

Київський національний університет імені Тараса Шевченка

Економічний факультет

Кафедра економічної кібернетики

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

«Моделювання розподілу фінансових ресурсів провідних технологічних компаній»

студента 4 курсу

спеціальності 051 «Економіка»

ОПП «Економічна кібернетика»

денної форми навчання

Компанця Богдана Вікторовича

Науковий керівник:

к.е.н., доцент економічних наук

Федоренко Ірина Костянтинівна

Засвідчую, що в цій роботі немає запозичень
із праць інших авторів без відповідних
посилань

Студент _____

Роботу допущено до захисту перед ЕК
рішенням кафедри економічної кібернетики
від 05.06.2024р., протокол № 15

Завідувач кафедри: доктор економічних наук,
професор Ляшенко Олена Ігорівна _____

РЕФЕРАТ

Кваліфікаційна робота бакалавра містить: 48 ст., 14 рис., 24 джерел, додатки

Ключові слова: розподіл фінансових ресурсів, стратегічне планування, операційна ефективність, інвестиції, емпіричний аналіз, моделі прогнозування.

Об'єкт дослідження: технологічні компанії

Мета дослідження: визначити оптимальний коефіцієнт розподілу фінансових ресурсів для технологічних компаній для максимізації зростання та конкурентної переваги

Методи дослідження: збір і аналіз фінансових даних із вибірки технологічних компаній, створення та емпіричне тестування прогнозової моделі впливу розподілу ресурсів. акт

Наукова новизна, теоретична значимість дослідження: розробка прогнозової моделі для розподілу фінансових ресурсів, адаптованої до технологічних компаній, пропонуючи новий підхід до збалансування інвестицій для сталого зростання

Практична цінність: рекомендації для технологічних компаній щодо оптимізації розподілу фінансових ресурсів, покращення показників ефективності та збереження конкурентних переваг.

RESUME

Taras Shevchenko National University of Kyiv,

Faculty of Economics, Department of Economic Cybernetics

Key words:

The graduation research of student Kompanets Bohdan deals with Modeling the distribution of financial resources of leading technology companies

The work is interesting for technology companies to benchmark and improve financial resource distribution, and for investors to make informed decisions based on financial behaviors and patterns of leading technology firms

Pages 48, bibliog. 24, append. 8.

ЗМІСТ

ВСТУП	4
РОЗДІЛ I ТЕОРЕТИКО-МЕТОДОЛОГІЧНЕ ОБГРУНТУВАННЯ НЕОБХІДНОСТІ РОЗПОДІЛУ ФІНАНСОВИХ РЕСУРСІВ	6
1.1. Поняття розподілу фінансових ресурсів	6
1.2. Методологія аналізу та обробки даних при розподілі фінансових ресурсів	9
1.3. Аналіз статистичних даних при розподілі фінансових ресурсів	14
РОЗДІЛ II ПІДГОТОВКА СТАТИСТИЧНИХ ДАНИХ ДЛЯ ПОДАЛЬШОГО АНАЛІЗУ	19
2.1. Усунення затримки між інвестицією і її результатом в статистиці... ..	19
2.2. Пошук точки кризового розриву даних	23
2.3. Очистка даних від впливу незалежних ринкових механізмів	27
РОЗДІЛ III ЗАСТОСУВАННЯ РЕГРЕСІЙНОГО АНАЛІЗУ ДЛЯ ПОШУКУ ОПТИМАЛЬНОГО ВАРІАНТУ РОЗПОДІЛУ ОБМЕЖЕНИХ РЕСУРСІВ	34
3.1. Побудова варіативних регресій та їх порівняння.....	34
3.2. Створення прогнозу на основі побудованої регресії	40
3.3. Знаходження оптимального розподілу фінансових ресурсів.	45
ВИСНОВКИ	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	53
ДОДАТКИ	55

ВСТУП

Зростання технологічних компаній за останні роки було феноменальним. Ці компанії порушили традиційні бізнес-моделі, запровадили нові способи ведення бізнесу та створили цінність для своїх клієнтів. Однак із зростанням конкуренції в технологічному просторі для компаній стало критично важливим визначати оптимальне співвідношення розподілу фінансових ресурсів, щоб максимізувати свій потенціал для зростання та успіху. Розподіл фінансових ресурсів є фундаментальним аспектом будь-якого бізнесу, і технологічні компанії не є винятком. Значна частина фінансових ресурсів цих компаній зазвичай спрямовується на дослідження та розробки, інновації, які є важливими для випередження конкурентів. Однак є кілька інших сфер, таких як маркетинг та операційні витрати, які також вимагають значних інвестицій. Визначення оптимального співвідношення розподілу фінансових ресурсів є складним завданням, яке включає кілька факторів, таких як стадія розвитку компанії, її конкурентне середовище та її фінансовий стан.

У середовищі технологічних досягнень, що швидко розвивається, ефективний розподіл фінансових ресурсів став першочерговим для компаній, які прагнуть зберегти конкурентоспроможність і досягти сталого зростання.

Актуальність роботи та підстави для її виконання - актуальність цього дослідження зумовлена кількома факторами:

Волатильність ринку: технологічні компанії працюють у середовищі постійних змін, де ринкові умови можуть швидко змінюватися через інновації, нормативні зміни та зміну споживчих уподобань. Ефективний розподіл фінансових ресурсів допомагає цим компаніям залишатися гнучкими та оперативними.

Цикли інновацій: Потреба в постійних інноваціях вимагає значних інвестицій у дослідження та розробки. Визначення оптимального розподілу гарантує достатнє фінансування без шкоди для інших критичних сфер.

Капіталомісткий характер: технологічні компанії часто потребують значного капіталу для інфраструктури, обладнання та залучення талантів. Оптимальна стратегія розподілу може запобігти надмірним інвестиціям в одну сферу за рахунок інших, забезпечуючи збалансоване зростання.

Стратегічне планування: правильний розподіл фінансових ресурсів має вирішальне значення для довгострокового стратегічного планування, допомагаючи компаніям визначати пріоритетність інвестицій, які відповідають їхнім стратегічним цілям і ринковим можливостям.

Конкурентна перевага: Компанії, які володіють мистецтвом розподілу фінансових ресурсів, можуть досягти конкурентної переваги, оптимізуючи свою операційну ефективність і використовуючи ринкові можливості більш ефективно, ніж інші учасники ринку.

Об'єкт дослідження - технологічні компанії, що працюють у різних сегментах технологічного сектора, включаючи, але не обмежуючись, розробку програмного забезпечення, виробництво апаратного забезпечення, інформаційно-технологічні послуги та нові технології, такі як штучний інтелект, блокчейн та Інтернет речей.

Предмет дослідження - моделі та методи розподілу обмежених ресурсів та їх прогнозування для технологічних компаній

Цілі дослідження

Для досягнення мети цієї роботи було поставлено та виконано такі задачі:

Огляд літератури про стратегії розподілу фінансових ресурсів, зокрема в контексті технологічних компаній. Збір даних та аналіз фінансових даних з репрезентативної вибірки технологічних компаній, для формування розуміння поточної практики розподілу та її результатів. Розробка моделі для прогнозування впливу різних стратегій розподілу на ефективність компанії. Емпіричний аналіз: Перевірка теоретичної моделі, використовуючи емпіричні дані, щоб підтвердити її точність і надійність. Визначення оптимального співвідношення розподілу фінансових ресурсів, які корелюють із покращеними показниками ефективності, такими як зростання доходу, прибутковість і ціна акцій.

РОЗДІЛ І ТЕОРЕТИКО-МЕТОДОЛОГІЧНЕ ОБГРУНТУВАННЯ НЕОБХІДНОСТІ РОЗПОДІЛУ ФІНАНСОВИХ РЕСУРСІВ

1.1. Поняття розподілу фінансових ресурсів

Розподіл фінансових ресурсів передбачає розподіл коштів всередині організації. Для технологічних компаній це означає визначення способу розподілу капіталу між різними функціями, проектами та підрозділами. Ефективний розподіл забезпечує оптимальне використання ресурсів, що безпосередньо впливає на продуктивність, зростання та інновації.

Правильний розподіл фінансових ресурсів є критичним з кількох причин:

Зменшення ризику. Диверсифікуючи інвестиції в різні сфери, компанії зменшують ризик, пов'язаний із надмірною концентрацією. Наприклад, розподіл ресурсів як на дослідження та розробки (R&D), так і на маркетинг забезпечує збалансований профіль ризику.

Стратегічне узгодження: розподіл ресурсів має відповідати стратегічним цілям. Критичні ініціативи, такі як розробка продукту чи розширення ринку, потребують належного фінансування. Стратегічне узгодження гарантує, що ресурси підтримують довгострокові цілі.

Ефективність: ефективний розподіл максимізує віддачу. Компанії повинні знайти баланс між інвестуванням у зростання (наприклад, дослідження та розробки) та підтриманням повсякденної діяльності (операційні витрати). Оптимізація цього балансу підвищує загальну ефективність.

Оптимальність означає досягнення найкращого можливого результату з урахуванням обмежень. У контексті розподілу фінансових ресурсів оптимальність передбачає знаходження правильного балансу між ризиком і винагородою. Основні міркування включають:

Толерантність до ризику: різні компанії мають різний апетит до ризику. Оптимальність враховує рівні стійкості до ризику під час розподілу ресурсів. Наприклад, стартап може терпіти вищий ризик потенційного швидкого зростання, тоді як відома компанія може надавати пріоритет стабільності.

Компроміси: Розподіл ресурсів передбачає компроміси. Компанії повинні вирішити, скільки інвестувати в інновації (наприклад, дослідження та розробки) проти збереження існуючої діяльності (наприклад, маркетинг). Оптимальність прагне максимізувати загальну цінність шляхом прийняття обґрунтованих компромісів.

Коефіцієнти відіграють вирішальну роль у розподілі ресурсів. Ось деякі відповідні співвідношення:

Капітальні витрати (CapEx) до операційних витрат (OpEx):

Капекс – це довгострокові інвестиції (наприклад, придбання обладнання або розширення потужностей). Операційні витрати включають повсякденні операційні витрати (наприклад, зарплати, комунальні послуги). Збалансування капітальних і операційних витрат забезпечує стабільне зростання без шкоди для поточних операцій.

Співвідношення боргу до власного капіталу:

Цей коефіцієнт порівнює борг компанії (позичені кошти) з її власним капіталом (інвестиції акціонерів). Високе відношення боргу до власного капіталу вказує на залежність від боргового фінансування, тоді як низьке співвідношення передбачає власне фінансування. Оптимізація цього коефіцієнта впливає на фінансову стабільність і управління ризиками.

Рентабельність інвестицій (ROI):

ROI вимірює ефективність розподілу ресурсів. Розраховується як

$$ROI = (\text{чистий прибуток} / \text{загальні інвестиції}) \times 100\%$$

він оцінює віддачу від інвестованого капіталу. Максимальна рентабельність інвестицій забезпечує ефективне використання ресурсів. Численні дослідження досліджують стратегії розподілу ресурсів у технологічних компаніях. [1]

Деякі ключові висновки включають:

Динамічний розподіл ресурсів: Компанії повинні адаптувати розподіл ресурсів відповідно до мінливих ринкових умов і конкурентного середовища. Гнучкість дозволяє перерозподіляти ресурси за потреби.

Теорія портфоліо:

Застосування принципів диверсифікації портфеля до розподілу ресурсів може підвищити стабільність. Диверсифікація інвестицій між різними проектами чи бізнес-підрозділами зменшує загальний ризик.[2]

Теорія реальних опціонів: Перегляд розподілу ресурсів як варіантів (подібно до фінансових варіантів) забезпечує гнучкість. Компанії можуть відкласти або розширити інвестиції залежно від розвитку ринку. Різні зацікавлені сторони отримують вигоду від результатів розподілу ресурсів:

Керівництво та ради: Особи, які приймають стратегічні рішення, покладаються на інформацію про розподіл ресурсів, щоб стимулювати зростання та інновації. Вони враховують ризик, рентабельність інвестицій і узгодженість зі стратегічними цілями.

Інвестори: Прозорість розподілу коштів впливає на довіру інвесторів. Інвестори оцінюють, наскільки ефективно використовуються ресурси для отримання прибутку.

Співробітники: Справедливий розподіл впливає на мораль і продуктивність. Співробітники цінують розподіл ресурсів, який підтримує їх роботу.

Регулятори та аналітики: Регулятори оцінюють дотримання фінансових правил. Аналітики оцінюють практику розподілу ресурсів для оцінки продуктивності.[3]

Фінансовий баланс дає миттєвий знімок фінансового стану компанії. Ключові компоненти включають: Активи: представляють те, чим володіє компанія (наприклад, готівка, інвентар, майно). Зобов'язання: відображають заборгованість компанії (наприклад, борги, кредиторська заборгованість). Акціонерний капітал: відображає залишкову частку після вирахування зобов'язань з активів. Бухгалтерське рівняння відображає фундаментальний зв'язок.

Активи = Зобов'язання + Акціонерний капітал

1.2. Методологія аналізу та обробки даних при розподілі фінансових ресурсів

Технічні індикатори, також відомі як «технічні показники», відіграють вирішальну роль в аналізі руху цін на акції. Ці математичні розрахунки базуються на історичних даних торгівлі (таких як ціна, обсяг і відкритий інтерес). Трейдери використовують технічні індикатори для виявлення моделей, трендів і потенційних точок входу або виходу. Давайте розглянемо деякі ключові технічні індикатори:

Accumulation Distribution Line (ADL):

$$ADL = ADL_{i-1} + \frac{((Close - Low) - (High - Close))}{High - Low} \cdot Volume$$

Де ADL_{i-1} - попереднє значення ADL.

Close - ціна закриття.

Low і High - найнижча і найвища ціни.

Volume - обсяг торгів.

ADL вимірює накопичення та розподіл активів, використовуючи обсяг та цінові дані. Він допомагає визначити, чи актив накопичується (купується) або розподіляється (продається) трейдерами. Цей індикатор може вказувати на майбутні зміни в ціні, якщо рух ADL не підтверджується ціною активу.[4]

Adaptive Moving Average (AMA):

$$AMA = \frac{ER \cdot (Close - AMA_{i-1}) + AMA_{i-1}}{SSC}$$

Де ER - коефіцієнт ефективності.

AMA_{i-1} - попереднє значення AMA.

Close - ціна закриття.

SSC - константа згладжування.

AMA адаптується до волатильності ринку, використовуючи коефіцієнт ефективності (ER) для регулювання швидкості рухомої середньої, щоб вона була більш чутливою до недавніх цінових змін. ER вимірюється як відношення між ціновим рухом та волатильністю, дозволяючи AMA швидко реагувати на ринкові зміни.

Aroon:

$$AroonUp = \frac{(n - DaysSinceHigh)}{n} \cdot 100$$

$$AroonDown = \frac{(n - DaysSinceLow)}{n} \cdot 100$$

Де n - період.

$DaysSinceHigh$ - кількість днів з моменту найвищого максимуму.

$DaysSinceLow$ - кількість днів з моменту найнижчого мінімуму.

Aroon визначає, чи актив перебуває в тренді, і силу цього тренду, вимірюючи час, що минув з моменту останнього максимуму та мінімуму ціни. Індикатор складається з двох ліній: Aroon Up та Aroon Down, які допомагають визначити, чи тренд рухається вгору чи вниз. [5]

Average True Range (ATR):

$$ATR = \frac{1}{n} \sum_{i=1}^n TR_i$$

Де TR_i - True Range для кожного періоду.

ATR вимірює волатильність активу, використовуючи “істинний діапазон” (TR), який включає поточні максимуми та мінімуми, а також попередні закриття цін. Цей індикатор корисний для визначення рівнів стоп-лосу та тейк-профіту

Awesome Oscillator (AO):

$$AO = SMA(MedianPrice, 5) - SMA(MedianPrice, 34)$$

$$\text{Де } MedianPrice = \frac{High+Low}{2}$$

АО - це індикатор моментуму, який використовує різницю між короткотерміною та довготерміною простими рухомими середніми з медіанних цін, щоб визначити ринкову динаміку. Зелені стовпці АО вказують на зростання моментуму, тоді як червоні стовпці вказують на його зниження. [6]

Середня Bollinger Band (MBB):

$$MBB = SMA(Close, n)$$

Верхня Bollinger Band (UBB):

$$UBB = MBB + d \cdot \text{Стандартне відхилення}$$

Нижня Bollinger Band (LBB):

$LBB = MBV - d \cdot \text{Стандартне відхилення}$

Де d - фактор відхилення.

Bollinger Bands складаються з трьох ліній: основної рухомої середньої (MBV) та двох ліній, що розташовані на певну кількість стандартних відхилень вище та нижче цієї середньої. Ці лінії допомагають визначити перекупленість або перепроданість активу.

Commodity Channel Index (CCI):

$$CCI \geq \frac{TP - SMA(TP, n)}{0.015 \cdot \text{Середнє відхилення}}$$

Де TP - Типова ціна.

$SMA(TP, n)$ - Проста рухома середня типової ціни.

Середнє відхилення - середнє відхилення типової ціни.

CCI використовується для визначення перекупленості або перепроданості активу, порівнюючи його поточну ціну з історичною середньою ціною. Цей індикатор може вказувати на потенційні точки повороту ціни.

Positive Directional Indicator (+DI):

$$+DI = \frac{\text{Smoothed } +DM}{ATR} \times 100$$

Negative Directional Indicator (-DI):

$$-DI = \frac{\text{Smoothed } -DM}{ATR} \times 100$$

Directional Movement Index (DX):

$$DX = \left(\frac{|+DI - (-DI)|}{(+DI) + (-DI)} \right) \times 100$$

Average Directional Index (ADX):

$$ADX = 14(\text{Prior}ADX \times 13) + \text{Current}DX$$

Де $(+DM)$ (Positive Directional Movement) =

$\text{Current High} - \text{Previous High}$

$(-DM)$ (Negative Directional Movement) = $\text{Previous Low} - \text{Current Low}$

$$Smoothed(\pm DM) = \left(\sum_{i=1}^{14} \pm DM_i \right) - \left(\frac{\sum_{i=1}^{14} \pm DM_i}{14} \right) + Current(\pm DM)$$

CDM = Current Directional Movement

ATR = Average True Range

Envelope:

Верхня Envelope:

(UpperEnvelope = MBV + d · Стандартне відхилення)

Нижня Envelope:

(LowerEnvelope = MBV - d · Стандартне відхилення)

Де MBV - Середня Bollinger Band.

d - фактор відхилення.

Envelope використовує рухомі середні для створення каналів вище та нижче ціни, що допомагає визначити потенційні рівні підтримки та опору.

Методологія аналізу та обробки даних:

Ефективний розподіл ресурсів починається з надійного збору даних. Компанії збирають фінансові дані з різних джерел, зокрема:

Фінансові звіти: Баланси, звіти про доходи та звіти про рух грошових коштів надають важливі дані про активи, зобов'язання, доходи та витрати. Ринкові дані: ціни на акції, обсяги торгів та ринкові індекси дають змогу зрозуміти настрої інвесторів і загальні ринкові умови. Внутрішні записи: журнали транзакцій, бюджетні звіти та дані про конкретний проект надають детальну інформацію.

Попередня обробка даних

Перед аналізом дані проходять попередню обробку для забезпечення точності та узгодженості: Очищення: видалення викидів, виправлення помилок і обробка відсутніх значень. Нормалізація: масштабування даних до загального діапазону (наприклад, від 0 до 1) для порівняння. Агрегація: узагальнення даних на відповідних рівнях (наприклад, щоквартально або щорічно).

Описова статистика надає огляд даних:

Показники центральної тенденції: середнє значення, медіана та мода. Міри дисперсії: дисперсія, стандартне відхилення та діапазон. Візуалізація даних: гістограми, діаграми розсіювання та прямокутні діаграми.

Економетричний інструментарій. Економетрика застосовує статистичні методи до економічних даних. Ключові інструменти включають:

Регресійний аналіз:

Лінійна регресія моделює зв'язки між змінними (наприклад, розподіл ресурсів і зростання доходу). Множинна регресія розглядає кілька предикторів одночасно. Аналіз часових рядів: Аналіз даних за певний час (наприклад, ціни на акції, тенденції доходів). Моделі авторегресії інтегрованого ковзного середнього (ARIMA) фіксують часові закономірності. Поєднання даних перехресного перерізу та часових рядів. Моделі фіксованих або випадкових ефектів враховують індивідуальні та часові ефекти. Прогнозування майбутніх потреб у ресурсах передбачає: Прогнозування часових рядів: використання історичних даних для прогнозування майбутніх тенденцій. Аналіз сценаріїв: Оцінка розподілу ресурсів за різними сценаріями (наприклад, зростання ринку, економічні спади).

1.3. Аналіз статистичних даних при розподілі фінансових ресурсів

Компоненти форми SEC 10-Q. Форма 10-Q є важливим джерелом інформації для фінансового аналізу технологічних компаній, оскільки вона надає детальну інформацію про фінансовий стан і результати їх діяльності. Особливу увагу при аналізі привертають такі ключові компоненти, як валове майно, заводи та обладнання (PP&E), накопичений знос і амортизація, чисте майно, заводи та обладнання, нараховані гарантії та супутні витрати, нарахована компенсація та виплати працівникам, відкладена маржа від продажу компонентів, нараховані витрати на маркетинг і розповсюдження, податки на прибуток до сплати, інші поточні зобов'язання, загальна сума нарахованих витрат, відстрочені податкові зобов'язання, інші непоточні зобов'язання, нараховані податки, нараховані витрати на маркетинг і збут, кумулятивний ефект зміни принципу бухгалтерського обліку та чистий прибуток.[7]

Валове майно, заводи та обладнання. Валове майно, заводи та обладнання (PP&E) є ключовим показником фізичної інфраструктури технологічних компаній, включаючи землю, будівлі, машини та обладнання. Цей показник відображає загальну вартість матеріальних активів, що належать компанії і є важливим для оцінки її виробничих можливостей та операційної ефективності.

Накопичений знос і амортизація. Накопичений знос і амортизація відображає зменшення вартості валового PP&E через його фізичне зношення та застаріння. Ця витрата впливає на фінансову звітність компанії, знижуючи її прибутковість та впливаючи на загальний фінансовий результат.

Чисте майно, заводи та обладнання. Чисте PP&E є різницею між валовим PP&E та накопиченою амортизацією і точніше відображає фактичну вартість активів компанії на даний момент. Цей показник враховує знос та амортизацію активів, що є важливим для правильного оцінювання майнових ресурсів підприємства.

Нарахована гарантія та супутні витрати. Нараховані гарантійні витрати відображають зобов'язання компанії щодо майбутніх витрат на ремонт або заміну дефектних продуктів протягом гарантійного періоду. Ця відповідальність впливає на фінансові результати компанії та її грошовий потік.

Нарахована компенсація та виплати працівникам. Нарахована компенсація включає заробітну плату, премії та інші виплати, які належать працівникам, але ще не були виплачені. Ця відповідальність гарантує, що працівники отримають належну компенсацію і впливає на фінансову стабільність підприємства.

Відкладена маржа від продажу компонентів. Відкладена маржа від продажу компонентів виникає внаслідок умов контракту або зобов'язань, що впливають на час визнання доходу. Цей показник впливає на фінансові результати компанії та показники прибутковості.

Нараховані витрати на маркетинг і розповсюдження. Нараховані витрати на маркетинг і розповсюдження охоплюють витрати, пов'язані з рекламою, просуванням та розповсюдженням продукції компанії. Ці зобов'язання впливають на ефективність маркетингових стратегій і фінансові результати.

Податки на прибуток до сплати. Податки на прибуток до сплати є поточними податковими зобов'язаннями компанії перед податковими органами. Цей показник відображає податкове становище компанії та його вплив на грошовий потік.

Інші поточні зобов'язання. Інші поточні зобов'язання включають різноманітні короткострокові зобов'язання, які не класифікуються в інших категоріях. Ці зобов'язання впливають на ліквідність компанії та управління оборотним капіталом.

Загальна сума нарахованих витрат. Загальна сума нарахованих витрат об'єднує різні нараховані зобов'язання, включаючи компенсацію, податки та гарантії. Цей показник надає загальне уявлення про короткострокові фінансові зобов'язання компанії.

Відстрочені податкові зобов'язання. Відстрочені податкові зобов'язання виникають внаслідок тимчасових різниць між бухгалтерським та податковим обліком. Ці зобов'язання впливають на майбутні податкові платежі та стратегії фінансового планування компанії.

Інші непоточні зобов'язання. Інші непоточні зобов'язання охоплюють довгострокові зобов'язання компанії, які не мають категорійного класифікації як поточні. Ці зобов'язання впливають на довгострокову фінансову стабільність підприємства.

Нараховані податки. Нараховані податки відображають податкові зобов'язання компанії, які вже заборговані, але ще не сплачені. Цей показник впливає на грошовий потік та фінансові результати компанії.

Нараховані витрати на маркетинг і збут. Нараховані витрати на маркетинг і збут включають витрати, пов'язані з продажами та рекламною діяльністю. Ці зобов'язання впливають на прибутковість компанії та ефективність маркетингових стратегій.

Кумулятивний ефект зміни принципу бухгалтерського обліку. Кумулятивний ефект зміни принципу бухгалтерського обліку відображає коригування, які виникають в результаті змін в правилах бухгалтерського обліку. Цей аспект забезпечує узгодженість фінансової звітності та впливає на історичну фінансову звітність компанії.

Чистий прибуток. Чистий прибуток є ключовим показником фінансової ефективності компанії, який відображає загальний прибуток після врахування всіх витрат, податків та відсотків. Цей показник критично важливий для оцінки фінансової стабільності та прибутковості підприємства.[8]

PART I — FINANCIAL INFORMATION

Item 1. Financial Statements

Apple Inc.

CONDENSED CONSOLIDATED STATEMENTS OF OPERATIONS (Unaudited)
(In millions, except number of shares which are reflected in thousands and per share amounts)

	Three Months Ended		Nine Months Ended	
	July 1, 2023	June 25, 2022	July 1, 2023	June 25, 2022
Net sales:				
Products	\$ 60,584	\$ 63,355	\$ 230,901	\$ 245,241
Services	21,213	19,604	62,886	58,941
Total net sales	81,797	82,959	293,787	304,182
Cost of sales:				
Products	39,136	41,485	146,696	155,084
Services	6,248	5,589	18,370	16,411
Total cost of sales	45,384	47,074	165,066	171,495
Gross margin	36,413	35,885	128,721	132,687
Operating expenses:				
Research and development	7,442	6,797	22,608	19,490
Selling, general and administrative	5,973	6,012	18,781	18,654
Total operating expenses	13,415	12,809	41,389	38,144
Operating income	22,998	23,076	87,332	94,543
Other income/(expense), net	(265)	(10)	(594)	(97)
Income before provision for income taxes	22,733	23,066	86,738	94,446
Provision for income taxes	2,852	3,624	12,699	15,364
Net income	\$ 19,881	\$ 19,442	\$ 74,039	\$ 79,082
Earnings per share:				
Basic	\$ 1.27	\$ 1.20	\$ 4.69	\$ 4.86
Diluted	\$ 1.26	\$ 1.20	\$ 4.67	\$ 4.82
Shares used in computing earnings per share:				
Basic	15,697,614	16,162,945	15,792,497	16,277,824
Diluted	15,775,021	16,262,203	15,859,263	16,394,937

Рис. 1.1. Приклад форми 10-Q технологічної компанії Джерело: [9]

Таким чином, аналіз статистичних даних форми 10-Q забезпечує глибоке розуміння розподілу фінансових ресурсів технологічних компаній і дозволяє здійснювати обґрунтовані фінансові рішення з максимальною достовірністю.

Форма SEC 10-Q є одним із ключових фінансових звітів, що подається публічними компаніями в США, відповідно до вимог Комісії з цінних паперів і бірж (SEC). Цей звіт має значний вплив на фінансовий аналіз і прийняття управлінських рішень, оскільки надає інвесторам і регулювальним органам інформацію про

фінансове становище, результати операцій та зміни в фінансових показниках підприємства.

Обов'язкові показники форми 10-Q. Основні обов'язкові показники, які повинні бути включені до форми 10-Q, включають фінансові звіти, які охоплюють звітний період, зокрема: Баланс стану справ на кінець періоду: включає активи, зобов'язання та власний капітал компанії на звітну дату. Звіт про прибутки і збитки за період: показує дохід, витрати і чистий прибуток (або збиток) за звітний період. Звіт про зміни в власному капіталі: включає рухи в капіталі компанії, такі як випуск акцій, дивіденди, переоцінка активів тощо. Звіт про готовність і ліквідність: показує здатність компанії здійснювати платежі в найближчому майбутньому та її поточні ліквідні активи.

Відрізняються показники в залежності від сфери компанії. Зміст та акценти форми 10-Q можуть варіюватися залежно від сфери діяльності компанії. Наприклад:

Технологічні компанії: можуть акцентувати увагу на витратах на дослідження та розробку, оцінці інтелектуальної власності та інноваційних процесах. Фінансові установи: мають показники, які відображають їхні кредитні операції, рівень ризику активів та пасивів, а також капіталовкладення. Зміни у стандартах оформлення форми

Останні роки свідчать про значні зміни у стандартах оформлення форми 10-Q. Основні зміни включають: Впровадження XBRL: обов'язкове подання фінансових звітів у форматі eXtensible Business Reporting Language для полегшення аналізу даних і їхнього порівняння. Посилення вимог до дисклозур: розширення і деталізація інформації про ризики, зміни управлінських структур та інші ключові аспекти, що впливають на фінансове становище. Адаптація до нових стандартів бухгалтерського обліку (наприклад, IFRS або нові стандарти FASB): впровадження нових підходів до оцінки активів, заборгованості та інші зміни, які можуть впливати на фінансову звітність компаній.

РОЗДІЛ II ПІДГОТОВКА СТАТИСТИЧНИХ ДАНИХ ДЛЯ ПОДАЛЬШОГО АНАЛІЗУ

2.1. Усунення затримки між інвестицією і її результатом в статистиці

Прогнозні моделі оцінюють майбутні результати, долаючи розрив між інвестиційними рішеннями та спостережуваними результатами. Вони керують розподілом ресурсів, прогнозуючи дохід, попит та інші КРІ. Прогнозування продажів: прогнозовані майбутні продажі на основі історичних моделей.

Очікуваний попит на продукт для управління запасами. Оцінка ризику: змодельований ризик для оптимізації рішень. Навчені моделі з використанням історичних даних. Гнучкі підходи дозволяють ітераційні коригування розподілу ресурсів. Менш часті рішення зменшують вплив затримок.

Аналіз даних за певний час виявив тенденції, сезонність і циклічність. Це допомогло визначити основні фактори продуктивності. Застосовані ковзні середні, експоненціальне згладжування та авторегресійні моделі. Перевірені моделі на основі даних поза вибіркою.[10]

Видалення сезонних коливань покращило аналіз трендів. Він ізолював базові візерунки від шуму. Застосовані методи сезонної декомпозиції (наприклад, декомпозиція STL). Оцінено сезонно скориговані дані для суттєвої інформації.

Прогнозування часових рядів відіграє ключову роль у різних сферах, від економіки до епідеміології, де точні прогнози майбутніх тенденцій і моделей є вирішальними для прийняття рішень. Серед арсеналу методів прогнозування моделі SARIMA (Seasonal Autoregressive Integrated Moving Average) виділяються своєю здатністю обробляти як несезонні, так і сезонні компоненти в даних часових рядів. У цьому розділі детально досліджується SARIMA, від теоретичних основ до практичного застосування, щоб забезпечити всебічне розуміння її можливостей і нюансів у прогнозуванні.

Теоретичні основи SARIMA

ARIMA та її розширення до SARIMA

Моделі авторегресійної інтегрованої ковзної середньої (ARIMA) утворюють основу, на якій побудовано SARIMA. Моделі ARIMA призначені для фіксації

автокореляції та стаціонарності даних часових рядів за допомогою трьох основних компонентів: авторегресії (AR), інтегрованої (I) і ковзного середнього (MA). Моделі ARIMA ефективні для несезонних даних, але потребують розширення для ефективної обробки сезонних моделей, що призводить до розробки SARIMA.

$$(1 - \phi_1 B)(1 - \Phi_1 B^s)(1 - B)(1 - B^s)y_t = (1 + \theta_1 B)(1 + \Theta_1 B^s)\varepsilon_t$$

Сезонний компонент в SARIMA

SARIMA вводить додаткові параметри для явного моделювання сезонних моделей:

Сезонна авторегресія (SAR): моделює зв'язок між даними спостереження та його сезонними лагованими значеннями.

Сезонне ковзне середнє (SMA): моделює залежність між спостереженням і залишковими помилками від моделі ковзного середнього, застосованої до сезонних лагованих значень.

Сезонна інтеграція (SI): включає розрізнення рядів за сезонним лагом для досягнення стаціонарності.[11]

Процес моделювання SARIMA

Підготовка даних

Перед застосуванням SARIMA важливо підготувати дані:

Виявлення сезонності: Візуальний огляд графіка часових рядів або методи декомпозиції, такі як декомпозиція STL, допомагають визначити сезонні моделі.

Стаціонарність: SARIMA вимагає стаціонарності; таким чином, методи розрізнення (як несезонні, так і сезонні) застосовуються, поки ряд не стане стаціонарним.

Ідентифікація моделі. Визначення відповідних параметрів $p, d, q, P, D, Q, p, d, q, P, D, Q$ та сезонного періоду ss передбачає: Аналіз ACF і PACF: графіки функції автокореляції (ACF) і функції часткової автокореляції (PACF) допомагають визначити початкові значення для AR, MA та сезонних порядків. Вибір параметрів: ітераційне уточнення параметрів на основі діагностики моделі та інформаційних критеріїв (AIC, BIC).

Оцінка та перевірка моделі. Підгонка моделі SARIMA: використовуйте такі методи оцінки, як оцінка максимальної правдоподібності (MLE), щоб підібрати модель SARIMA до попередньо оброблених даних.

Діагностична перевірка: оцініть адекватність моделі, перевіряючи залишки на випадковість, автокореляцію та нормальність за допомогою графіків ACF, тесту Лjung-Бокса та графіків Q-Q.

Прогнозування та вибір моделі. Генерація прогнозу: після перевірки моделі SARIMA використовуйте її для створення прогнозів для майбутніх часових точок.

Вибір моделі: порівняйте різні специфікації SARIMA, використовуючи інформаційні критерії, щоб вибрати модель із найкращим балансом відповідності та складності.

Порівняльний аналіз: ARIMA проти SARIMA. У той час як моделі ARIMA підходять для несезонних даних, SARIMA чудово фіксує сезонні коливання. Розширення SARIMA дає змогу робити точніші прогнози в наборах даних, що демонструють повторювані сезонні закономірності, наприклад дані про щомісячні продажі або сезонні епідеміологічні тенденції. Тематичні дослідження та застосування в реальному світі. [12]

Фінансове прогнозування: моделі SARIMA широко використовуються у фінансах для прогнозування цін на акції, де сезонні моделі та історичні тенденції відіграють вирішальну роль.

Складність моделі та можливість інтерпретації. Ефективність SARIMA залежить від правильного визначення та моделювання сезонних моделей. Однак підвищена складність порівняно з ARIMA вимагає ретельного вибору моделі та інтерпретації результатів.

Вимоги до даних і припущення. SARIMA припускає, що основний часовий ряд є стаціонарним після розрізнення та що майбутні спостереження можна передбачити на основі минулої поведінки. Адекватні історичні дані та періодична перевірка є

важливими для підтримки релевантності та точності моделі.

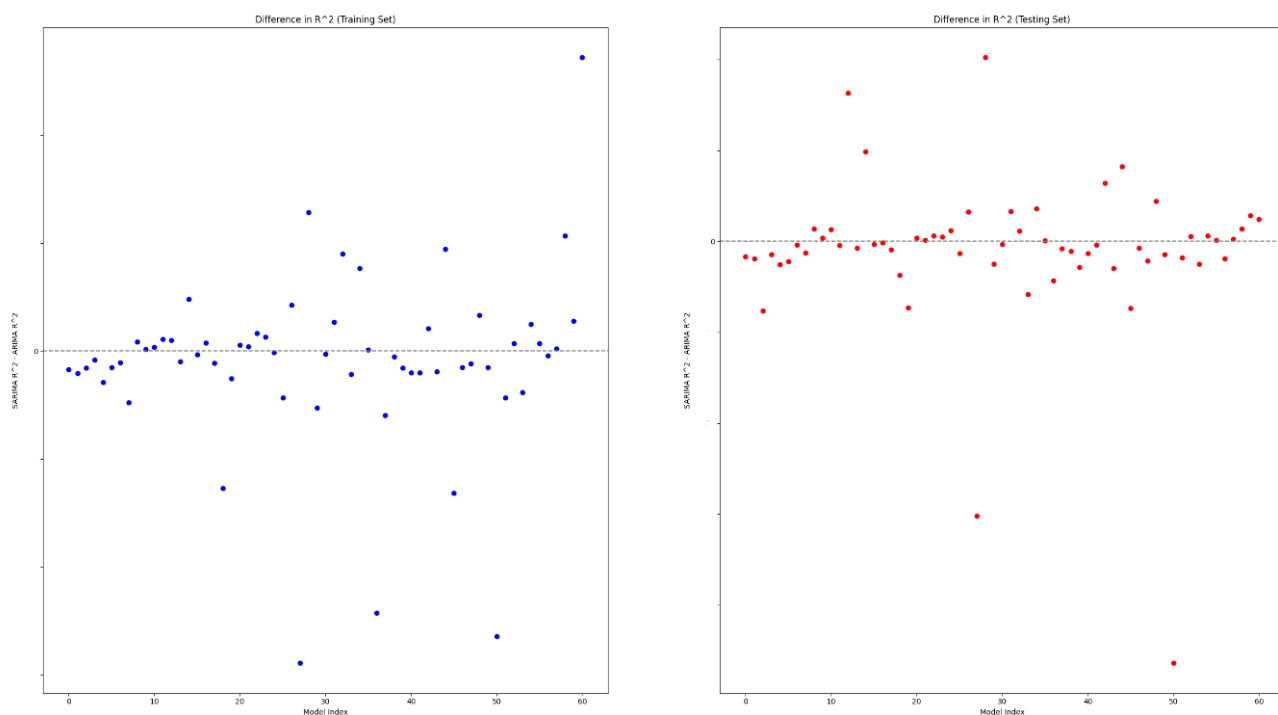


Рис. 2.1. Різниця R^2 між SARIMA та ARIMA на тренувальній та тестовій множині Джерело: додаток Ж

Моделі SARIMA є потужним інструментом у наборі інструментів прогнозування, особливо придатним для даних часових рядів із чіткими сезонними моделями. Розширюючи можливості ARIMA для адаптації до сезонних коливань, SARIMA забезпечує точніші та глибші прогнози для різноманітних програм. Постійне вдосконалення діагностики моделей і обчислювальних методів обіцяє подальше підвищення ефективності та застосовності SARIMA в практиці прогнозування.

2.2. Пошук точки кризового розриву даних

Призначення алгоритму Пельта:

Виявлення точки зміни: Pelt спеціально розроблено для виявлення точок зміни в даних часових рядів, коли статистичні властивості (середнє значення, дисперсія тощо) раптово змінюються. Ці зміни часто свідчать про зрушення або перерви в основному процесі генерування даних.

Ключові характеристики Pelt:

Pelt одночасно точний і ефективний, що дозволяє йому знаходити точки зміни оптимально в лінійному часі, що робить його придатним для великих наборів даних. Стратегія скорочення: Pelt використовує стратегію скорочення, щоб зменшити складність обчислень, зберігаючи при цьому точність. Він обрізає субоптимальні сегментації на основі функції вартості. Розрахунок вартості сегмента: Pelt оцінює вартість кожного потенційного сегмента (розділу часового ряду) за допомогою обраної моделі або функції витрат. Вартість сегмента зазвичай включає штраф за додавання нового сегмента та придатний термін, який визначає, наскільки сегмент відповідає даним. [13]

Формули, що використовуються в Pelt:

Функція витрат:

Алгоритм Пелта мінімізує функцію вартості, де $cost(t, \tau)$ – поточний часовий індекс, а τ – потенційна точка зміни:

$$cost(t, \tau) = cost(\tau) + penalty(t - \tau)$$

Вартість сегмента $cost(t, \tau)$: цей термін оцінює, наскільки добре дані до t відповідають вибраній моделі. Наприклад, у сценарії виявлення середнього зсуву (як-от використання ядра "rbf") це може виміряти квадрат помилки між спостережуваними значеннями та прогнозованими значеннями за стаціонарного припущення.

Штраф $penalty(t - \tau)$: Цей термін штрафує за введення нового сегмента (точка зміни) у t . Це врівноважує компроміс між правильною підгонкою даних і додаванням складності.

Стратегія оптимізації:

Pelt оптимізує цю функцію витрат за допомогою динамічного програмування. Він ефективно обчислює сегментацію мінімальних витрат, враховуючи потенційні точки зміни та рекурсивно оновлюючи вартість у міру її просування в часовому ряду.

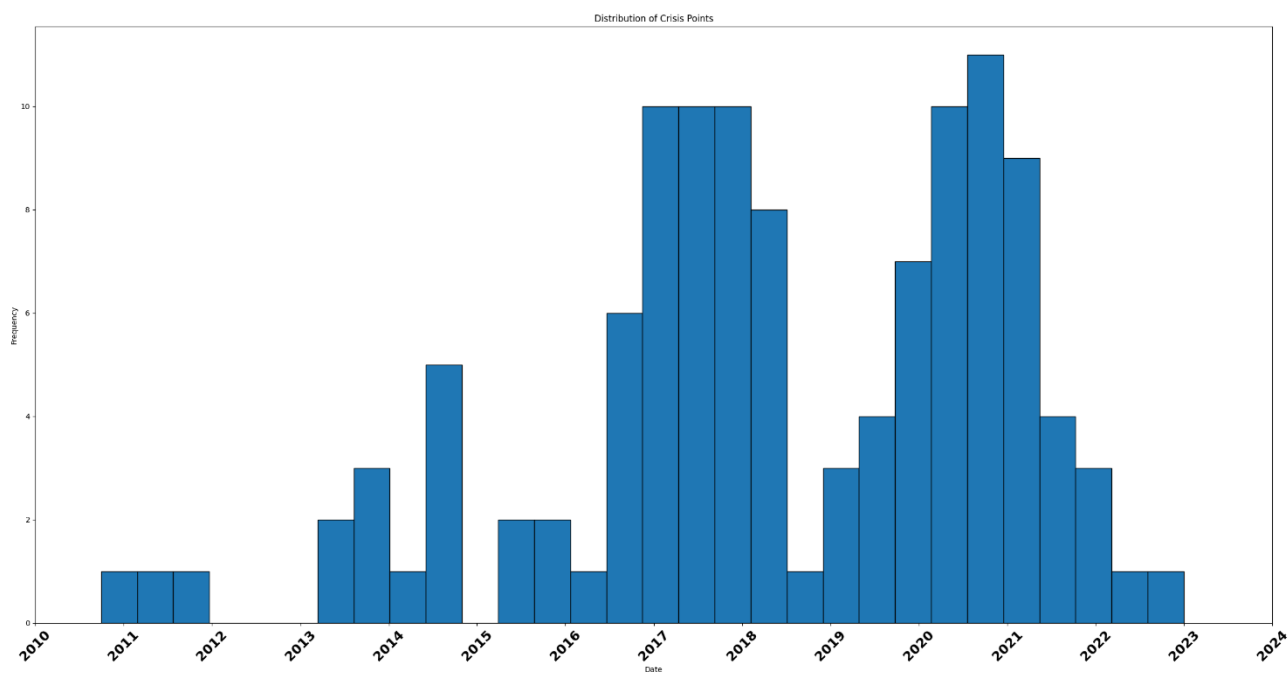


Рис. 2.2. Розподіл точок кризового розриву даних на часовому ряді. Джерело: додаток E

Набір даних, що аналізується, включає кількість кризових точок, зареєстрованих у 60 технологічних компаніях за квартал, з 2010 по 2024 рік. Мета полягає в тому, щоб визначити тенденції та потенційні причинні фактори, що стоять за коливаннями кризових точок протягом цього періоду. Наступний аналіз заглиблюється в дані, висвітлюючи значні піки, тенденції та правдоподібні пояснення спостережуваних моделей.

Ключові зауваження

Пік 2 кварталу 2017 року: набір даних вказує на найвищу кількість кризових точок у другому кварталі 2017 року, загалом 11 криз.

Вторинні піки: додаткові значні сплески спостерігаються в першому кварталі 2021 року (8 криз) і в трьох окремих кварталах — 4 квартал 2016 року, 2 квартал 2020 року та 3 квартал 2020 року — у кожному з них зафіксовано 7 криз.

Помірна активність: кілька кварталів у межах набору даних демонструють помірну активність кризових точок, коливається від 4 до 6 криз.

Низька активність: численні квартали, особливо на початку десятиліття, демонструють нижчу активність у кризовій точці з показниками від 1 до 3.

Незначна активність на початку десятиліття: з 2010 по 2015 роки кризові точки відносно рідкісні, що свідчить про менше збоїв протягом цього періоду.

Аналіз тенденцій

2010-2015: початковий період характеризується низькими показниками кризових точок, які періодично перериваються піками. Це свідчить про менше збоїв на ринку або менш інтенсивний контроль за технологічними фірмами. Відносно стабільний період можна пояснити регулятивним середовищем, що зароджується, і появою структур кібербезпеки.

2016-2017: цей інтервал знаменує значне зростання кризових точок, кульмінацією якого став різкий пік у 2 кварталі 2017 року. Цей період може відображати посилену регулятивну перевірку, сплеск інцидентів із кібербезпекою або підвищену волатильність ринку, що впливає на технологічні підприємства.

2018-2019: набір даних вказує на відносно стабільну фазу з помірною активністю кризових точок. Ця стабільність може свідчити про те, що фірми адаптувалися до попереднього сплеску регуляторних та кібербезпекових проблем, що призвело до більш керованого середовища ризиків.

2020-2024 роки: спостерігається ще одне помітне збільшення кризових точок, особливо у 2020Q2 і 2020Q3. Ці сплески відповідають початку та прогресуванню пандемії COVID-19, яка, ймовірно, призвела до значних збоїв у роботі та вимагала швидкої адаптації до безпрецедентної глобальної кризи.

Потенційні пояснення для піків

Пік 2 кварталу 2017 року:

Регуляторні зміни: запровадження суворої нормативної бази могло призвести до необхідності перегляду відповідності, що призвело до операційних криз.

Інциденти з кібербезпекою: у цей період могли статися часті серйозні порушення кібербезпеки, які вплинули на кілька компаній і загострили кризові точки.

Волатильність ринку: коливання економічних умов могли створити нестабільне ринкове середовище, що призвело до фінансових труднощів у всьому секторі.

Пікові показники 2020 року (Q2 і Q3):

Пандемія COVID-19. Початок пандемії спричинив масові збої, вплинувши на ланцюги поставок, динаміку робочої сили та прискоривши цифрову трансформацію, тим самим посиливши кризові точки.

Економічна невизначеність. Економічна невизначеність, спричинена пандемією, суттєво вплинула на доходи та фінансову стабільність.

Перехід на віддалену роботу: швидкий перехід до налаштувань віддаленої роботи виявив технологічну вразливість і операційну неефективність, що спричинило кризи.

Пік 1 кварталу 2021 року:

Наслідки пандемії: тривалі наслідки пандемії, ймовірно, призвели до тривалих перебоїв у роботі та проблем адаптації.

Проблеми з ланцюгом постачань. Постійні збої в ланцюзі постачання, які спостерігалися, продовжували впливати на виробництво та розподіл, що призводило до криз.

Загальні спостереження

Дані показують, що технологічні компанії протягом багатьох років зазнавали посиленого контролю, привертаючи більше уваги регуляторів, ЗМІ та громадськості, що, ймовірно, сприяло зростанню кількості повідомлень про кризи. Швидкий технологічний прогрес часто супроводжує проблеми зростання, такі як вразливість системи безпеки та проблеми впровадження. Крім того, динаміка ринку, включаючи економічні буми та спади, безпосередньо впливає на стабільність технологічних компаній.

2.3. Очистка даних від впливу незалежних ринкових механізмів

Було проведено аналіз даних компаній відносно індексу S&P500, інтервали значень цих змінних дають уявлення про ефективність компанії та її сприйнятливість до зовнішніх ринкових сил: R-квадрат і скоригований R-квадрат: ці значення вказують на те, яку частину варіації залежної змінної можна пояснити незалежними змінними. Більш високі значення вказують на сильніші внутрішні фактори, що впливають на ефективність, нижчі значення можуть вказувати на значний вплив зовнішніх факторів. R-квадрат: Коефіцієнт детермінації в діапазоні від 0 до 1, який вимірює частку дисперсії залежної змінної, яку можна передбачити на основі незалежних змінних. Вищий R-квадрат вказує на кращу відповідність моделі даним.

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Де SS_{res} - сума квадратів залишків

SS_{tot} - загальна сума квадратів

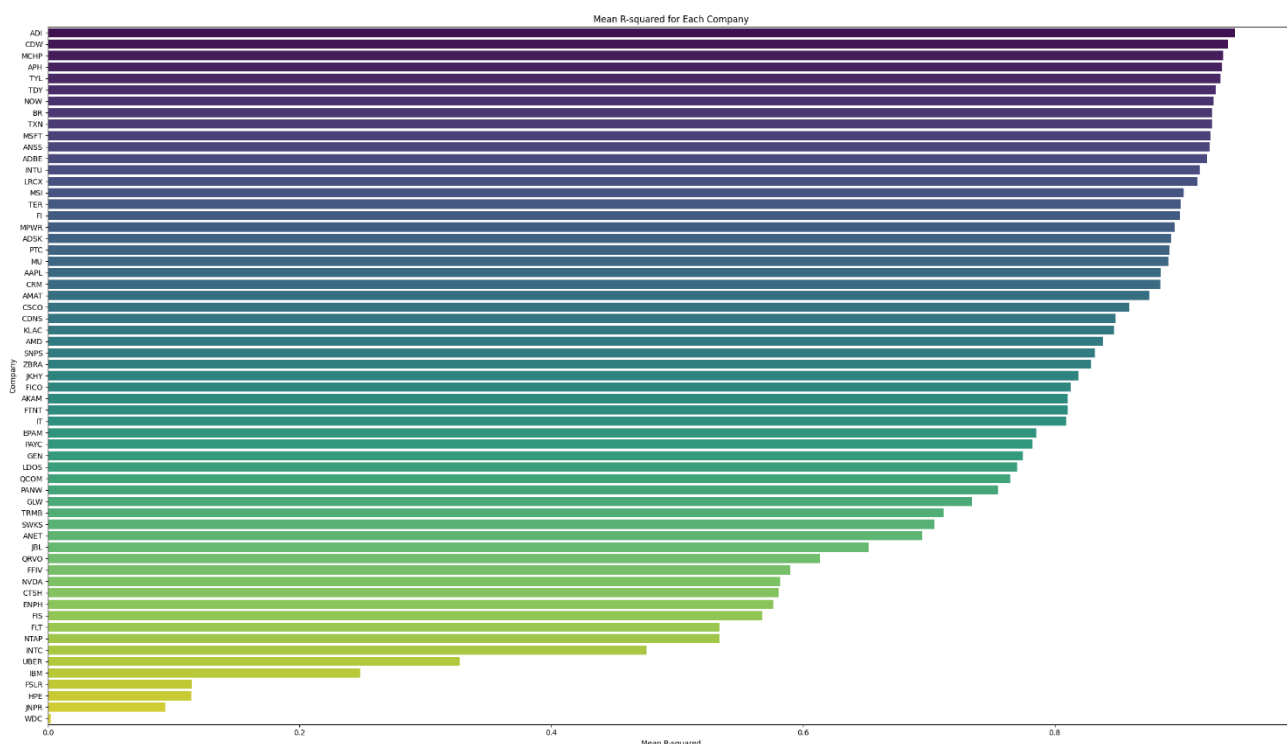


Рис. 2.3. Коефіцієнти детермінації індексу S&P500 та технологічних компаній

Джерело: додаток В

Скоригований R-квадрат враховує кількість незалежних змінних у моделі, штрафуючи за надмірну складність. Це більш консервативний показник відповідності моделі порівняно з R-квадратом.[14]

$$\text{Adjusted } R^2 = 1 - \frac{(1 - R^2)(n - 1)}{n - k - 1}$$

Де n - кількість спостережень

k - кількість незалежних змінних

F-статистика та Prob (F-статистика): F-статистика перевіряє загальну значущість регресійної моделі. Значна F-статистика (низька ймовірність) свідчить про те, що незалежні змінні сукупно впливають на залежну змінну, вказуючи на надійну внутрішню динаміку або вплив зовнішнього ринку. F-статистика: співвідношення, яке перевіряє загальну значущість регресійної моделі. Він порівнює пояснену дисперсію моделі з нез'ясованою дисперсією та оцінює, чи є модель у цілому статистично значущою. Імовірність (F-статистика): ймовірність, яка вказує на теб чи спостережувана F-статистика може мати місце за нульовою гіпотезою (немає істотного зв'язку між залежною та незалежною змінними).[15]

$$F = \frac{\frac{SSR}{k}}{\frac{SSE}{n-k-1}}$$

Де SSR – це сума квадратів, пояснена регресією.

SSE – це сума квадратів залишків.

k – кількість незалежних змінних.

n – кількість спостережень.

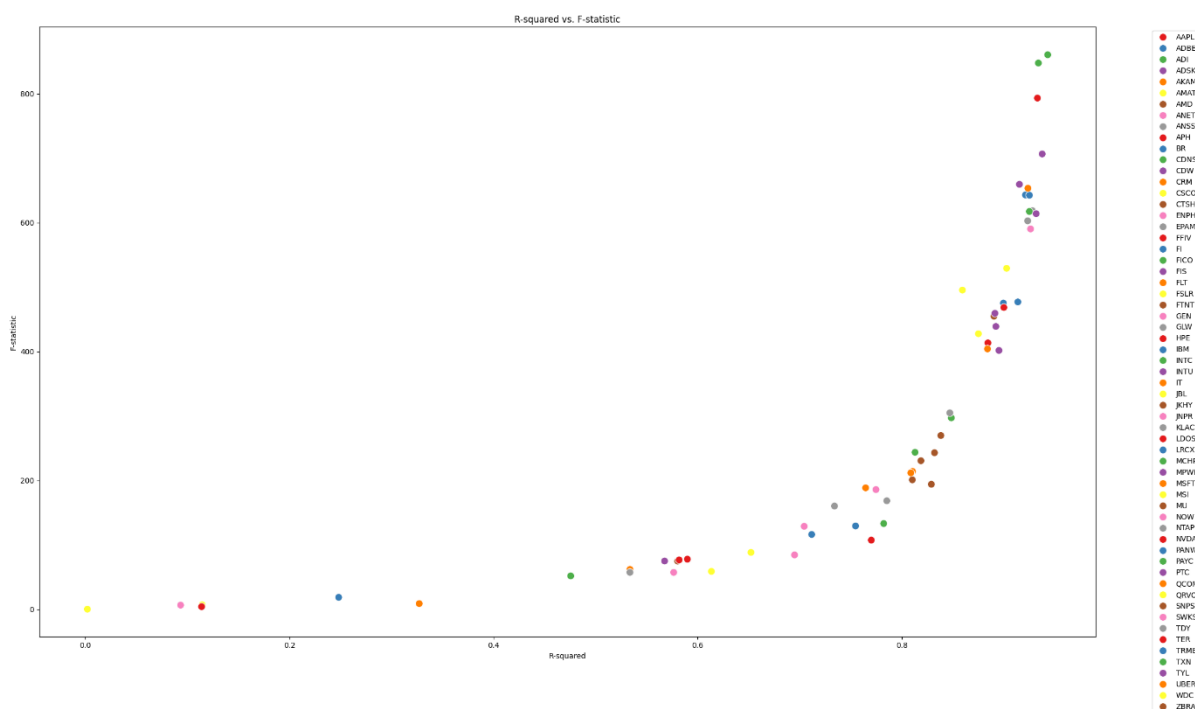


Рис. 2.4. Залежність коефіцієнтів детермінації індексу S&P500 та технологічних компаній від F-статистики Джерело: додаток В

Параметри (Params) і t-статистика (t): ці показники оцінюють значущість окремих незалежних змінних. Високі t-значення (і низькі P-значення) вказують на те, що конкретні фактори (наприклад, коефіцієнти прибутковості, ринкова волатильність) значно впливають на результати діяльності компанії, потенційно відображаючи тиск зовнішнього ринку. Параметри: кількість параметрів, оцінених у регресійній моделі, які відповідають коефіцієнтам незалежних змінних і члену перехоплення в рівнянні регресії. t: t-статистика вимірює значущість кожної незалежної змінної в регресійній моделі. Він розраховується як розрахунковий коефіцієнт, поділений на стандартну помилку. [16]

$$t = \frac{\hat{\beta}_j}{SE(\hat{\beta}_j)}$$

Де $\hat{\beta}_j$ - розрахунковий коефіцієнт для незалежної змінної j

$SE(\hat{\beta}_j)$ – коефіцієнт стандартної помилки $\hat{\beta}_j$

Довірчі інтервали: довірчі інтервали навколо оцінок параметрів забезпечують діапазони, в межах яких справжнє значення параметрів, ймовірно, потраплятиме.

Вузькі інтервали пропонують точні оцінки впливу зовнішніх ринкових сил на діяльність компанії. Інтервал 95% (нижній): нижня межа 95% довірчого інтервалу для розрахункового коефіцієнта незалежної змінної. Він надає діапазон, у межах якого вірогідно потрапить справжній параметр населення. Інтервал 95% (верхній): верхня межа 95% довірчого інтервалу для розрахункового коефіцієнта незалежної змінної. Разом із нижньою межею він визначає діапазон вірогідних значень параметра.

$$[\hat{\beta}_j - t_{\alpha/2} \cdot SE(\hat{\beta}_j), \hat{\beta}_j + t_{\alpha/2} \cdot SE(\hat{\beta}_j)]$$

Де $\hat{\beta}_j$ - представляє розрахунковий коефіцієнт для незалежної змінної j .

$SE(\hat{\beta}_j)$ - позначає стандартну похибку коефіцієнта $\hat{\beta}_j$.

$t_{\alpha/2}$ — це критичне значення t-розподілу для даного рівня довірчої вірогідності α , зазвичай $t_{0,025}$ для рівня довірчої вірогідності 95%.

Тести Omnibus і Jarque-Bera: Ці тести оцінюють загальну відповідність моделі та припущення щодо розподілу залишків. Відхилення від нормального (на що вказують значні тести Omnibus і Jarque-Bera) можуть свідчити про невраховані зовнішні ринкові сили, що впливають на надійність моделі. Омнібус: тестова статистика, яка оцінює загальну значущість коефіцієнтів моделі. Він перевіряє гіпотезу про те, що всі коефіцієнти регресії одночасно дорівнюють нулю. Ймовірність (Omnibus): ймовірність, пов'язана зі статистикою тесту Omnibus, яка вказує на ймовірність того, що спостережувана статистика може мати місце за нульовою гіпотезою (тобто відсутність значущих коефіцієнтів у моделі). Жарке-Бера (JB): Тест на нормальність залишків у регресійній моделі. Він оцінює, чи мають залишки асиметрію та ексцес, які відповідають нормальному розподілу. Ймовірність (JB): ймовірність, пов'язана зі статистикою тесту Жарке-Бера, яка вказує на ймовірність того, що спостережуване відхилення від нормального значення залишків може статися за нульової гіпотези (тобто залишки розподілені нормально).[17]

Асиметрія та ексцес: ці показники описують форму та розподіл залишкових помилок. Високі значення асиметрії або ексцесу можуть вказувати на нелінійні залежності або викиди, викликані зовнішніми ринковими шоками. Асиметрія: міра асиметрії розподілу залишків навколо середнього значення. Позитивна асиметрія

вказує на довший правий хвіст, тоді як негативна асиметрія вказує на довший лівий хвіст. Ексцес: міра «хвостатості» розподілу залишків. Вищий ексцес передбачає більш екстремальні значення в хвостах порівняно з нормальним розподілом.

AIC (інформаційний критерій Акаїке): міра відносної якості статистичної моделі для заданого набору даних. Він збалансовує придатність і складність моделі, надаючи можливість порівнювати різні моделі.[18]

$$AIC = k - 2 \ln(\hat{L})$$

Де \hat{L} - максимальне значення функції правдоподібності для моделі.

k – кількість параметрів у моделі

BIC (байєсівський інформаційний критерій): Подібно до AIC, BIC є ще одним критерієм вибору моделі. Він карає моделі з більшою кількістю параметрів суворіше, ніж AIC, потенційно віддаючи перевагу простішим моделям.[19]

$$BIC = k * \ln(n) - 2 \ln(\hat{L})$$

Де \hat{L} - максимальне значення функції правдоподібності для моделі.

k – кількість параметрів у моделі

n – кількість спостережень

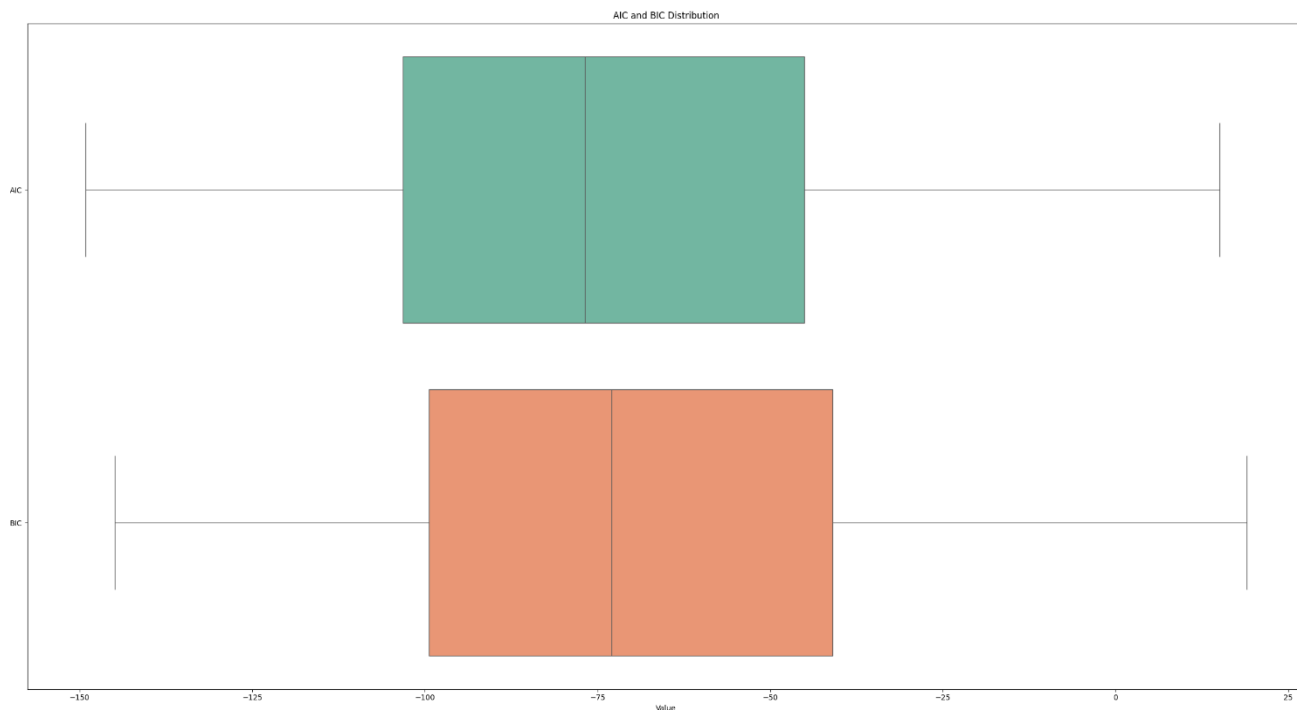


Рис. 2.5. BOX-plot для показників AIC і BIC Джерело: додаток В

Номер умови: вказує на мультиколінеарність між незалежними змінними. Високе число умови свідчить про те, що надійність моделі в ізоляції впливу кожної незалежної змінної може бути порушена, потенційно через корельовані зовнішні ринкові фактори, які впливають на декілька змінних одночасно. Номер умови: діагностична міра, яка оцінює мультиколінеарність між незалежними змінними в регресійній моделі. Високе число умови свідчить про потенційні проблеми мультиколінеарності, коли незалежні змінні сильно корельовані.

Target - ціна акцій у цьому дослідженні, була нормалізована за допомогою мінімально-максимальної техніки нормалізації. Цей метод масштабує цільову змінну до фіксованого діапазону (зазвичай від 0 до 1), гарантуючи, що всі точки даних пропорційно порівнюються між різними компаніями та періодами часу.

Функції були побудовані для кожної компанії на основі їхніх відповідних даних часових рядів. Ці функції інкапсулювали зв'язок між часом і нормалізованими курсами акцій, фіксуючи конкретні тенденції та моделі, унікальні для фінансових показників кожної компанії протягом аналізованого періоду.

Ці окремі функції були використані для побудови середньої гладкої функції залежності target від часу. Ця сукупна функція надала зведене уявлення про поведінку середньої ціни акцій у всіх досліджуваних компаніях, демонструючи ширші ринкові тенденції та зміни в часі.

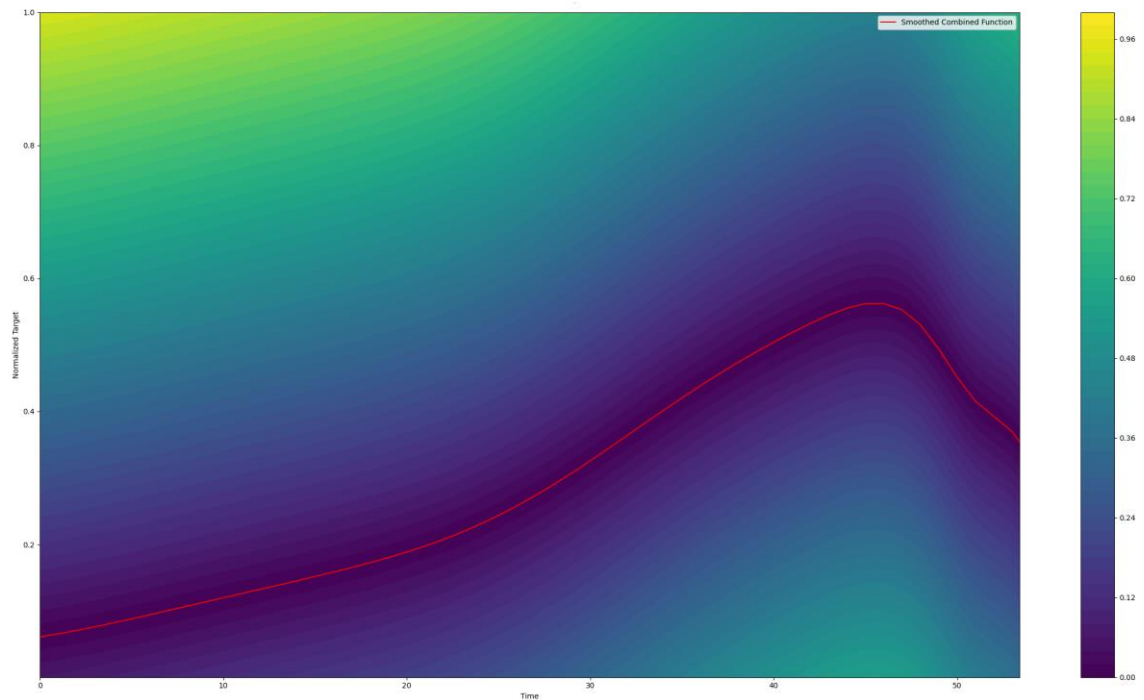


Рис. 2.6. Теплова карта нормалізованих цін акцій компаній Джерело: додаток Д

Теплова карта даних компаній була створена у зв'язку з гладкою середньою функцією цільового часу. Ця візуалізація методично ілюструє, як фінансові дані кожної компанії відхилялися від агрегованої тенденції, представлені функцією середнього. Завдяки відображенню цих відхилень теплова карта дозволила отримати всебічне розуміння ринкової ситуації

РОЗДІЛ III ЗАСТОСУВАННЯ РЕГРЕСІЙНОГО АНАЛІЗУ ДЛЯ ПОШУКУ ОПТИМАЛЬНОГО ВАРІАНТУ РОЗПОДІЛУ ОБМЕЖЕНИХ РЕСУРСІВ

3.1. Побудова варіативних регресій та їх порівняння

У цьому розділі ми оцінюємо різні моделі для прогнозування розподілу фінансових ресурсів провідних технологічних компаній. Розглянуті моделі включають сезонну авторегресійну інтегровану ковзну середню (SARIMA), дисперсійний аналіз (ANOVA), k-найближчих сусідів (KNN), дерева рішень C5.0, процеси Гауса [22] та лінійну регресію. Ці моделі оцінювали за допомогою метрики середньої абсолютної відсоткової помилки (MAPE) і статистики R-квадрат (R^2). Найефективніша модель, лінійна регресія, була обрана для подальшого аналізу через її чудову продуктивність і потенціал для оптимізації в майбутньому.

Метрики оцінки:

Середня абсолютна похибка (MAE)

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

MAE вимірює середню величину помилок у наборі прогнозів.

Це середнє значення абсолютних різниць між прогнозованими значеннями \hat{y}_i і справжніми значеннями y_i .

Інтерпретація: нижчий MAE вказує на кращу відповідність моделі даним.

Середня квадратична помилка (MSE)

$$MAE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

MSE вимірює середнє значення квадратів помилок.

Це середнє значення квадратів різниць між прогнозованими та справжніми значеннями.

Зведення помилок у квадрат карає більші помилки більше, ніж менші.

Інтерпретація: нижчий MSE вказує на кращу продуктивність моделі.

Середньоквадратична помилка (RMSE)

$$RMSE = \sqrt{MSE}$$

RMSE – це квадратний корінь із середньої квадратичної помилки.

Він має ті самі одиниці, що й цільова змінна, що полегшує його інтерпретацію.

Інтерпретація: нижча RMSE вказує на кращу відповідність моделі даним. [23]

R-квадрат

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2}$$

R^2 вимірює частку дисперсії залежної змінної, яку можна передбачити на основі незалежних змінних.

Порівнює залишкову суму квадратів $\sum_{i=1}^n (y_i - \hat{y}_i)^2$ із загальною сумою квадратів $\sum_{i=1}^n (y_i - \bar{y}_i)^2$ — середнє значення істинних значень.

Інтерпретація: значення R^2 коливаються від 0 до 1, при цьому вищі значення вказують на кращу відповідність. R^2 , що дорівнює 1, означає, що модель пояснює всю дисперсію цільової змінної.

Середня абсолютна відсоткова похибка (MAPE)

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\%$$

MAPE вимірює точність прогнозів у відсотках.

Це середнє значення абсолютних процентних похибок між прогнозованими та справжніми значеннями.

Інтерпретація: нижчий MAPE означає кращу модель. Це особливо корисно, коли цільові значення змінної ненульові та їх можна порівняти в різних масштабах.

Середня абсолютна похибка

$$\text{Median Absolute Error} = \text{median}(|y_i - \hat{y}_i|)$$

Середня абсолютна похибка – це медіана абсолютних різниць між прогнозованими та справжніми значеннями.

Він забезпечує надійний показник центральної тенденції помилок передбачення, менш чутливий до викидів, ніж MAE.

Інтерпретація: менша середня абсолютна похибка вказує на кращу модель.

Квантильна втрата (втрата пінболу)

$$\text{Quantile Loss} = \sum_{i=1}^n \max(q(y_i - \hat{y}_i), (1 - q)(\hat{y}_i - y_i))$$

Втрата квантиля використовується під час прогнозування конкретного квантиля цільової змінної.

Він карає за переоцінку та заниження на основі вказаного квантиля q .

Інтерпретація: менша втрата квантиля вказує на кращу відповідність зазначеному квантилю. Наприклад, квантильна втрата 0,1 для 10-го процентяля вказує на точне передбачення для нижнього кінця розподілу.

Щоб порівняти продуктивність моделей, ми використали два ключові показники: MAPE та R^2 .

Аналіз результатів:

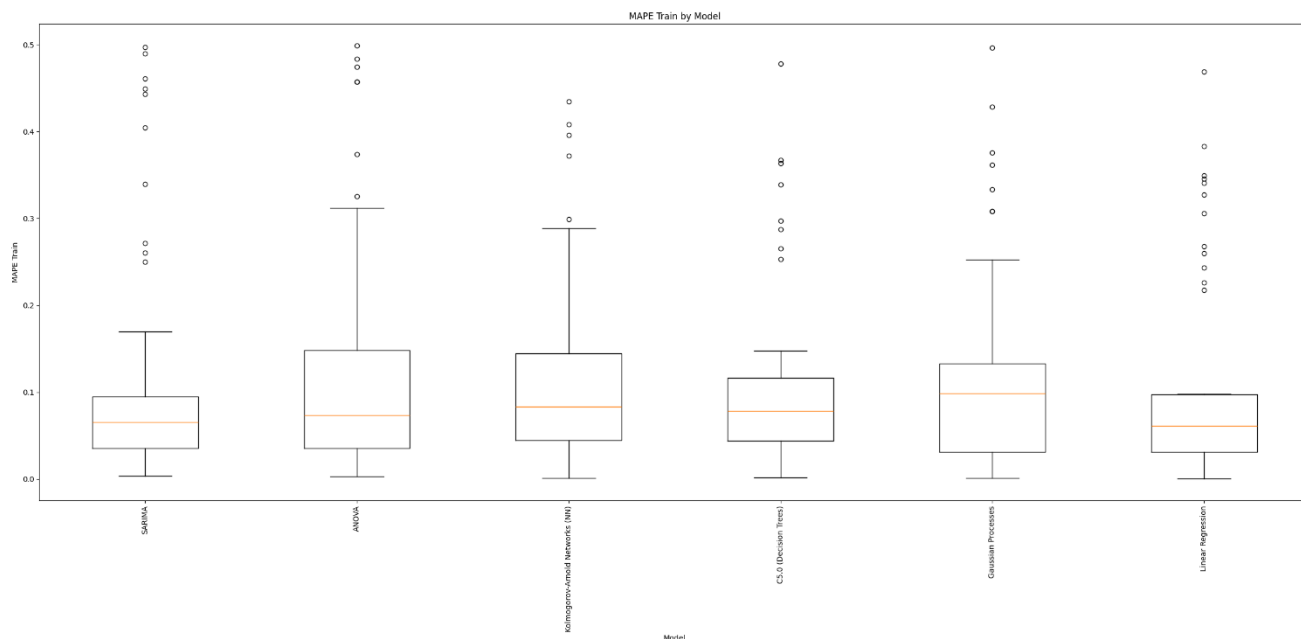


Рис. 3.1. Box-plot результатів моделей на тренувальній множині (MAPE)

Джерело: додаток А

SARIMA: Модель SARIMA продемонструвала стабільну продуктивність із відносно низькими значеннями MAPE, що свідчить про хорошу точність прогнозування.

ANOVA: модель ANOVA+SARIMA продемонструвала ширший розкид значень MAPE, що свідчить про варіабельність її продуктивності.

KNN: Модель KNN також показала високу варіабельність, з багатьма викидами, що вказують на погані прогнози в певних випадках.

C5.0 Древа рішень: ця модель мала продуктивність, подібну до KNN, із широким діапазоном значень MAPE та кількома викидами.

Процеси Гауса: ця модель показала помірну точність з деякими викидами, що вказує на випадкову низьку продуктивність.

Лінійна регресія: Лінійна регресія постійно досягала найнижчих значень MAPE, що вказує на вищу точність прогнозування порівняно з іншими моделями.

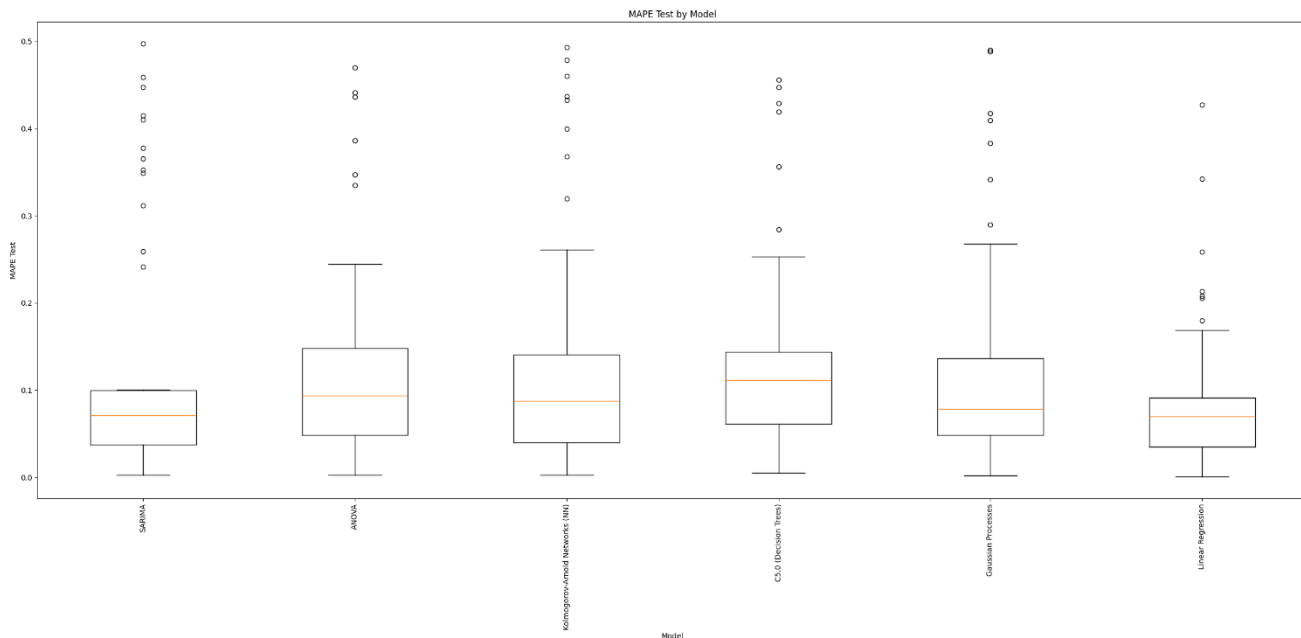


Рис. 3.2. Box-plot результатів моделей на тестувальній множині (MAPE)

Джерело: додаток А

Результати навчального набору показують подібні тенденції до тестового набору. Моделі SARIMA та лінійної регресії продемонстрували нижчі значення MAPE, що свідчить про хорошу підгонку під час тренування.

ANOVA+SARIMA, KNN і C5.0 знову показали ширші розкиди, що свідчить про те, що ці моделі можуть бути певною мірою переобладнаними.

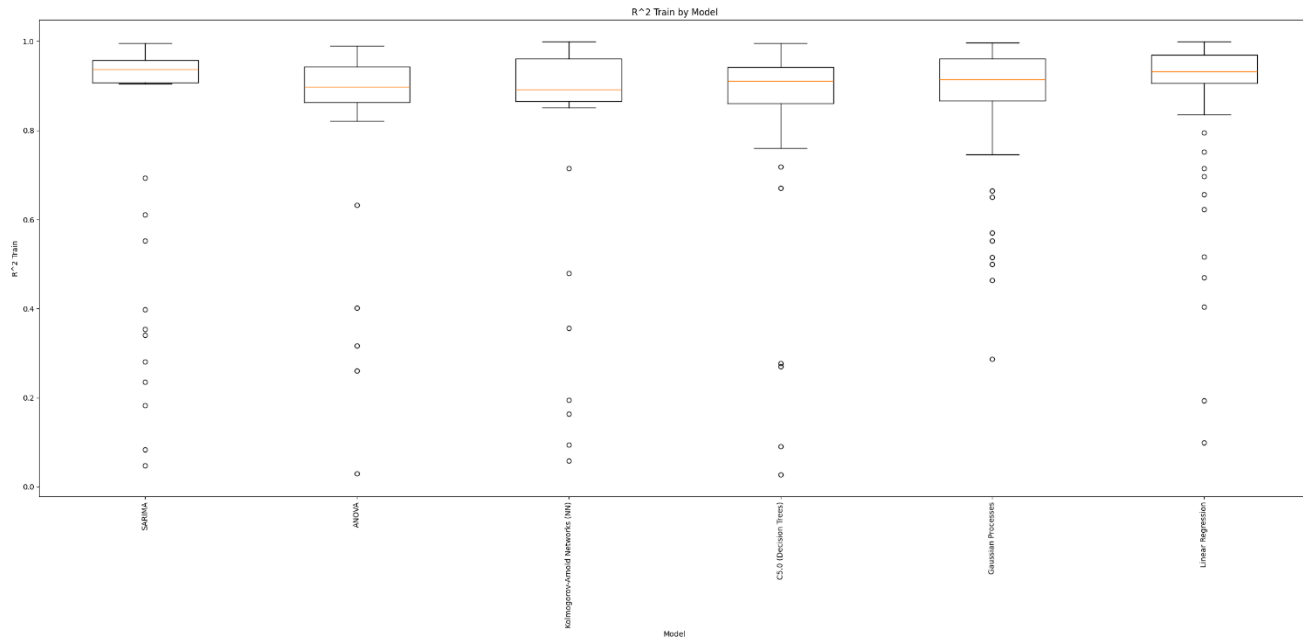


Рис. 3.3. Box-plot результатів моделей на тренувальній множині (R^2) Джерело: додаток А

SARIMA та лінійна регресія: обидві моделі демонструють високі значення R^2 , що вказує на те, що вони охоплюють значну частину дисперсії в даних навчання.

ANOVA+SARIMA, KNN і C5.0: ці моделі демонструють більшу варіабельність значень R^2 , з численними викидами, які показують, що в деяких випадках вони погано відповідають навчальним даним.

Процеси Гаусса: ця модель також показує хороші значення R^2 , але з деякими викидами, які іноді вказують на погане відповідність.

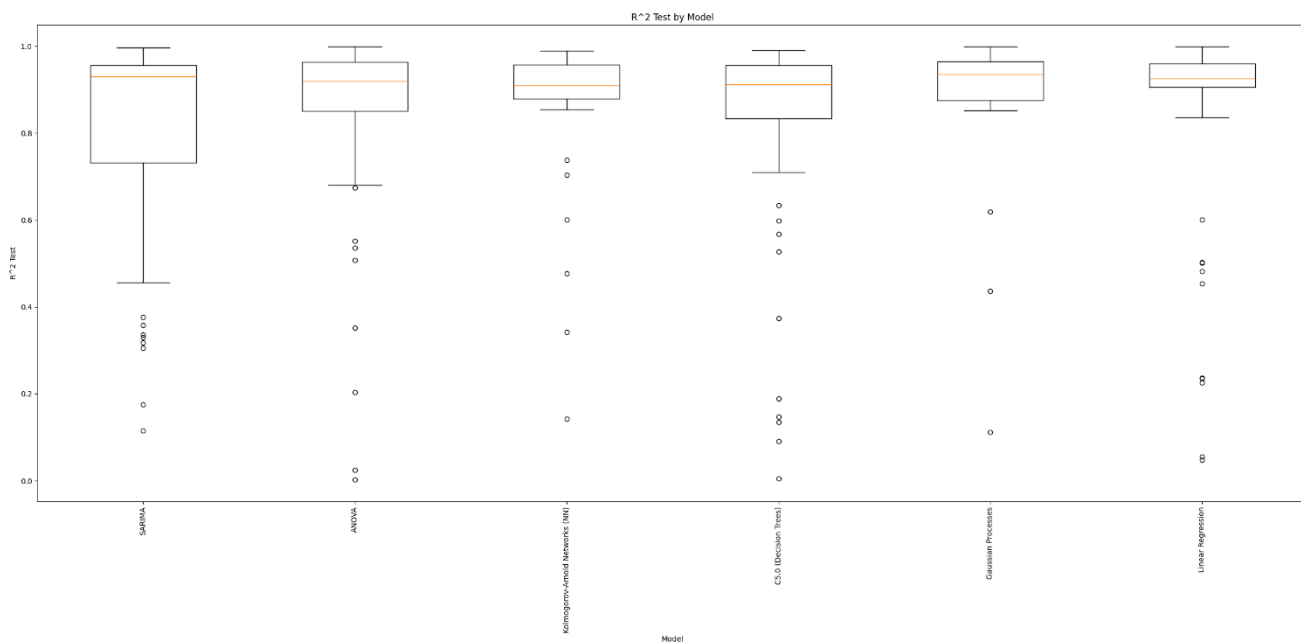


Рис. 3.4. Box-plot результатів моделей на тестувальній множині (R^2) Джерело: додаток А

SARIMA: модель SARIMA продемонструвала високі значення R^2 , що вказує на хорошу відповідність і точні прогнози на основі даних тестування.

ANOVA: модель ANOVA+SARIMA показала широкий діапазон значень R^2 , що відображає непостійну продуктивність і недостатню надійність у певних сценаріях.

KNN: модель KNN показала значну варіабельність значень R^2 із численними викидами, що вказує на те, що їй важко узагальнити дані тесту.

C5.0 Дерева рішень: ця модель також мала широкий розкид значень R^2 із деякими випадками дуже низької точності прогнозування.

Процеси Гауса: ця модель показала відносно високі значення R^2 , хоча деякі викиди свідчать про випадкову низьку продуктивність.

Лінійна регресія: Лінійна регресія постійно досягала найвищих значень R^2 , що вказує на те, що вона найкраще вловлює відхилення в даних і забезпечує точні прогнози. Було побудовано ряд рівнянь.

3.2. Створення прогнозу на основі побудованої регресії

У сфері розподілу фінансових ресурсів між технологічними компаніями неможливо переоцінити стратегічну важливість прогнозування. Прогнозування дозволяє приймати обґрунтовані рішення, надаючи розуміння майбутніх фінансових тенденцій, що має вирішальне значення для планування та стратегії розподілу ресурсів. Серед різноманітних методів прогнозного моделювання лінійна регресія виділяється своєю надійністю при створенні прогнозних моделей на основі історичних даних.

Лінійна регресія особливо підходить для визначення лінійних зв'язків між залежними та незалежними змінними, що робить її методом вибору в багатьох сценаріях фінансового прогнозування. Рішення використовувати лінійну регресію було підкріплено її ефективністю, продемонстрованою на попередніх етапах дослідження. Цей методологічний вибір узгоджується з метою прогнозування майбутніх фінансових тенденцій на основі минулих показників і ринкових умов.

Для впровадження лінійної регресії використовувався модуль `LinearRegression` з бібліотеки `scikit-learn`, популярний вибір в екосистемі Python завдяки надійній підтримці завдань регресії та бездоганній інтеграції в конвеєри обробки даних. Цей вибір відображає практичний підхід до розробки та розгортання моделі, використовуючи можливості бібліотеки для ефективної обробки завдань регресії. [20]

Використання коефіцієнтів регресії безпосередньо для важливості предиктора може бути проблематичним у певних контекстах. Ось кілька причин, чому коефіцієнти регресії не завжди можуть бути придатними для визначення важливості предиктора: Проблеми масштабування: Коефіцієнти регресії залежать від масштабу предикторів. Якщо предиктори знаходяться на різних шкалах, коефіцієнти можуть вводити в оману з точки зору їхньої абсолютної величини. Функції з більшими числовими діапазонами можуть мати більші коефіцієнти, навіть якщо їх фактична передбачувана сила не є пропорційно більшою. Мультиколінеарність: коли предиктори сильно корельовані (мультиколінеарні), коефіцієнти регресії можуть стати нестабільними або навіть змінювати знаки. Це ускладнює інтерпретацію коефіцієнтів як показників важливості для окремих предикторів. Нелінійні зв'язки:

Лінійна регресія передбачає наявність лінійного зв'язку між предикторами та цільовою змінною. Якщо залежність є нелінійною, коефіцієнт може неточно відображати важливість предиктора. Навпаки, такі методи, як Random Forest або Gradient Boosting Machines, можуть ефективніше фіксувати нелінійні зв'язки. Припущення моделі: Лінійна регресія передбачає певні припущення моделі (наприклад, лінійність, гомоскедастичність, незалежність помилок), які можуть не виконуватися на практиці. Порушення цих припущень може вплинути на надійність коефіцієнтів регресії як показників важливості предиктора. Складність моделі: у більш складних моделях, де важливі взаємодії та нелінійності (наприклад, у випадкових лісах або нейронних мережах), важливість предикторів часто виходить за рамки простих лінійних залежностей, які визначаються коефіцієнтами регресії.

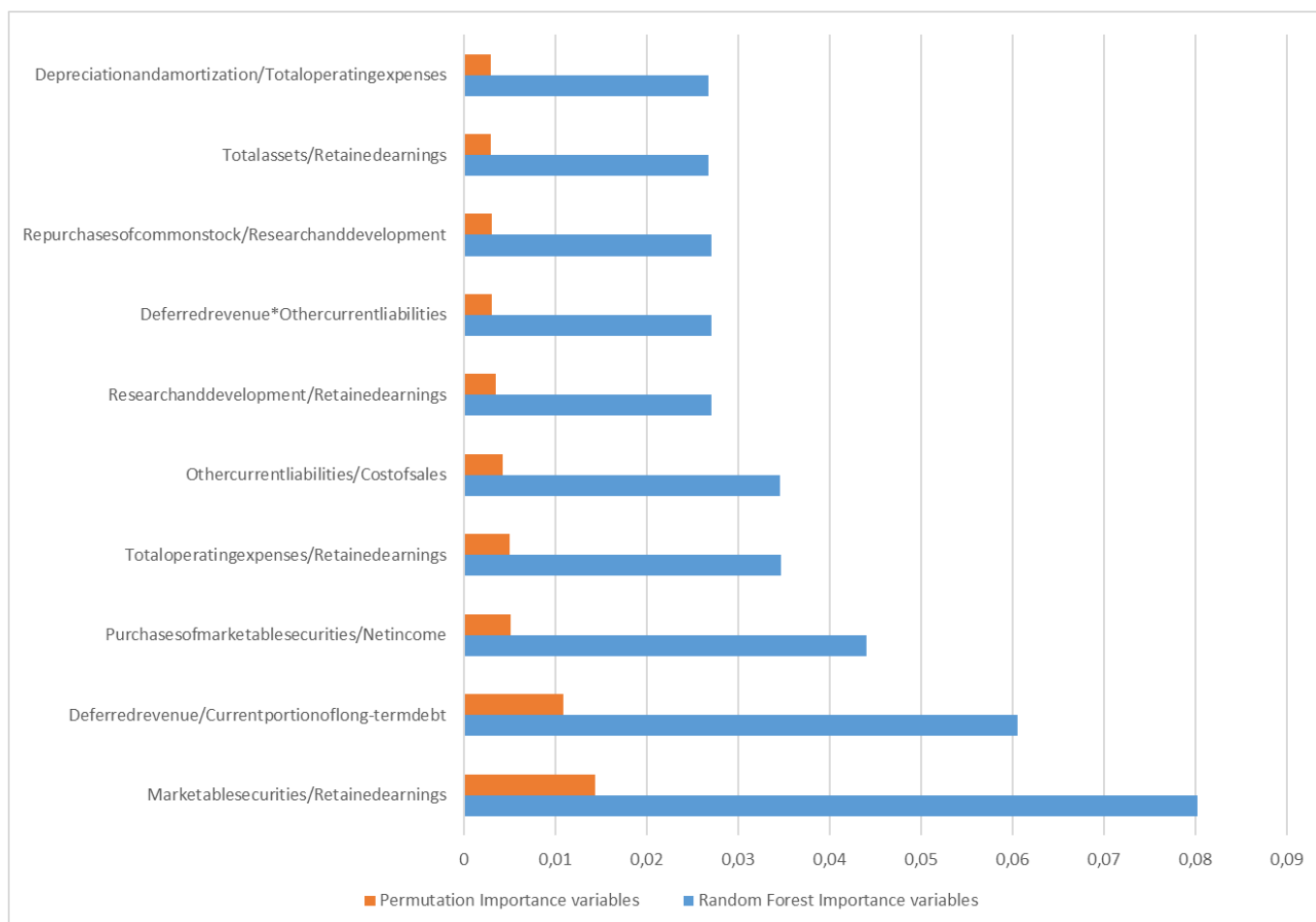


Рис. 3.5. Графік важливості змінних та їх впливу на результати регресії

Джерело: додаток 3

Важливість випадкового лісу, також відома як важливість функції з випадкових лісів, є показником того, наскільки кожна функція сприяє зменшенню невизначеності (наприклад, дисперсії в регресії) у моделі випадкового лісу.

Обчислення на основі дерева: у кожному дереві рішення у випадковому лісі вузли розділені на основі різних характеристик, щоб зменшити невизначеність. Важливість Джині (або середнє зменшення невизначеності): для кожної функції ми накопичуємо загальне зменшення невизначеності, викликане всіма розподілами, де використовується ця функція. Це робиться на всіх деревах у лісі. Усереднення: важливість кожної функції усереднюється по всіх деревах у лісі, щоб отримати остаточну оцінку важливості.

Використання: Вибір функцій: визначення найважливіших функцій, які мають найбільший вплив на прогнози моделі. Інтерпретація моделі: розуміння того, на які функції модель найбільше покладається для своїх прогнозів.

Важливість перестановки — це модельно-агностичний метод визначення важливості ознак. Він оцінює важливість, вимірюючи зміну продуктивності моделі, коли значення однієї функції випадково перемішуються. Базова продуктивність: спочатку продуктивність моделі вимірюється за допомогою метрики оцінки (наприклад, R-квадрат для регресії, точність для класифікації).

Перемішування функцій: для кожної функції значення перемішуються випадковим чином, порушуючи зв'язок між функцією та цільовою змінною. Зниження продуктивності: продуктивність моделі вимірюється знову за допомогою того самого показника оцінки. Зниження продуктивності вказує на важливість функції перемішування. Усереднення: процес повторюється кілька разів (`n_repeats`), щоб отримати стабільну оцінку важливості.

Використання: Оцінка впливу функції: визначення того, наскільки кожна функція впливає на прогнози моделі. Інтерпретація моделі: надання інформації про те, які функції є найважливішими для продуктивності моделі.

Відмінності між важливістю випадкового лісу та важливістю перестановки

Важливість випадкового лісу: специфічно для моделей на основі дерева та використовує внутрішню механіку роботи цих моделей.

Важливість перестановки: не залежить від моделі та може застосовуватися до будь-якої прогнозної моделі, а не лише до моделей на основі дерева. Важливість випадкового лісу: надає внутрішню перспективу на основі того, як часто функція використовується для поділу даних у деревах. Важливість перестановки: пропонує зовнішню перспективу, засновану на практичному впливі функції на продуктивність моделі, коли її зв'язок із цільовою метою порушено.

Додаткова інформація: використання обох методів разом може забезпечити більш повне розуміння важливості функції. Важливість випадкового лісу: дає уявлення про структуру моделі та внутрішню важливість функцій під час навчання моделі. Важливість перестановки: демонструє практичний вплив кожної функції на прогнози моделі, наголошуючи на реальних наслідках змін функції. Перевірка: важливість перестановки може служити перевіркою важливості функції, отриманої з моделі випадкового лісу, гарантуючи, що функції, які модель вважає важливими, справді впливають на її прогнозу ефективність.

Ключове покращення передбачуваної сили моделі включало розробку умов взаємодії. Терміни взаємодії – це нові змінні, створені шляхом множення або ділення існуючих змінних, спрямовані на охоплення більш складних зв'язків у даних. Наприклад, взаємодія між різними показниками фінансової ефективності та ринковими умовами може забезпечити більш глибоке розуміння того, як ці фактори разом впливають на динаміку розподілу ресурсів.

Оцінюючи продуктивність побудованої моделі, результатом стала середня абсолютна відсоткова помилка (MAPE) 9,74% для набору даних перевірки. Цей показник вказує на здатність моделі добре узагальнювати нові дані, припускаючи її надійність у прогнозуванні розподілу фінансових ресурсів протягом горизонту

прогнозу.

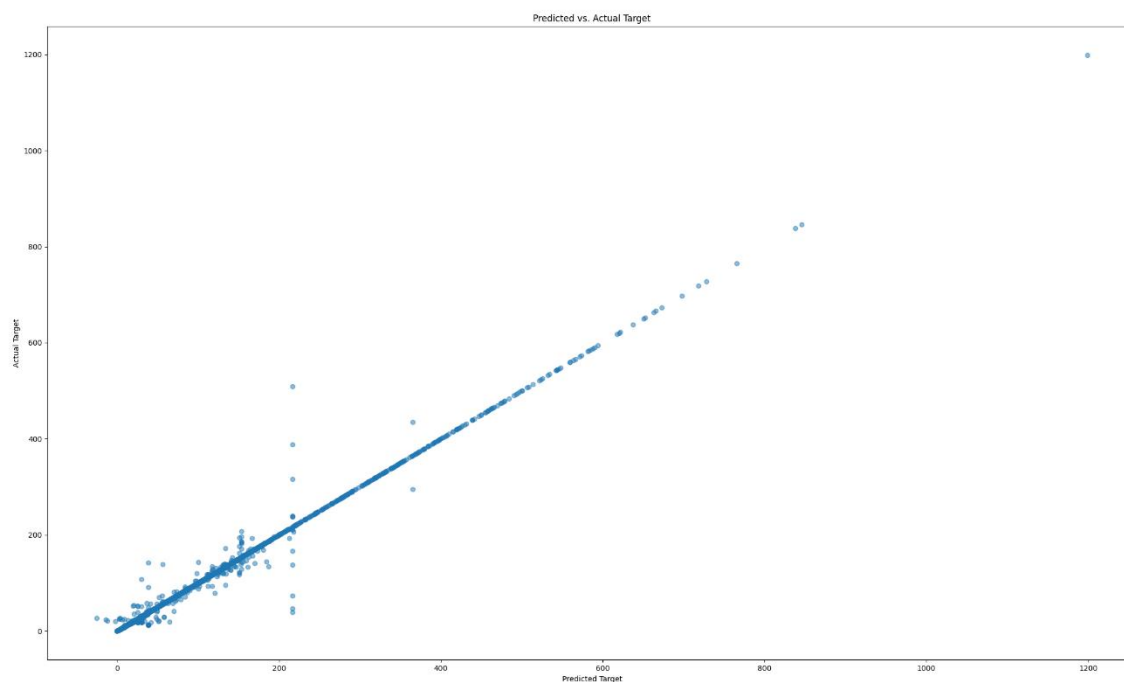


Рис. 3.6. Діаграма розкиду передбачених регресією значень Джерело: додаток Б

Використовуючи лінійну регресію, доповнену умовами взаємодії та перевірену за допомогою суворих статистичних заходів, таких як MAPE, сприяє вдосконаленню розуміння та застосуванню прогнозного моделювання в стратегічному фінансовому плануванні для технологічних компаній. Відомості, отримані в результаті цього підходу, дають змогу зацікавленим сторонам приймати обґрунтовані рішення щодо ефективного розподілу ресурсів в умовах динамічних ринкових умов і бізнес-ландшафту, що розвивається.

3.3. Знаходження оптимального розподілу фінансових ресурсів.

На попередньому етапі цього дослідження були розроблені функції регресії для провідних технологічних компаній, щоб зрозуміти зв'язок між розподілом фінансових ресурсів і ціною акцій. Ці функції послужили основою для оптимізації розподілу фінансових ресурсів.

Обмеження для змінних. Для побудови задачі оптимізації обмеження для змінних були розраховані на основі історичних даних. Розглядалися такі варіанти:

Середнє значення ± 3 стандартних відхилення

$$Limit = \frac{1}{n} \sum_{i=1}^n x_i \pm \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n}}$$

Цей підхід мав на меті охопити широкий діапазон можливих значень для забезпечення статистичної значущості. Однак отримані значення часто були надто нереалістичними та непрактичними через фінансові обмеження.

Останнє значення \pm середня зміна:

$$Limit = x_n \pm x_n \times \frac{1}{n-1} \sum_{i=1}^{n-1} x_{i+1} - x_i$$

Цей підхід забезпечив незначні зміни, тому не запропонував значущої оптимізації.

Останнє значення \pm подвійна зміна:

$$Limit = x_n \pm 2 \times x_n \times (x_n - x_{n-1})$$

Цей метод створив більш практичні та значні варіації для оптимізації.

Методи максимізації цільової функції. Для максимізації цільової функції було розглянуто декілька методів оптимізації:

Лінійне програмування (LP): незважаючи на свій потенціал, LP було відхилено через його низьку швидкість обчислення, яка є недостатньою для прийняття рішень у реальному часі на швидкозмінних ринках.

Диференціальна еволюція (DE): цей алгоритм глобальної оптимізації розглядався, але в кінцевому підсумку був відхилений через його тенденцію сходиться до локальних максимумів, що призводить до неоптимальних рішень. [24]

L-BFGS-B: цей алгоритм обмеженої пам'яті для обмеженої оптимізації було обрано через його баланс між ефективністю та здатністю вирішувати масштабні проблеми. Це забезпечило більш надійний шлях до оптимального розподілу фінансових ресурсів, гарантуючи, що рішення були практичними та значущими.

L-BFGS-B – алгоритм мінімізації, тому цільову функцію було помножено на -1

Алгоритм починається з початкової оцінки x_0 оптимального значення та уточнює її ітераційно через послідовність покращених оцінок x_1, x_2, \dots . Похідні $g_k = \nabla f(x_k)$ керують алгоритмом, вказуючи напрямок найкрутішого спуску та допомагаючи апроксимувати матрицю Гессе H_k .

L-BFGS, метод квазі-Ньютона, особливо відрізняється тим, як він обчислює напрямок $d_k = -H_k \times g_k$. Замість того, щоб безпосередньо формувати H_k , він використовує історію оновлень для формування цього вектора напрямку, зазвичай за допомогою методу, відомого як «двоциклова рекурсія».

Починаючи з x_k і $g_k = \nabla f(x_k)$ і зберігаючи останні оновлення $s_k = x_{k-1} - x_k$ і $y_k = x_{k-1} - x_k$, алгоритм обчислює $\rho_k = \frac{1}{y_k^T s_k}$ та ініціалізує H_k^0 як початкову оцінку зворотного Гессе.

Формула оновлення для H_{k+1} ітеративно уточнює H_k за допомогою:

$$H_{k+1} = (I - \rho_k s_k y_k^T) H_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T$$

Щоб обчислити напрямок спуску $z_k = -H_k g_k$, L-BFGS використовує рекурсивні алгоритми для q_k і z_k :

$$q = q_k$$

Для $i = k - 1, k - 2, \dots, k - m$:

$$\alpha_i = \rho_i s_i^T q$$

$$q = q - \alpha_i y_i$$

$$\gamma_k = \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}}$$

$$H_k^0 = \gamma_k I$$

$$z = H_k^0 q$$

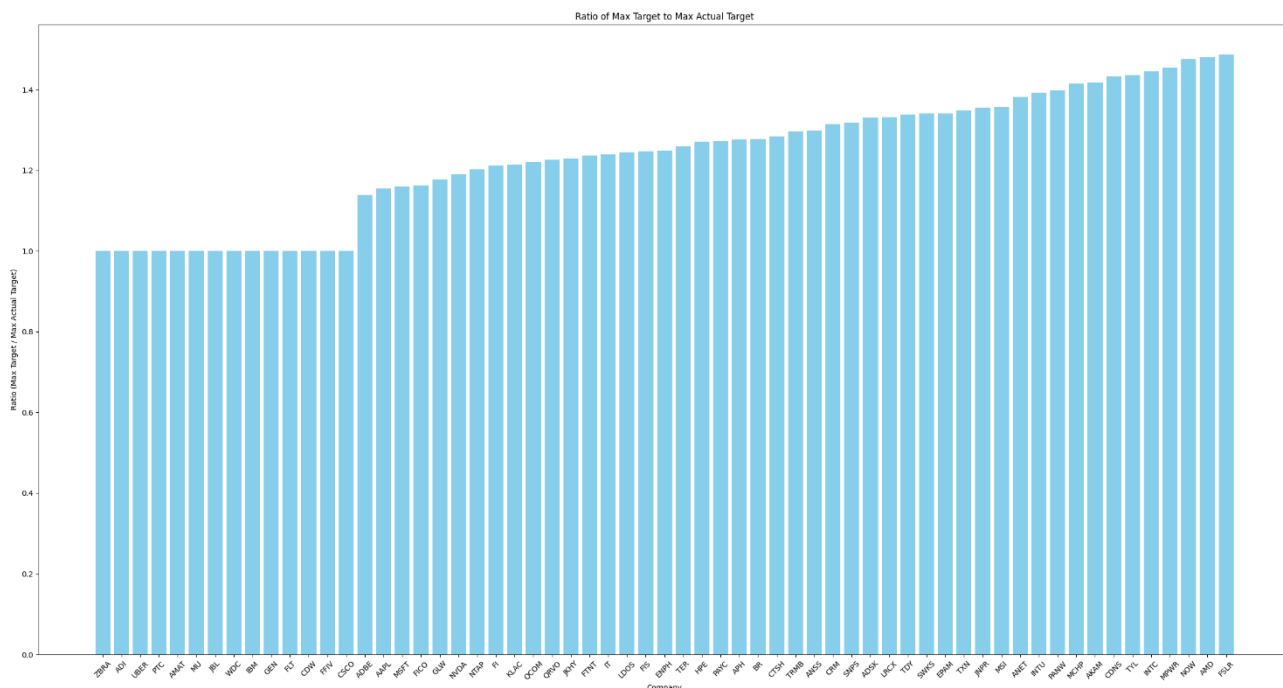
Для $i = k - m, k - m + 1, \dots, k - 1$:

$$\beta_i = \rho_i y_i^T z$$

$$z = z + s_i(\alpha_i - \beta_i)$$

$$z = -z$$

Це формулювання ефективно обчислює напрямок пошуку $z = -H_k g_k$ для мінімізації $f(x)$. Для задач максимізації використовується $-z$. H_k^0 зазвичай масштабується, щоб забезпечити добре масштабований напрямок пошуку, а рядковий пошук (наприклад, Wolfe або Armijo) забезпечує стабільність і конвергенцію. Хоча двоконтурна рекурсія є ефективною для зворотних обчислень Гессе, вона явно не формує H_k або H_k^{-1} і несумісна з обмеженнями, що не є прямокутними. Для оптимізації з обмеженнями такі методи, як низькорангові наближення Гессе, дозволяють використовувати L-BFGS у більш складних налаштуваннях.[21]



У лівій частині графіка є кластер компаній із співвідношенням 1.

Співвідношення 1 вказує на те, що прогноз моделі дорівнює максимальному історичному значенню. яке виникає, коли $R^2 < 0,9$. Це означає, що для цих компаній модель не могла покращити історичний максимум через низьку точність (що вказує на можливі обмеження моделі або недостатність даних).

Коефіцієнти, більші за 1, вказують на те, що модель передбачає вищу максимальну ціну акцій порівняно з історичними даними, що означає потенціал зростання, якщо розподіл фінансових ресурсів оптимізовано відповідно до моделі.

Помітні стрибки у співвідношенні вказують на компанії, у яких модель передбачає значно вищі максимальні ціни акцій порівняно з історичними даними. Це можуть бути ключові компанії, які слід досліджувати для потенційної оптимізації ресурсів та інвестиційних можливостей.

Компанії крайнього правого краю мають найвищі коефіцієнти, що вказує на те, що модель бачить значні можливості для покращення цін на їхні акції. Ці викиди можуть представляти особливий інтерес для подальшого фінансового аналізу.

ВИСНОВКИ

Проведене дослідження мало на меті змодельовати розподіл фінансових ресурсів у провідних технологічних компаніях, зосередившись на прогнозуванні цін на акції на основі різних фінансових показників і витрат. Під час цього дослідження було розглянуто кілька ключових компонентів, включаючи розробку прогнозних моделей, формулювання рівняння, що пов'язує фінансові результати з курсами акцій, і оптимізації результатів курсу акцій.

Прогнозні моделі для цін на акції: Дослідження успішно розробило моделі для прогнозування цін на акції для провідних технологічних компаній. Враховуючи фінансові результати та витрати, ці моделі продемонстрували значну прогностичну силу, підкреслюючи зв'язок між розподілом фінансових ресурсів і ринковою оцінкою.

Формулювання прогнозних рівнянь: основним досягненням стала побудова формули, яка вводить фінансові результати та вартість витрат для виведення цін на акції. Ця формула ґрунтується на надійних статистичних методах і відображає складну взаємодію між різними фінансовими змінними та показниками ринку.

Оптимізація: за допомогою статистичного аналізу було визначено обмеження для формування задачі оптимізації. Вирішення цієї задачі оптимізації дало стратегії максимізації можливих цін на акції. Підхід забезпечив систематичний метод визначення оптимального розподілу фінансових ресурсів, збалансувавши інвестиції між різними критичними сферами, такими як дослідження та розробки, маркетинг та операції.

Результати цього дослідження мають кілька важливих наслідків для технологічних компаній:

Стратегічне фінансове планування: розроблені прогнозні моделі та система оптимізації можуть допомогти технологічним компаніям приймати більш обґрунтовані рішення щодо розподілу фінансових ресурсів. Розуміючи, як різні стратегії розподілу впливають на ціни акцій, компанії можуть визначити пріоритетність інвестицій, які з більшою ймовірністю підвищать ринкову оцінку.

Стратегії збалансованого інвестування: підхід лінійного програмування підкреслює важливість збалансованого інвестування в різних функціональних сферах. Надмірне інвестування в одну сферу за рахунок інших може бути згубним. Натомість комплексна стратегія розподілу, яка підтримує інновації, маркетинг і операційну ефективність, має вирішальне значення для сталого зростання та конкурентної переваги.

Адаптивність на нестабільному ринку: динамічний характер технологічного сектора вимагає гнучкості у фінансовому плануванні. Розроблені в цьому дослідженні моделі можна адаптувати для реагування на зміни ринкових умов, допомагаючи компаніям залишатися гнучкими та стійкими в умовах нестабільності.

Прийняття рішень на основі даних: це дослідження підкреслює цінність емпіричного аналізу та стратегій на основі даних у фінансовому плануванні. Технологічні компанії можуть використовувати ці моделі для тестування різних сценаріїв і оптимізації розподілу ресурсів відповідно до своїх стратегічних цілей.

Майбутні напрямки досліджень: Хоча це дослідження забезпечує надійну структуру для моделювання розподілу фінансових ресурсів, кілька областей вимагають подальшого вивчення:

Включення нефінансових показників: майбутні дослідження могли б інтегрувати нефінансові показники, такі як задоволеність клієнтів, сила бренду та залученість співробітників, щоб забезпечити більш цілісне уявлення про фактори, що впливають на ціни акцій. Секторальні моделі: розробка моделей, адаптованих до конкретних сегментів у технологічному секторі (наприклад, компанії, що займаються програмним забезпеченням чи апаратним забезпеченням), може дати точнішу інформацію та рекомендації. Поздовжні дослідження: Довгострокові дослідження, що відстежують ефективність компаній, які приймають рекомендовані стратегії розподілу, були б цінними для перевірки ефективності цих моделей з часом.

Вплив нових технологій: оскільки нові технології, такі як штучний інтелект і блокчейн, продовжують розвиватися, оцінка їхнього впливу на розподіл фінансових ресурсів і ринкову оцінку буде критично важливою для майбутніх досліджень.

Визначення оптимального співвідношення розподілу фінансових ресурсів є складним завданням, на яке впливають кілька факторів, включаючи етап розвитку компанії, її конкурентне середовище та її фінансовий стан. У технологічному середовищі, яке швидко розвивається, ефективний розподіл фінансових ресурсів став головним пріоритетом для компаній, які прагнуть залишатися конкурентоспроможними та досягати сталого зростання.

Це дослідження спрямоване на технологічні компанії, що працюють у різних сегментах технологічного сектора, включаючи, але не обмежуючись, розробку програмного забезпечення, виробництво апаратного забезпечення, інформаційно-технологічні послуги та нові технології, такі як штучний інтелект, блокчейн та Інтернет речей. Дослідження зосереджено на стратегії розподілу фінансових ресурсів у цих компаніях, зокрема на визначенні оптимального співвідношення розподілу між різними функціональними сферами, такими як дослідження та розробки, маркетинг, операції, інфраструктура та людські ресурси.

Для досягнення цілей цього дослідження були поставлені такі завдання:

Проведення комплексного огляду літератури про стратегії розподілу фінансових ресурсів, зокрема в контексті технологічних компаній.

Збір та аналіз фінансових даних із репрезентативної вибірки технологічних компаній, щоб зрозуміти поточну практику розподілу та її результати.

Розробка моделі для прогнозування впливу різних стратегій розподілу на ефективність компанії.

Виконання емпіричного аналізу для перевірки теоретичної моделі з використанням реальних даних, забезпечення її точності та надійності.

Визначення оптимальних коефіцієнтів розподілу фінансових ресурсів, які співвідносяться з покращеними показниками ефективності, такими як зростання доходу, прибутковості та курсу акцій.

Завдяки цим завданням дослідження має на меті надати технологічним компаніям практичну інформацію та стратегії для оптимізації розподілу їхніх фінансових ресурсів, тим самим покращуючи їх зростання та конкурентоспроможність на ринку.

На завершення це дослідження пропонує комплексний підхід до розуміння та оптимізації розподілу фінансових ресурсів у технологічних компаніях. Пов'язуючи фінансові стратегії з показниками курсу акцій, дослідження дає практичну інформацію, яка може сприяти прийняттю стратегічних рішень і сприяти сталому зростанню в технологічному секторі, що постійно розвивається.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Pearce, J. M. (2016). "Return on investment for open source scientific hardware development" (PDF). *Science and Public Policy*. 43 (2): 192–195. doi:10.1093/scipol/scv034
2. David Lando and Rolf Poulsen's lecture notes, Chapter 9, "Portfolio theory"
3. Lazo, J.G.L., Pacheco, M.A.C., Vellasco, M.M.B.R. (2009). Real Options Theory. In: Pacheco, M.A.C., Vellasco, M.M.B.R. (eds) *Intelligent Systems in Oil Field Development under Uncertainty*. Studies in Computational Intelligence, vol 183. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-93000-6_2
4. Accumulation distribution. Fidelity. (n.d.). <https://www.fidelity.com/learning-center/trading-investing/technical-analysis/technical-indicator-guide/accumulation-distribution>
5. Mitchell, C. (n.d.). Aroon indicator: Formula, calculations, interpretation, limits. Investopedia. <https://www.investopedia.com/terms/a/aroon.asp>
6. Williams, B. (1998). *New trading dimensions: How to profit from Chaos in stocks, bonds, and Commodities*. Wiley.
7. Form 10-Q. (n.d.). <https://www.sec.gov/files/form10-q.pdf>
8. Kenton, W. (n.d.). SEC form 10-Q: Definition, deadlines for filing, and components. Investopedia. <https://www.investopedia.com/terms/1/10q.asp>
9. Form 10-Q - apple inc.. (n.d.). https://s2.q4cdn.com/470004039/files/doc_earnings/2023/q3/filing/_10-Q-Q3-2023-As-Filed.pdf
10. Brownlee, J. (2017). *Introduction to time series forecasting with python: How to prepare data and develop models to predict the future*. Jason Brownlee.
11. Hamilton, J. D. (2020). *Time Series analysis*. Princeton University Press.
12. SAS Institute (Carolina del Norte). (1998). SAS. SAS Institute.
13. Killick, Rebecca, Fearnhead, Paul, & Eckley, Idris. (2012). Optimal detection of changepoints with a linear computational cost.
14. Glantz, S. A., Slinker, B. K., & Neilands, T. B. (2016). *Primer of Applied Regression & Analysis of variance*. McGraw-Hill Education.

15. Hypothesis testing: Lecture 4. (2018). . Palomar Community College District.
16. Hora, S. (2022, January 10). Parametric tests -the T-test. Medium.
<https://towardsdatascience.com/parametric-tests-the-t-test-c9b17faabfb0>
17. X. Jarque-Bera test - MATLAB. (n.d.).
<https://www.mathworks.com/help/stats/jbtest.html;jsessionid=44b4c34b23677410e4c27a0ca413>
18. Akaike, H. (1992). Information theory and an extension of the maximum likelihood principle. s.n.
19. Stoica, Peter, & Selén, Yngve. (2004). Model-order selection: A Review of Information Criterion Rules. Institutionen för informationsteknologi Regler teknik Systems and Control Systems and Control.
20. 1.1. Linear Models. scikit. (n.d.). https://scikit-learn.org/stable/modules/linear_model.html
21. Liu, D. C., & Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1–3), 503–528.
<https://doi.org/10.1007/bf01589116>
22. 1.7. gaussian processes. scikit. (n.d.-b). https://scikit-learn.org/stable/modules/gaussian_process.html
23. Tofallis, C. (2015). A better measure of relative prediction accuracy for model selection and model estimation. *Journal of the Operational Research Society*, 66(8), 1352–1362. <https://doi.org/10.1057/jors.2014.103>
24. Differential Evolution algorithm. differential evolution algorithm - an overview | ScienceDirect Topics. (n.d.). <https://www.sciencedirect.com/topics/computer-science/differential-evolution-algorithm>

ДОДАТКИ

ДОДАТОК А

```

import os
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_percentage_error, r2_score
from sklearn.ensemble import RandomForestRegressor
import matplotlib.pyplot as plt

def process_file(file_path):
    df = pd.read_excel(file_path)
    numeric_cols = df.select_dtypes(include=[float, int]).columns
    df[numeric_cols] = df[numeric_cols].replace(0, df[numeric_cols].mean())
    scaler = MinMaxScaler()
    df[numeric_cols] = scaler.fit_transform(df[numeric_cols])
    X_interact_df = df.copy()
    selected_features = []
    for i in range(2, X_interact_df.shape[1]):
        for j in range(2, X_interact_df.shape[1]):
            if i != j:
                col1_name = X_interact_df.columns[i]
                col2_name = X_interact_df.columns[j]
                new_feature_name = f"{col1_name}*{col2_name}"
                X_interact_df[new_feature_name] = X_interact_df[col1_name] *
X_interact_df[col2_name]
                X = X_interact_df[[new_feature_name]]

```

```

y = X_interact_df['target']
model = RandomForestRegressor(random_state=42)
model.fit(X, y)
if model.feature_importances_[0] > 0.001:
    selected_features.append(new_feature_name)
X_interact_df = X_interact_df[selected_features]
X_interact_df.fillna(0, inplace=True)
return X_interact_df

```

```

def load_data(root_folder):
    all_data = []
    file_paths = [os.path.join(folder_name, file_name)
                  for folder_name, subfolders, files in os.walk(root_folder)
                  for file_name in files if file_name.endswith('.xlsx')]
    for file_path in file_paths:
        data = process_file(file_path)
        all_data.append(data)
    combined_data = pd.concat(all_data, ignore_index=True)
    return combined_data

```

```

def train_and_evaluate(data):
    X = data.drop(columns=['target'])
    y = data['target']
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
    model = LinearRegression(n_jobs=-1)
    X_train.fillna(0, inplace=True)
    X_test.fillna(0, inplace=True)
    model.fit(X_train, y_train)
    y_train_pred = model.predict(X_train)

```

```

y_test_pred = model.predict(X_test)
train_mape = mean_absolute_percentage_error(y_train, y_train_pred)
test_mape = mean_absolute_percentage_error(y_test, y_test_pred)
train_r2 = r2_score(y_train, y_train_pred)
test_r2 = r2_score(y_test, y_test_pred)
return model, train_mape, test_mape, train_r2, test_r2, X_train, y_train_pred,
X_test, y_test_pred

```

```

def plot_actual_vs_predicted(y_actual, y_predicted, title):
    plt.figure(figsize=(8, 6))
    plt.scatter(y_actual, y_predicted, alpha=0.5)
    plt.plot([y_actual.min(), y_actual.max()], [y_actual.min(), y_actual.max()], 'k--',
lw=2)
    plt.xlabel('Actual')
    plt.ylabel('Predicted')
    plt.title(title)
    plt.grid(True)
    plt.show()

```

```

def plot_residuals(y_actual, y_predicted, title):
    residuals = y_actual - y_predicted
    plt.figure(figsize=(8, 6))
    plt.scatter(y_predicted, residuals, alpha=0.5)
    plt.axhline(y=0, color='r', linestyle='--', linewidth=2)
    plt.xlabel('Predicted')
    plt.ylabel('Residuals')
    plt.title(title)
    plt.grid(True)
    plt.show()

```

```
if __name__ == "__main__":
    root_folder = "E:/Desktop/work"
    combined_data = load_data(root_folder)
    model, train_mape, test_mape, train_r2, test_r2, X_train, y_train_pred, X_test,
y_test_pred = train_and_evaluate(combined_data)
    print("\nTraining MAPE:", train_mape)
    print("Testing MAPE:", test_mape)
    print("Training R^2:", train_r2)
    print("Testing R^2:", test_r2)
    plot_actual_vs_predicted(y_train, y_train_pred, title="Training Set: Actual vs.
Predicted")
    plot_actual_vs_predicted(y_test, y_test_pred, title="Testing Set: Actual vs.
Predicted")
    plot_residuals(y_train, y_train_pred, title="Training Set: Residual Plot")
    plot_residuals(y_test, y_test_pred, title="Testing Set: Residual Plot")
```

ДОДАТОК Б

```
import os
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt

def find_excel_files(directory):
    excel_files = []
    for root, dirs, files in os.walk(directory):
        for file in files:
            if file.endswith('.xlsx') or file.endswith('.xls'):
                excel_files.append(os.path.join(root, file))
    return excel_files

def evaluate_model(file_path):
    try:
        df = pd.read_excel(file_path)
        X = df.drop(['date', 'target'], axis=1)
        y = df['target']
        interaction_features = pd.DataFrame(index=X.index)
        for col1 in X.columns:
            for col2 in X.columns:
                if col1 != col2:
                    interaction_features[f"{col1}*{col2}"] = X[col1] * X[col2]
                    interaction_features[f"{col1}/{col2}"] = X[col1] / (X[col2] + 1e-10)
        predictors_interact = pd.concat([X, interaction_features], axis=1)
        model = LinearRegression()
        model.fit(predictors_interact, y)
```

```

    y_pred = model.predict(predictors_interact)
    return y_pred, y
except Exception as e:
    print(f"Error processing {file_path}: {str(e)}")
    return None, None

def process_all_files(root_folder):
    try:
        excel_files = find_excel_files(root_folder)
        if not excel_files:
            print(f"No Excel files found in {root_folder}.")
            return
        all_y_pred = []
        all_y_true = []
        for file in excel_files:
            y_pred, y_true = evaluate_model(file)
            if y_pred is not None and y_true is not None:
                all_y_pred.append(y_pred)
                all_y_true.append(y_true)
        combined_y_pred = np.concatenate(all_y_pred)
        combined_y_true = np.concatenate(all_y_true)
        plt.figure(figsize=(10, 6))
        plt.scatter(combined_y_pred, combined_y_true, alpha=0.5)
        plt.xlabel('Predicted Target')
        plt.ylabel('Actual Target')
        plt.title('Predicted vs. Actual Target')
        plt.show()
    except Exception as e:
        print(f"Error processing files: {str(e)}")

```

```
if __name__ == "__main__":  
    root_folder = r'E:\Desktop\work'  
    process_all_files(root_folder)
```

ДОДАТОК В

```

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

file_path = r'E:\Desktop\analysis_results.xlsx'
df = pd.read_excel(file_path, sheet_name='Sheet1', decimal=',')

numeric_columns = ['R-squared', 'Adj. R-squared', 'F-statistic', 'Prob (F-statistic)',
                   'Log-Likelihood', 'AIC', 'BIC', 'No. Observations', 'Params',
                   'Std. Error', 't', 'P>|t|', 'Conf. Interval 95% (Lower)', 'Conf. Interval 95%
(Upper)',
                   'Omnibus', 'Prob(Omnibus)', 'Jarque-Bera (JB)', 'Prob(JB)', 'Skew',
'Kurtosis', 'Condition Number']

for col in numeric_columns:
    df[col] = pd.to_numeric(df[col], errors='coerce')

rsquared_groups = df.groupby('Company')['R-
squared'].mean().sort_values(ascending=False)
print("Companies grouped by R-squared:")
print(rsquared_groups)

fstatistic_groups = df.groupby('Company')['F-
statistic'].mean().sort_values(ascending=False)
print("\nCompanies grouped by F-statistic (influence of S&P 500):")
print(fstatistic_groups)

aic_groups = df.groupby('Company')['AIC'].mean().sort_values()

```

```

bic_groups = df.groupby('Company')['BIC'].mean().sort_values()
print("\nCompanies grouped by AIC (Akaike Information Criterion):")
print(aic_groups)
print("\nCompanies grouped by BIC (Bayesian Information Criterion):")
print(bic_groups)

plt.figure(figsize=(10, 6))
sns.scatterplot(x='R-squared', y='F-statistic', data=df, hue='Company', palette='Set1',
s=100)
plt.title('R-squared vs. F-statistic')
plt.xlabel('R-squared')
plt.ylabel('F-statistic')
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.tight_layout()
plt.show()

plt.figure(figsize=(12, 8))
mean_rsquared = df.groupby('Company')['R-
squared'].mean().sort_values(ascending=False)
sns.barplot(x=mean_rsquared.values, y=mean_rsquared.index, palette='viridis')
plt.xlabel('Mean R-squared')
plt.ylabel('Company')
plt.title('Mean R-squared for Each Company')
plt.tight_layout()
plt.show()

plt.figure(figsize=(10, 6))
sns.boxplot(data=df[['AIC', 'BIC']], orient='h', palette='Set2')
plt.xlabel('Value')
plt.title('AIC and BIC Distribution')

```

```
plt.tight_layout()
```

```
plt.show()
```

ДОДАТОК Г

```
import pandas as pd
import matplotlib.pyplot as plt

file_path = r'E:\Desktop\results.xlsx'
df = pd.read_excel(file_path)
df['File'] = df['File'].str.replace('Consolidated', '').str.replace('.xlsx', '')

sorted_df = df.sort_values(by='Max Actual Target')

df['Ratio'] = df['Max Target'] / df['Max Actual Target']
sorted_ratio_df = df.sort_values(by='Ratio')

plt.figure(figsize=(10, 6))
plt.bar(sorted_ratio_df['File'], sorted_ratio_df['Ratio'], color='skyblue')
plt.xticks(rotation=45)
plt.xlabel('Company')
plt.ylabel('Ratio (Max Target / Max Actual Target)')
plt.title('Ratio of Max Target to Max Actual Target')
plt.tight_layout()
plt.show()
```

ДОДАТОК Д

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import UnivariateSpline

def normalize_data(x):
    return (x - np.min(x)) / (np.max(x) - np.min(x))

def distance_to_combined(x, y, smoothed_combined):
    return np.abs(y - smoothed_combined(x))

root_folder = r"E:\Desktop\work"
all_normalized_targets = []
max_length = 0

for subdir, dirs, files in os.walk(root_folder):
    for file in files:
        if file.endswith(".xlsx"):
            filepath = os.path.join(subdir, file)
            company_data = pd.read_excel(filepath)

            if 'date' in company_data.columns and 'target' in company_data.columns:
                dates = company_data['date']
                targets = company_data['target']

                normalized_targets = normalize_data(targets)
```

```

if len(normalized_targets) > max_length:
    max_length = len(normalized_targets)
    all_normalized_targets.append(normalized_targets)

for i in range(len(all_normalized_targets)):
    current_length = len(all_normalized_targets[i])
    if current_length < max_length:
        padding_length = max_length - current_length
        all_normalized_targets[i] = np.concatenate((all_normalized_targets[i],
np.zeros(padding_length)))

all_normalized_targets = np.array(all_normalized_targets)

smoothed_combined = UnivariateSpline(np.arange(max_length),
np.mean(all_normalized_targets, axis=0), s=0.01, k=4)

x = np.linspace(0, max_length, 200)
y = np.linspace(np.min(all_normalized_targets), np.max(all_normalized_targets),
100)
X, Y = np.meshgrid(x, y)
distance = distance_to_combined(X, Y, smoothed_combined)

plt.figure(figsize=(8, 6))
plt.contourf(X, Y, distance, levels=50, cmap='viridis')
plt.plot(np.arange(max_length), smoothed_combined(np.arange(max_length)),
color='red', label='Smoothed Combined Function')
plt.title('Distance to Smoothed Combined Function')
plt.xlabel('Time')
plt.ylabel('Normalized Target')
plt.legend()

```

```
plt.tight_layout()  
plt.colorbar(label='Distance')  
plt.show()
```

ДОДАТОК Е

```
import os
import pandas as pd
import matplotlib.pyplot as plt
import ruptures as rpt

def detect_change_points(company_data):
    target = company_data['target'].values
    model = "rbf" # Radial Basis Function kernel for mean shifts
    algo = rpt.Pelt(model=model).fit(target)
    result = algo.predict(pen=3)
    return result

def process_company_file(file_path):
    try:
        df = pd.read_excel(file_path)
        df['date'] = pd.to_datetime(df['date'], format='%Y.%m.%d')

        change_points = detect_change_points(df)

        if len(change_points) == 0:
            print(f"No change points found for file: {file_path}")
            return pd.Series(dtype='datetime64[ns]')

        valid_change_points = [idx for idx in change_points if idx < len(df)]
        change_dates = df.iloc[valid_change_points]['date']

    return change_dates
```

```

except Exception as e:
    print(f"Error processing file {file_path}: {e}")
    return pd.Series(dtype='datetime64[ns]')

def process_all_company_files(root_folder):
    all_change_dates = []
    for subdir, dirs, files in os.walk(root_folder):
        for file in files:
            file_path = os.path.join(subdir, file)
            if file_path.endswith('.xlsx'):
                change_dates = process_company_file(file_path)
                all_change_dates.extend(change_dates)

    quarterly_counts = pd.Series(all_change_dates).dt.to_period('Q').value_counts()

    for quarter, count in quarterly_counts.items():
        print(f"{quarter} - {count}")

plt.figure(figsize=(12, 6))
plt.hist(all_change_dates, bins=30, edgecolor='black')
plt.title('Distribution of Crisis Points')
plt.xlabel('Date')
plt.ylabel('Frequency')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

root_folder = r'E:\Desktop\work'
process_all_company_files(root_folder)

```

ДОДАТОК Ж

```
import os
import pandas as pd
import numpy as np
from sklearn.metrics import r2_score
import matplotlib.pyplot as plt
from statsmodels.tsa.statespace.sarimax import SARIMAX
from statsmodels.tsa.arima.model import ARIMA

def load_data(root_folder):
    data = {}
    for subdir, _, files in os.walk(root_folder):
        for file in files:
            if file.endswith('.xlsx'):
                file_path = os.path.join(subdir, file)
                df = pd.read_excel(file_path)
                df.columns = df.columns.str.strip().str.lower()
                for col in df.columns:
                    if col != 'date':
                        if df[col].dtypes == 'float64' or df[col].dtypes == 'int64':
                            df[col].fillna(0, inplace=True)
                            column_mean = df[col].replace(0, pd.NA).mean()
                            df[col] = df[col].replace(0, column_mean)
                if 'date' in df.columns and 'target' in df.columns:
                    df['date'] = pd.to_datetime(df['date'], format='%Y.%m.%d')
                    df = df.set_index('date').resample('Q').mean()
                    df['target'].fillna(0, inplace=True)
                    column_mean = df['target'].replace(0, pd.NA).mean()
                    df['target'] = df['target'].replace(0, column_mean)
```

```

        data[file] = df
    return data

```

```

def split_data(data, test_size=0.2):
    train_test_data = {}
    for key, df in data.items():
        train_size = int(len(df) * (1 - test_size))
        train, test = df.iloc[:train_size], df.iloc[train_size:]
        train_test_data[key] = (train, test)
    return train_test_data

```

```

def evaluate_sarima(train, test, order, seasonal_order):
    model = SARIMAX(train['target'], order=order, seasonal_order=seasonal_order,
enforce_stationarity=False, enforce_invertibility=False)
    sarima_results = model.fit(dispatch=False)
    train_pred = sarima_results.predict(start=train.index[0], end=train.index[-1])
    test_pred = sarima_results.predict(start=test.index[0], end=test.index[-1])
    r2_train = r2_score(train['target'], train_pred)
    r2_test = r2_score(test['target'], test_pred)
    return r2_train, r2_test

```

```

def evaluate_arima(train, test, order):
    model = ARIMA(train['target'], order=order)
    arima_results = model.fit()
    train_pred = arima_results.predict(start=train.index[0], end=train.index[-1])
    test_pred = arima_results.predict(start=test.index[0], end=test.index[-1])
    r2_train = r2_score(train['target'], train_pred)
    r2_test = r2_score(test['target'], test_pred)
    return r2_train, r2_test

```

```

if __name__ == "__main__":
    root_folder = "E:\\Desktop\\work"
    data = load_data(root_folder)
    train_test_data = split_data(data)
    results = {'model': [], 'r2_train_arima': [], 'r2_train_sarima': [], 'r2_test_arima': [],
'r2_test_sarima': []}
    for key, (train, test) in train_test_data.items():
        arima_order = (1, 1, 1)
        r2_train_arima, r2_test_arima = evaluate_arima(train, test, arima_order)
        sarima_order = (1, 1, 1, 4)
        r2_train_sarima, r2_test_sarima = evaluate_sarima(train, test, sarima_order[:3],
sarima_order)
        results['model'].append(key)
        results['r2_train_arima'].append(r2_train_arima)
        results['r2_train_sarima'].append(r2_train_sarima)
        results['r2_test_arima'].append(r2_test_arima)
        results['r2_test_sarima'].append(r2_test_sarima)
    results_df = pd.DataFrame(results)
    results_df['r2_diff_train'] = results_df['r2_train_sarima'] -
results_df['r2_train_arima']
    results_df['r2_diff_test'] = results_df['r2_test_sarima'] - results_df['r2_test_arima']
    plt.figure(figsize=(10, 6))
    plt.subplot(1, 2, 1)
    plt.scatter(results_df.index, results_df['r2_diff_train'], color='blue')
    plt.axhline(y=0, color='gray', linestyle='--')
    plt.title('Difference in R^2 (Training Set)')
    plt.xlabel('Model Index')
    plt.ylabel('SARIMA R^2 - ARIMA R^2')
    plt.subplot(1, 2, 2)
    plt.scatter(results_df.index, results_df['r2_diff_test'], color='red')

```

```
plt.axhline(y=0, color='gray', linestyle='--')
plt.title('Difference in R^2 (Testing Set)')
plt.xlabel('Model Index')
plt.ylabel('SARIMA R^2 - ARIMA R^2')
plt.tight_layout()
plt.show()
```

ДОДАТОК 3

```
import os

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.ensemble import RandomForestRegressor

from sklearn.inspection import permutation_importance

def find_excel_files(directory):

    excel_files = []

    for root, dirs, files in os.walk(directory):

        for file in files:

            if file.endswith('.xlsx') or file.endswith('.xls'):

                excel_files.append(os.path.join(root, file))

    return excel_files

def evaluate_and_plot(file_path, output_folder):

    try:

        df = pd.read_excel(file_path)

        X = df.drop(['date', 'target'], axis=1)

        y = df['target']

        interaction_features = pd.DataFrame(index=X.index)
```

```

for col1 in X.columns:
    for col2 in X.columns:
        if col1 != col2:
            interaction_features[f"{col1}*{col2}"] = X[col1] * X[col2]
            interaction_features[f"{col1}/{col2}"] = X[col1] / (X[col2] + 1e-10)

predictors_interact = pd.concat([X, interaction_features], axis=1)

model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(predictors_interact, y)

rf_importances = model.feature_importances_

perm_importance = permutation_importance(model, predictors_interact, y,
n_repeats=10, random_state=42)

perm_importances = perm_importance.importances_mean

plot_top_n_importances(predictors_interact.columns, rf_importances, file_path,
output_folder, "Random Forest Importance", "rf_importance")

plot_top_n_importances(predictors_interact.columns, perm_importances, file_path,
output_folder, "Permutation Importance", "perm_importance")

except Exception as e:
    print(f"Error processing {file_path}: {str(e)}")

def plot_top_n_importances(predictors, importances, file_path, output_folder, title,
suffix):

```

```

n = 10

plt.figure(figsize=(12, 8))

indices = np.argsort(importances)[::-1][:n]

plt.bar(range(len(indices)), importances[indices], align='center')
plt.xticks(range(len(indices)), [predictors[i] for i in indices], rotation=90, fontsize=20)
plt.xlabel('Predictor Variables', fontsize=20)
plt.ylabel('Importance', fontsize=20)
plt.title(f'Top {n} Most Useful Predictor Variables ({title})', fontsize=20)
plt.tight_layout()

file_name = os.path.basename(file_path)
plot_file_path = os.path.join(output_folder,
f"{os.path.splitext(file_name)[0]}_{suffix}.png")
plt.savefig(plot_file_path)
plt.show()

print(f"Top {n} {title} variables for {os.path.basename(file_path)}:")
for i in indices:
    print(f"{predictors[i]}: {importances[i]:.4f}")

def process_all_files(root_folder):
    try:
        excel_files = find_excel_files(root_folder)

```

```
if not excel_files:
```

```
    print(f"No Excel files found in {root_folder}.")
```

```
    return
```

```
output_folder = os.path.join(root_folder, 'output_plots')
```

```
os.makedirs(output_folder, exist_ok=True)
```

```
for file in excel_files:
```

```
    evaluate_and_plot(file, output_folder)
```

```
except Exception as e:
```

```
    print(f"Error processing files: {str(e)}")
```

```
if __name__ == "__main__":
```

```
    root_folder = r'C:\Users\B0K1\Desktop\work'
```

```
    process_all_files(root_folder)
```

Календарний план виконання кваліфікаційної роботи бакалавра

№	Етапи роботи	Терміни виконання	Відмітка керівника про виконання
1	Розробка та затвердження завдання кваліфікаційної роботи бакалавра	До 22 листопада 2023 року	
2	Ознайомлення з темою та формування цілей дослідження	До 10 січня 2024 року	
3	Збір та аналіз літератури	До 20 січня 2024 року	
4	Формування теоретичної бази дослідження	До 5 лютого 2024 року	
5	Визначення методології дослідження	До 17 лютого 2024 року	
6	Збір даних та підготовка вхідних даних	До 5 березня 2024 року	
7	Розробка моделей прогнозування	До 20 березня 2024 року	
8	Розв'язання задачі оптимізації	До 3 квітня 2024 року	
9	Аналіз отриманих результатів	До 20 квітня 2024 року	
10	Написання основних розділів роботи	До 15 травня 2024 року	
11	Редагування та доопрацювання тексту роботи	До 30 травня 2024 року	
12	Подання роботи на попередній захист	До 3 червня 2024 року	

Науковий керівник: Федоренко Ірина Костянтинівна

Студент: Компанець Богдан Вікторович

ЗАВДАННЯ

на кваліфікаційну роботу бакалавра

студента 4 курсу спеціальності 051 «Економіка», ОПП «Економічна кібернетика»

Компанця Богдана Вікторовича

1. Тема роботи: Моделювання розподілу фінансових ресурсів провідних технологічних компаній
2. Термін завершення роботи: 30 травня 2024 року
3. Попередній захист роботи: 3 червня 2024 року
4. Об'єкт дослідження: технологічні компанії, що працюють у різних сегментах технологічного сектора, включаючи, але не обмежуючись, розробку програмного забезпечення, виробництво апаратного забезпечення, інформаційно-технологічні послуги та нові технології, такі як штучний інтелект, блокчейн та Інтернет речей
5. Предмет дослідження: моделі та методи розподілу обмежених ресурсів та їх прогнозування для технологічних компаній
6. Мета дослідження: дослідження розподілу обмежених ресурсів та їх прогнозування для технологічних компаній
7. Завдання дослідження:
 - 7.1. Збір та аналіз фінансових даних з репрезентативної вибірки технологічних компаній.
 - 7.2. Пошук точок кризового розриву даних.
 - 7.3. Очистка даних від впливу незалежних ринкових механізмів.
 - 7.4. Розробка моделей для прогнозування впливу різних стратегій розподілу на ефективність компанії.
 - 7.5. Реалізація порівняльного аналізу моделей.
 - 7.6. Створення прогнозу на основі побудованої моделі.
 - 7.7. Знаходження оптимального розподілу фінансових ресурсів.

Науковий керівник: к.е.н., доцент економічних наук Федоренко Ірина Костянтинівна

Студент: Компанець Богдан Вікторович

Затверджено на засіданні кафедри економічної кібернетики

протокол № 4 від 22.11.2023 р.