

Міністерство освіти і науки України  
Національний технічний університет  
«Дніпровська політехніка»

Факультет інформаційних технологій  
(факультет)

Кафедра системного аналізу та управління  
(повна назва)

**ПОЯСНОВАЛЬНА ЗАПИСКА**  
кваліфікаційної роботи ступеня бакалавра

Здобувача вищої освіти Куницького Владислава Вікторовича  
академічної групи 124-21-2  
спеціальності 124 Системний аналіз  
за освітньо-професійною програмою Системний аналіз  
на тему: «Системний аналіз та оптимізація фінансової діяльності  
маркетплейсу з використанням систем штучного інтелекту»

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	Інституційною	
кваліфікаційної роботи	<i>к.т.н., доц. Алексєєв О.М.</i>			
розділів:				
Інформаційно-аналітичний	<i>к.т.н., доц. Алексєєв О.М.</i>			
Спеціальний розділ	<i>к.т.н., доц. Алексєєв О.М.</i>			
Рецензент				
Нормоконтролер	<i>к.ф.-м.н., доц. Хом'як Т.В.</i>			

Дніпро  
2025

ЗАТВЕРДЖЕНО:

завідувач кафедри  
Системного аналізу та управління  
(повна назва)

\_\_\_\_\_ к.т.н., доц. Желдак Т.А.  
(підпис) (прізвище, ініціали)

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_\_ року

**ЗАВДАННЯ**  
**на кваліфікаційну роботу**  
**ступеня бакалавра**

здобувачу вищої освіти Куницькому В.В. академічної групи 124-21-2

спеціальності: 124 Системний аналіз

за освітньо-професійною програмою Системний аналіз

на тему «Системний аналіз та оптимізація фінансової діяльності  
маркетплейсу з використанням систем штучного інтелекту»

затверджену наказом ректора НТУ «Дніпровська політехніка» від 05.05.2025  
р. №336-с

Розділ	Зміст	Терміни виконання
1. Інформаційно-аналітичний розділ	<i>Дослідження існуючих джерел щодо обраної теми.</i>	15.01.2025 – 08.03.2025
2. Спеціальний розділ	<i>Аналіз та формалізація задачі. Порівняння існуючих програмних реалізацій. Порівняння існуючих мов програмування та IDE для вибору найоптимальнішої. Вибір найоптимальнішої системи штучного інтелекту. Розробка моделі та програмної реалізації завдання.</i>	09.03.2025 – 10.06.2025

Завдання видано \_\_\_\_\_

(підпис)

доц. Алексєєв О.М.  
(прізвище, ініціали)

Дата видачі: 03.02.2025 р.

Дата подання до екзаменаційної комісії: \_\_\_\_\_

Прийнято до виконання \_\_\_\_\_

(підпис студента)

Куницький В.В.  
(прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка: 69 с., 27 рис., 7 табл., 4 додатки, 17 джерел.

*Об'єктом дослідження* в роботі є процес розробки методології та програми, що оптимізує та автоматизує пошук критичних параметрів в фінансовій діяльності підприємства ROZETKA.

*Предметом дослідження* є методологія, що виявляє критичні параметри в фінансовій діяльності підприємства та визначає необхідні покращення для зменшення ризику банкрутства.

*Метою* даної кваліфікаційної роботи є розробка методології, що оптимізує та автоматизує пошук слабких місць в фінансовій діяльності підприємства ROZETKA та написання для неї розрахункової програми.

*Методи дослідження:* методи системного аналізу – для розглядів фінансової діяльності підприємства, методи машинного навчання – для покращення процесів управління фінансами, методи прогнозування – для передбачення фінансових результатів підприємства.

*У інформаційно-аналітичному* розділі наведено огляд створення алгоритмів машинного навчання та систем штучного інтелекту.

*У спеціальному розділі* були визначені рішення щодо оптимізації фінансової діяльності підприємства та вибране середовище реалізації, також були обрані системи штучного інтелекту, створено моделі для розв'язку задачі та написаний програмний код.

Практична цінність отриманих результатів полягає в тому, що розроблена методологія оптимізує час на знаходження критичних параметрів та автоматизує розрахунки за допомогою програмної реалізації.

Ключові слова: Системний аналіз, Машинне навчання, Системи штучного інтелекту, Фінансова діяльність, Ймовірність банкрутства, Критичні параметри.

## ABSTRACT

Cover letter: 69 pages, 27 figures, 7 tables, 4 appendices, 17 sources.

*The object of the research* in the paper is the process of developing a methodology and program that optimizes and automates the search for critical parameters in the financial activities of the ROZETKA enterprise.

*The subject of the study* is a methodology that identifies critical parameters in the financial activities of the enterprise and determines necessary improvements to reduce the risk of bankruptcy.

*The aim* of this qualification work is to develop a methodology that optimizes and automates the search for weaknesses in the financial activities of the company ROZETKA and to write a computational program for it.

*Research methods:* methods of system analysis – for examining the financial activities of the enterprise, machine learning methods – for improving financial management processes, forecasting methods – for predicting the financial results of the enterprise.

The *information and analytical section* examines the development of algorithms for machine learning and artificial intelligence systems.

In a *special section*, a decision was made to optimize the financial activity of the enterprise and the middle stage of implementation was selected, as well as the creation of a system of artificial intelligence, a model was created for unraveling the task and writing program code.

The practical value of the obtained results lies in the fact that the developed methodology optimizes the time for identifying critical parameters and automates calculations through software implementation.

**Keywords:** System Analysis, Machine Learning, Artificial Intelligence Systems, Financial Activity, Probability of Bankruptcy, Critical Parameters.

## ЗМІСТ

ВСТУП.....	6
ІНФОРМАЦІЙНО-АНАЛІТИЧНИЙ РОЗДІЛ .....	7
1.1 Теоретичні дані щодо машинного навчання .....	7
1.2 Принципи роботи машинного навчання.....	9
1.3 Види машинного навчання.....	100
1.4 Типи машинного навчання.....	144
1.5 Машинне навчання та статистика .....	15
1.6 Регресія у машинному навчанні .....	17
1.7 Існуючі алгоритми машинного навчання .....	20
СПЕЦІАЛЬНИЙ РОЗДІЛ .....	322
2.1 Побудова Z-моделей визначення ймовірності настання банкрутства ТОВ «ROZETKA» та визначення на основі них проблемних зон управління фінансовою діяльністю та їх оптимізація.....	322
2.2 Мотивація та аналіз задачі .....	355
2.3 Формалізація задачі (виділення вхідних-вихідних параметрів, зав'язків між ними тощо) .....	366
2.4 Вибір технології штучного інтелекту – обґрунтування вибору, огляд можливих варіантів, що розглядались .....	377
2.5 Обґрунтування вибору середовища реалізації (Python).....	422
2.6 Вибір IDE для розробки.....	444
2.7 Вибір додаткових бібліотек для розробки.....	466
2.8 Розробка моделі – створення формальної моделі для розв'язку задачі на основі обраної технології .....	48
2.9 Практична реалізація – короткий огляд програмної реалізації моделі в середовищі Python.....	49
2.10 Опис основних функцій програмної реалізації.....	500
ВИСНОВОК.....	655
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	67
ДОДАТКИ.....	69
Додаток А. Відомість матеріалів кваліфікаційної роботи .....	69
Додаток Б. Фінансові дані підприємства ROZETKA .....	760
Додаток В. Відгук керівника кваліфікаційної роботи... <b>Ошибка! Закладка         не определена.6</b>	

## ВСТУП

У сучасних умовах економічного розвитку питання фінансової стабільності та здатності підприємств до виживання стають все більш актуальними. Особливо це стосується компаній, що працюють у секторі роздрібно́ї торгівлі, де конкуренція надзвичайно висока, а вимоги споживачів постійно зростають.

Одним із ключових інструментів для оцінки фінансової стійкості підприємства є побудова Z-моделей та визначення ймовірності настання банкрутства. На прикладі ТОВ «ROZETKA» можна провести аналіз та виявити проблемні зони в управлінні фінансовою діяльністю, а також розробити стратегії для їх оптимізації.

У даному дослідженні ми розглянемо процес побудови Z-моделей для визначення ймовірності банкрутства ТОВ «ROZETKA» та проаналізуємо отримані результати. Важливим етапом буде визначення проблемних зон у фінансовому управлінні компанії та розробка пропозицій щодо їх оптимізації. Дослідження виявиться корисним для розуміння та управління фінансовими ризиками в умовах сучасного бізнесу, а також для підвищення ефективності стратегічного планування та управління підприємством.

## ІНФОРМАЦІЙНО-АНАЛІТИЧНИЙ РОЗДІЛ

### 1.1 Теоретичні дані щодо машинного навчання

Машинне навчання — це сфера штучного інтелекту, яка розробляє алгоритми та статистичні моделі, які дозволяють комп'ютерам працювати без чітко визначених інструкцій, покладаючись на аналіз даних і розпізнавання образів. Іншими словами, це дозволяє комп'ютерам адаптуватися на основі накопиченого досвіду, як і люди.

Теорія машинного навчання включає завдання передбачення майбутньої поведінки складних систем, навіть за відсутності точних моделей, що описують поведінку подібних систем. Машинне навчання є важливим інструментом для багатьох компаній, який допомагає їм розвиватися, знаходити нові джерела доходу та вирішувати складні проблеми. Дані керують бізнес-рішеннями, і компанії часто отримують інформацію з таких джерел, як відгуки клієнтів, відгуки співробітників і фінансова статистика. Дослідження машинного навчання можуть посилити та покращити цей процес. Програмне забезпечення, яке швидко обробляє великі обсяги даних, дозволяє підприємствам швидше досягати результатів. Це підвищує гнучкість, ефективність і конкурентоспроможність, дозволяючи швидко й точно реагувати на зміни та потреби.

В даний час машинне навчання застосовується в багатьох сферах завдяки своїй універсальності та високій ефективності при вирішенні різноманітних завдань. Це особливо важливо в таких сферах:

#### 1. Виробництві

Машинне навчання використовується для профілактичного обслуговування, QA та інноваційних досліджень у виробництві. Ці технології допомагають компаніям покращити логістику, управління активами, ланцюг постачання та запаси. Наприклад, 3M AWS використовує машинне навчання для вдосконалення процесу виробництва наждачного

паперу. Використовуючи алгоритми машинного навчання, дослідники ЗМ можуть оцінити, як невеликі зміни у формі, розмірі та орієнтації частинок можуть покращити здатність продукту до дезінтеграції та довговічність. Це позитивно впливає на виробничий процес і допомагає розробляти кращу, більш конкурентоспроможну продукцію. Крім того, машинне навчання покращує автоматизований контроль якості, щоб швидко виявляти дефекти, зменшувати кількість браку та підвищувати загальну ефективність виробництва.

## 2. Охороні здоров'я та у медико-біологічних розробках

У медицині машинне навчання має багато застосувань, від діагностики захворювань до розробки нових ліків. Алгоритми обробляють великі обсяги медичних даних і дозволяють швидше й точніше діагностувати захворювання, наприклад раннє виявлення раку. У біологічних дослідженнях машинне навчання аналізує генетичні дані, щоб виявити нові біомаркери та розробити персоналізовані методи лікування. Наприклад, IBM Watson використовує машинне навчання для аналізу медичних записів і надання лікарям рекомендацій щодо лікування.

## 3. Фінансових послугах

У фінансовій галузі машинне навчання використовується для виявлення шахрайства, управління ризиками, автоматизації транзакцій і персоналізації банківських послуг. Алгоритми аналізують транзакції та виявляють підозрілу активність, щоб запобігти шахрайству. В управлінні ризиками машинне навчання може допомогти передбачити платіжні квитанції та проаналізувати якість клієнтів. Крім того, автоматизовані торгові системи використовують машинне навчання для прийняття інвестиційних рішень у режимі реального часу, щоб максимізувати прибуток і мінімізувати ризик. Наприклад, JPMorgan Chase використовує алгоритми машинного навчання для автоматизації торгівлі та аналізу ринкових даних.

#### 4. Роздрібній торгівлі

Машинне навчання широко використовується в сучасній галузі роздрібною торгівлі для покращення обслуговування клієнтів, мерчандайзингу та багатоканального маркетингу. Наприклад, Amazon Fulfillment (AFT) зміг знизити витрати на інфраструктуру на 39%, використовуючи потужні моделі машинного навчання для виявлення непотрібних запасів. Це дозволило компанії Amazon виконувати дану обіцянку, забезпечувати всім клієнтам прозорі покупки та своєчасну доставку, незважаючи на великий обсяг поставок, які вона обробляє.

#### 5. Мультимедіа та розвагах

У мультимедіа та розвагах машинне навчання використовується для створення персоналізованого контенту, покращення взаємодії з користувачем і розробки нових форм розваг. Алгоритми аналізують поведінку користувачів і рекомендують контент, який може їх зацікавити, наприклад фільми, музику та книги. Приклади використання машинного навчання для створення персоналізованих рекомендацій включають Netflix і Spotify. Крім того, машинне навчання використовується в розробці відеоігор для створення реалістичних інтерактивних світів. Наприклад, віртуальні персонажі можуть навчатися та коригувати свою поведінку на основі рухів гравця, що робить гру більш захоплюючою та непередбачуваною [1].

### 1.2 Принципи роботи машинного навчання

Основна ідея сучасного машинного навчання полягає у встановленні математичних залежностей між будь-якою комбінацією вхідних та вихідних даних. Хоча сучасна модель машинного навчання заздалегідь не знає цих зв'язків, вона здатна їх згенерувати, якщо буде надано достатньо наборів цифрових даних. Це означає, що будь-який алгоритм машинного навчання базується на адаптованій математичній функції, наприклад:

1. Під час навчання алгоритму передбачте комбінації введення та виведення у вигляді [input/output (i,o)]: (2,10), (5,19) та (9,31).
2. Алгоритм встановлює зв'язок між введенням і виведенням, наприклад так:  $o = 3 \times i + 4$ .
3. Потім вводиться значення 7 і попросимо алгоритм передбачити результат. Алгоритм автоматично визначає вихідне значення 25.

Це базове пояснення, але суть машинного навчання полягає в тому, що можна інтегрувати дані будь-якої складності, якщо є достатньо даних і обчислювальних ресурсів для їх обробки. Таким чином, точність вихідних прогнозів прямо пропорційна величині кількості наданих вхідних даних [2].

### 1.3 Види машинного навчання

Існує багато варіантів методів машинного навчання, включаючи декілька основних таких як: Контрольоване навчання (supervised learning), Неконтрольоване навчання (unsupervised learning), Навчання з підкріпленням (reinforcement learning), Частково контрольоване навчання (Semi-Supervised learning) та Глибинне навчання (Deep learning).

Кожен із цих методів є унікальним, адже має свої переваги та недоліки та використовується залежно від бажаних результатів.

#### 1. Контрольоване Навчання

Метод контрольованого навчання є оптимальним, коли ми точно знаємо, що саме потрібно навчитися машині. Цей метод дозволяє використовувати великі обсяги даних для навчання комп'ютера та коригування параметрів за потреби для досягнення бажаних результатів. Після завершення навчання ми можемо точно перевірити, чому навчилася машина, попросивши її передбачити результати на основі даних тестування, які раніше не використовувалися.

Цей метод зазвичай використовується для класифікації та прогнозування. Наприклад, у сфері фінансів він може допомогти визначити фінансові ризики для окремих осіб і організацій на основі минулої фінансової діяльності. Крім того, системи, навчені за допомогою цього підходу, можуть робити розширені прогнози купівельної поведінки на основі минулих тенденцій.

## 2. Неконтрольоване Навчання

Під час неконтрольованого навчання машина аналізує заданий набір даних, щоб виявити можливі зв'язки та закономірності між різними змінними без використання спеціальних умов навчання. Цей метод часто використовується для автоматичної класифікації даних на основі статистичних властивостей.

Одним із найпоширеніших застосувань неконтрольованого навчання є алгоритми кластеризації, які ранжують набори даних на основі подібності. Алгоритм виявляє можливі зв'язки між елементами даних, дозволяючи виявляти зв'язки між людьми та компаніями в реальному та віртуальному світі. Цей підхід дуже привабливий для компаній, які мають великі обсяги даних з різних джерел і різних відділів, оскільки допомагає створити цілісне уявлення про клієнтську базу [2].

Неконтрольоване навчання також може використовувати самостійне навчання для аналізу аудіо та визначення емоційного стану людей на основі їх активності в соціальних мережах.

Сьогодні компанії, що спеціалізуються на фінансових послугах, часто використовують неконтрольоване навчання для вимірювання задоволеності клієнтів і ймовірності повернення клієнтів.

## 3. Навчання з Підкріпленням

Навчання з підкріпленням в машинному навчанні є одним із найефективніших і практичних способів змусити машини виконувати різні

завдання за допомогою ітераційного процесу та перевірки правильності вибраних операцій.

Під час цього процесу машина може взаємодіяти з навколишнім середовищем, наприклад, поміщати дефектні продукти з конвеєрної стрічки в кошик, а потім отримувати «винагородження», коли робота виконана ідеально. Автоматизовані розрахунки винагород дозволяють машинам почати навчатися та використовувати це як основу для подальшого покращення продуктивності.

Одним із найпоширеніших застосувань цього типу машинного навчання є класифікація товарів у магазинах. Там потрібно швидко й ефективно сортувати велику кількість однотипних товарів.

Деякі роздрібні торговці вже експериментують з автоматизованими системами вибору товарів, зокрема для одягу, взуття та аксесуарів. Завдяки навчанню на основі винагороди робот може точно визначати оптимальну ступінь стиснення під час захоплення об'єктів і вибирати найкращий тип захоплення для кожного об'єкта, допомагаючи зробити процес сортування більш ефективним.

#### 4. Частково Контрольоване Навчання

Спосіб частково контрольованого навчання – це поєднання навчання під наглядом і навчання без нагляду. Люди позначають невелику частину даних, а машина знає, як сортувати решту.

Цей метод зазвичай використовується для розслідування різних типів шахрайства, де спроби представити щось підроблене або неправдиве відрізняються від нормальної, стабільної активності. Машинне навчання з мікроконтролем дозволяє створювати моделі, які виявляють ті самі аномалії з часом, класифікуючи ті самі дані та знаходячи відмінності. Відповідні системи також все частіше використовуються для виявлення шахрайства під час купівлі та продажу в Інтернеті.

Даний метод також часто застосовується у випадках, коли у людей вже є готові набори даних часткової розмітки, особливо це характерно для великих підприємств.

Яскравим прикладом використання цього підходу є компанія Amazon, яка використовує частково контрольоване навчання для свого голосового асистента Alexa. Цей підхід дозволяє алгоритмам штучного інтелекту Alexa розуміти природу мови на основі комбінації позначених і не позначених даних, що забезпечує точніші відповіді та краще розуміння мови користувачів.

## 5. Глибинне Навчання

Глибинне навчання стало важливим підходом серед різноманітних методів машинного навчання. Цей метод можна використовувати як без участі людини, так і з підкріпленням.

Глибинне навчання використовує нейронні мережі, які імітують методи людського навчання, щоб покращити властивості даних. Нейронні мережі глибинного навчання часто використовуються для швидкого аналізу великих обсягів даних, наприклад, під час пошуку нових ліків. Він може обробляти велику кількість зображень і виявляти важливі функції, які важко виявити іншими методами [2].

Ця технологія машинного навчання також широко використовується для виявлення шахрайства, допомагаючи фільтрувати дані та підвищувати точність виявлення помилок. Глибинне навчання використовується в автомобільній промисловості для виконання різних завдань, таких як перевірка та ремонт транспортних засобів [3].

Глибинне навчання відкриває широкі можливості для розвитку штучного інтелекту та обробки даних. Складні нейронні мережі та алгоритми можна використовувати для вирішення завдань у різноманітних областях, включаючи розпізнавання образів, обробку мови та прогнозування.

#### 1.4 Типи машинного навчання

Загалом, існує два основних типи навчання: індуктивне (на основі прецедентів) і дедуктивне. Деякі індуктивні методи навчання були розроблені як альтернатива традиційним статистичним методам.

При індуктивному навчанні, яке також називають навчанням на основі процентів, система вивчає шаблони та правила на основі конкретних прикладів. Тоді як ми використовуємо отримані знання для узагальнення та висновків на основі спостережень, тематичне навчання робить висновки щодо конкретних ситуацій на основі загальних принципів і правил. Цей метод використовується, коли відомі або встановлені правила застосовуються до нових ситуацій для досягнення результату.

Перший тип навчання характеризується великою кількістю даних і відсутністю або непотрібністю попереднього досвіду. Це означає, що модель навчається на великій кількості різних прикладів без огляду на попередній досвід або знання. З іншого боку, другий тип навчання характеризується невеликою кількістю даних або невеликим вибраним набором даних. Такий підхід вимагає глибокого розуміння предмета чи теми, що вивчається. Він заснований на аналізі невеликих обсягів інформації, але з великою увагою до деталей і характеристик [4].

Машинне навчання використовує дві різні техніки для внутрішнього аналізу. Статистика, з іншого боку, використовує менше підходів і дуже консервативна у своїх методах. Дедуктивний тип часто асоціюється з експертними системами, звідси загальний термін «машинне навчання» для штучного навчання. Передумова або навчальна вибірка — це набір вхідних об'єктів і відповідних їм виходів. Наразі не існує чіткої формули, яка б аналітично описувала зв'язок між усіма виходами та вхідними даними.

Наприклад, для прогнозування погоди на завтра, потрібно врахувати багато параметрів, таких як географічні координати, рельєф місцевості та переміщення теплих і холодних вітрів, тощо. Точність результатів у цьому

випадку визначається функцією оцінки якості. Рішення емпірично створюються на основі аналізу накопиченого досвіду. Навчальні системи повинні мати можливість узагальнювати та відповідним чином реагувати на дані поза наявними навчальними прикладами [4].

На практиці вхідні дані можуть бути неповними, неправильними або різнорідними. Зараз, хоча і існує багато видів методів машинного навчання, можна сказати, що машинне навчання завжди використовує метод аналогії, який називається Case Based Reasoning (CBR) - це спосіб робити висновки з подібних випадків (прикладів) і розв'язувати проблеми за аналогією.

### **1.5 Машинне навчання та статистика**

Машинне навчання та статистика мають багато спільних властивостей і характеристик, через що їх часто можуть плутати, порівнювати під час використання. Основна відмінність між статистикою і машинним навчанням полягає в їх підході та меті. Статистика використовує дані для пояснення випадкових явищ за допомогою методів, які досліджують структуру даних. Машинне навчання передбачає розробку алгоритмів, які можуть вивчати дані та робити прогнози без явних правил.

Зараз існує багато моделей, які роблять прогнози на основі статистичних даних, але ці прогнози часто не дуже точні. Подібним чином моделі машинного навчання мають різні рівні розуміння, від чітко визначеної Регресії Ласо до складних нейронних мереж, але навіть останні загалом мають високу прогностичну силу, хоча для цього і жертвують зрозумілістю. Щоб пояснити різницю між традиційним машинним навчанням і статистичними моделями, ми можемо використати приклад лінійної регресії.

Лінійна регресія – це статистичний метод, який використовується для дослідження зв'язку між двома чи більше змінними. Цей метод дозволяє встановити лінійну залежність між змінними і передбачити значення однієї

змінної на основі значень інших змінних. Лінійна регресія намагається знайти лінійні параметри (або гіперплощини в багатовимірному просторі), які мінімізують суму квадратів відхилень прогнозованих значень моделі від фактичних значень. Тут використовуються такі методи, як найменші квадрати, щоб знайти пряму лінію, яка мінімізує відхилення між фактичним і прогнозованим значенням залежної змінної. У цьому випадку основною метою машинного навчання є отримання найкращих результатів на тестовому наборі. [5].

Статистичні моделі дозволяють знаходити пряму лінію, яка найкраще відповідає зв'язку між змінними, і робити точні прогнози. Цей процес знаходить лінію, яка найкраще представляє зв'язок між змінними. У цьому випадку немає необхідності використовувати навчальні та тестові набори. Часто, особливо в дослідженнях, основною метою моделі є не прогнозування майбутніх даних, а дослідження зв'язку між даними та змінами в результатах.

Цей процес називається статистичним висновком, а не прогнозом. Однак за допомогою моделі можна робити прогнози, і в деяких випадках прогнозування може бути основною метою. Однак замість того, щоб оцінювати модель за допомогою стандартизованих тестів, ми оцінюємо значущість і надійність параметрів моделі.

Основна мета навчання машинного навчання – знайти моделі, які можуть робити систематичні прогнози. Хоча інтерпретація моделі, як правило, менш важлива, все ж важливо перевірити її, щоб переконатися, що прогнози моделі дійсно мають сенс. Підсумувати описане вище можна такими словами: Машинне навчання є ефективним підходом для компаній, які потребують точних прогнозів. З іншого боку, статистичне моделювання менше пов'язане з прогнозуванням, а більше з вивченням зв'язків між змінними та визначенням їх значущості.

Незважаючи на відмінності між двома методами, вони мають багато спільного. Бо слід зауважити, що машинне навчання часто базується на статистиці, оскільки воно передбачає обробку даних.

Сучасні технологічні відкриття немислимі без використання машинного навчання, ефективність якого стає все більш визнаною порівняно зі статистичними методами та статистикою в цілому.

Багато хто вважає, що статистичні методи застаріли, але насправді вони все ще широко використовуються, хоча часто для досягнення інших цілей.

Коли компаніям і організаціям потрібно розробити алгоритми, які можуть точно прогнозувати ціни на нерухомість та їх коливання, або аналізувати дані для визначення ймовірності певної хвороби, машинне навчання є найкращим вибором. Однак, якщо ви хочете встановити зв'язки між різними змінними або зробити висновки з даних, краще використовувати статистичні моделі.

Машинне навчання тісно пов'язане з аналізом даних і статистичними методами. Методи аналітики допомагають зрозуміти та створити шаблони даних для навчання моделей машинного навчання. Крім того, визначено достовірність та статистичну значущість отриманих результатів.

## **1.6 Регресія у машинному навчанні**

Основною метою машинного навчання є виявлення зв'язків між різними факторами, які можна виявити (іноді їх називають регресорами або ознаками), і конкретним результатом, який ми хочемо передбачити. Наприклад, якщо у вас є інформація про дозу ліків і температуру тіла пацієнта, то можна передбачити, як це вплине на артеріальний тиск. У цьому випадку дозування та температура є тим, що ми спостерігаємо, а артеріальний тиск - тим, що ми хочемо передбачити.

Задача встановлення зв'язку між парами цих змінних називається регресією. Оскільки зв'язок між спостереженнями та цільовими результатами може бути складним, ми використовуємо моделі, щоб спробувати оцінити значення цільової змінної на основі підмножини можливих моделей. Однією з найпростіших моделей є лінійна модель. У машинному навчанні модель — це математична функція, яка збирає спостереження, обробляє їх і поєднує з функцією регресії для прогнозування значення цільової змінної [6].

Лінійна регресія — це модель машинного навчання, заснована на припущенні, що залежна змінна має лінійний зв'язок із незалежними змінними. У цій моделі зміни значення однієї змінної змінюються пропорційно і завжди впливають на зміни значень інших змінних. Це найбазовіша модель, яка може пояснити багато різних ситуацій. Крім того, лінійна регресія досить добре вивчена і має чимало корисних теоретичних характеристик, і завдяки своїй простоті легко піддається інтерпретації.

Лінійна модель — це інструмент, який можна використовувати для дослідження зв'язку між незалежними та залежними змінними. Ця залежність показує, що її можна представити у вигляді лінійної функції, тобто прямої лінії на графіку. Ця простота робить результати легшими для розуміння, легшими для інтерпретації та дозволяє краще аналізувати зв'язки в даних. Загальний вигляд лінійних функцій однієї змінної може бути записаний наступним чином:

$$h(x) = k * x + b \quad (1.1)$$

де  $k$  і  $b$  - довільні числа. Приклад лінійної залежності - залежність відстані від часу ( $s = v*t$ ).

Діаграма, що ілюструє використання лінійної моделі, а саме, пряму лінію однієї змінної, надає можливість аналізу взаємозв'язку між різними вимірюваними факторами, показана на рисунку 1.1.

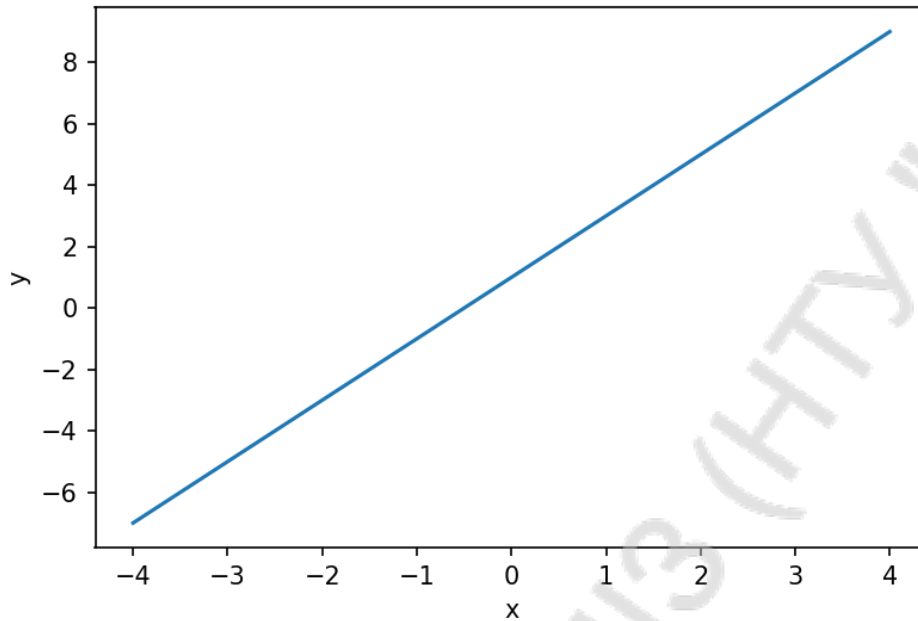


Рисунок 1.1 – Графік лінійної функції однієї змінної

У разі, коли є кілька спостережуваних змінних, лінійні функції мають вигляд:

$$h(x_1, x_2, \dots, x_N) = a_0 + a_1 * x_1 + a_2 * x_2 + \dots + a_N * x_N, \quad (1.2)$$

де  $a_0, a_1, \dots, a_N$  - також довільні числа. Графік цієї функції - теж пряма, але вже в N-вимірному просторі, приклад графіку можна побачити на рисунку 1.2.

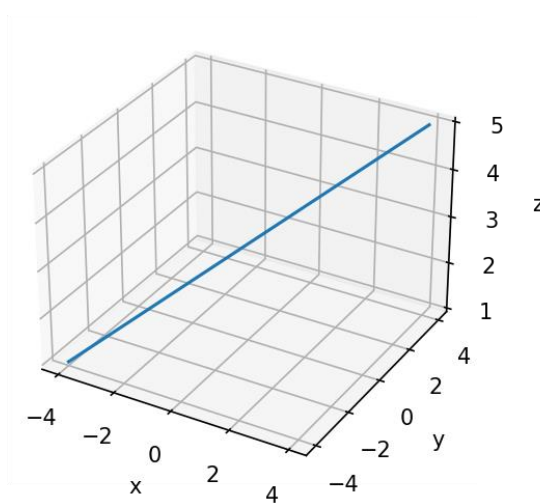


Рисунок 1.2 – Графік лінійної функції двох змінних

Підсумовуючи, ми робимо наступні висновки:

Лінійна регресія — це метод, який дозволяє вивчати зв'язок між спостережуваною змінною та цільовою змінною за допомогою лінійної залежності.

Існують спеціальні моделі, які враховують доступні дані, тобто набір значень  $x$ -ів (називаються спостережуваними змінними) і відповідних значень  $y$ -ів (цільова змінна).

Оптимальним кроком є вибір найкращих коефіцієнтів для кожної змінної, щоб визначити їх вплив на модель. Якість змінної визначається функцією втрат, яка вказує, наскільки точно прогнози моделі порівнюються з цільовим значенням. Процес підбору коефіцієнтів називається навчанням моделі і здійснюється за допомогою спеціальних алгоритмів оптимізації.

Ці параметри, що використовуються в конкретній моделі, називаються вагами. Вони вказують на важливість кожної змінної для прогнозу. Чим більше значення ваги, тим більший вплив відповідної змінної на передбачення. Це робить лінійну регресію інтуїтивно зрозумілою моделлю, оскільки вона дозволяє визначити, які фактори важливі, а які ні.

Після завершення процесу навчання модель готова до використання - потрібно використати отриману функцію (модель), ввести в неї набір цікавих для нас  $x$ -змінних і обчислити значення, які і будуть представляти наші прогнози. [7].

## **1.7 Існуючі алгоритми машинного навчання**

### **1. Лінійна регресія**

Лінійна регресія - це статистичний метод, який досліджує зв'язок між однією або кількома незалежними змінними та залежною змінною, припускаючи, що між ними існує лінійний зв'язок. Лінійну регресію можна виразити у вигляді рівняння, яке представляє пряму лінію, яка найбільш точно описує зв'язок між вхідною змінною  $X$  і вихідною змінною  $Y$ .

Основна ідея полягає в мінімізації різниці між фактичними та передбаченими значеннями залежної змінної для створення лінійної моделі, яка найкраще відповідає спостережуваним даним. Фактичні та прогнозовані залежні змінні. Лінійна регресія знаходить оптимальні параметри цієї лінійної залежності, які найкраще описують зміни залежної змінної від незалежних змінних. Таким чином, цей метод дозволяє використовувати просту лінійну модель для прогнозування значення залежної змінної на основі інших змінних. Результат побудови подібного рівняння буде мати вигляд подібний до рисунку 1.3.

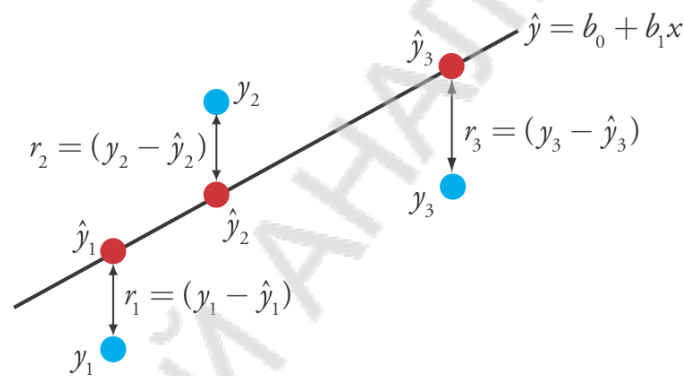


Рисунок 1.3 – Приклад Лінійної Регресії

Функція  $Y = B_0 + B_1 * X$  використовується, щоб знайти значення  $Y$  за відомим значенням  $X$ . Основна мета лінійної регресії - знайти значення коефіцієнтів  $B_0$  і  $B_1$ .

Можна оцінити моделі регресії за допомогою різних методів, зокрема лінійної алгебри та методів найменших квадратів.

За допомогою лінійної регресії можна використовувати деякі практичні прийоми для підвищення точності прогнозів. Якщо можливо, важливо усунути спільні (корелюючі) фактори та зменшити шум у даних [7].

Лінійна регресія є швидким і простим методом, який чудово підходить як перший алгоритм для вивчення [8].

## 2. Логістична регресія

Логістична регресія - це статистичний метод, який використовується для моделювання відносної ймовірності події, пов'язаної з кількома залежними змінними. Функція може бути представлена символом "S" і може перетворювати будь-яке значення в числові значення які знаходяться в діапазоні від 0 до 1. Це важливо, оскільки це правило дозволяє призначати значення 0 і 1 виходу логістичної функції (наприклад, якщо результат функції менше 0,5, класифікувати його як 1) і робити прогнози класу. Основна ідея полягає у використанні логістичної функції для перетворення результатів у відповідний діапазон і пошуку найкращої лінійної функції, яка може розділити два класи даних [9]. Логістична регресія використовує методи оптимізації, щоб знайти параметри лінійної функції, яка найкраще відповідає даним. Таким чином, цей метод дозволяє нам передбачити ймовірність події та класифікувати об'єкти на основі цієї ймовірності. Приклад побудови логістичної функції зображений на рисунку 1.4.

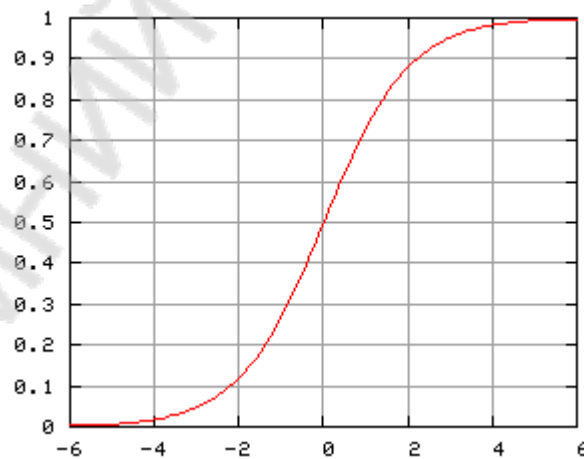


Рисунок 1.4 – Приклад логістичної функції

Логістичний регресійний прогноз може вказати на ймовірність того, що зразок належить до класу 0 або класу 1 завдяки методу навчання. Це корисно, коли для прогнозу потрібно більше даних.

Як і лінійну регресію, логістичну регресію найкраще використовувати, коли видаляються надлишкові або схожі фактори. Логістична регресія має швидко навчається та підходить для задач бінарної класифікації [9].

### 3. Лінійний Дискримінантний Аналіз(LDA)

Лінійний дискримінантний аналіз (LDA) - це метод статистичного машинного навчання, який використовується для визначення ліній або площин, які найкраще відрізняють два або більше типів об'єктів або подій. Основна ідея полягає в тому, щоб збільшити варіативність між класами та зменшити її всередині кожного класу, щоб краще диференціювати групи. LDA шукає в просторі ознак напрямки, у якому оцінка даних найкраще підходить для класифікації. Таким чином, LDA можна використовувати для зменшення розміру даних і полегшення подальшого аналізу без значних втрат інформації. Цей підхід часто використовується в задачах розпізнавання образів і системах рекомендацій, де важливо враховувати відмінності між категоріями даних [9].

Загальний вид отриманих результатів у вигляді дисперсії можливо отримати завдяки великій кількості програм, зазвичай вона буде мати вигляд подібний до рисунку 1.5.

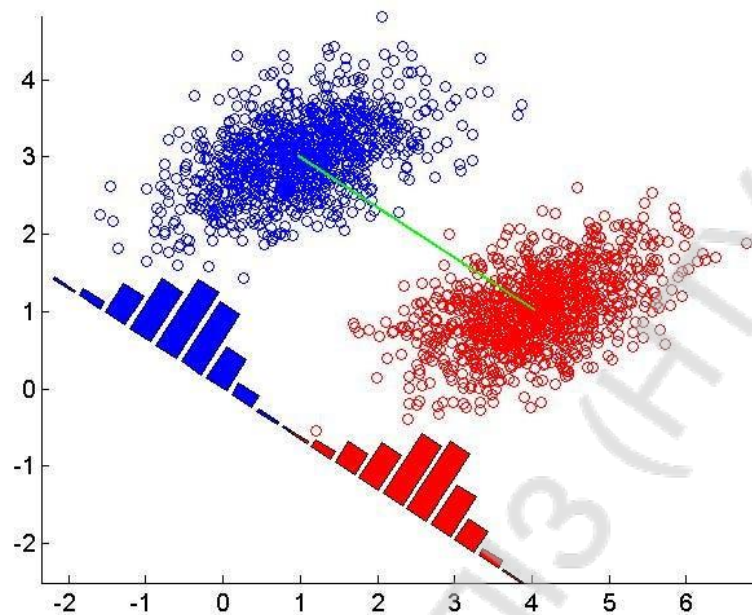


Рисунок 1.5 – Приклад роботи Лінійного Дискримінантного Аналізу

Прогноз обчислює значення кожного класу та вибирає клас з найбільшою вагою. Для цього важливо, щоб дані були нормально розподілені. Тому перед початком роботи рекомендується перевірити та виправити аномалії.

Лінійний дискримінантний аналіз є простим і ефективним методом класифікації в задачах машинного навчання [9].

#### 4. Дерево прийняття рішень

Дерево прийняття рішень - це метод машинного навчання, який використовується для створення моделей класифікації та регресії. Основна ідея цього підходу полягає в тому, щоб розділити дані на менші частини на основі певних умов або правил, що дозволяє приймати рішення на кожному рівні дерева. Дерева рішень знаходять найкращі точки розділення, щоб мінімізувати варіацію або дисперсію в кожній підмножині даних. У результаті цей підхід може чітко виразити та візуалізувати процеси прийняття рішень, роблячи навіть складні завдання інтуїтивно зрозумілими. Приклад найпростішого дерева наведений на рисунку 1.6.

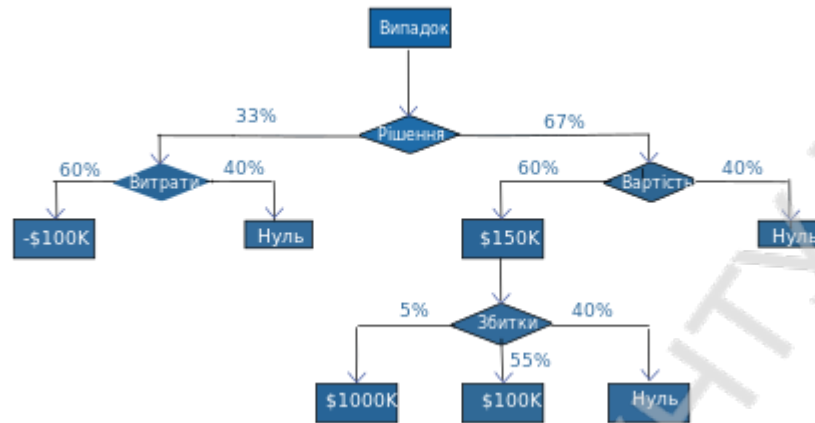


Рисунок 1.6 – Приклад Дерева Прийняття Рішень

Термін "листові вузли" відноситься до кінцевої точки дерева, що використовується для прогнозування. Під час процесу прогнозування модель рухається від кореня до кожного термінального вузла та визначає клас, що відповідає цьому вузлу [10].

Дерева характеризуються швидкістю навчання та ефективністю передбачення. Крім того, вони часто дуже точні та не потребують спеціальної підготовки даних.

## 5. Наївний Байєсівський Класифікатор

Наївний Байєс - це простий, але потужний метод машинного навчання, заснований на застосуванні теореми Байєса, припускаючи незалежність ознак. Основна ідея полягає в тому, щоб оцінити ймовірність того, що об'єкт належить до певної категорії за наявними атрибутами, навіть якщо ці атрибути вважаються незалежними. Наївний Байєсівський Класифікатор знаходить найімовірніший клас на основі атрибутів кожного об'єкта. Таким чином, цей метод забезпечує ефективну та швидку класифікацію та особливо ефективний для великих наборів даних і завдань класифікації тексту [9].

Якщо дані неперервні, то можна використовувати нормальний розподіл для обчислення ймовірностей. Принцип роботи наївного байєсівського класифікатора можна побачити на прикладі, зображеному на рисунку 1.7.

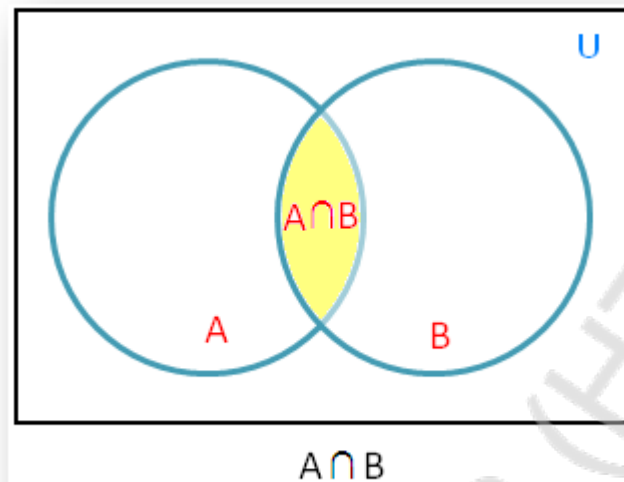


Рисунок 1.7 – Приклад Байєсівського Класифікатора

Термін "наївний" Байєсівський класифікатор походить від припущення, що всі вхідні елементи незалежні. Це припущення може бути занадто спрощеним і не завжди відповідає фактичним даним. Однак цей прийом ефективний для вирішення багатьох складних завдань, таких як фільтрації спаму або розпізнавання рукописних цифр і символів [9].

#### 6. K-найближчих сусідів (KNN)

Алгоритм K-найближчих сусідів (KNN) - це проста та інтуїтивно зрозуміла техніка машинного навчання для класифікації та пошуку. Основна ідея полягає в тому, щоб класифікувати об'єкти відповідно до категорій їхніх найближчих сусідів у просторі ознак. KNN знаходить k найближчих точок у навчальному наборі даних і визначає тип нового об'єкта більшістю голосів цих сусідів. Таким чином, цей метод забезпечує простий і ефективний метод прогнозування, особливо коли дані добре розділені в просторі ознак, але він може бути обчислювально витратним для великих наборів даних. Роботу алгоритму KNN можна побачити на рисунку 1.8. [10].

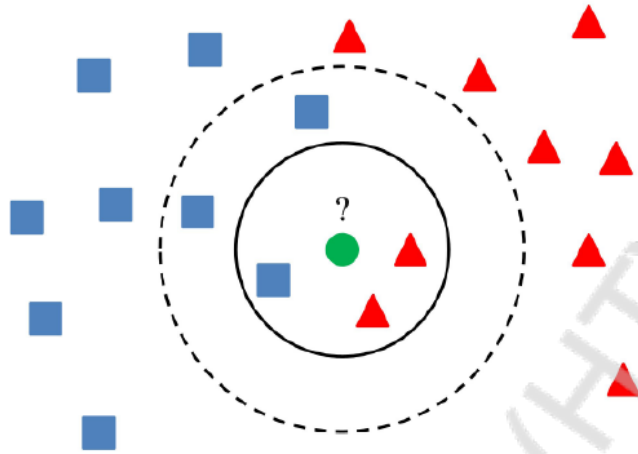


Рисунок 1.8 – Приклад K-найближчих сусідів

Незважаючи на те, що алгоритму KNN потрібно багато пам'яті для зберігання всіх минулих даних, прогнози він може генерувати швидко. Крім того, навчальні дані можна регулярно оновлювати для забезпечення точних прогнозів у майбутньому.

У випадку наборів даних з багатьох вимірів концепція найближчого сусіда зазвичай виявляється неефективною через велику кількість вхідних змінних. Ця ситуація відома як прокляття розміру і може негативно вплинути на ефективність вирішення проблем алгоритмом. Тому для прогнозу рекомендується використовувати лише найважливіші параметри.

#### 7. Мережі Векторного Квантування (LVQ)

Мережі Векторного Квантування (LVQ) - це техніка машинного навчання для класифікації на основі принципів модельного навчання. Основна ідея полягає в тому, щоб представити кожен клас набором прототипів, типових для цього класу, і використовувати ці прототипи для класифікації нових об'єктів. LVQ шукає найкращу позицію прототипу в просторі ознак, щоб мінімізувати помилку класифікації. Цей алгоритм працює найкраще, коли всі дані знаходяться в одному діапазоні, наприклад, від 0 до 1. Таким чином, цей метод забезпечує зручний та інтуїтивно зрозумілий метод класифікації, який дозволяє вам агрегувати дані та робити точні прогнози на основі невеликої кількості корисних зразків [9].

Роботу алгоритму LVQ можна побачити на рисунку 1.9.

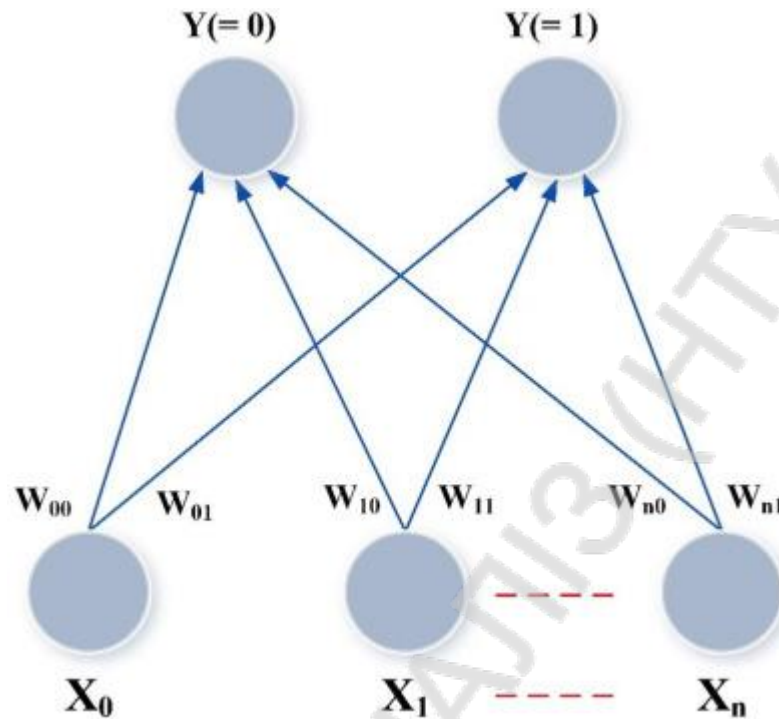


Рисунок 1.9 – Приклад Мереж Векторного Квантування

Після завершення процесу навчання можна використовувати метод LVQ для класифікації нового набору даних. Для цього вхідний зразок порівнюється з набором кодових векторів і визначається кодовий вектор, до якого він найбільш схожий. Передбачається, що прогнозований клас вхідного зразка є класом, найближчим до цього кодового вектора.

## 8. Метод Опорних Векторів (SVM)

Метод Опорних Векторів (SVM) - це потужний метод машинного навчання для класифікації та регресії. Основна ідея цього методу полягає в тому, щоб знайти гіперплощину в багатовимірному просторі ознак, яка ефективно розділяє класи об'єктів шляхом збільшення відстані (або «зазор») між суміжними точками різних класів (так званими опорними векторами). Ця гіперплощина розділяє точки даних у просторі за їх класом, наприклад, класом 0 або 1. SVM шукає цю найкращу комбінацію поділу, щоб забезпечити найточнішу класифікацію. Таким чином, цей метод забезпечує високу точність прогнозів, оскільки він може добре розрізняти класи та

отримувати більш значущі розділові межі навіть у випадках високої складності [10].

Графічний приклад роботи методу Опорних Векторів зображений на рисунку 1.10.

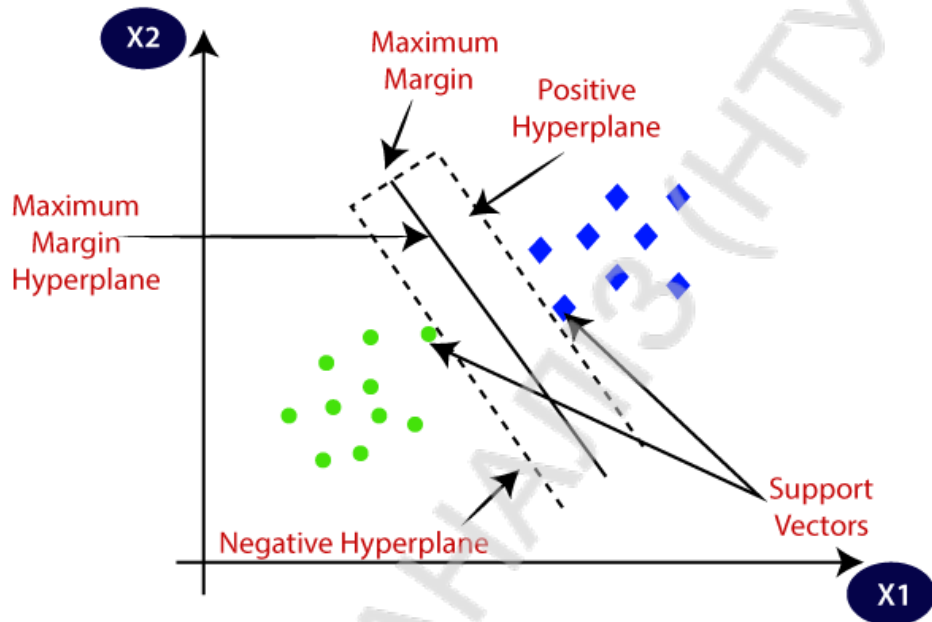


Рисунок 1.10 – Приклад Методу Опорних Векторів

Відстань між гіперплощиною та найближчою точкою даних визначається шляхом вимірювання їх різниці. Оптимальною гіперплощиною, яка розділяє ці два класи, є лінія максимального розриву. Тільки ці точки важливі для визначення гіперплощини та створення класифікатора, які називають опорними векторами. Для визначення значення коефіцієнта збільшення розриву використовується спеціальний алгоритм оптимізації.

Метод опорних векторів є одним із найбільш ефективних стандартних класифікаторів, які слід враховувати.

## 9. Беггінг та Випадковий Ліс

Беггінг (Bagging) та Випадковий Ліс (Random Forest) - це два популярних методи навчання, які використовуються для підвищення точності моделей машинного навчання. Основна ідея беггінгу полягає в

тому, щоб побудувати кілька моделей (зазвичай дерев рішень) на різних підмножинах даних і поєднати їхні прогнози для зменшення мінливості та покращення узгодженості результатів.

Випадковий ліс — це розширення беггінгу, яке вводить додаткові міркування у вибір функцій під час побудови кожного дерева та може зменшити кореляції між окремими деревами. Разом ці методи здатні створювати потужні та надійні моделі, ефективно зменшувати зайве навчання та покращувати загальну точність прогнозування [8].

Тренувальні дані розділяються на безліч випадкових підвбірок, для кожної з яких створюється окрема модель, цей процес можна побачити на рисунку 1.11.

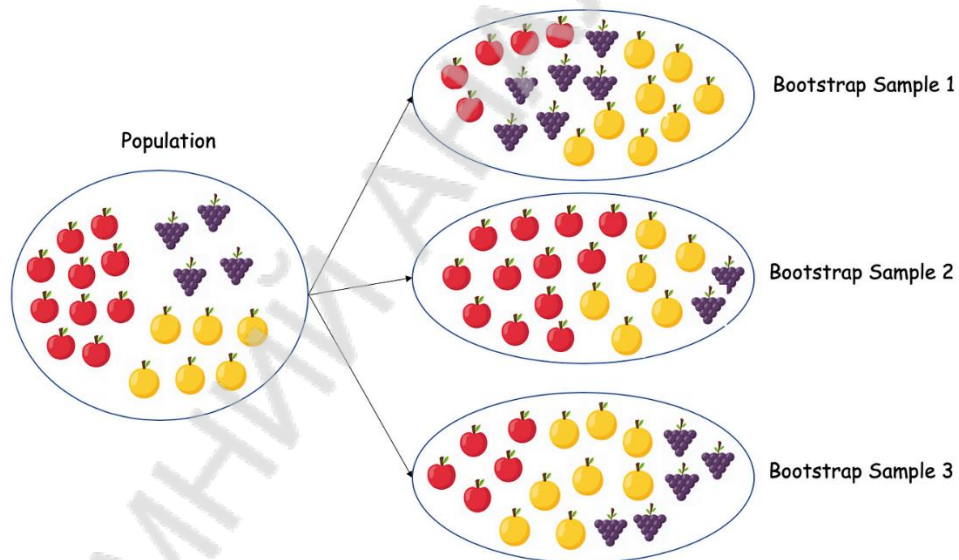


Рисунок 1.11 – Приклад Беггінгу та Випадкового Лісу

У алгоритмі випадкового лісу дерево створюється на основі кожної випадкової вибірки навчальних даних і використовується для прийняття рішень. Під час побудови кожного дерева елементи кожного вузла вибираються випадковим чином. Хоча окремі моделі недостатньо точні, їх поєднання може значно покращити прогнози.

Якщо алгоритми з великими відмінностями між прогнозами, наприклад дерева рішень, добре працюють із набором даних, часто покращити ці результати можна за допомогою методу бегтінгу.

## 10. Бустинг і AdaBoost

Бустинг - це метод навчання, який поєднує кілька слабких моделей для створення надійної моделі для покращення класифікації або точності пошуку. Основна ідея полягає в тому, що кожна нова модель враховує, де попередня модель припустилася найбільших помилок, і компенсує ці помилки, тим самим підвищуючи загальну точність [9].

AdaBoost є одним із найпоширеніших варіантів бустингу, де кожна наступна модель навчається на найважливіших даних для правильної класифікації за вагами. Цей процес триває, доки не буде досягнуто певної точності або іншої межі. Таким чином, бустинг можна використовувати для створення складних і потужних моделей, які є високоточними та можуть забезпечити надійні прогнози навіть на дуже складних даних. Процес перевірки першого дерева зображений на рисунку 1.12.

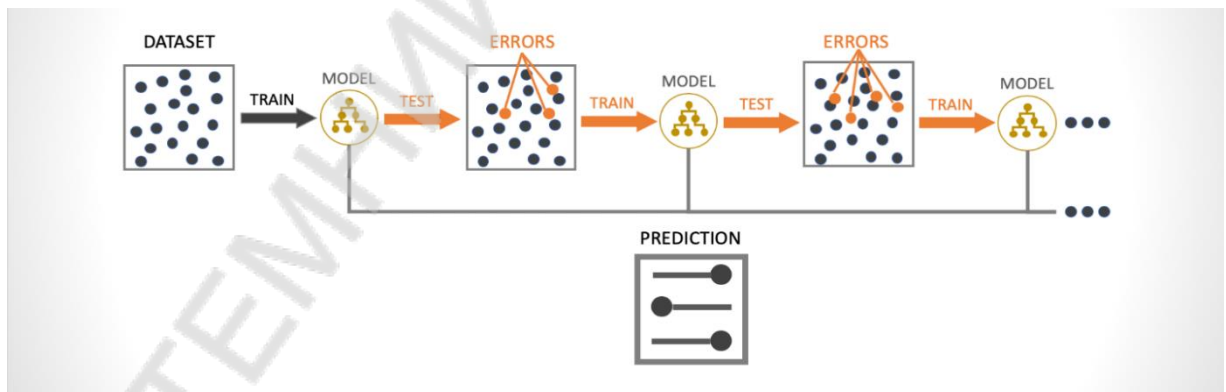


Рисунок 1.12 – Приклад Бустингу і AdaBoost

Процес створення проходить серію ітерацій, на кожному кроці адаптуючи ваги для наступної моделі. Після завершення всіх ітерацій моделі використовуються для прогнозування нових даних, а продуктивність кожної моделі оцінюється на основі її точності на навчальних даних.

Оскільки цей метод активно коригує помилки в моделі, тому важливо, щоб вхідні дані були чистими та без викидів.

## СПЕЦІАЛЬНИЙ РОЗДІЛ

### **2.1 Побудова Z-моделей визначення ймовірності настання банкрутства ТОВ «ROZETKA» та визначення на основі них проблемних зон управління фінансовою діяльністю та їх оптимізація**

Сформулюємо постановку задачі наступним чином:

На основі Z-моделей (які отримані засобами регресійного аналізу) визначимо імовірність банкрутства ТОВ «ROZETKA» у найближчі 1-3 роки. Вхідні дані взяти з відкритих фінансових звітів.

Задача визначення ймовірності банкрутства найбільшого підприємства галузі є актуальною, адже достовірна й своєчасна ідентифікація негативних факторів впливу на фінансово-господарську діяльність компанії дає можливість сформувати відповідну фінансову політику та розробити заходи, які направлені на підвищення ефективності використання фінансових ресурсів. А також підвищити платоспроможність, забезпечити конкурентоспроможність у довгостроковому періоді, а головне – попередити ризик банкрутства підприємства. Діагностика банкрутства – це своєчасне виявлення неплатоспроможності, збитковості, фінансової залежності від зовнішніх джерел фінансування, низької ділової активності.

Причини банкрутства підприємств можуть бути найрізноманітнішими. Їх можна розділити на дві групи:

- зовнішні, які практично дуже важко (іноді неможливо) врахувати;
- внутрішні, що безпосередньо залежать від форм, методів та організації роботи на самому підприємстві.

Усі методи діагностики банкрутства поділяються на кількісні та якісні. В дослідженнях найбільш поширеними серед кількісних методів прогнозування банкрутства є зарубіжні та вітчизняні факторні моделі, де факторами виступають окремі показники фінансової діяльності підприємства [11].

Після аналізу розроблених зарубіжних методик діагностики ймовірності банкрутства для розрахунків обрано п'ять моделей.

**1. П'ятифакторна модель Е. Альтмана (1968 рік).**

$$Z = 1,2X_1 + 1,4X_2 + 3,3X_3 + 0,6X_4 + X_5, \quad (2.1)$$

де  $X_1$  – відношення оборотного капіталу до валюти балансу;

$X_2$  – відношення нерозподіленого прибутку до валюти балансу;

$X_3$  – відношення валового прибутку до валюти балансу;

$X_4$  – відношення вартості власного капіталу до вартості всіх зобов'язань;

$X_5$  – відношення доходу від реалізації до валюти балансу.

Якщо значення  $Z < 1,8$  – тоді ймовірність банкрутства дуже висока (понад 80%);  $1,81 < Z < 2,7$  – висока (від 40% до 50%);  $2,71 < Z < 2,99$  – можлива (від 15% до 20%);  $Z > 3$  – дуже низька.

**2. Модель Р. Ліса (1972 рік).**

$$Z = 0,063X_1 + 0,092X_2 + 0,057X_3 + 0,001X_4, \quad (2.2)$$

де  $X_1$  – відношення оборотного капіталу до всіх активів;

$X_2$  – відношення прибутку від реалізації до всіх активів;

$X_3$  – відношення нерозподіленого прибутку до всіх активів;

$X_4$  – відношення власного капіталу до позичкового капіталу.

Якщо значення  $Z < 0,037$  – висока ймовірність банкрутства;  $Z = 0,037$  – граничне значення;  $Z > 0,037$  – низька ймовірність банкрутства.

**3. Модель Р. Таффлера (1977 рік).**

$$Z = 0,53X_1 + 0,13X_2 + 0,18X_3 + 0,16X_4, \quad (2.3)$$

де  $X_1$  – відношення прибутку від реалізації продукції до короткострокових зобов'язань;

$X_2$  – відношення оборотних активів до зобов'язань;

$X_3$  – відношення короткострокових зобов'язань до всіх активів;

$X_4$  – відношення власного капіталу до позичкового капіталу.

Якщо значення  $Z < 0,2$  – ймовірність банкрутства досить висока;  $0,2 < Z < 0,3$  – можливе банкрутство;  $Z > 0,3$  – ймовірність банкрутства низька.

#### 4. Модель Г. Спрінгейта (1978 рік).

$$Z = 1,03X_1 + 3,07X_2 + 0,66X_3 + 0,4X_4, \quad (2.4)$$

де  $X_1$  – частка оборотного капіталу в структурі активів;

$X_2$  – відношення чистого прибутку до виплати податків і відсотків до сукупних активів;

$X_3$  – відношення чистого прибутку до виплати відсотків до поточних зобов'язань;

$X_4$  – відношення доходу від реалізації продукції до всіх активів.

Якщо значення  $Z < 0,862$  – підприємство є потенційним банкрутом;  $Z = 0,862$  – граничне значення ймовірності банкрутства;  $Z > 0,862$  – відсутність ймовірності банкрутства. Після аналізу найбільш поширених зарубіжних методик діагностики ймовірності банкрутства можна зазначити, що основна їх перевага полягає в простоті розрахунків та можливості використання під час проведення зовнішнього діагностичного аналізу. Так п'ятифакторна модель Е. Альтмана відображає різні аспекти господарської діяльності підприємства. Вона надає можливість динамічного прогнозування змін фінансової стійкості та інтерпретації інтегрального показника. В моделі Г. Спрінгейта похибка прогнозу не перевищує 10%; тобто забезпечується достатній рівень надійності прогнозування ймовірності банкрутства.

Головним недоліком цих зарубіжних моделей є те, що вони розроблені за результатами дослідження підприємств США й Західної Європи. Вагові коефіцієнти не адаптовані до сучасних умов трансформації вітчизняної економіки й не враховують специфіку діяльності українських підприємств, а саме: систему бухгалтерського обліку, податкове законодавство, значний вплив інфляції на формування фінансових показників, галузеву приналежність підприємства тощо. Ці моделі мають кілька суттєвих

обмежень. Їх можна використовувати лише як додаткові моделі паралельно з сучасними вітчизняними моделями [4].

### 5. Модель О.О. Терещенка (2004 р.).

$$Z = 1,5X_1 + 0,08X_2 + 10X_3 + 5X_4 + 0,3X_5 + 0,1X_6, \quad (2.5)$$

де  $X_1$  – відношення грошових надходжень до зобов'язань;

$X_2$  – відношення валюти балансу до зобов'язань;

$X_3$  – відношення чистого прибутку до середньорічної суми активів;

$X_4$  – відношення прибутку до доходу від реалізації;

$X_5$  – відношення виробничих запасів до доходу від реалізації;

$X_6$  – відношення доходу від реалізації до основного капіталу.

Якщо  $Z > 2$  – банкрутство не загрожує,  $1 < Z < 2$  – фінансова стійкість порушена,  $0 < Z < 1$  – існує загроза банкрутства.

## 2.2 Мотивація та аналіз задачі

Дана задача може бути вирішена засобами штучного інтелекту. Постановка задачі включає аналіз фінансових даних ТОВ «ROZETKA» для визначення його ймовірності банкрутства протягом наступних 1-3 років на основі Z-моделей, отриманих за допомогою регресійного аналізу. Для виконання цієї задачі потрібно врахувати кілька ключових етапів, а саме:

1. Збір та підготовка даних: На цьому етапі потрібно зібрати фінансові дані ТОВ «ROZETKA» з фінансових звітів за певний період часу. Це можуть бути дані про прибуток, витрати, активи, зобов'язання, а також будь-які інші фінансові показники, які можуть бути важливими для аналізу банкрутства.

2. Побудова Z-моделей: За допомогою регресійного аналізу необхідно побудувати Z-моделі, які будуть використовуватися для прогнозування ймовірності банкрутства. Z-моделі зазвичай базуються на статистичних залежностях між фінансовими показниками та ризиком банкрутства.

3. Аналіз результатів моделювання: Після побудови Z-моделей необхідно проаналізувати результати, щоб оцінити їх ефективність та адекватність. Це може включати аналіз точності прогнозів, виявлення факторів, які найбільше впливають на ймовірність банкрутства, та інші аспекти.

4. Прогнозування ймовірності банкрутства: На основі побудованих Z-моделей можна прогнозувати ймовірність банкрутства ТОВ «ROZETKA» протягом наступних 1-3 років.

Мотивація застосування такого аналізу полягає у можливості заздалегідь виявити фінансові проблеми та ризики банкрутства компанії, що дозволить вчасно прийняти стратегічні рішення для запобігання негативним наслідкам і збереження фінансової стабільності.

### **2.3 Формалізація задачі (виділення вхідних-вихідних параметрів, зав'язків між ними тощо)**

Формалізація задачі передбачає виділення вхідних та вихідних параметрів, а також визначення залежностей між ними. У визначенні ймовірності банкрутства ТОВ «ROZETKA» можна виділити наступні параметри:

#### **1. Вхідні параметри:**

- Фінансові дані звітності: це може включати дані про прибуток, витрати, активи, зобов'язання, оборотний капітал і т.д.
- Показники ліквідності: наприклад, поточний відношення, швидкого відношення тощо.
- Показники рентабельності: чистий прибуток до виручки, рентабельність активів, рентабельність власного капіталу та інші.
- Фінансові показники заборгованості: наприклад, коефіцієнт покриття відсотків, коефіцієнт заборгованості тощо.

- Інші ключові фінансові та економічні показники, які можуть впливати на фінансову стійкість компанії.

## 2. Вихідний параметр:

- Ймовірність банкрутства: числове значення, що вказує на ймовірність банкрутства ТОВ «ROZETKA» протягом наступних 1-3 років.

## 3. Залежності між параметрами:

- Використання регресійного аналізу для побудови Z-моделей: залежності між фінансовими показниками та ризиком банкрутства можуть бути встановлені шляхом аналізу історичних даних.

- Врахування взаємозв'язків між фінансовими показниками: різні показники можуть бути взаємозалежними, тому їх взаємовплив слід враховувати при побудові моделей.

- Аналіз результатів та вибір найбільш важливих параметрів: після побудови моделей важливо проаналізувати їх ефективність та визначити, які фінансові показники найбільше впливають на ймовірність банкрутства.

- Прогнозування ймовірності банкрутства: на основі побудованих моделей можна здійснити прогноз ймовірності банкрутства ТОВ «ROZETKA» на певний період часу.

Формалізація цих параметрів дозволить створити модель для оцінки фінансової стійкості компанії та прогнозування її ризику банкрутства.

## **2.4 Вибір технології штучного інтелекту – обґрунтування вибору, огляд можливих варіантів, що розглядались**

Вибір технології штучного інтелекту для прогнозування ймовірності банкрутства підприємства є критичним етапом, оскільки від правильності цього вибору залежить точність та ефективність прогнозу. Нижче розглянемо обґрунтування вибору технології та оглянемо можливі варіанти для розв'язку цієї задачі:

### 1. Обґрунтування вибору технології:

- Відповідність завданню: Вибрана технологія повинна мати можливість для обробки великого обсягу даних і проведення складного аналізу для прогнозування банкрутства.

- Точність: Технологія має забезпечувати високу точність прогнозування, щоб уникнути помилкових рішень та забезпечити вчасну реакцію на можливі ризики.

- Інтеграція: Важливо, щоб обрана технологія могла інтегруватися з існуючими системами та даними підприємства для отримання повного образу ситуації.

- Масштабованість: Технологія повинна бути гнучкою та масштабованою для роботи з різноманітними типами даних та великими обсягами інформації.

## 2. Можливі варіанти для розв'язку задачі:

- Методи машинного навчання: Використання алгоритмів класифікації та регресії, таких як логістична регресія, випадковий ліс, градієнтний бустинг тощо.

- Нейронні мережі: Використання глибокого навчання для аналізу великих обсягів даних та виявлення складних залежностей між параметрами.

- Аналітика тексту: Використання методів обробки природної мови для аналізу текстової інформації з фінансових звітів та новинних джерел для отримання додаткових ознак банкрутства.

- Системи експертних знань: Розробка баз знань та правил на основі експертного досвіду для автоматизованого прийняття рішень щодо банкрутства.

Обираючи технологію штучного інтелекту для прогнозування банкрутства, важливо враховувати специфіку підприємства, його потреби та особливості даних. Також рекомендується провести тестування різних варіантів технологій на власних даних для вибору оптимальної моделі.

Логістична регресія, випадковий ліс та градієнтний бустинг - це три різні методи машинного навчання, які часто використовуються для розв'язання задач класифікації та регресії. Нижче наведено короткий огляд кожного з цих методів:

1. Логістична регресія:

- Призначення: Логістична регресія використовується для розв'язання задач класифікації, коли потрібно передбачити ймовірність належності об'єкта до певного класу.
- Основна ідея: Модель логістичної регресії використовує логістичну функцію для оцінки ймовірності того, що екземпляр належить до певного класу, в залежності від значень ознак.
- Переваги: Простота та інтерпретованість моделі, добре підходить для бінарної класифікації.

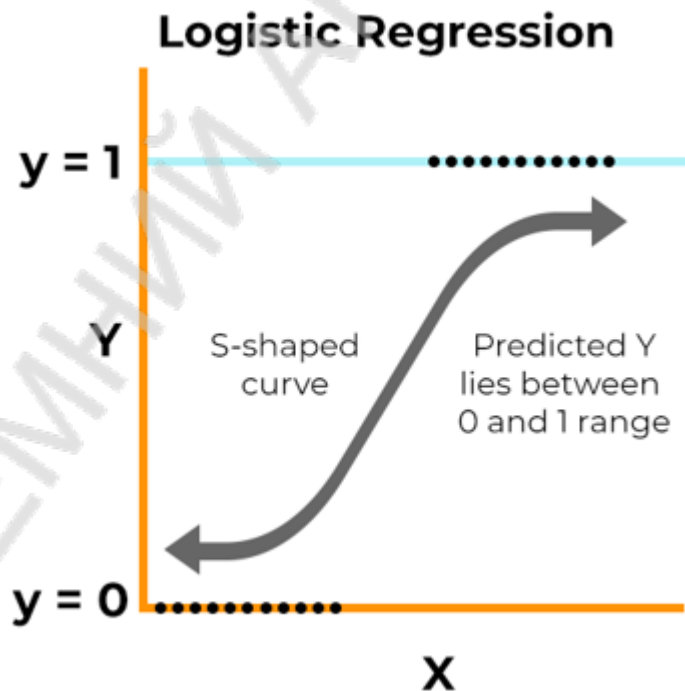


Рисунок 2.1 – Модель логістичної регресії

2. Випадковий ліс (Random Forest):

- Призначення: Випадковий ліс також використовується для задач класифікації та регресії. Він базується на ідеї ансамблю алгоритмів.
- Основна ідея: Випадковий ліс використовує кілька рішачих дерев для прийняття рішення. Кожне дерево вирішує задачу класифікації, а потім результати об'єднуються для отримання кінцевого результату.
- Переваги: Здатність роботи з великим обсягом даних та високою точністю, стійкість до перенавчання.

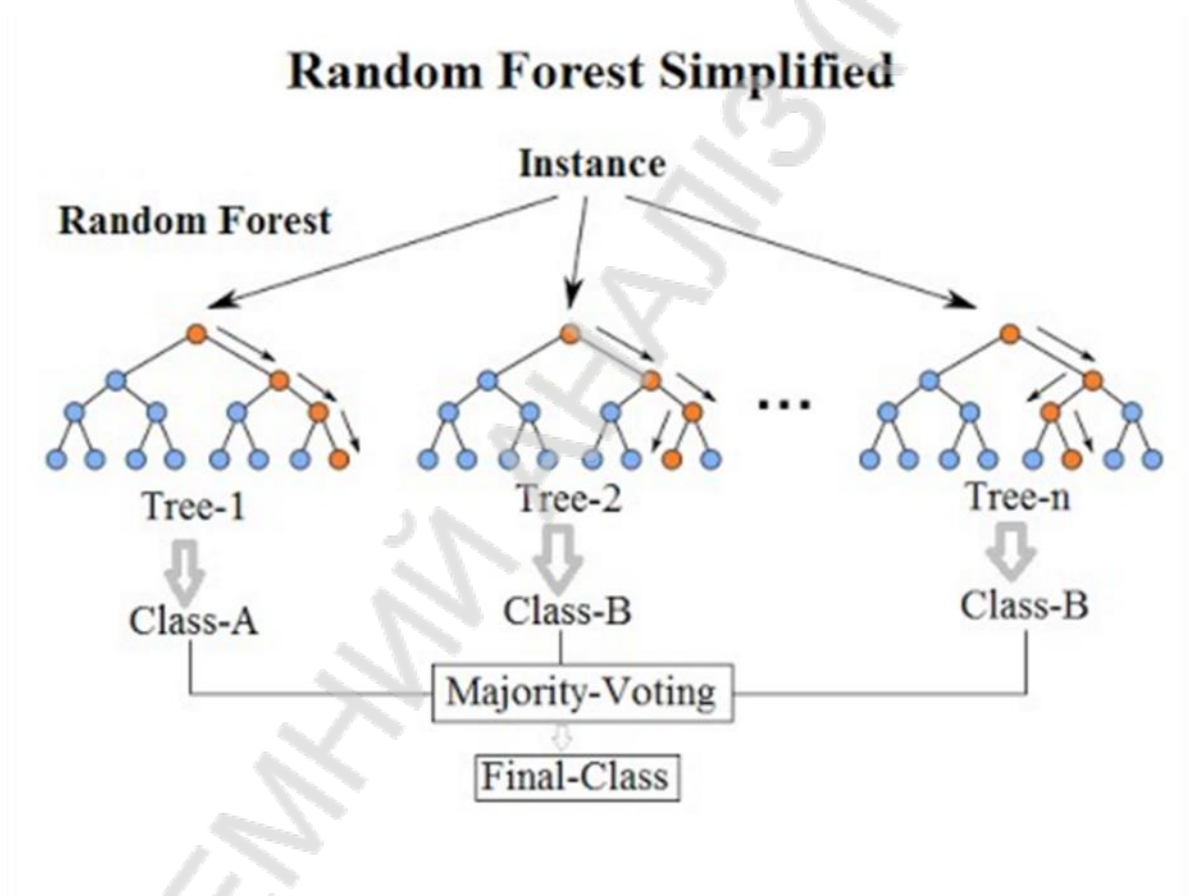


Рисунок 2.2 – Модель випадково лісу

### 3. Градієнтний бустинг (Gradient Boosting):

- Призначення: Градієнтний бустинг також використовується для класифікації та регресії. Він також є ансамблевим методом.
- Основна ідея: Градієнтний бустинг використовує послідовне навчання слабких моделей, які покращуються з кожним кроком, адаптуючись до помилок попередніх моделей.

- Переваги: Висока точність, здатність розробляти складні моделі з великою кількістю параметрів.

У кожного з цих методів є свої переваги та обмеження, тому вибір методу залежить від конкретного завдання, обсягу даних та доступних ресурсів. Часто застосовується підход "спробуйте всі три" для порівняння їх ефективності в конкретному контексті задачі.

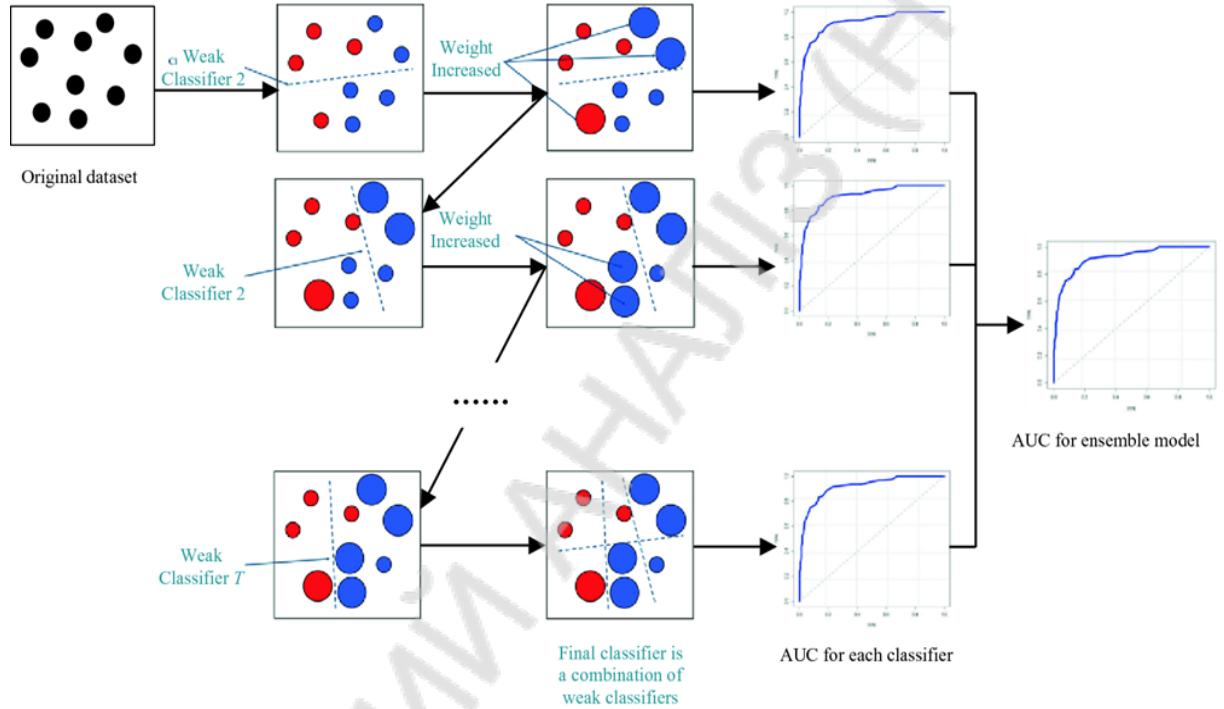


Рисунок 2.3 – Модель градієнтного бустингу

Серед трьох моделей ШІ було обрано саме модель логістичної регресії та нейронмережі на основі неї. Саме її використаємо для прогнозування та оцінки ймовірності банкрутства.

## 2.5 Обґрунтування вибору середовища реалізації (Python)

Вибір мови програмування Python для реалізації даної задачі може бути обґрунтований наступними аргументами:

1. Простота вивчення та використання: Python має простий і зрозумілий синтаксис, який дозволяє швидко оволодіти мовою навіть початківцям. Це робить його ідеальним вибором для широкого кола розробників.

2. Широкий вибір бібліотек: Python має велику кількість розширень та бібліотек для обробки даних, статистичного аналізу, машинного навчання та інших суміжних завдань. Наприклад, бібліотеки, такі як Pandas, NumPy, SciPy, Scikit-learn, Matplotlib, дозволяють легко і ефективно працювати з фінансовими даними та будувати моделі прогнозування.

3. Спільнота розробників: Python має велику та активну спільноту розробників, яка постійно розширює функціональність мови, підтримує і вдосконалює бібліотеки та допомагає вирішувати труднощі, які можуть виникнути під час розробки.

4. Переносимість: Python є переносимою мовою програмування, тобто код, написаний на Python, може працювати на різних операційних системах, що робить його універсальним і зручним для розгортання на різних платформах.

5. Інтеграція з іншими технологіями: Python добре інтегрується з іншими мовами програмування та технологіями, що може бути корисним у випадку, якщо для реалізації деяких частин задачі необхідно використовувати інші мови або середовища.

Загалом, Python є потужним та гнучким інструментом, який може ефективно використовуватися для аналізу фінансових даних, побудови моделей прогнозування та розв'язання багатьох інших завдань, пов'язаних з обробкою і аналізом даних.

STATISTICA - це потужне програмне забезпечення для статистичного аналізу та візуалізації даних, яке широко використовується у наукових дослідженнях, бізнес-аналітиці, фінансовому аналізі та інших галузях. Ось деякі аргументи на користь використання STATISTICA:

1. **Потужність і гнучкість:** STATISTICA має широкий спектр статистичних методів та інструментів для аналізу даних, включаючи описивну статистику, регресійний аналіз, аналіз варіант, кластерний аналіз, факторний аналіз та інші. Це дозволяє ефективно вирішувати різноманітні завдання аналізу даних.

2. **Зручний інтерфейс користувача:** STATISTICA має інтуїтивно зрозумілий інтерфейс користувача, який дозволяє легко виконувати аналіз даних без необхідності великих навичок програмування. Це робить його доступним для користувачів з різним рівнем підготовки та досвіду.

3. **Велика кількість вбудованих функцій і модулів:** STATISTICA постачається з багатьма вбудованими функціями та модулями для виконання різних завдань аналізу даних, що включає в себе візуалізацію даних, побудову графіків, підготовку даних для аналізу, статистичний аналіз та багато іншого.

4. **Підтримка і документація:** STATISTICA має велику базу знань, документацію та підтримку спільноти, що дозволяє користувачам швидко знайти відповіді на свої запитання та вирішити проблеми, які можуть виникнути під час роботи з програмним забезпеченням.

5. **Підтримка мов програмування:** STATISTICA підтримує вбудовані мови програмування, такі як Python та R, що дозволяє розширити його функціональність та виконувати складні аналітичні завдання, використовуючи додаткові засоби програмування.

Загалом, STATISTICA є потужним інструментом для статистичного аналізу даних, який може бути використаний для вирішення різних завдань у сфері аналітики та досліджень.

## 2.6 Вибір IDE для розробки

Після визначення мови програмування, в даному випадку Python, наступним кроком є вибір середовища розробки(Integrated Development Environment). Є багато популярних IDE(Integrated Development Environment) на вибір.

По-перше, давайте розглянемо IDE, яке можна завантажити разом з Python з офіційного веб-сайту - це Integrated Development and Learning Environment. Це просте середовище з обмеженою функціональністю. Зазвичай використовується для створення простих інструментів. Але, як на мене, це не найкращий вибір для нашого проекту. Нижче наведено загальний вигляд вікна середовища IDLE:

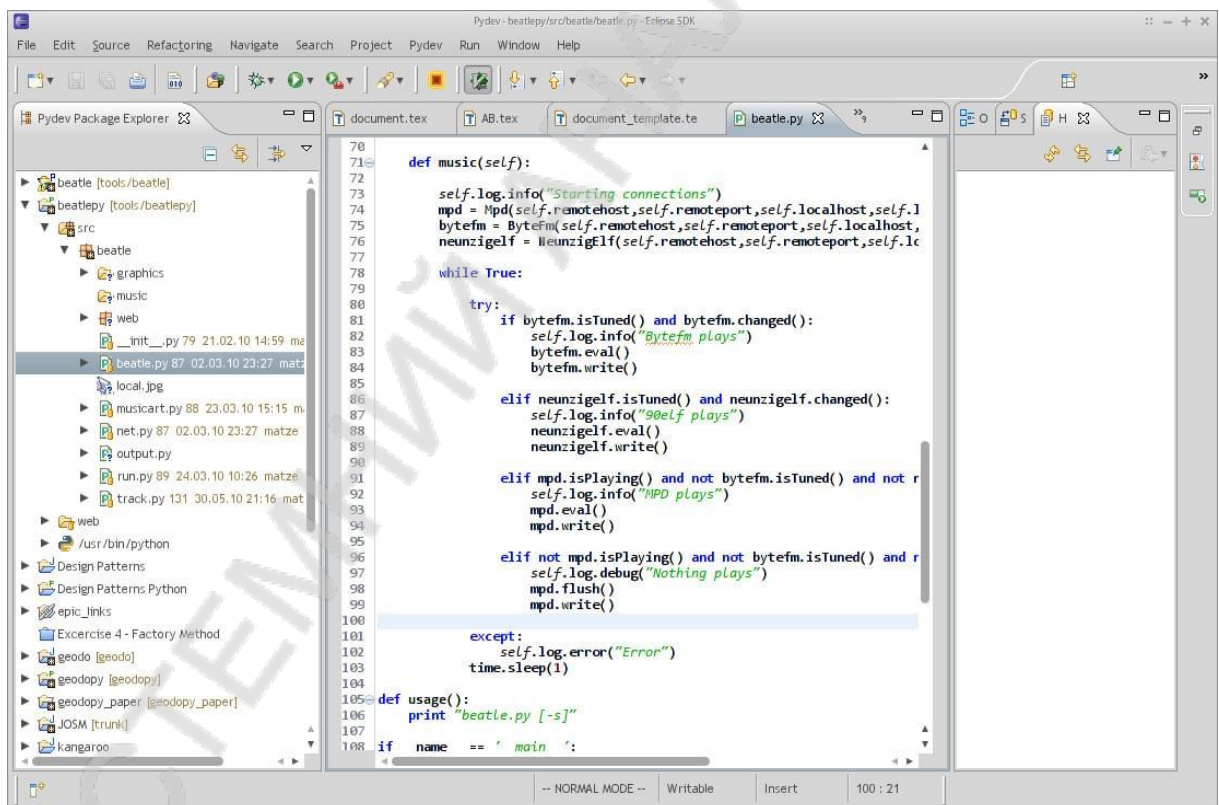


Рисунок 2.4 – Зовнішній вигляд IDE Integrated Development and Learning Environment

Ще один варіант – PyCharm. Це середовище розробки відоме своєю зручністю, оскільки воно може автозаповнювати код, аналізувати, позначати потенційні та фактичні помилки та миттєво їх виправляти.

PyCharm також надає різні способи реорганізації коду за допомогою таких рефакторингів, як: Rename, Delete, Introduce Variable, Inline Variable, Extract Method, Inline Method та інші. Завдяки цим рефакторингам можна вносити зміни у всьому проєкті, враховуючи особливості мови програмування та фреймворку.

PyCharm має один із найбільших наборів інструментів для роботи з кодом, включаючи інструмент для запуску тестів, вбудований відладчик, профільник Python, багато інструментів для роботи з базами даних, повноцінний вбудований термінал та багато іншого. Середовище розробки інтегрується безпосередньо з усіма основними системами контролю версій, має вбудований SSH-термінал і підтримує можливості віддаленої розробки та віддалених інтерпретаторів, а також може інтегруватися з Docker.

PyCharm дозволяє підключатися безпосередньо до Oracle, SQL Server, PostgreSQL, MySQL та інших баз даних. Крім того, в PyCharm можна самостійно редагувати код SQL.

Варто зазначити, що PyCharm підтримує багато фреймворків, таких як Flask, Django, web2py та інші. Таким чином, ми можемо працювати з Django-шаблонами, використовувати велику кількість утиліт manage.py і використовувати багато специфічних для цих фреймворків навігацій, не виходячи з IDE.

Окрім фреймворків, PyCharm пропонує повну підтримку платформам для розробки на мові Python, а також може підтримувати інші мови шаблонів, такі як HTML/CSS, Node.js, CoffeeScript, AvaScript, TypeScript та AngularJS.

PyCharm можна встановити на Windows, macOS та Linux. Інтерфейс IDE розроблено так, щоб задовольнити всі потреби користувача. Розробники можуть налаштовувати своє робоче середовище, зокрема вибирати колірні схеми та відповідні комбінації клавіш.

Нижче наведено загальний вигляд вікна IDE:



Рисунок 2.5 – Зовнішній вигляд IDE PyCharm

Враховуючи всі вищезазначені переваги, я вважаю, що PyCharm буде найефективнішим та найкращим варіантом для розробки нашого проєкту. Крім того, PyCharm має дуже зручну функцію швидкого встановлення бібліотек, необхідних для проєкту.

## 2.7 Вибір додаткових бібліотек для розробки

У попередньому пункті було обрано PyCharm як середовище розробки, яке має можливість підключення до великої кількості додаткового програмного забезпечення. За допомогою PyCharm можна працювати з Jupyter ноутбуками, запускати Python команди в інтерактивній консолі, підключати Anaconda бібліотеки, а також працювати з іншими бібліотеками для наукових обчислень та аналізу даних, зокрема з Matplotlib та NumPy.

Надалі розглянемо бібліотеки, які будемо використовувати для створення нашого проєкту:

- NumPy (np):
- Опис: NumPy - це бібліотека для мови програмування Python, яка додає підтримку великих масивів і матриць разом з високорівневими математичними функціями для роботи з цими масивами. Це надає

можливість виконувати операції лінійної алгебри, обробки сигналів, та рандомізації даних, що робить його дуже корисним для наукових обчислень та аналізу даних.

- Використання: Часто використовується для обчислень над великими наборами даних, векторизації коду, а також для побудови і обробки матриць.

- Matplotlib (plt):

- Опис: Matplotlib - це бібліотека для створення двовимірних графіків в мові програмування Python. Вона дозволяє будувати різноманітні види графіків, такі як лінійні графіки, гістограми, точкові графіки та багато інших.

- Використання: Використовується для візуалізації даних, щоб зробити їх зрозумілими та легкими для аналізу.

- Scikit-learn (sklearn):

- Опис: Scikit-learn - це бібліотека машинного навчання для мови програмування Python. Вона містить реалізації багатьох алгоритмів навчання з учителем та без учителя, таких як класифікація, регресія, кластеризація, а також інструменти для оцінки та підбору моделей.

- Використання: Використовується для побудови, навчання та оцінки моделей машинного навчання.

- Scipy.stats (linregress):

- Опис: Scipy - це бібліотека для наукових обчислень та технічних обчислень для мови програмування Python. Модуль stats в Scipy містить статистичні функції та інструменти для роботи з даними.

- linregress: Функція linregress використовується для побудови простої лінійної регресії та отримання інформації про цю регресію, таку як нахил, перетин, коефіцієнт кореляції та р-значення.

Використання: Часто використовується для аналізу статистичних даних, побудови регресійних моделей та оцінки їхньої ефективності.

## 2.8 Розробка моделі – створення формальної моделі для розв'язку задачі на основі обраної технології

Для розробки моделі прогнозування банкрутства на основі логістичної регресії і отримання моделі типу Альтмана, нам знадобиться:

1. Збір даних: Зібрати фінансові дані компанії ROZETKA, які включають показники звітності, такі як баланс, звіт про прибутки та збитки, кошторисні дані тощо. Для розрахунку моделі типу Альтмана основними фінансовими показниками є рівень заборгованості, рентабельність активів, оборотність активів, коефіцієнт власного капіталу та ринкова капіталізація.

2. Підготовка даних: Провести попередню обробку даних, таку як видалення відсутніх значень, нормалізація, видалення викидів тощо.

3. Вибір ознак: Визначити набір факторів, що впливають на банкрутство компанії. Для моделі типу Альтмана ці ознаки включають показники фінансового здоров'я, які згадані вище.

4. Побудова моделі логістичної регресії: Використовуючи оброблені дані та вибраний набір ознак, побудуйте модель логістичної регресії для прогнозування ймовірності банкрутства компанії. У моделі типу Альтмана можна використовувати коефіцієнти, які були розроблені самим Альтманом: Z-бал, що обчислюється як лінійна комбінація фінансових показників.

5. Налаштування та валідація моделі: Налаштуйте параметри моделі та проведіть валідацію для перевірки її точності та ефективності. Для цього можна використовувати різні методи, такі як хресна перевірка або розділення на навчальний та тестовий набори даних.

6. Оцінка результатів: Оцініть результати моделі, провівши аналіз важливості ознак та інші метрики ефективності, такі як точність, чутливість, специфічність тощо.

7. Впровадження та моніторинг: Після успішної побудови та перевірки моделі впровадіть її в робоче середовище. Необхідно також

встановити механізми моніторингу та оновлення моделі для забезпечення її актуальності в майбутньому.

## 2.9 Практична реалізація – короткий огляд програмної реалізації моделі в середовищі Python

Спочатку сформуємо масив вихідних даних та на основі нього розрахуємо коефіцієнти, які характеризують фінансовий стан ТОВ «ROZETKA».

Для українських реалій підходить шестифакторна модель О.О. Терещенка, хоча досі є застарілою, але саме ті коефіцієнти, які у неї входять візьмемо за основу.

$$Z = 1,5X_1 + 0,08X_2 + 10X_3 + 5X_4 + 0,3X_5 + 0,1X_6, \quad (3.1)$$

де  $X_1$  – відношення грошових коштів до зобов'язань;

$X_2$  – відношення валюти балансу до зобов'язань;

$X_3$  – відношення чистого прибутку до середньорічної суми активів;

$X_4$  – відношення прибутку до доходу від реалізації;

$X_5$  – відношення виробничих запасів до доходу від реалізації;

$X_6$  – відношення доходу від реалізації до основного капіталу.

Оцінити ймовірність банкрутства можна за формулою:

$$p = \frac{\exp Z}{1 + \exp Z} \quad (3.2)$$

Таблиця 3.1

### Фінансово-економічні показники (абсолютні) для розрахунку моделі

№	Показник	2021 рік	2022 рік	2023 рік
1	Грошові кошти	139820	137732	791591
2	Зобов'язання	30481+3988495	30896+3294600	61345+4858167
3	Чистий прибуток	113210	33962	119062
4	Середньорічна сума активів	(65128+4167619)/2	(89744+3562733)/2	(85095+4773072)/2
5	Валовий прибуток	3236826	3549851	4663963

6	Дохід від реалізації	8390610	18388104	25463875
7	Виробничі запаси	337	446	1035
8	Валюта балансу	4232747	3652477	4858167

## 2.10 Опис основних функцій програмної реалізації

### 1. Імпорт бібліотек:

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression, LinearRegression
from scipy.stats import linregress
```

- `numpy`: Ця бібліотека використовується для роботи з масивами та числовими даними.
- `matplotlib.pyplot`: Ця бібліотека використовується для візуалізації даних.
- `sklearn.linear_model`: Ця бібліотека використовується для машинного навчання, зокрема для лінійної та логістичної регресії.
- `scipy.stats`: Ця бібліотека використовується для статистичного аналізу даних.

### 2. Вхідні дані:

```
# Вхідні дані за 2021, 2022 і 2023 роки
# 1 показник - Грошові кошти
# 2 показник - Забов'язання
# 3 показник - Чистий прибуток
# 4 показник - Середньорічна сума активів
# 5 показник - Валовий прибуток
# 6 показник - Дохід від реалізації
# 7 показник - Виробничі запаси
# 8 показник - Валюта балансу
data_2021 = (139820, 4018976, 113210, 2116373.5, 3236826, 8390610, 337, 4232747)
data_2022 = (137732, 3325496, 33962, 1826238.5, 3549851, 18388104, 446, 3652477)
data_2023 = (791591, 4919512, 119062, 2429083.5, 4663963, 25463875, 1035, 4858167)
```

Ці рядки коду містять дані про фінансові показники компанії за 2021, 2022 та 2023 роки. Дані представлені у вигляді кортежів, де кожен елемент відповідає певному показнику.

### 3. Розрахунок коефіцієнтів:

```
# Розрахунок коефіцієнтів для кожного року
coef_2021 = [
    data_2021[0] / data_2021[1],
    data_2021[7] / data_2021[1],
    data_2021[2] / data_2021[3],
    data_2021[4] / data_2021[5],
    data_2021[6] / data_2021[5],
    data_2021[5] / data_2021[7]
]

coef_2022 = [
    data_2022[i] / data_2022[j] for i, j in [(0, 1), (7, 1), (2, 3), (4, 5), (6, 5), (5, 7)]
]

coef_2023 = [
    data_2023[i] / data_2023[j] for i, j in [(0, 1), (7, 1), (2, 3), (4, 5), (6, 5), (5, 7)]
]
```

Розраховуються фінансові коефіцієнти для кожного року. Коефіцієнти розраховуються як співвідношення між різними фінансовими показниками.

Назви коефіцієнтів:

- Відношення грошових коштів до зобов'язань
- Відношення валюти балансу до зобов'язань
- Відношення чистого прибутку до середньорічної суми активів
- Відношення прибутку до доходу від реалізації
- Відношення виробничих запасів до доходу від реалізації
- Відношення доходу від реалізації до основного капіталу

### 4. Виведення розрахованих коефіцієнтів:

```
# Виведення розрахованих коефіцієнтів для кожного року
coef_names = [
    "Відношення грошових коштів до зобов'язань",
    "Відношення валюти балансу до зобов'язань",

```

```

"Відношення чистого прибутку до середньорічної суми активів",
"Відношення прибутку до доходу від реалізації",
"Відношення виробничих запасів до доходу від реалізації",
"Відношення доходу від реалізації до основного капіталу"
]

print("Розраховані коефіцієнти для кожного року:")
print("Коефіцієнти 2021 року:")
for name, coef in zip(coef_names, coef_2021):
    print(f"{name}: {coef}")
print("\nКоефіцієнти 2022 року:")
for name, coef in zip(coef_names, coef_2022):
    print(f"{name}: {coef}")
print("\nКоефіцієнти 2023 року:")
for name, coef in zip(coef_names, coef_2023):
    print(f"{name}: {coef}")

```

Вивід розрахованих коефіцієнтів для кожного року.

Вводиться список `coef_names`, який містить назви відповідних коефіцієнтів. Цикл `for` використовується для друку назви та значення кожного коефіцієнта.

```

Розраховані коефіцієнти для кожного року:
Коефіцієнти 2021 року:
Відношення грошових коштів до зобов'язань: 0.03478995644661725
Відношення валюти балансу до зобов'язань: 1.0531904146727922
Відношення чистого прибутку до середньорічної суми активів: 0.05349244828476637
Відношення прибутку до доходу від реалізації: 0.38576766170755167
Відношення виробничих запасів до доходу від реалізації: 4.016394517204351e-05
Відношення доходу від реалізації до основного капіталу: 1.982308415787667

Коефіцієнти 2022 року:
Відношення грошових коштів до зобов'язань: 0.04141697960244126
Відношення валюти балансу до зобов'язань: 1.0983254828753366
Відношення чистого прибутку до середньорічної суми активів: 0.01859669479095967
Відношення прибутку до доходу від реалізації: 0.1930514967720435
Відношення виробничих запасів до доходу від реалізації: 2.4254811697823767e-05
Відношення доходу від реалізації до основного капіталу: 5.03442020305672

Коефіцієнти 2023 року:
Відношення грошових коштів до зобов'язань: 0.16090843969889695
Відношення валюти балансу до зобов'язань: 0.9875302672297578
Відношення чистого прибутку до середньорічної суми активів: 0.04901519441386021
Відношення прибутку до доходу від реалізації: 0.1831599864513944
Відношення виробничих запасів до доходу від реалізації: 4.0645816868014e-05
Відношення доходу від реалізації до основного капіталу: 5.24145732330733

```

Рисунок 2.6 – Скріншот з розрахованими коефіцієнтами для 2021, 2022 та 2023 років

## 5. Підготовка даних для моделювання:

```
# Створення X_train
X_train = np.array([coef_2021, coef_2022, coef_2023])

# Задання років
years = np.array([2021, 2022, 2023])

# Мітки класів: 0 - не банкрутство, 1 - банкрутство
y_train = np.array([0, 0, 1])
```

Підготовка даних для навчання моделей машинного навчання:

- `X_train` - це масив, що містить розраховані коефіцієнти для кожного року.
- `years` - це масив, що містить роки, до яких належать дані в `X_train`.
- `y_train` - це масив міток класів, де 0 означає не банкрутство, а 1 - банкрутство. Дані за 2023 рік марковані як 1, що означає, що компанія зазнала банкрутства.

## 6. Навчання моделей:

```
# Ініціалізація та навчання моделей
logistic_model = LogisticRegression()
logistic_model.fit(X_train, y_train)

linear_model = LinearRegression()
linear_model.fit(X_train, years)
```

Навчання двох моделей машинного навчання:

- `logistic_model`: Це модель логістичної регресії, яка використовується для прогнозування ймовірності банкрутства на основі розрахованих коефіцієнтів.
- `linear_model`: Це модель лінійної регресії, яка використовується для прогнозування року на основі розрахованих коефіцієнтів.

Функція `fit` використовується для навчання моделей на тренувальних даних (`X_train` та `y_train`), де `X_train` - це набір фінансових показників за

роки, а `y_train` - мітки класів, де 1 відповідає банкрутству, а 0 - не банкрутству.

Також тут створюється та навчається модель лінійної регресії для прогнозування року для 2024 року на основі фінансових показників.

#### 7. Прогнозування ймовірності банкрутства:

```
# Прогнозування ймовірності банкрутства для 2023 року за допомогою
логістичної регресії
X_test = np.array([
    [data_2023[0] / data_2023[1], data_2023[7] / data_2023[1],
     data_2023[2] / data_2023[3], data_2023[4] / data_2023[5],
     data_2023[6] / data_2023[5], data_2023[5] / data_2023[7]]
]) # Дані за 2023 рік
probability_of_bankruptcy = logistic_model.predict_proba(X_test)[0][1]
print("\nЙмовірність банкрутства для 2023 року:", probability_of_bankruptcy)
```

Тут використовується навчена модель `logistic_model`, щоб передбачити ймовірність банкрутства для 2023 року на основі даних `X_test`:

Створюється масив `X_test`, що містить розраховані коефіцієнти для 2023 року. Функція `predict_proba` моделі логістичної регресії використовується для прогнозування ймовірності того, що значення мітки класу буде 1 (тобто банкрутство) для даних у `X_test`. Результат зберігається в змінній `probability_of_bankruptcy` та друкується.

Ймовірність банкрутства для 2023 року: 0.44944840582447226

Рисунок 2.7. Скріншот ймовірності банкрутства для 2023 року

#### 8. Прогнозування року за даними 2023:

```
# Прогнозування року для 2024 року за допомогою лінійної регресії
predicted_year_2024 = linear_model.predict(X_test)[0]
print("Прогнозований рік для 2024 року:", int(predicted_year_2024))
```

Тут використовується навчена модель лінійної регресії для прогнозування року для 2024 року на основі даних `X_test`.

Прогнозується рік на основі даних за 2023 рік. Функція `predict` моделі лінійної регресії використовується для прогнозування значення на основі

даних у `X_test`. Оскільки модель навчалася прогнозувати роки, отримане значення відповідає прогнозованому року. Результат зберігається в змінній `predicted_year_2024` та друкується після округлення до цілого числа.

```
Прогнозований рік для 2024 року: 2023
```

Рисунок 2.8 – Скріншот прогнозування року за даними 2023

9. Обчислення лінійної регресії для прогнозування ймовірності банкрутства:

```
# Обчислення лінійної регресії для прогнозування ймовірності банкрутства
slope, intercept, _, _, _ = linregress(years,
logistic_model.predict_proba(X_train)[:, 1])
```

Обчислення коефіцієнтів лінійної регресії для зв'язку між роками (`years`) та прогнозованою ймовірністю банкрутства

(`logistic_model.predict_proba(X_train)[:, 1]`). Функція `linregress` з бібліотеки `scipy.stats` використовується для обчислення цих коефіцієнтів.

Збережені значення:

- `slope`: кутовий коефіцієнт лінії регресії
- `intercept`: точка перетину лінії регресії з віссю `Y`

10. Прогнозування ймовірності банкрутства для 2024 року на основі лінійної регресії:

```
# Прогнозування ймовірності банкрутства для 2024 року
predicted_probability_2024 = slope * predicted_year_2024 + intercept
print("Прогнозована ймовірність банкрутства для 2024 року:",
predicted_probability_2024)
```

Тут використовується оновлена модель логістичної регресії для прогнозування ймовірності банкрутства для 2024 року на основі оновлених фінансових даних `X_test_modified`.

Використання раніше обчислених коефіцієнтів лінійної регресії (`slope` та `intercept`) для прогнозування ймовірності банкрутства на 2024 рік.

predicted\_year\_2024 - це прогнозований рік, отриманий раніше. Результат зберігається в змінній predicted\_probability\_2024 та друкується.

Прогнозована ймовірність банкрутства для 2024 року: 0.49264738412995257

Рисунок 2.9 – Скріншот ймовірності банкрутства для 2024 року

### 11. Визначення критичних параметрів:

```
# Критичні параметри, які найбільше впливають на ймовірність банкрутства
parameters = coef_names
coefficients = logistic_model.coef_[0]
critical_params = sorted(zip(parameters, coefficients), key=lambda x: abs(x[1]),
reverse=True)

# Вивід критичних параметрів та їх впливу
print("\nКритичні параметри, які найбільше впливають на ймовірність банкрутства:")
for param, coef in critical_params:
    print(f"{param}: {coef}")
```

Визначаються критичні параметри, які найбільше впливають на ймовірність банкрутства. coef\_names містить назви параметрів (коефіцієнтів). logistic\_model.coef\_[0] містить ваги моделі логістичної регресії для кожного параметра. sorted використовується для сортування пари (назва параметра, вага) за абсолютним значенням ваги в порядку спадання (reverse=True). Отриманий список critical\_params містить пари назва-вага для критичних параметрів. Цикл for друкує назву та вагу кожного критичного параметра.

```

Критичні параметри, які найбільше впливають на ймовірність банкрутства:
Відношення доходу від реалізації до основного капіталу: 0.5132473938467106
Відношення грошових коштів до зобов'язань: 0.06664452993344192
Відношення валюти балансу до зобов'язань: -0.05507603103317839
Відношення прибутку до доходу від реалізації: -0.030651448709568818
Відношення чистого прибутку до середньорічної суми активів: 0.01218032336260315
Відношення виробничих запасів до доходу від реалізації: 6.942106062094543e-06

```

Рисунок 2.10 – Скріншот критичних параметрів

## 12. Ітеративний підбір параметрів для зниження ймовірності банкрутства

```

# Ітеративний підбір параметрів для зниження ймовірності банкрутства
# Використовується простий підбір коефіцієнтів для перших трьох
критичних параметрів
target_probability = probability_of_bankruptcy_2023
max_iterations = 1000
tolerance = 1e-6

X_test_modified = X_test.copy()
critical_param_indices = [parameters.index(param[0]) for param in
critical_params[:3]]

for i in range(max_iterations):
    probability = logistic_model.predict_proba(X_test_modified)[0][1]
    error = target_probability - probability
    if abs(error) < tolerance:
        break
    for index in critical_param_indices:
        # Регулюємо параметр, використовуючи коефіцієнт пропорційності
        X_test_modified[0, index] += error * 0.01 / len(critical_param_indices)

```

Цей фрагмент коду описує процес ітеративного підбору параметрів моделі логістичної регресії з метою зниження ймовірності банкрутства. Він використовує простий метод підбору коефіцієнтів для перших трьох критичних параметрів.

### 1. Визначення змінних:

- target\_probability: Ця змінна зберігає цільове значення ймовірності банкрутства, яке ми хочемо досягти. У даному випадку це ймовірність банкрутства для 2023 року, отримана з моделі.

- `max_iterations`: Ця змінна визначає максимальну кількість ітерацій, які буде виконано алгоритмом підбору параметрів.
- `tolerance`: Ця змінна визначає допустиму похибку між цільовою ймовірністю та прогнозованою ймовірністю.
- `X_test_modified`: Ця змінна є копією масиву `X_test`, який використовується для прогнозування ймовірності банкрутства. Її буде модифікувати алгоритм підбору параметрів.
- `critical_param_indices`: Цей список містить індекси перших трьох критичних параметрів, які будуть піддаватися регулюванню.

## 2. Цикл ітерацій:

- Цикл `for` виконується до `max_iterations` разів або до тих пір, поки похибка між цільовою ймовірністю та прогнозованою ймовірністю не стане меншою за `tolerance`.
- В кожній ітерації:
  - Виконується прогнозування ймовірності банкрутства за допомогою модифікованого масиву `X_test_modified` та моделі логістичної регресії.
  - Розраховується похибка між цільовою ймовірністю та прогнозованою ймовірністю.
  - Якщо похибка менша за `tolerance`, цикл завершується.
  - В іншому випадку, для кожного з критичних параметрів:
    - Параметр регулюється шляхом додавання до нього значення, пропорційного похибці та кількості критичних параметрів.

## 13. Зміна параметрів для уникнення банкрутства:

```
# Зміна параметрів для уникнення банкрутства
max_values = {parameters[index]: X_test_modified[0, index] for index in
critical_param_indices}
print("\nДля уникнення банкрутства потрібно змінити:")
for param, value in max_values.items():
    print(f"{param} на {value:.6f}.")

predicted_probability_2024_modified =
```

```
logistic_model.predict_proba(X_test_modified)[0][1]
print("\nПрогнозована ймовірність банкрутства для 2024 року з оновленими даними:", predicted_probability_2024_modified)
```

Демонстрування, як можна змінити критичні параметри, щоб знизити ймовірність банкрутства. `max_values` - це словник, який містить пари назва-значення для змінених критичних параметрів. Друкується список параметрів, які потрібно змінити, та їх цільові значення. Створюється масив `X_test_modified` з оновленими значеннями критичних параметрів. Ймовірність банкрутства для 2024 року прогнозується з використанням оновлених даних. Результат зберігається в змінній `predicted_probability_2024_modified` та друкується.

```
Для уникнення банкрутства потрібно змінити:
Відношення доходу від реалізації до основного капіталу на 5.241457.
Відношення грошових коштів до зобов'язань на 0.160908.
Відношення валюти балансу до зобов'язань на 0.987530.

Прогнозована ймовірність банкрутства для 2024 року з оновленими даними: 0.44944840582447226
```

Рисунок 2.11 – Скріншот параметрів, які треба змінити та оновленої ймовірності банкрутства на 2024 рік

#### 14. Візуалізація даних:

```
# Графіки для кожного параметра
fig1, axs1 = plt.subplots(1, 2, figsize=(12, 6))
fig2, axs2 = plt.subplots(1, 2, figsize=(12, 6))
fig3, axs3 = plt.subplots(1, 2, figsize=(12, 6))
axs = axs1.flatten().tolist() + axs2.flatten().tolist() + axs3.flatten().tolist()

for i, (param, coef) in enumerate(critical_params[:6]):
    axs[i].plot(years, X_train[:, i], marker='o')
    axs[i].set_title(param)
    axs[i].set_xlabel('Рік')
    axs[i].set_ylabel('Значення параметра')
    axs[i].grid(True)

# Прогноз ймовірності банкрутства для 2024 року для кожного параметра
X_test_param = np.array([[0.0] * len(parameters)])
X_test_param[0, i] = 1.0
probability_of_bankruptcy =
logistic_model.predict_proba(X_test_param)[0][1]
```

```
axs[i].scatter(2024, probability_of_bankruptcy, color='red')

# З'єднання точки для 2024 року лінією
axs[i].plot([years[-1], 2024], [X_train[-1, i], probability_of_bankruptcy],
color='red', linestyle='--')

plt.tight_layout()

# Графік ймовірності банкрутства на проміжку з 2021 по 2024
plt.figure(figsize=(8, 6))

# Графік для існуючих даних
probabilities_existing = logistic_model.predict_proba(X_train)[: , 1]
plt.plot(years, probabilities_existing, marker='o', label='Ймовірність
банкрутства (існуючі дані)', color='blue')

# Прогнозована ймовірність банкрутства для 2024 року
plt.scatter(2024, predicted_probability_2024, color='red',
label='Прогноз ймовірності банкрутства (2024)')
plt.plot([years[-1], 2024], [probabilities_existing[-1],
predicted_probability_2024], linestyle='--',
color='red')

# Прогнозована ймовірність банкрутства для 2024 року з оновленими
даними
plt.scatter(2024, predicted_probability_2024_modified, color='purple',
label='Прогноз ймовірності банкрутства (зі зміненими даними)')
plt.plot([years[-1], 2024], [probabilities_existing[-1],
predicted_probability_2024_modified],
linestyle='--', color='purple')

plt.xlabel('Рік')
plt.ylabel('Ймовірність банкрутства')
plt.title('Прогноз ймовірності банкрутства на проміжку з 2021 по 2024')
plt.legend()
plt.grid(True)

# Додати відступ від країв
plt.subplots_adjust(left=0.1, right=0.9, top=0.9, bottom=0.1)

plt.show()
```

Ці рядки коду створюють декілька графіків для візуалізації даних та прогнозів.

Графіки для кожного параметра:

- Кожен графік складається з 2 підграфіків, розміщених на фігурі розміром 12x6.
  - Відображається динаміка одного з критичних параметрів протягом років з 2021 по 2024.
  - Використовуються маркери 'o' для позначення фактичних значень параметра за кожен рік.
  - Додатково відображається прогнозована ймовірність банкрутства на 2024 рік, якби даний параметр мав максимальне значення. Це дозволяє візуально оцінити вплив цього параметра на ймовірність банкрутства.

Графік ймовірності банкрутства на проміжку з 2021 по 2024:

- Цей графік розміщений на фігурі розміром 8x6.
- Відображає динаміку ймовірності банкрутства протягом років з 2021 по 2024.
- Використовуються маркери 'o' для позначення фактичних значень ймовірності банкрутства за кожен рік.
  - Додатково відображаються:
    - Прогнозована ймовірність банкрутства на 2024 рік (червона точка).
    - Прогнозована ймовірність банкрутства на 2024 рік з урахуванням змінених критичних параметрів (фіолетова точка).
  - Графік має заголовок "Прогноз ймовірності банкрутства на проміжку з 2021 по 2024", вісь x з позначенням років та вісь y з позначенням ймовірності банкрутства.
    - Легенда допомагає розрізнити різні лінії та точки.
    - На графіку використовується сітка для покращення візуального сприйняття.

Функції, що використовуються для побудови графіків:

- `plt.subplots` - створює фігуру та підграфіки.
- `axs.flatten().tolist()` - перетворює список підграфіків у плоский список.
- `plt.plot` - будує лінії на графіку.
- `plt.scatter` - будує точки на графіку.
- `plt.set_title`, `plt.set_xlabel`, `plt.set_ylabel` - задають назви осей та заголовка графіка.
- `plt.grid` - додає сітку на графік.
- `plt.tight_layout` - автоматично підбирає розміри та розташування графіків.
- `plt.legend` - додає легенду до графіка.
- `plt.subplots_adjust` - налаштовує відступи від країв графіка.
- `plt.show` - візуалізує графіки.

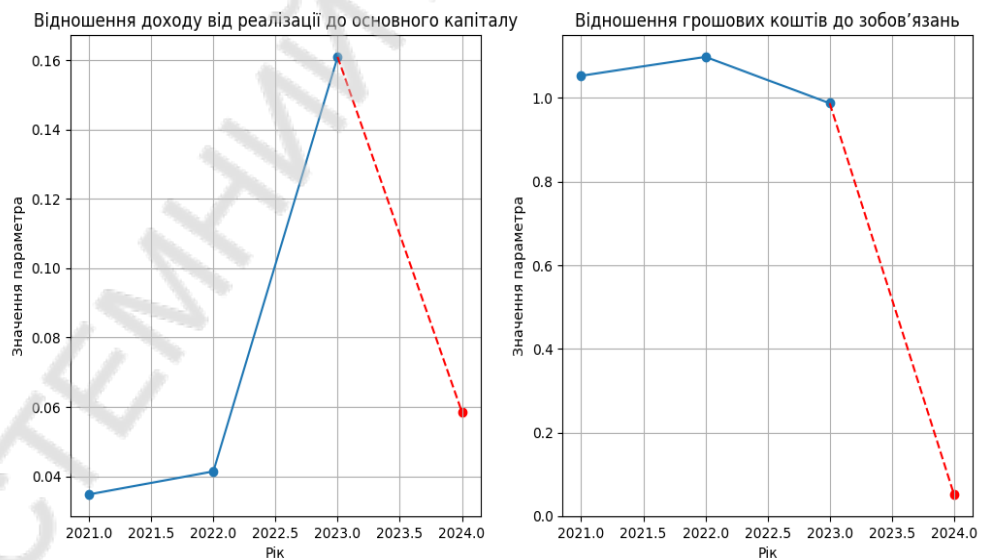


Рисунок 2.12 – Тенденції першого та другого критичного параметру

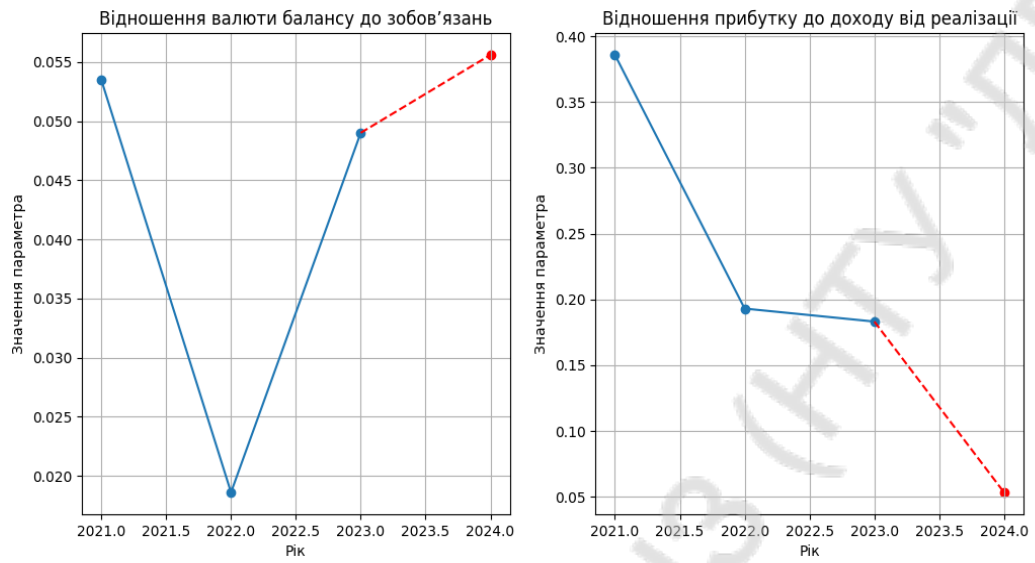


Рисунок 2.13 – Тенденції третього та четвертого критичного параметру

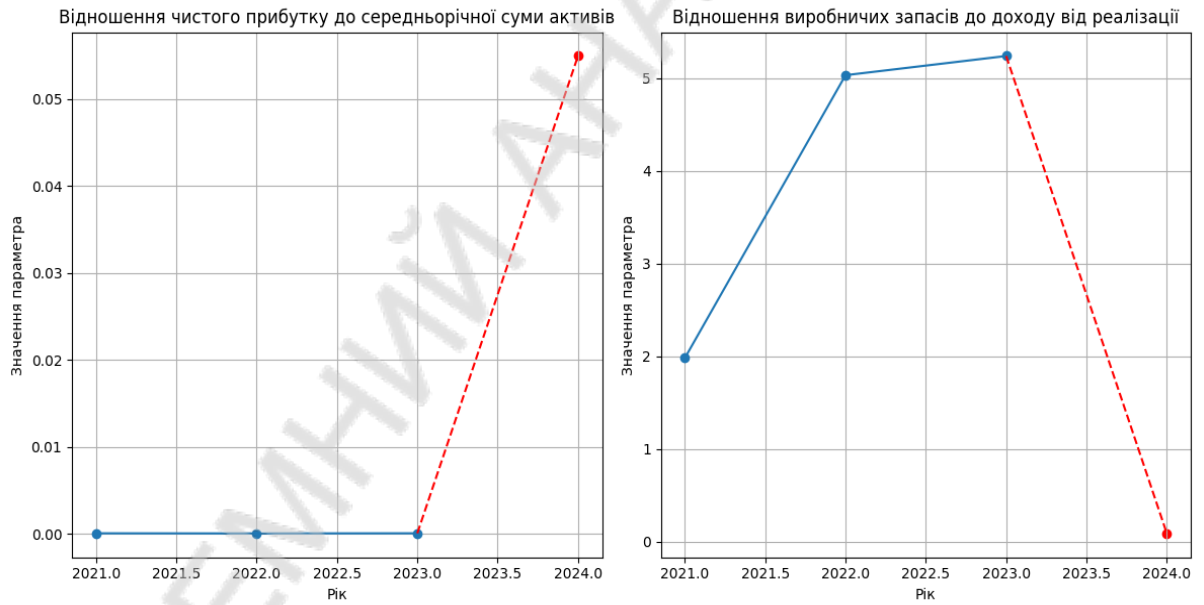


Рисунок 2.14 – Тенденція п'ятого та шостого критичного параметру

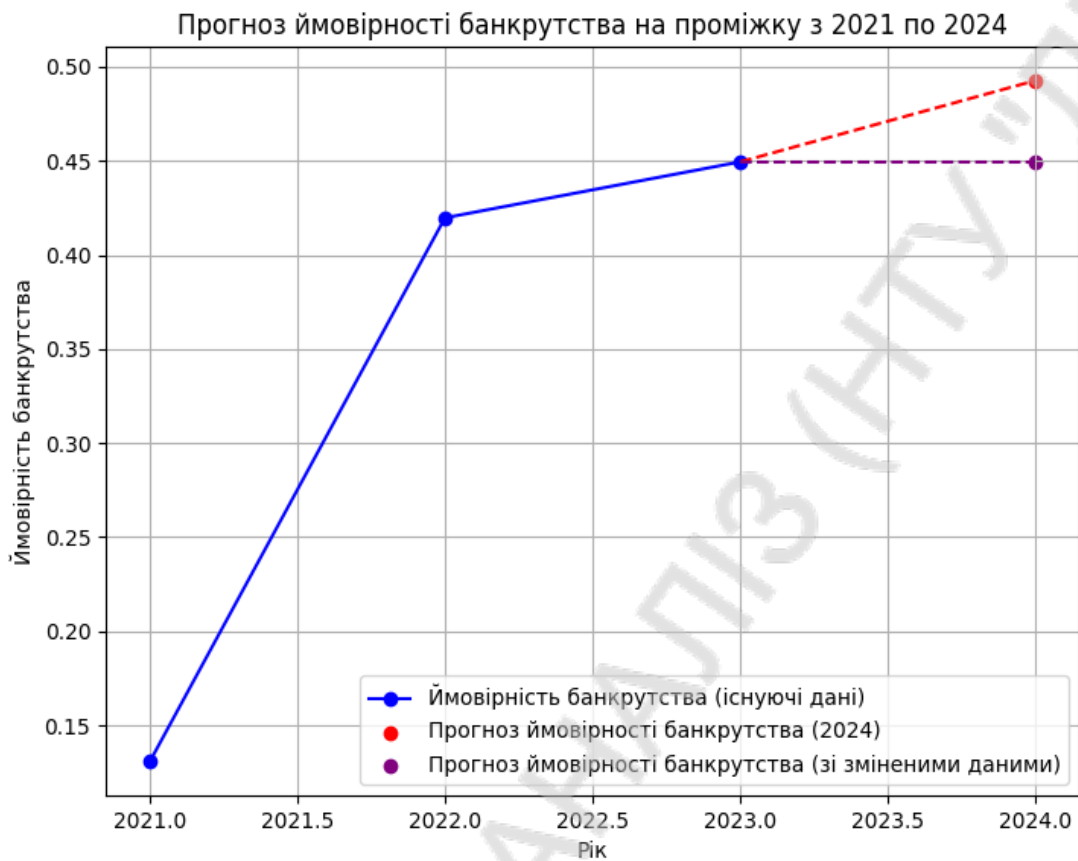


Рисунок 2.15 – Ймовірність банкрутства з 2021 по 2024 роки

Підсумовуючи усю інформацію з розділу можемо сказати, що у останньому розділі було описано процес проєктування та розробки програми, яка використовує технології машинного навчання.

Була описана логіка програми, описаний її алгоритм, розглянуті основні функції, які представляють собою логіку програми та реалізацію алгоритму у цілому.

## ВИСНОВОК

У даній роботі було проведено глибокий аналіз теоретичних аспектів машинного навчання та створення штучного інтелекту з метою їхнього застосування у сфері фінансового управління.

Розділ 1 присвячений дослідженню основних концепцій машинного навчання, включаючи типи, види та принципи роботи. Важливий акцент було зроблено на регресії у машинному навчанні, яка є одним із ключових інструментів для розв'язання задач прогнозування та оптимізації.

Розділ 2 є практичною частиною роботи, де розглянуто проєктні рішення з оптимізації фінансової діяльності підприємства на прикладі ТОВ «ROZETKA». Була розроблена Z-модель для визначення ймовірності банкрутства та виявлення проблемних зон управління фінансами, що є критичними для ефективного функціонування підприємства. В цьому контексті великий акцент був зроблений на аналізі задачі та виборі технологій для реалізації розроблених моделей.

Розділ 3 присвячений програмній реалізації розроблених моделей. Він включає вибір технології штучного інтелекту, розробку формальної моделі та практичну реалізацію за допомогою мови програмування Python. Важливим етапом було обґрунтування вибору середовища реалізації та вибір додаткових бібліотек для забезпечення ефективності та точності моделі.

Підсумовуючи описане вище створені модель з програмою наразі мають можливість розраховувати наступні параметри: "Відношення грошових коштів до зобов'язань", "Відношення валюти балансу до зобов'язань", "Відношення чистого прибутку до середньорічної суми активів", "Відношення прибутку до доходу від реалізації", "Відношення виробничих запасів до доходу від реалізації", "Відношення доходу від реалізації до основного капіталу" "Ймовірності банкрутства", має змогу

навчатися за допомогою технології машинного навчання та здатна надавати критичні параметри, які найбільше впливають на ймовірність банкрутства.

У результаті цієї роботи я поглибив свої знання щодо машинного навчання, його типів, способів та підходів, був ознайомлений з можливими його реалізаціями та використанням у різних стежах. Також було показано, що машинне навчання може бути успішно використане для вирішення складних завдань управління фінансами підприємств. Отримані результати можуть бути корисні для управлінських рішень та стратегій в контексті фінансової стабільності та ефективності підприємства. Даний дослід може послужити основою для подальших досліджень у цій області та розвитку нових методів оптимізації фінансових процесів.

**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Christopher Bishop "Pattern Recognition and Machine Learning", 2005, 534с.
2. Michael Negnevitsky David L. Poole and Alan K. Mackworth, "Artificial Intelligence: A Systems Approach" , 2008, 304с.
3. Weng, L. (2018). Introduction to Reinforcement Learning: An Overview / Lilian Weng URL: <https://lilianweng.github.io/lil-log/2018/02/19/a-long-peek-into-reinforcement-learning.html>
4. Brownlee, J. (2009). A Gentle Introduction to Natural Language Processing URL: <https://machinelearningmastery.com/natural-language-processing/>
5. Heron Oreljen "Deep Learning for Natural Language Processing", 2018,161с
6. David L. Poole and Alan K. Mackworth "Artificial Intelligence: Foundations of Computational Agents", 2011, 612с.
7. He, K., Zhang, X., Ren, S., Sun, J. (2015). Deep Residual Learning for Image Recognition URL: <https://arxiv.org/abs/1512.03385>
8. Heron Oreljen "Deep Learning for Natural Language Processing", 2018,161с.
9. Sebastian Raschka and Vahid Mirjalili "Python Machine Learning", 2017, 442с.
10. Christopher Bishop "Pattern Recognition and Machine Learning", 2006, 147с.
11. Степаненко О.П. Інтелектуальні системи підтримки управління діяльністю організації. Культура Причорномор'я. 2008. № 140. С. 119 - 122
12. Chollet, F. (2019). Getting Started with Keras URL: [https://keras.io/guides/sequential\\_model/](https://keras.io/guides/sequential_model/)
13. Francois Chollet "Deep Learning with Python", 2017, 192с.

14. Ситник В.Ф. Системи підтримки прийняття рішень : навч. посібник. Київ : КНЕУ, 2004. 614 с.
15. Єфіменко С. М., Степашко В. С. Прогнозна аналітика як ефективний інструмент підтримки рішень у системах цифрової економіки. Управляючі системи та машини. 2018. № 6.
16. Балацький О.Ф., Теліженко О.М., Соколов М.О. Управління інвестиціями : навч. посібник для студ. вищих навч. закл. Суми : Університетська книга, 2004. 231 с.
17. Фінансова звітність ТОВ «РОЗЕТКА». Електронний ресурс. Режим доступу: [https://clarity-project.info/edr/37193071/finances?current\\_year=2024](https://clarity-project.info/edr/37193071/finances?current_year=2024)
18. . Кваліфікаційна робота бакалавра [Електронний ресурс] : методичні рекомендації для здобувачів ступеня бакалавра освітньо-професійної програми «Системний аналіз» зі спеціальності 124 Системний аналіз / уклад.: Т.А. Желдак, Т.В. Хом'як, А.В. Малієнко ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2025. – 32 с.

## ДОДАТКИ

## Додаток А. Відомість матеріалів кваліфікаційної роботи

## Відомість матеріалів кваліфікаційної роботи

№ з/п	Позначення				Найменування	Кількість аркушів	Примітки		
1									
2					Документація				
3									
4	САУ.КР.25.33.ПЗ				Пояснювальна записка	77	Формат А4		
5									
6					Демонстраційний матеріал		Презентація на CD-R		
7									
8					Копія роботи	1	Диск CD-R		
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
					САУ.КР.25.33.ДА.ПЗ.				
Змін.	Аркуш	№ докум.	Підпис	Дата					
Розроб.		Куницький В.В.			<b>Матеріали кваліфікаційної роботи</b>	Літ.	Аркуш	Аркушів	
К. розд.		Алексеев О.М.							
Керівн.		Алексеев О.М.				НТУ «ДП», 12; 124-21-2			
Н.контр.		Хом'як Т.В.							
Зав. каф.		Желдак Т.А.							

## Додаток Б. Фінансові дані підприємства ROZETKA

Таблиця Б.1

## J0900108 Баланс (Звіт про фінансовий стан)( Актив)

Назва рядка	Код рядка	На початок звітної періоду, тис. грн	На кінець звітної періоду, тис. грн
I. Необоротні активи Нематеріальні активи	1000	219.00	27.00
первісна вартість	1001	1 384.00	1 241.00
накопичена амортизація	1002	-1 165.00	-1 214.00
Незавершені капітальні інвестиції	1005	160.00	160.00
Основні засоби	1010	62 180.00	72 045.00
первісна вартість	1011	138 578.00	174 099.00
знос	1012	-76 398.00	-102 054.00
Відстрочені податкові активи	1045	20 179.00	12 863.00
Усього за розділом I	1095	82 738.00	85 095.00
II. Оборотні активи Запаси	1100	2 159 188.00	2 505 961.00
Виробничі запаси	1101	446.00	1 035.00
Товари	1104	2 158 742.00	2 504 926.00
Дебіторська заборгованість за продукцію, товари, роботи, послуги	1125	878 734.00	1 286 662.00
Дебіторська заборгованість за розрахунками: за виданими авансами	1130	279 458.00	53 271.00
з бюджетом	1135	62 903.00	65 723.00
Інша поточна дебіторська заборгованість	1155	8.00	551.00
Гроші та їх еквіваленти	1165	137 732.00	791 591.00
Рахунки в банках	1167	137 732.00	791 591.00
Інші оборотні активи	1190	41 538.00	69 313.00
Усього за розділом II	1195	3 559 561.00	4 773 072.00
Баланс	1300	3 642 299.00	4 858 167.00

**Ю900108 Баланс (Звіт про фінансовий стан)( Пасив)**

Назва рядка	Код рядка	На початок звітної періоду, тис. грн	На кінець звітної періоду, тис. грн
I. Власний капітал Зареєстрований (пайовий) капітал	1400	1.00	1.00
Нерозподілений прибуток (непокритий збиток)	1420	332 832.00	344 848.00
Усього за розділом I	1495	332 833.00	344 849.00
Інші довгострокові зобов'язання	1515	27 184.00	61 345.00
Усього за розділом II	1595	27 184.00	61 345.00
III. Поточні зобов'язання і забезпечення Короткострокові кредити банків	1600	74 000.00	
Поточна кредиторська заборгованість за: довгостроковими зобов'язаннями	1610	50 461.00	51 347.00
товари, роботи, послуги	1615	2 617 180.00	4 058 034.00
розрахунками з бюджетом	1620	122 400.00	81 050.00
у тому числі з податку на прибуток	1621	37 401.00	25 772.00
розрахунками зі страхування	1625	5 146.00	5 930.00
розрахунками з оплати праці	1630	44 695.00	70 698.00
за одержаними авансами	1635	127 400.00	184 914.00
Інші поточні зобов'язання	1690	241 000.00	
Усього за розділом III	1695	3 282 282.00	4 451 973.00
Баланс	1900	3 642 299.00	4 858 167.00

**Ю900207 Звіт про фінансові результати (Звіт про сукупний дохід)(  
Фінансові результати)**

Назва рядка	Код рядка	За звітний період, тис. грн	За аналогічний період попереднього року, тис. грн
Чистий дохід від реалізації продукції (товарів, робіт, послуг)	2000	25 463 875.00	18 388 104.00
Собівартість реалізованої продукції (товарів, робіт, послуг)	2050	20 799 912.00	14 838 253.00
Валовий: прибуток	2090	4 663 963.00	3 549 851.00
Інші операційні доходи	2120	31 018.00	38 203.00
Адміністративні витрати	2130	181 245.00	167 524.00
Витрати на збут	2150	4 356 271.00	2 996 497.00
Інші операційні витрати	2180	154 434.00	254 323.00
Фінансовий результат від операційної діяльності: прибуток	2190	3 031.00	169 710.00
Інші фінансові доходи	2220	15 159.00	2 142.00
Інші доходи	2240	30.00	
Фінансові витрати	2250	22 396.00	21 621.00
Інші витрати	2270	-51 254.00	3 346.00
Фінансовий результат до оподаткування: прибуток	2290	47 078.00	146 885.00
Витрати (дохід) з податку на прибуток	2300	-35 062.00	-27 823.00
Чистий фінансовий результат: прибуток	2350	12 016.00	119 062.00

**Ю900108 Баланс (Звіт про фінансовий стан)( Актив)**

Назва рядка	Код рядка	На початок звітнього періоду, тис. грн	На кінець звітнього періоду, тис. грн
I. Необоротні активи Нематеріальні активи	1000	82.00	219.00
первісна вартість	1001	1 047.00	1 384.00
накопичена амортизація	1002	-965.00	-1 165.00
Незавершені капітальні інвестиції	1005	248.00	160.00
Основні засоби	1010	54 531.00	69 186.00
первісна вартість	1011	94 567.00	146 985.00
знос	1012	-40 036.00	-77 799.00
Відстрочені податкові активи	1045	10 267.00	20 179.00
Усього за розділом I	1095	65 128.00	89 744.00
II. Оборотні активи Запаси	1100	2 495 144.00	2 158 662.00
Виробничі запаси	1101	337.00	446.00
Товари	1104	2 494 807.00	2 158 216.00
Дебіторська заборгованість за продукцію, товари, роботи, послуги	1125	1 207 246.00	877 942.00
Дебіторська заборгованість за розрахунками: за виданими авансами	1130	162 098.00	280 669.00
з бюджетом	1135	122 632.00	64 979.00
Інша поточна дебіторська заборгованість	1155	45.00	1 212.00
Гроші та їх еквіваленти	1165	139 820.00	137 732.00
Рахунки в банках	1167	139 820.00	137 732.00
Інші оборотні активи	1190	40 634.00	41 537.00
Усього за розділом II	1195	4 167 619.00	3 562 733.00
Баланс	1300	4 232 747.00	3 652 477.00

**Ю900108 Баланс (Звіт про фінансовий стан)( Пасив)**

Назва рядка	Код рядка	На початок звітного періоду, тис. грн	На кінець звітного періоду, тис. грн
I. Власний капітал Зареєстрований (пайовий) капітал	1400	1.00	1.00
Нерозподілений прибуток (непокритий збиток)	1420	213 770.00	326 980.00
Усього за розділом I	1495	213 771.00	326 981.00
Інші довгострокові зобов'язання	1515	30 481.00	30 896.00
Усього за розділом II	1595	30 481.00	30 896.00
III. Поточні зобов'язання і забезпечення Короткострокові кредити банків	1600	74 000.00	
Поточна кредиторська заборгованість за: довгостроковими зобов'язаннями	1610	24 826.00	65 010.00
товари, роботи, послуги	1615	3 624 487.00	2 616 585.00
розрахунками з бюджетом	1620	4 178.00	121 384.00
у тому числі з податку на прибуток	1621	3 055.00	36 386.00
розрахунками зі страхування	1625	1 251.00	5 146.00
розрахунками з оплати праці	1630	9 646.00	44 695.00
за одержаними авансами	1635	154 105.00	126 767.00
Інші поточні зобов'язання	1690	170 002.00	241 013.00
Усього за розділом III	1695	3 988 495.00	3 294 600.00
Баланс	1900	4 232 747.00	3 652 477.00

**Ю0900207 Звіт про фінансові результати (Звіт про сукупний дохід)(  
Фінансові результати)**

Назва рядка	Код рядка	За звітний період, тис. грн	За аналогічний період попереднього року, тис. грн
Чистий дохід від реалізації продукції (товарів, робіт, послуг)	2000	18 390 610.00	21 172 647.00
Собівартість реалізованої продукції (товарів, робіт, послуг)	2050	14 840 759.00	17 935 821.00
Валовий: прибуток	2090	3 549 851.00	3 236 826.00
Інші операційні доходи	2120	33 102.00	35 912.00
Адміністративні витрати	2130	168 879.00	90 682.00
Витрати на збут	2150	2 996 510.00	3 122 791.00
Інші операційні витрати	2180	80 139.00	18 499.00
Фінансовий результат від операційної діяльності: прибуток	2190	337 425.00	40 766.00
Інші фінансові доходи	2220	2 142.00	618.00
Інші доходи	2240	1.00	11.00
Фінансові витрати	2250	21 986.00	6 121.00
Інші витрати	2270	177 564.00	1 964.00
Фінансовий результат до оподаткування: прибуток	2290	140 018.00	33 310.00
Витрати (дохід) з податку на прибуток	2300	-26 808.00	652.00
Чистий фінансовий результат: прибуток	2350	113 210.00	33 962.00

## Додаток В. Відгук керівника кваліфікаційної роботи

### Відгук на кваліфікаційну роботу бакалавра студента групи 124-21-2 Куницького Владислава Вікторовича на тему «Системний аналіз та оптимізація фінансової діяльності маркетплейсу з використанням систем штучного інтелекту» спеціальності 124 Системний аналіз

Тема кваліфікаційної роботи: «Системний аналіз та оптимізація фінансової діяльності маркетплейсу з використанням систем штучного інтелекту».

Обсяг кваліфікаційної роботи 77 стор.

Мета кваліфікаційної роботи кваліфікаційної роботи є розробка методології, що оптимізує та автоматизує пошук слабких місць в фінансовій діяльності підприємства ROZETKA та написання для неї розрахункової програми.

Актуальність теми полягає в тому, що дослідження виявиться корисним для розуміння та управління фінансовими ризиками в умовах сучасного бізнесу, а також для підвищення ефективності стратегічного планування та управління підприємством.

Тема кваліфікаційної роботи безпосередньо пов'язана з об'єктом діяльності бакалавра спеціальності 124 Системний аналіз.

Завдання, що виконані в кваліфікаційній роботі відповідають вимогам ступеня бакалавра спеціальності 124 Системний аналіз.

Практичне значення результатів кваліфікаційної роботи полягає в тому, що розроблена методологія оптимізує час на знаходження критичних параметрів та автоматизує розрахунки за допомогою програмної реалізації.

Висновки підтверджують можливість використання результатів роботи в подальшому для рішення подібних задач.

Оформлення пояснювальної записки та демонстраційного матеріалу до неї виконано згідно з вимогами. Роботу виконано самостійно, відповідно до завдання та у повному обсязі.

У роботі відзначено такі недоліки: Є зауваження до оформлення роботи.

Кваліфікаційна робота в цілому заслуговує оцінки: добре і з урахуванням висловлених зауважень автор заслуговує присвоєння освітньої кваліфікації «бакалавр з системного аналізу».

Керівник кваліфікаційної  
роботи бакалавра, к.т.н., доц.

\_\_\_\_\_ Алексєєв О.М.

**Рецензія**  
**на кваліфікаційну роботу бакалавра**  
студента групи 124-21-2 Куницького Владислава Вікторовича  
на тему «Системний аналіз та оптимізація фінансової діяльності  
маркетплейсу з використанням систем штучного інтелекту»  
спеціальності 124 Системний аналіз

Тема кваліфікаційної роботи: «Системний аналіз та оптимізація фінансової діяльності маркетплейсу з використанням систем штучного інтелекту».

Обсяг кваліфікаційної роботи 77 стор.

Висновок про відповідність кваліфікаційної роботи завданню та освітньо-професійній програмі спеціальності відповідає.

Загальна характеристика кваліфікаційної роботи, ступінь використання нормативно-методичної літератури та передового досвіду

Позитивні сторони кваліфікаційної роботи: Одним із ключових інструментів для оцінки фінансової стійкості підприємства є побудова Z-моделей та визначення ймовірності настання банкрутства. На прикладі ТОВ «ROZETKA» можна провести аналіз та виявити проблемні зони в управлінні фінансовою діяльністю, а також розробити стратегії для їх оптимізації.

Основні недоліки кваліфікаційної роботи: Є зауваження до оформлення роботи.

Оформлення пояснювальної записки та демонстраційного матеріалу до неї виконано згідно з вимогами. Роботу виконано самостійно, відповідно до завдання та у повному обсязі.

Кваліфікаційна робота в цілому заслуговує оцінки добре. \_\_\_\_

З урахуванням висловлених зауважень автор заслуговує присвоєння освітньої кваліфікації «бакалавр з системного аналізу».

Рецензент,