE)
```

Тут використовується функція read_delim пакету **readr**, яка читає дані із зазначеним роздільником «;». Параметр locale(encoding="cp1251") забезпечує коректне зчитування кирилиці в Windows-кодуванні. Після виконання цього коду таблиця Data_03_08_24_proba містить всі сирі дані вступної кампанії (понад 6000 записів і 24 колонки, як показано в прикладі на початку розділу 2.1).

Далі в скрипті визначаються можливі значення статусу заявки і відбираються необхідні нам рівні:

```
# Визначення можливих статусів заяв

level <- levels(as.factor(Data_03_08_24_proba$`Статус заявки`))[2:3]

# Визначення можливих спеціальностей

special <- levels(as.factor(Data_03_08_24_proba$Спеціальність))

# Задання списку спеціальностей

sp <- special[c(3, 34, 35, 33, 32)]
```

Тут команда levels(as.factor(...)) вилучає унікальні значення фактору "Статус заявки". У вихідних даних цих статусів було чотири (як згадано вище), але скрипт бере підмножину [2:3], тобто другий і третій рівні, – ними виявляються "Допущено" та "Рекомендовано (бюджет)" (факторні рівні можуть впорядковуватися за алфавітом або порядком появи, тому конкретні індекси підібрані таким чином). Таким чином, вектор level містить дві рядкові константи, які будемо використовувати для фільтрації даних за статусом.

Аналогічно визначається фактор спеціальностей і з нього формується вектор special усіх назв спеціальностей у даних. Потім вручну обрано п’ять потрібних спеціальностей – у коді це зроблено шляхом вибору елементів з індексами 3,34,35,33,32. Така, на перший погляд, незвична послідовність індексів зумовлена порядком спеціальностей у факторі. В результаті змінна sp містить вектор з назвами п’яти спеціальностей, які ми вирішили аналізувати (перелічені в 3.1). Надалі sp використовується для фільтрації датасету.

Після визначення класів і спеціальностей відбувається формування робочого піднабору даних:

```
# Формування бази даних

Data <- Data_03_08_24_proba[, c(5, 9:11)]

Data <- Data[is.element(Data$`Статус заявки`, level), ]
```

Ці рядки створюють таблицю Data, вибираючи лише потрібні стовпці з оригінальної (5-й стовпець – Спеціальність, 9-й – Статус заявки, 10-й – Конкурсний_бал, 11-й – Пріоритет) і відфільтровуючи тільки ті рядки, в яких статус заявки належить до вектору level (тобто "Допущено" або "Рекомендовано (бюджет)"). Функція is.element(..., level) повертає булевий вектор, який ми використовуємо для індексації рядків. Після цього в Data зберігаються саме ті 1196 записів, які відповідають вибраним класам статусу. Змінна RowN <- row.names(Data) зберігає індекси рядків цього піднабору (щоб зручно робити вибірку за номерами).

2. Розбиття на навчальну і тестову вибірки. На наступному етапі скрипта реалізовано вибір підмножин даних для навчання моделі і для її перевірки:

```
# Побудова вибірки для навчання системи прогнозу

train1 <- sample(RowN[is.element(Data$`Статус заявки`, level[1])], 100)

train2 <- sample(RowN[is.element(Data$`Статус заявки`, level[2])], 100)



# Формування навчальної та тестової вибірки

DataTrain <- rbind(Data[train1, ], Data[train2, ])

DataTest <- Data[!is.element(row.names(Data), c(train1, train2)), ]
```

Тут використано функцію sample() для випадкового вибору індексів. RowN[...] відбирає індекси рядків Data, що належать до першого класу (level[1] – "Допущено"), і випадково обирає 100 з них для навчальної вибірки (train1). Аналогічно train2 обирає 100 індексів другого класу ("Рекомендовано

(бюджет)"). Потім функцією rbind() ці рядки об’єднуються у новий датафрейм DataTrain (при цьому row.names задається явно, щоб уникнути дублювання індексів). У результаті DataTrain містить 200 рядків (100 бюджетних і 100 небюджетних заяв). Змінна DataTest визначається як всі решта даних Data, індекси яких не належать до вибраних train1 і train2. Таким чином, DataTest – це тестовий набір із 996 спостережень (переважно контрактних, оскільки бюджетних менше).

Важливо зазначити, що на момент формування моделей навчальна вибірка ще не фільтрувалася за спеціальностями – тобто ми спершу забезпечили баланс класів загалом, а вже під час побудови моделі будемо враховувати обмеження по спеціальностях. У скрипті це зроблено через параметр subset у функції моделювання (див. далі). Альтернативно можна було відразу відфільтрувати Data за спеціальностями sp і вже з тієї підмножини вибирати train/test, – обидва підходи дають схожий результат. У підсумку навчальні дані містять приблизно по 20–50 прикладів кожної з п’яти спеціальностей (в сумі по 100 кожного класу), а тестові дані – решту заяв, в тому числі всі заявки інших спеціальностей і невідібраних на тренування.

3. Побудова моделі дискримінантного аналізу. Після підготовки даних переходимо до навчання моделі. У даному випадку, відповідно до заголовку розділу, спершу будується лінійна дискримінантна модель:

```
# Дискримінантний аналіз (lda - лінійний)

lda_model <- MASS::lda(`Статус заявки` ~ Конкурсний_бал + Пріоритет,

   data = DataTrain, subset = is.element(DataTrain$Спеціальність, sp),

   prior = c(0.5, 0.5))
```

Тут ми використовуємо функцію lda() з пакету **MASS** (виклик через MASS:: не обов’язковий, якщо пакет підключено, але показує, з якого пакету функція). Формула моделі задається як Статус заявки ~ Конкурсний_бал + Пріоритет, тобто цільова змінна залежить від двох предикторів. Параметр data = DataTrain вказує, що беремо дані з навчальної вибірки. Важливим є параметр subset = is.element(DataTrain$Спеціальність, sp), який обмежує навчання лише тими рядками DataTrain, спеціальність яких входить у наш список sp вибраних спеціальностей. Хоча DataTrain і так в основному складається з цих спеціальностей (оскільки більшість бюджетних саме там), цей параметр гарантує, що модель буде сфокусована тільки на них. Нарешті, prior = c(0.5, 0.5) задає рівні апріорні ймовірності класів «контракт» і «бюджет» відповідно, як ми планували для збалансованого навчання.

В результаті виконання цього коду об’єкт lda_model містить параметри побудованої LDA-моделі: оцінки середніх значень конкурсного бала і пріоритету для кожного класу, спільну коваріаційну матрицю та інформацію про апріорі. За потреби ми могли б переглянути ці параметри (наприклад,

lda_model$scaling містить коефіцієнти лінійної дискримінантної функції). Знаючи їх, можна явно записати рівняння дискримінантної функції. Але безпосередньо в коді ми цього не робили, а перейшли одразу до застосування моделі.

4. Класифікація тестових даних та оцінка моделі. Наступним кроком скрипт здійснює прогнозування на тестовій вибірці і порівняння з фактичними значеннями:

```
# Формування даних для перевірки класифікації

newData <- DataTest[is.element(DataTest$Спеціальність, sp), ]



# Класифікація тестових даних

res <- predict(lda_model, newData)



# Таблиця результатів класифікації та реальних статусів

tbl <- table(Статус = newData$`Статус заявки`, Прогноз = res$class)
```

Тут створюється таблиця newData, яка є підмножиною DataTest, обмеженою тими ж спеціальностями sp. Таким чином, ми оцінюємо модель лише на тих тестових даних, для яких вона була навчена (заявки інших спеціальностей, яких модель *не бачила*, ми виключаємо з оцінки, оскільки не розраховували модель для них). Далі predict(lda_model, newData) застосовує натреновану LDA-модель до кожного спостереження з newData. Результат res – це список, що містить прогнозовані класи (res$class) і матрицю постеріor-імовірностей належності до кожного класу (res$posterior) для кожного випадку. Насамперед нас цікавлять класи. Функція table() будує крос-таблицю, де по рядках – **фактичний статус** заявки (newData$Статус заявки), а по стовпцях – **прогнозований статус** моделі (res$class). Ця таблиця tbl фактично і є матрицею змішування (confusion matrix), приклад якої наведено в таблиці 2.1 для LDA.

Далі в скрипті виконуються розрахунки точності:

```
# Відсотки правильних класифікацій за статусами

(tab_res <- cbind(tbl, round(c(tbl[1,1], tbl[2,2]) / apply(tbl, 1, sum) * 100, 2)))

# Загальний відсоток правильних класифікацій

round((tbl[1,1] + tbl[2,2]) / sum(tbl) * 100, 2)
```

Перший рядок обчислює відсоток правильних класифікацій **для кожного класу окремо**. Вираз c(tbl[1,1], tbl[2,2]) витягає діагональні елементи матриці (кількість правильних прогнозів для класу "Допущено" та для класу "Рекомендовано (бюджет)"). apply(tbl, 1, sum) рахує суму по кожному рядку (тобто реальну кількість об’єктів у кожному класі). Ділення першого на друге і множення на 100 дає відсоток правильно класифікованих у кожному класі. Функція round(..., 2) округлює до 2 знаків після коми, а cbind(tbl, ...) поєднує ці відсотки як додаткову колонку поруч із таблицею tbl. У результаті tab_res – це таблиця, яка точно відповідає нашим таблицям 2.1 чи 2.2 у тексті (де додано стовпець "% правильно").

Друга команда обчислює **загальну точність**: вона сумує правильні передбачення для обох класів (tbl[1,1] + tbl[2,2]) і ділить на загальне число прогнозів sum(tbl) (що дорівнює числу випадків в newData). Цей відсоток також округлюється до 0.01%. У підсумку скрипт виводить ці значення – для LDA-моделі ми отримали ~60,4% правильних для "контракт", ~71,8% для "бюджет" і ~62,3% загалом (як було описано). Для QDA-моделі, якщо замінити lda_model на qda_model і повторити, були отримані показники ~68,5%, ~64,4% і ~67,9% відповідно.

Слід зазначити, що для QDA-моделі у скрипті код аналогічний, лише виклик моделювання був:

```
qda_model <- MASS::qda(`Статус заявки` ~ Конкурсний_бал + Пріоритет,
DataTrain,

    subset = is.element(DataTrain$Спеціальність, sp)
```

та подальший predict(qda_model, newData). У нашому аналізі ми не наводимо повний повтор коду для QDA, оскільки логіка абсолютно та сама, різниця лише в тому, що функція qda() враховує окремі коваріації. Результати QDA ми вже детально розглянули у розділі 2.3-2.4.

Таким чином, на цьому етапі ми побудували в R дві класифікаційні моделі (LDA і QDA) та оцінили їх точність на тестових даних. Було підтверджено, що QDA дещо кращий, тому надалі використовуватимемо саме його для інтерпретації та прогнозування.

3.3 Оцінка моделі

Розроблену дискримінантну модель слід оцінити не лише за відсотком правильних класифікацій, але й за іншими критеріями, важливими для практичного застосування. В контексті вступної кампанії критичною є здатність моделі правильно ідентифікувати тих абітурієнтів, які отримають бюджет (тобто **чутливість** або повнота для класу "бюджет"), а також точно відсіювати тих, хто бюджету не отримає (**специфічність** для класу "контракт").

Як показали результати на тестовій вибірці (див. табл. 2.1 і 2.2), наша обрана модель QDA досягає балансу між цими показниками: вона розпізнає близько 64,4% бюджетників (пропускаючи приблизно кожного третього – ті, кого модель помилково віднесла до контракту), і правильно визначає ~68,5% контрактників (помилково зараховуючи на бюджет близько 31,5% контрактних заяв). Такий рівень чутливості і специфічності є компромісним. Як обговорювалося, модель навмисно налаштована (через рівні апріорі) на підвищену чутливість до бюджету, тому пожертвувала деякою специфічністю.

Для практики важливо врахувати, що помилки двох типів мають різні наслідки. Помилка першого роду в нашій задачі – це **незапропонований бюджетнику** (модель не виявила абітурієнта, який насправді отримав бюджет). Така помилка означає, що модель недооцінила шанси сильного абітурієнта. Помилка другого роду – **помилкове прогнозування бюджету для контрактника** – означає, що модель дала хибну надію там, де її не мало б бути. З точки зору рекомендацій абітурієнтам (розглядається далі), другий тип помилки може бути гіршим, адже абітурієнт може переоцінити свої шанси. Тому, можливо, варто налаштувати модель дещо консервативніше (зменшити хибнопозитивні результати) – це можна зробити, змінивши поріг класифікації або врахувавши реальні пропорції класів.

Загальна точність ~68% сама по собі не є надто високою, але, як вже зазначалося, вона нижча за тривіальний класифікатор "усі контрактники" саме через наше прагнення виявляти бюджетників. Тому загальна точність тут не є найінформативнішим показником успіху моделі. Більш показовими є наведені вище чутливість (64,4%) і специфічність (68,5%), а також **Precision** для класу "бюджет" – частка справжніх бюджетників серед тих, кого модель визначила як бюджет. Для QDA precision (точність позитивного прогнозу) дорівнює $\frac{105}{105+262} \approx 28{,}6\%$. Це досить низько, що очікувано при рідкісному позитивному класі і такій налаштованості моделі: лише кожен третій "прогнозований бюджетник" дійсно отримує бюджет, а двоє з трьох насправді ні. Для LDA цей показник був ще меншим (~26%). Ці цифри підтверджують, що модель "захоплює" багато зайвих випадків у клас бюджету. Однак у вступній кампанії такий підхід виправданий, оскільки кількість бюджетних місць обмежена і, вірогідно, нас більше цікавить не пропустити гідного абітурієнта (помилка I роду) навіть ціною додаткового шуму.

Враховуючи зазначені метрики, модель QDA можна вважати прийнятною як інструмент для аналізу. Вона адекватно відображає загальні тенденції і дає змогу робити певні прогнози щодо шансів конкретного абітурієнта. Надалі ми проаналізуємо, як інтерпретувати порогові значення моделі і що вони означають з практичної точки зору, а також наведемо рекомендації, які можна дати вступникам, спираючись на результати моделювання.

3.4 Інтерпретація порогів і практичні рекомендації

Одним з ключових результатів дискримінантного аналізу є визначення порогового правила, за яким відбувається класифікація. У випадку двох класів це правило можна виразити через критичне значення дискримінантної функції або, еквівалентно, через **поріг апостеріорної ймовірності** належності до класу "бюджет". При рівних апріорних ймовірностях та рівних втратах помилка класифікації поріг для післярозподільної ймовірності становить 0.5 – тобто модель відносить спостереження до бюджету, якщо $P(\text{бюджет}|x) > 0.5$. У нашій QDA-моделі ми використали саме такий критерій. Це дозволяє інтерпретувати результати наступним чином: модель визначає, чи перевищує очікувана ймовірність отримання бюджетного місця 50%. Звичайно, 50% – умовна межа; її можна коригувати залежно від того, яку мету ставить користувач моделі (максимізувати точність чи виявити більше позитивних випадків). У нашому аналізі поріг 0.5 було використано для оцінки точності, але для рекомендацій абітурієнтам може бути корисно розглянути й інші порогові значення.

На основі побудованої моделі можна знайти **критичні значення конкурсного бала** для різних сценаріїв. Наприклад, визначимо, при якому балові модель дає $P(\text{бюджет}|x) \approx 0.5$ для певного пріоритету. З огляду на результати моделювання, можна зробити такі висновки:

- **Поріг для пріоритету №1.** Якщо дана спеціальність – перший пріоритет для абітурієнта, то пороговий конкурсний бал, при якому шанси на бюджет ~50/50, складає приблизно **150 балів**. За нашою моделлю, при балові 150 і пріоритеті 1 апостеріорна ймовірність бюджету трохи перевищує 50%. Тобто абітурієнт з балом 150, який подав заявку як першим пріоритетом, вже має приблизно рівні шанси отримати бюджет або залишитися на контракті. При балові нижче 145 шанс різко падає (наприклад, при 140 балах модель дає ~30% на бюджет для пріоритету 1). Навпаки, при балові вище 155 шанси стають більшими за 50%. Для балів ~160 і вище модель прогнозує вже >70% ймовірності бюджету на пріоритеті 1.
- **Вплив пріоритету.** Модель чітко показує, що для однакового конкурсного бала нижчий пріоритет заявки зменшує шанси на бюджет. Це узгоджується з правилами відбору: якщо спеціальність стоїть у абітурієнта другим чи третім пріоритетом, це означає, що першим він обрав іншу спеціальність, і якщо він достатньо сильний, то може отримати бюджет саме на першому, а на цю (другу) спеціальність уже не претендуватиме. З точки зору моделі, пріоритет виступає як ознака, опосередковано пов’язана з конкурентністю. Наш QDA-підхід дозволяє кількісно оцінити цей вплив. Зокрема, для **пріоритету 2** пороговий бал для 50% ймовірності трохи вищий – близько **155**. Тобто абітурієнту, який поставив цю програму другим пріоритетом, потрібно десь на 5 балів більше, щоб мати ті самі 50% шансу. Для **пріоритету 3** поріг ще вищий, орієнтовно **160+** балів для 50% шансу. А при пріоритеті 4–5 навіть дуже

високі бали не гарантують наближення до 50% шансу, оскільки, ймовірно, якщо абітурієнт має дуже високий бал, він реалізує бюджет на одному з пріоритетів 1–3 до того, як дійде до 4–5 (і в наших даних більшість бюджетників мали пріоритет 1 або 2).

Іншим способом інтерпретації моделі є **визначення найімовірнішого результату для конкретного абітурієнта** з певним балом і набором пріоритетів. Використовуючи функцію прогнозування, ми можемо для кожного абітурієнта отримати ймовірність бюджету. Це дозволяє побудувати рекомендації. Наприклад, в скрипті *Script.R* реалізовано підбір оптимального варіанту пріоритетів для абітурієнта з фіксованим балом. Фрагмент коду:

```
# Можливі перестановки пріоритетів (для 5 заяв)

prior <- combinat::permn(5)

bal <- 180

maxprob <- 0.5

for(i in 1:length(prior)) {

  prd <- predict(da_res, data.frame(Конкурсний_бал = rep(bal, 5),

                                    Пріоритет = prior [[i]]))
names(prd[["class"]])<-sp[prior[[i]]]

  mp<-max(prd[["posterior"]][,"Рекомендовано (бюджет)"])

  if(is.element("Рекомендовано (бюджет)", prd[["class"]]))

    result<-c(result, list(prd))

  if(mp>maxprob)

  {

    maxprob<-mp

    result_opimal<-prd

  }

}
```

Цей код генерує всі можливі порядки пріоритетів 1–5 (всього 120 перестановок) і для гіпотетичного абітурієнта з балом 155 перевіряє, який порядок дає максимальну ймовірність отримати хоча б десь бюджет. В кожній ітерації створюється штучний набір з 5 заяв (бал фіксований 155, а пріоритети беруться в певному порядку, представляючи собою «набір» виборів абітурієнта). Далі

predict обчислює posterior-імовірності для кожної з цих 5 заяв (в межах наших 5 спеціальностей). mp <- max(prd$posterior[,"Рекомендовано (бюджет)"]) знаходить максимальну ймовірність бюджету серед цих 5 – фактично, це ймовірність того, що даний абітурієнт отримає бюджет хоча б за одним зі своїх пріоритетів (найкраща з п’яти). Алгоритм фіксує ту перестановку пріоритетів, яка дала найбільше значення mp. За результатами цього перебору виявилося, що для бала 180 оптимально (з точки зору максимізації шансу на бюджет) розставити пріоритети в натуральному порядку (1,2,3,4,5). Це означає: абітурієнт повинен ранжувати спеціальності від найбажанішої до найменш бажаної, тобто фактично **нічого не виграється від перестановки пріоритетів** – що цілком відповідає офіційній політиці, яка закликає вступників ставити спеціальності в порядку щирої бажаності, адже алгоритм розподілу є максимально справедливим і немає сенсу намагатися його "обхитрити" перестановкою пріоритетів. Наш аналіз підтверджує, що якщо ціль абітурієнта – максимізувати шанс *отримати бюджет десь*, то йому просто слід використати всі 5 можливостей і поставити наищиріший пріоритет першим. У цьому випадку модель оцінює його сумарний шанс близько 59,6%. Спроби поставити, наприклад, менш конкурентну спеціальність на пріоритет 1, а більш бажану – на 2 чи 3, не підвищують цю сумарну ймовірність (у моделі це теж ~59-60%, просто перерозподіляється ймовірність між різними заявами).

Отримані результати дозволяють сформулювати **практичні рекомендації для абітурієнтів**:

1. **Звернути увагу на свій конкурсний бал.** Модель показує, що існують орієнтовні порогові значення балів для кожної спеціальності. Якщо ваш бал суттєво нижчий за мінімальний прохідний бал минулих років для бажаної спеціальності (для наших п’яти програм ці мінімальні бюджети були ~130–160), то ймовірність отримати бюджет невисока. В такому разі варто завчасно продумати план дій на випадок навчання за контрактом або вибрати менш конкурсну спеціальність, де ваш бал буде ближче до верхніх позицій рейтингу.
2. **Пріоритети слід виставляти за реальними бажаннями.** Наш аналіз підтверджує, що система пріоритетів працює належним чином: найвищий шанс отримати бюджет – на спеціальності, поставленій першим пріоритетом (за інших рівних умов). Тому не варто ставити "для підстраховки" менш бажану, але потенційно легшу спеціальність вище, ніж більш бажану. Якщо абітурієнт достатньо сильний для другої, то він тим більше пройде на першу; якщо ж для першої не вистачить балів, система автоматично розгляне другу. Таким чином, **стратегічно викривляти пріоритети немає сенсу** – це не підвищує загальної ймовірності отримання бюджету, а може призвести до ситуації, коли вступник отримає бюджет на менш бажаний напрям і вже не зможе претендувати на більш бажаний, хоча, можливо, міг би туди пройти на контракт (або навіть на бюджет при інших обставинах). Тож пріоритет №1 варто віддавати тій спеціальності, де ви найбільше хочете навчатися.

3. **Оцінити свої шанси з допомогою моделі.** Використовуючи результати дискримінантного аналізу, абітурієнт може приблизно оцінити свою ситуацію. Наприклад, маючи конкурсний бал X, він може зіставити його з орієнтовними порогами: якщо X на 10 і більше балів нижчий за прогнозований поріг для бажаної спеціальності, шанс отримати там бюджет мінімальний. Якщо ж бал на рівні або вищий порога, шанси хороші. Для наших даних, скажімо, вступник з балом 170+, який подає документи на математику чи комп'ютерні науки, майже напевно буде рекомендований на бюджет (модель дає >80% ймовірності). Навпаки, вступник з балом 140 на ці ж спеціальності практично не має шансів (ймовірність <20%). Такі оцінки допоможуть прийняти зважені рішення: можливо, зосередитися на підготовці до наступної сесії ЗНО/НМТ, якщо метою є саме бюджет в конкурентній галузі, або ж розглянути альтернативні спеціальності/заклади з нижчим конкурсним бар'єром.
4. **Врахувати специфіку спеціальностей.** Хоча наша узагальнена модель неявно усереднює різні спеціальності, відомо, що пороги на різних програмах різняться. Наприклад, в реальності на «Інженерію програмного забезпечення» прохідний бал на бюджет значно вищий, ніж на «Середню освіту (математика)». Тому, плануючи пріоритети, слід дивитися статистику минулих років для кожної спеціальності окремо. Наша модель це теж відобразила: для досягнення однакової ймовірності бюджету вимагались різні бали (ми фактично побачили це через різні мінімальні бюджетні бали в даних). Тож рекомендація – **корелювати свої бали з історичними порогами конкретних спеціальностей**. Якщо ваш бал ледь перевищує торішній поріг спеціальності, шанс ~50–60%; якщо сильно перевищує – шанс близький до 100%; якщо нижчий – розраховувати на бюджет ризиковано.

Загалом, дискримінантний аналіз підтвердив очевидні, але важливі для абітурієнтів речі: високий бал відкриває двері на бюджет, а правильна розстановка пріоритетів гарантує, що шанс буде використано максимально. Абітурієнтам з середніми балами варто завжди подавати всі 5 заяв (щоб не втратити можливість отримати бюджет на менш конкурентній програмі) і бути готовими до конкуренції на топових напрямах. Наш модельний підхід може стати в нагоді приймальним комісіям для **прогнозування кількості контрактників і бюджетників** при певному розподілі балів, а самим вступникам – для **оцінки ризиків** при виборі тієї чи іншої траєкторії подачі документів.

Список використаних джерел

1. Hastie T., Tibshirani R., Friedman J. **The Elements of Statistical Learning**. – 2nd ed. – Springer, 2009. (розд. 4.3 "Linear Discriminant Analysis").
2. James G., Witten D., Hastie T., Tibshirani R. **An Introduction to Statistical Learning with R**. – Springer, 2013. (гл. 4 "Classification", секція 4.4.1–4.4.3 про LDA, QDA, Logistic).
3. Press S. J., Wilson S. **Choosing between logistic regression and discriminant analysis** // *Journal of the American Statistical Association*, 1978, 73(364), p. 699–705.
4. **Умови прийому до закладів вищої освіти України у 2024 році** / Наказ МОН України №1098 від 15.10.2023. (Офіційний документ, доступний на сайті МОН).
5. R Core Team (2024). _R: A Language and Environment for Statistical Comp uting_. R Foundation for Statistical Computing, Vienna, Austria. <https://www .R-project.org/>.

Додатки

```
#Завантаження даних

library(readr)

Data_03_08_24_proba <- read_delim("Data_03_08_24_proba.csv",
                                  delim = ";",
                                  escape_double = FALSE,
                                  col_types = cols(`Ід заявки` = col_character(),
                                                   `Ід персони` = col_character(),
                                                   `ЗНО.Українська мова` = col_double(),
                                                   `ЗНО.Французька мова` = col_double(),
                                                   `ЗНО.Німецька мова` = col_double(),
                                                   `ЗНО.Іспанська мова` = col_double()),
                                  locale = locale(encoding = "cp1251"),
                                  trim_ws = TRUE)


#Визначення можливих статусів заяв

level<-levels(as.factor(Data_03_08_24_proba$`Статус заявки`))[2:3]


#Визначення можливих спеціальностей

special<-levels(as.factor(Data_03_08_24_proba$Спеціальність))


#Задання списку спеціальностей (за потреби, замість конкретних номерів може бути "all()")

sp<-special[c(3, 34, 35, 33, 32)]#all()


#Формування бази даних
```

```
Data<-Data_03_08_24_proba[,c(5,9:11)]
Data<-Data[is.element(Data$`Статус заявки`, level),]
Data <- Data[is.element(Data$Спеціальність, sp), ]
RowN <- row.names(Data)

# Побудова вибірки для навчання системи прогнозу
train1 <- sample(RowN[is.element(Data$`Статус заявки`, level[1])], 100)
train2 <- sample(RowN[is.element(Data$`Статус заявки`, level[2])], 100)

# Формування навчальної та тестової вибірки
DataTrain <- rbind(Data[train1, ], Data[train2, ])
DataTest <- Data[!is.element(row.names(Data), c(train1, train2)), ]

#Дискримінантний аналіз (qda - квадратичний, lda - лінійний)
lda_model <- MASS::lda(`Статус заявки` ~ Конкурсний_бал + Пріоритет,
                       data = DataTrain, subset = is.element(DataTrain$Спеціальність, sp),
                       prior = c(0.5, 0.5))

#Формування даних для перевірки процедури класифікації
newData<-DataTest[is.element(DataTest$Спеціальність,sp),]

#Класифікація тестових даних
res<-predict(lda_model,newData)
```

```
#Таблиця результатів класифікації та реальних статусів
tbl <- table(Статус = newData$`Статус заявки`, Прогноз = res$class)

#Відсотки правильних класифікацій за статусами
(tab_res<-cbind(tbl, round(c(tbl[1,1], tbl[2,2])/apply(tbl, 1, sum)*100,
                           digits = 2)))

#Загальний відсоток правильних класифікацій
(round((tbl[1,1]+tbl[2,2])/sum(tbl)*100,digits = 2))


## СПРОБА ЗАСТОСУВАННЯ
#Побудова процедури класифікації
da_res<-MASS::qda(`Статус заявки` ~ Конкурсний_бал+Пріоритет, Data,
                  subset = is.element(Data$Спеціальність,sp))

#Коефіцієнти дискримінантної функції
da_res[["scaling"]]

#Можливі перестановки пріоритетів
prior<-combinat::permn(5)

#Конкурсний бал (результат ЗНО/НМТ)
bal<-180
```

```
#Визначення найкращого варіанту вибору пріоритетів
result<-list()
result_opimal<-list()
maxprob<-0.5
for(i in 1:length(prior))
{
  prd<-predict(da_res,data.frame(Конкурсний_бал=rep(bal,5),
                                 Пріоритет=prior[[i]]))
  names(prd[["class"]])<-sp[prior[[i]]]
  mp<-max(prd[["posterior"]][,"Рекомендовано (бюджет)"])
  if(is.element("Рекомендовано (бюджет)", prd[["class"]]))
    result<-c(result, list(prd))
  if(mp>maxprob)
  {
    maxprob<-mp
    result_opimal<-prd
  }
}

# Всі результати і найкращий за ймовірністю результат
result_full<-list(result,result_opimal)
```