

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу**

До захисту допущено
В. о. завідувача кафедри
_____ О.Л. Тимошук
«___» _____ 20__ р.

**Дипломна робота
на здобуття ступеня бакалавра
за освітньо-професійною програмою «Системний аналіз і управління»
спеціальності 124 «Системний аналіз»
на тему: «Дослідження ринку криптовалют за допомогою статистичного
арбітржу та глибокого хеджування»**

Виконав:

Студентка IV курсу, групи КА-92
Катасанова Карина Олександрівна

Керівник:

доц., канд. фіз.-мат. наук Каніовська І.Ю.

Консультант з економічного розділу:

доцент, к.е.н. Рощина Н.В.

Консультант з нормоконтролю:

Доцент, к.т.н. Статкевич В.М.

Рецензент:

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань.

Студент _____

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

Інститут прикладного системного аналізу

Кафедра математичних методів системного аналізу

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 124 «Системний аналіз»

Освітньо-професійна програма «Системний аналіз і управління»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ О.Л. Тимошук

«__» _____ 20__ р.

ЗАВДАННЯ

на дипломну роботу студентці

Катасановій Карині Олександрівні

1. Тема роботи «Дослідження ринку криптовалют за допомогою статистичного арбітражу та глибокого хеджування», керівник роботи доцент, канд. фіз.-мат. наук Каніовська І.О., затверджені наказом по університету від «__» _____ 20__ р. № _____

2. Термін подання студентом роботи

3. Вихідні дані до роботи:

4. Зміст роботи:

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): презентація

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Рощина Н.В., доцент		

7. Дата видачі завдання: _____

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Затвердження теми ДР		виконано
2	Ознайомлення зі структурою БДР згідно з Положенням про державну атестацію студентів НТУУ «КПІ ім. І. Сікорського»		виконано
3	Ознайомлення з ДСТУ 3008-95 та стандарти ЄСПД		виконано
4	Проведення дослідження за темою БДР під керівництвом керівника		виконано
5	Завершення роботи над першим варіантом частини БДР		виконано
6	Проведення роботи над експериментальною частиною БДР		виконано
7	Проведення роботи над програмним продуктом		виконано
8	Оформлення БДР та аналіз отриманих результатів		виконано

Студент

К.О. Катасанова

Керівник

І.Ю. Каніовська

РЕФЕРАТ

Дипломна робота містить сторінок текстової частини, ілюстрацій, таблиць, 2 додатки, 29 бібліографічних посилань.

Ключові слова: КРИПТОВАЛЮТИ, АРБІТРАЖ, ХЕДЖУВАННЯ, МАШИННЕ НАВЧАННЯ, ТОРГОВІ СТРАТЕГІЇ, КОІНТЕГРАЦІЯ, СТАЦІОНАРНІСТЬ.

Об'єкт дослідження: ринок криптовалют.

Мета дослідження: визначити можливості застосування статистичного арбітражу на глибокого хеджування на крипторинку, реалізувати трейдингову стратегію з використанням динамічного стоп-лосу та інтерпретувати отримані результати для загального випадку.

Актуальність дослідження: популяризація крипторинку та поява нових можливостей для торгівлі.

Методи дослідження: методи прогнозування: нейронна мережа CNN-LSTM; методи дослідження коінтеграції: доповнений коінтеграційний тест Дікі-Фуллера, тест Квятковського-Філіпса-Шмідта-Шина.

Отримані результати: було створено механізм дослідження ринку криптовалют методами статистичного арбітражу та глибокого хеджування. Було побудовано арбітражну стратегію з використанням динамічного стоп-лосу та виявлено, що вона є досить прибутковою та має низькі ризики. Також, було показано, що застосування даного механізму дослідження потребує попереднього аналізу криптонабору.

В рамках подальшого дослідження пропонується побудувати більш універсальний алгоритм застосування статистичного арбітражу та глибокого хеджування, що дозволить дослідити ринок більш глибоко, а також розширити даний механізм на більшу кількість арбітражних стратегій.

ABSTRACT

The theme: ‘Cryptocurrencies market research using statistical arbitrage and deep hedging’

Diploma work: p., fig., tabl., 2 appendixes, 29 references.

Keywords: CRYPTOCURRENCIES, ARBITRAGE, HEDGING, MACHINE LEARNING, TRADING STRATEGIES, COINTEGRATION, STATIONARITY.

The object of the study: the cryptocurrencies market.

The purpose of the study: determination of the possibilities of applying statistical arbitrage and deep hedging on the crypto market, and implement a trading strategy using dynamic stop-loss, interpret the obtained results for the general case.

The relevance of the study: the popularization of the crypto market and the emergence of new trading opportunities.

Research methods: forecasting methods: CNN-LSTM neural network; cointegration testing methods: augmented Dickey-Fuller cointegration test, Kwiatkowski-Phillips-Schmidt-Shine test.

The results obtained: a mechanism for researching the cryptocurrency market using statistical arbitrage and deep hedging methods was created. An arbitrage strategy using dynamic stop-loss was built and defined that it is quite profitable and low-risk. Also, it was shown that the application of this research mechanism requires a preliminary analysis of the cryptographic set.

As part of further research, proposed to build a more universal algorithm for the usage of statistical arbitrage and deep hedging, which allows to explore the market more deeply. Also, opportunities for the further research are defined in expanding built mechanism to a larger number of arbitrage strategies.

ЗМІСТ

ВСТУП	8
1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ.....	11
1.1 Основні поняття ринку криптовалют	11
1.2 Існуючі ризики на ринку криптовалют.....	14
1.3 Перспективи ринку криптовалют.....	16
1.4 Аналіз існуючих підходів до вирішення поставленої задачі	18
1.4.1. Огляд основних методів дослідження ринку криптовалют	19
1.4.2. Комбінації підходів до дослідження ринку криптовалют ..	21
1.4.3. Обґрунтування комбінації глибокого хеджування та статистичного арбітражу	29
Висновки	32
2 ПРИНЦИП РЕАЛІЗАЦІЇ СТАТИСТИЧНОГО АРБІТРАЖУ ТА ГЛИБОКОГО ХЕДЖУВАННЯ НА РИНКУ КРИПТОВАЛЮТ	34
2.1. Основні математичні визначення.....	34
2.2. Реалізація глибокого хеджування	35
2.2.1. Структура і принцип роботи нейронної мережі CNN-LSTM	36
2.2.2. Дослідження коінтеграції.....	40
2.3. Статистичний арбітраж як метод торгівлі криптовалютами.....	43
2.4. Показники для оцінювання якості проведеного дослідження ..	45
2.4.1. Оцінювання роботи нейронної мережі CNN-LSTM	45
2.4.2 Оцінювання якості арбітражної стратегії.....	48
Висновки	50

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ	52
3.1. Опис розробленого програмного продукту.....	52
3.2. Аналіз вхідних даних.....	53
3.2.1 Аналіз першого датасету.....	54
3.2.2. Аналіз другого датасету	57
3.3. Архітектура програмного продукту.....	61
3.4. Аналіз результатів дослідження.....	63
3.4.1. Аналіз результатів для першого датасету	63
3.4.2. Аналіз результатів для другого датасету.....	69
Висновки	77
4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ	79
4.1. Постановка задачі проектування.....	80
4.2. Обґрунтування функцій програмного продукту.....	80
4.3. Обґрунтування системи параметрів ПП	83
4.4. Аналіз експертного оцінювання параметрів	86
4.5 Аналіз рівня якості варіантів реалізації функцій.....	91
4.6 Економічний аналіз варіантів розробки ПП	92
4.7 Вибір кращого варіанту ПП техніко-економічного рівня	98
Висновки	99
ВИСНОВКИ	101
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	102
ДОДАТОК А.....	105
ДОДАТОК Б	119

ВСТУП

Ринок криптовалют є об'єктом багатьох дискусій у сучасному світі. Даний ринок є доволі непередбачуваним, що експерти розцінюють як його перевагу так і суттєвий недолік. Крім того, в останні декілька років криптовалюти набули популярності у суспільства та перебувають у стадії активного розвитку – з'являються нові цифрові валюти, NFT та токени, крім того по всьому світу відкриваються підприємства, що займаються виключно дослідженням ринку.

Дослідження ринку криптовалют є доволі важливим для світової економіки. Більшість досліджень розглядають криптовалюти як спосіб хеджування фіатних валют або як поле для інвестицій та зберігання коштів. Цікавим дослідженням є розгляд ринку криптовалют як спосіб для захисту від впливу інфляції на світову економіку та порівняння цифрових валют із золотом з метою визначити, що має більшу ефективність для хеджування [1]. Метою дослідження, представленого у цій роботі є розглянути стратегію статистичного арбітражу з використанням глибокого хеджування та визначити ефективність даної комбінації на ринку та можливості її застосування.

Ринок криптовалют відомий своєю високою волатильністю, що впливає на довіру користувачів до ринку та, власне, на популярність і швидкість розвитку ринку. Крім того, для цифрових валют досить важко застосувати стандартні способи хеджування з використанням ф'ючерсів та опціонів, оскільки подібні контракти не є популярними на ринку. Крім того, дослідження арбітражних можливостей на ринку шляхом порівняння різних криптовалютних бірж також не є ефективним, в силу високої волатильності та потребує швидкого реагування користувача та постійного моніторингу ринку. Саме тому застосування методів машинного навчання є вкрай важливим для досліджень, оскільки дані методи дозволяють робити більш точні прогнози та розширюють можливості статистичних методів дослідження.

Актуальність дослідження полягає в тому, що ринок криптовалют є досить молодим ринком, тому його перспективи та ризики до кінця невідомі. Постає питання – які інструменти, що показують високу ефективність на інших, більш стабільних фінансових ринках, можуть бути застосовані до ринку цифрових валют та до яких результатів вони приведуть. Крім того, дослідження ринку є важливим і для світової економіки в силу питань, що наразі виникають до банківської системи та її проблем, що змушує людство шукати альтернативні способи зберігання та примноження власних коштів.

Оскільки ринок криптовалют є доволі молодим, проведені раніше дослідження спираються на властивості більш традиційних фінансових ринків та на наявну інформацію ринку криптовалют, зокрема ціни активів, як наслідок отримані результати є релевантними для окремих наборів криптовалют, які вже існують на ринку довгий час, наприклад висновки можуть бути зроблені на основі даних про Біткоїн [26]. Новизна даної роботи полягає у реалізації механізму дослідження, який можна буде застосувати до будь-яких криптовалютних наборів, в тому числі і для тих, що містять нові активи, історичні дані про які доступні за невеликий проміжок часу. Це допоможе зробити більш точні висновки про ринок та дозволить розглянути торговельні можливості не лише для тих цифрових валют, що існують на ринку деякий час, а й для більш нових активів, інформації про які може бути недостатньо для інтерпретації тих досліджень, що були представлені раніше, для цього ми будемо спиратися на результати прогнозування цін методами машинного навчання.

Перший розділ буде присвячено огляду предметної області. У ньому будуть висвітлені основні поняття ринку криптовалют, розглянуті ризики і перспективи його розвитку, а також буде проведено огляд основних методів дослідження ринку. У другому розділі буде представлений апарат дослідження, його математичні основи та нейронна мережа, що буде застосована у ході дослідження. У цьому розділі буде розглянуто механізм реалізації глибокого хеджування та арбітражної стратегії з використанням

динамічного стоп-лосу на основі результатів отриманих хеджуванням. Також, буде описано метрики, що застосовуються у роботі для оцінки якості дослідження та буде обґрунтовано їх використання.

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Основні поняття ринку криптовалют

Ринком криптовалют називають сукупність усіх існуючих криптовалют, засобів для роботи з ними та криптовалютних продуктів, які наразі є доступними для продажу, купівлі та обміну на криптовалютних біржах. Даний ринок не підпорядковується центральному банку та іншим фінансовим органом, тобто є децентралізованим.

Криптовалюта, в свою чергу, є прикладом цифрового активу, який використовує принципи криптографії для шифрування та захисту даних. Дана валюта використовується як засіб для зберігання значних сум фіатних валют та для їхнього обміну. Кількість випущеної криптовалюти є обмеженою, що дозволяє зберегти цінність цифрової валюти з економічної точки зору, а також застосувати певні технології для роботи з нею, такі як блокчейн, що дозволяє створити обмежену кількість нових блоків, що містять транзакції.

Технологія блокчейн являє собою розподілену базу даних, яка зберігає, власне, записи про транзакції криптовалют. Працює ця технологія наступним чином – кожен новий блок містить у собі інформацію про попередній, записуючи дані про транзакцію до загального реєстру, тобто, з точки зору програмування, блокчейн є в деякому сенсі подібним до контейнера. З міркувань безпеки та забезпечення стійкості мережі, жодна з транзакцій не може бути змінена та, як зазначалося раніше, таких записів може бути обмежена кількість.

Створення нових блоків у блокчейні криптовалют називають майнінгом. В основі майнінгу лежить розв'язання багатьох складних математичних задач, які наразі не можуть бути розв'язані автоматичним шляхом. Тож, для створення нових блоків, залучаються користувачі –

майнери. Оскільки даний процес потребує значних зусиль, майнери отримують певну винагороду за розв'язання задач – це може бути створення власної криптовалюти та комісійні кошти від користувачів.

Важливим елементом функціонування блокчейн технології, та криптовалюти в цілому є токен. Токен, фактично, є видом цифрових активів і служить для забезпечення доступу до послуг, функцій та ресурсів на криптоплатформах. Кожній криптовалюті присвоюється власний токен, який використовується в межах даної криптовалюти, тобто є засобом платежу, власне, активом, який надає право на отримання певних бонусів від володіння криптовалютою.

Одним з ключових понять, яке безпосередньо впливає на функціонування ринку криптовалют є процес ІСО – одна з форм залучення інвесторів для проектів, що пов'язані з криптовалютами та блокчейн. Ідея полягає у тому, що компанія випускає новий токен, який надаватиме певну послугу у майбутньому, та пропонує його в обмін на іншу криптовалюту. У користувача цей токен може зберігатися як White paper, що містить детальний опис проекту, інші токени та переваги, які він надає. Використати такий токен можна як доступ до продукту, що розробляється, бонусів або як цінний папір, що потенційно зростатиме у вартості.

Всі придбані криптовалюти, а також інформація про них зберігаються у гаманці. Гаманцем виступає програмне забезпечення, яке надає користувачеві можливість безпосередньо зберігати криптовалюти, надсилати їх іншим користувачам та отримувати цифрову валюту від інших користувачів. Останні дві дії можливі з використанням унікальної адреси, яка присвоюється кожному гаманцю. Крім того, за допомогою аналізу окремого гаманця, можна зробити висновок про загальні тренди ринку криптовалют, оскільки більшість з них є взаємозалежними, однак це не є єдиним чинником. Використання гаманця має безліч нюансів, оскільки кожен з них пропонує різні функції та має свої переваги та недоліки.

Продаж чи купівля криптовалют за ринковою ціною здійснюється на криптовалютних біржах. На даних біржах можна як купити нову криптовалюту за фіатні валюти, або за інші криптовалюти, також можна придбати фіатні гроші за криптовалюти. Також на даних біржах часто реалізують угоди між користувачами, які готові придбати або продати криптовалюту на індивідуальних умовах. Деякі біржі також пропонують функції гаманця.

Як вже було зазначено раніше, нові криптовалюти можуть з'являтися у результаті ІСО процесу. Іншим процесом для створення криптовалют є форк – подія, в результаті якої блокчейн поділяється на дві версії, які продовжують функціонувати незалежно одна від одної, тобто безпосередньо утворюється нова криптовалюта. Однак, існує два види років – хардфорк та софтфорк. У першому випадку, зміни у блокчейні є доволі суттєвими, нова гілка функціонує за зовсім інакшими правилами та протоколами, в такому випадку користувачі повинні оновити свій програмний простір для продовження роботи. У випадку ж софтфорка, зміни є сумісними з попередньою версією. В такому випадку користувач може продовжити роботу з попереднім ланцюгом блоків та не переходити до нової криптовалюти.

Основними прикладами криптовалют на ринку є біткоїн та альткоїн. Біткоїн є найбільш відомою криптовалютою та був випущений у 2009 році. Дана криптовалюта працює за технологією блокчейн. Термін альткоїн був розроблений для опису всіх інших цифрових валют, які були випущені після біткоїну. Деякі з них можуть різнитися за принципами роботи, наприклад ІОТА є криптовалютою, яка не використовує технологію блокчейн. Однак, головні відмінності альткоїнів від біткоїна полягають у функціоналі, наприклад Ripple є криптовалютою для швидких міжнародних переказів, що не підтримується біткоїном.

1.2 Існуючі ризики на ринку криптовалют

Ринок криптовалют, на сьогоднішній день, є доволі зрозумілим у функціонуванні та не потребує значних зусиль для того щоб почати на ньому працювати. Однак, поверхнева простота ринку є досить привабливою для шахраїв та злочинців, тож власне виникає перший ризик – кібербезпека. Оскільки всі криптовалюти зберігаються на цифрових платформах, існує ризик злому та відповідно втрати всіх активів. Це може статися через втрату приватного ключа, який надає доступ до гаманця або через слабкий захист даних користувачів безпосередньо на обраній платформі. Нерідко бувають ситуації, коли програмне забезпечення як криптовалютна біржа або гаманець розробляється, безпосередньо, зловмисниками і кожна покупка не надає користувачеві реального діючого токenu, тож надалі працювати з придбаною цифровою валютою неможливо.

Також криптовалюти, оскільки з їх використанням зберігається майже повна анонімність, можуть бути використані у цілях фінансування терористичних операцій. Даний фактор призводить до ризику невизнання криптовалют деякими країнами, компаніями тощо. Це призводить до того, що використання цифрових валют може бути заборонене у фізичному світі або матиме певні серйозні обмеження. Це може призвести до зниження цінності такого типу валют, а отже і падіння у ціні.

З зазначеного фактору вище впливає наступний – ризик банкрутства. На ринку криптовалют втратити свої активи ризикують усі – від користувачів бірж до майнерів та власників сервісів. Якщо говорити про користувачів, то ризик банкрутства пояснюється високою волатильністю ринку. Волатильність – це тенденція цін до несподіваних змін, яка вимірюється як дисперсія прибутків активу. У контексті криптовалют, волатильність може призвести до значних збитків, власне, як і будь-яка інвестиція у акції, однак криптовалюти є менш стійкими до зовнішніх факторів таких як, зміни в законодавстві,

хакерські атаки та кількість згадувань у новинах. Банкрутство сервісів, таких як криптовалютні біржі та гаманці пояснюється недостатньою кількістю інвестицій та відсутності стабільного фінансування. Оскільки крипторинки є досить молодим, інвестори досить довго обдумують чи варто інвестувати у подібні продукти. Ситуацію тут дещо регулюється процесом ICO, однак для нових бірж цього часто недостатньо.

Маніпулювання на ринку криптовалют є одним з основних ризиків. В першу чергу, маніпулювання може відбуватися з метою підвищення ціни на криптовалюту. Цей метод називається *pump and dump* і застосовується задля того аби вигідно продати свої активи. Певна група людей розповсюджує інформацію про криптовалюту, підіймаючи інтерес спільноти, зокрема інвесторів, в результаті чого ціна штучно зростає. Після цього починається масовий продаж віртуальних монет, в результаті чого ціна різко падає. Даний процес представлений на прикладі Dogecoin (рисунок 1.1).



Рис. 1.1. Часовий ряд ціни на криптовалюту Dogecoin з сайту Coinbase.com

Іншим способом маніпулювання на ринку є розповсюдження неправдивої інформації про криптовалюту. Зазвичай метою є збільшення ціни криптовалюти, що є вигідним для тих, хто планує вигідно продати активи, однак стає проблемою для користувачів, що розраховують на подальший ріст ціни і інвестують у дану валюту. Маніпулювання в будь-якому випадку призводить до зниження рівня довіри до ринку та може призвести до його краху та витіснення криптовалют звичними фіатними валютами.

Ще одним суттєвим ризиком на ринку криптовалют є крекдаун – дії правоохоронних або регуляторних органів, що спрямовані на припинення злочинної діяльності. Проведення крекдауну може призвести до закриття криптовалютних бірж, вилучення всіх користувацьких гаманців та заборону інвестиційних проектів. Даний процес, знову ж таки, тягне за собою збитки для користувачів, втрату інвестованих коштів підприємців та підриває довіру до ринку криптовалют. За версією Bloomberg крекдаун допомагає створити більш безпечні умови на ринку[2], оскільки в першу чергу регулює шахрайську діяльність, однак, за версією Forbes даний процес регуляції є досить шкідливим не лише для шахрайських бірж, а й для тих, що функціонують довгий період часу та мають певну репутацію[3]. Так, за 2023 рік від крекдауну постраждали відомі криптовалютні біржі – Coinbase, Binance та Kraken. У результаті висунутих обвинувачень, біржі виплатили великі суми штрафів та обмежили свою діяльність, оскільки в деяких країнах попит на криптовалюту значно впав через проведений крекдаун.

1.3 Перспективи ринку криптовалют

Питання перспектив ринку криптовалют наразі є доволі дискусійним. З огляду на молодість ринку та велику кількість ризиків описаних раніше однозначно відповісти на питання чи дійсно криптовалюты можуть стати

основною формою валюти не можна. На сьогоднішній день продовжує активно розвиватися хоча і має більш обмежений функціонал у порівнянні з традиційними фінансовими ринками.

Однак, розглядаючи крипторинки як платформу для інвестицій можна зробити певні висновки. В першу чергу це потенціал високого зростання цін. Висока волатильність, яка є недоліком ринку криптовалют, водночас є його перевагою, що притягує потенційних інвесторів. Крім того, даний фактор позитивно впливає на перспективи розвитку ринку, оскільки перевірені криптовалюти приносять відносно стабільний дохід, а легкість доступу робить процес інвестування, та роботу з криптовалютами, зрозумілим для середньостатистичного користувача, що сприяє швидкій популяризації даного ринку. Крім того, ринок криптовалют має високу ліквідність – активи можна швидко купити або продати за ринкову ціну, без значних комісійних втрат. Криптовалюти є одними з небагатьох активів, часті операції над якими не лише не приводять до збитків, а й навпаки можуть примножити капітал.

Говорячи про збереження власних коштів у криптовалютах, теж можна відмітити певні переваги. По-перше, криптовалюти є швидкими засобами для оплати і деякі компанії вже визнали цифрові валюти на рівні з фіатними. У комплексі з підвищеною анонімністю, яку забезпечують криптовалюти, це дозволяє швидше укладати купівлі дорогих об'єктів, наприклад нерухомості. Це дає підстави вважати, що криптовалюти здатні витіснити звичні для нас фіатні гроші на другий план, однак, як вже було зазначено раніше, однозначно відповісти на дане питання наразі не можна.

Позитивно на розвиток ринку криптовалют впливає банківська криза, яка поширюється світом з 2022 року. Наразі більшість фіатних валют показують досить високу нестабільність, зокрема це можна помітити з курсу євро до долара. Ще у червні 2022 року Forbes повідомив, що у разі, якщо долар дожене євро у ціні можна буде говорити про початок Всесвітньої банківської кризи, яка вже була у 2008 році. Більшість фінансових аналітиків наразі радить не зберігати свої кошти у доларах, євро та інших фіатних валютах, а натомість

обрати інші засоби, як для збереження коштів та і для інвестицій. Дана ситуація, а також оголошення всесвітньо відомих банків банкротами та втрати банківських депозитів у березні 2023 року призвели до збільшення інтересу до ринку криптовалют на 5%. Проблеми у світовій економіці є позитивними для розвитку крипторинку, оскільки призводять до зростання їх ціни, однак і збільшують їхню волатильність, що може створити й негативний ефект на перспективи розвитку[28].

На сьогоднішній день, дослідники крипторинку та провідні економісти вважають, що ринок криптовалют наразі перебуває у стадії зростання, як це колись було з фіатними валютами під час розвитку світових банків. На думку Forbes така картина свідчить про те, що крипторинки активно рухається у напрямку закріплення свого місця в світовій економіці[4].

1.4 Аналіз існуючих підходів до вирішення поставленої задачі

Метою дослідження ринку криптовалют з використанням статистичного арбітражу та глибокого хеджування, як було зазначено раніше, є виявлення торговельних можливостей на крипторинку та оцінювання ефективності хеджування на непередбачуваних активах. Важливо зазначити, що розгляд кожної криптовалюти окремо не має сенсу, оскільки більшість з них не є широко вживаними і, в силу короткого терміну існування, не впливають на загальні тенденції ринку. Крім того, це неможливо, оскільки їх наразі існує дуже велика кількість, що зростає щодня, і для збору необхідної, в рамках дослідження, інформації цифрові валюти повинні проіснувати на біржах хоча б декілька діб.

Дослідження ринку може проводитися на основі збору фундаментальних даних про криптовалюту, з залученням певних статистичних метрик або за допомогою аналізу часових рядів. Кожен спосіб

приводить до різного результату наприкінці та дає змогу визначитись з можливостями подальшої роботи на ринку.

1.4.1. Огляд основних методів дослідження ринку криптовалют

Одним з методів дослідження ринку криптовалют є фундаментальний аналіз. В ході його застосування розглядається команда проекту конкретної криптовалюти, технології та протоколи за якими вона функціонує та інші базові показники. У результаті можна висунути припущення, які є ризики інвестування у дану криптовалюту та її перспективи росту. Також, фундаментальний аналіз має на увазі роботу з бізнес-планом компанії, що випустила криптовалюту, та кількістю користувачів та проведених транзакцій, що дозволяє оцінити ціну та прибутковість конкретної криптовалюти. Даний метод є доволі широко застосованим та популярним серед інвесторів, оскільки дозволяє приймати рішення, що будуть релевантні в довготривалій перспективі, з використанням регресійного аналізу, часових рядів та статистичних методів для формування оцінок.

Методи управління ризиками дають можливість оцінити, власне, ризики, що пов'язані з різними інвестиційними стратегіями та обрати одну, що дозволить досягти найкращих результатів. Дані методи часто застосовують методи машинного навчання та базуються на знаннях про часові ряди. Суттєвою перевагою даних методів є те, що окрім побудови сценаріїв виходячи з досвіду дослідника, застосовується також математичний апарат. Так, глибоке хеджування, що відноситься до методів управління ризиками застосовує регресійний аналіз, аналіз часових рядів, методи Монте-Карло тощо [7-8]. Однак, методи управління ризиками у якості основного апарату для захисту криптовалютного гаманця потребують постійного аналізу результатів та повторних досліджень. Це пов'язано з динамічністю ринку

криптовалют, та досить вузьким набором можливих хеджів на ринку, що робить застосування методів управління ризиками, що є звичними для традиційних фінансових ринків, досить складною задачею.

Оптимізація гаманця є досить популярним методом дослідження ринку криптовалют. Основна ідея полягає у виборі оптимальної комбінації цифрових валют для досягнення максимальної прибутковості портфелю. Зазвичай, даний метод використовує певні засоби фінансової математики, наприклад моделювання Марковіца або модель мінімального ризику. Основний недолік оптимізації гаманця полягає в обмеженості методів, які можна застосувати у контексті криптовалют та у високій волатильності самого ринку. В першу чергу це призводить до різких змін у ціні окремих криптовалют у гаманці, що позначиться на загальній прибутковості. По-друге, висока волатильність активів ускладнює й сам процес оптимізації гаманця, оскільки криптовалюта може впасти у ціні на значний відсоток за дуже короткий термін, що не завжди можна врахувати під час математичних розрахунків. Однак, даний метод можна застосувати як частину процесу управління ризиками, що навпаки розширить їхні можливості, проте зазвичай даний процес є доволі кропітким та вимагає залучення методів машинного навчання, особливо якщо мається на увазі робота з великою кількістю активів.

Одним з ключових методів аналізу ринку є аналіз технічних показників. В ході застосування цього методу проводиться дослідження поведінки руху цін за допомогою графіків, статистичних показників, таких як середня ціна, рівень підтримки, дисперсія, а також якісні показники ринку – кількість проданої валюти, транзакцій тощо. До аналізу технічних показників, у контексті криптовалют відноситься, статистичний арбітраж, з допомогою якого можна виявити кращі можливості отримання прибутку та виявити недооцінені чи переоцінені активи, що є на ринку. Технічний аналіз показників дозволяє користувачам розуміти не тільки можливості росту криптовалют, а й визначити коли краще входити на ринок, а коли вийти з нього та яку біржу краще обрати.

Застосування кожного з зазначених методів має свої переваги та недоліки. В першу чергу, можна помітити різну спрямованість методів – або прогнозування цін валют, що допомагає визначити найбільш вигідні для покупки активи, або зменшення ризиків щодо вже наявних у портфелі криптовалют. Зрозуміло, що визначення можливостей на ринку вимагає врахування обох факторів, тож зазвичай для дослідження ринку застосовують комбінації методів.

1.4.2. Комбінації підходів до дослідження ринку криптовалют

Для досягнення максимальної ефективності від дослідження ринку криптовалют часто застосовують декілька методів аналізу, в залежності від задач, які ставить перед собою дослідник. Основною метою комбінацій методів є врахувати вплив ризиків на вартість гаманця та розглянути торгові можливості на ринку.

Однією з найбільш поширених комбінацій методів є фундаментальний аналіз та технічний аналіз показників. У сукупності ці два методи дозволяють отримати більш широке уявлення про ринок криптовалют та точніші прогнози цін на активи. Ці результати досягаються, оскільки в ході фундаментального аналізу враховуються зовнішні фактори, які впливають на рівень зацікавленості у криптовалютах, а також на надійність ринку. Комбінуючи результати отримані під час вивчення фундаментальних факторів з результатами аналізу технічних показників ринку, можна зробити висновки як ті чи інші соціальні показники впливають на коливання ціни на криптовалюту, та відповідно зрозуміти чи є дана криптовалюта перспективною для торгівлі та чи варто вкладати у неї кошти.

Розглянемо наступний приклад застосування комбінації фундаментального аналізу та технічного аналізу показників для аналізу

нещодавно випущених tokenів. Криптовалюта DIMO зараз є новою на ринку, тож відповідно недорого коштує та є досить привабливою для інвестицій. Дана криптовалюта була випущена у результаті процесі ICO, однойменною компанією, що займається створенням віртуальної інфраструктури для водіїв. З «білого паперу» криптовалюти, відомо, що даний продукт є корисним як для власників автомобілів так і для розробників програмного забезпечення. Крім того, компанія заздалегідь забезпечила DIMO схемою підвищення зацікавленості токеном у суспільстві (рисунок 1.2).

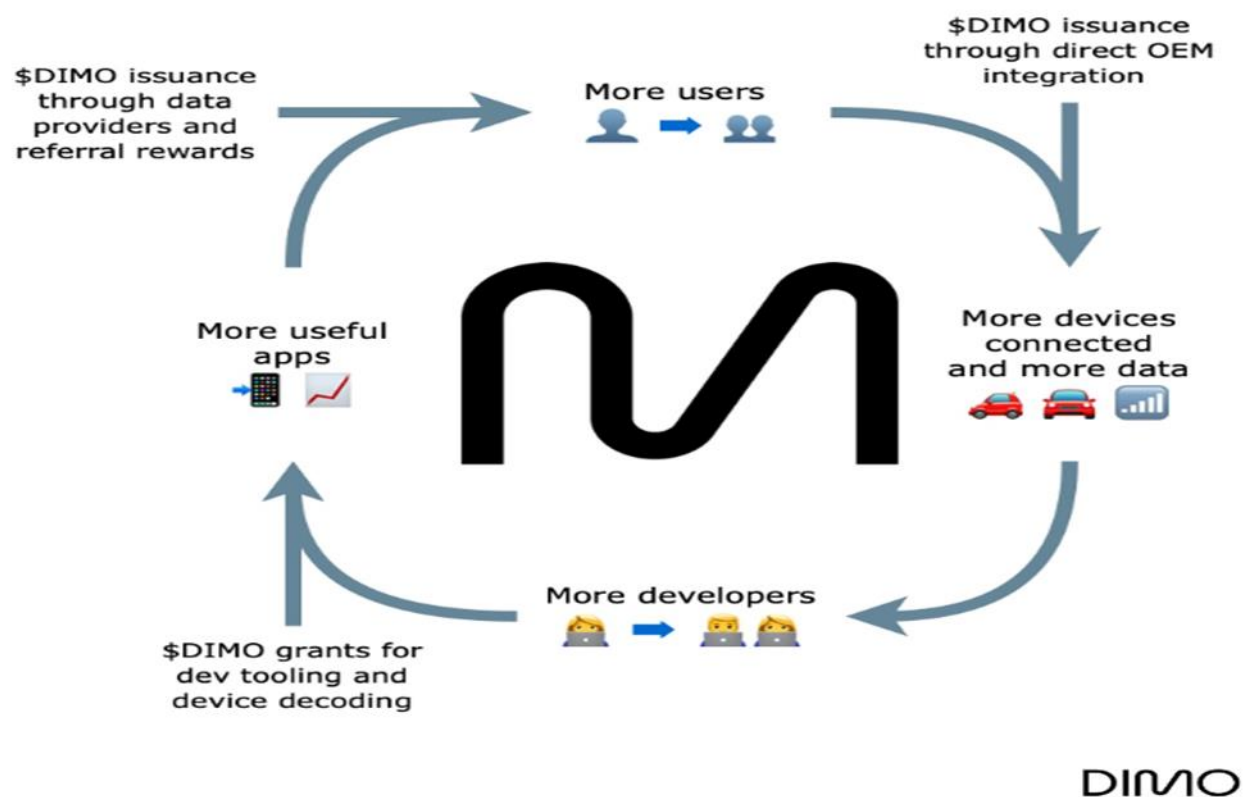


Рис. 1.2. Схема розвитку токenu з whitepaper DIMO

Однак, наразі згадок про DIMO у новинах та статтях фактично немає та продукт майже не просувається, що свідчить про низьку зацікавленість потенційних користувачів. Сам продукт наразі є на стадії розробки, тож неможливо повною мірою оцінити чи дійсно застосунок, та відповідний token користуватимуться популярністю.

Технічні характеристики DIMO наразі не демонструють позитивної динаміки. Згідно з даними сайту Coinbase найвища ціна на криптовалюту

становила 11.35 грн, 18 квітня 2023 року. З того часу ціна впала приблизно на 50% і дорівнювала 6.37 грн станом на 5 травня 2023 року. Хоча, з моменту випуску даної криптовалюти її ціна зросла на 219.76%, за період з 22 квітня по 5 травня 2023 року даний актив тримав тенденцію на втрату цінності. Також, розглянувши статистику з сайту Coinbase станом на 8 травня 2023 року (рисунок 1.3), можна побачити, що за останню добу з даною валютою було зроблено транзакції на загальну суму 9.8 мільйонів гривень, однак судячи з динаміки показника зміни ціни, це були транзакції на продаж валюти. Той самий висновок можна зробити з аналізу графіка зміни ціни за останню добу (рисунок 1.4), де видно, що вартість криптовалюти більшість часу залишалась на рівні 6.37 грн, після чого, не дивлячись на короткочасні підйоми у вартості йшла на спад.

Market stats

MARKET CAP ⓘ

UAH 564.8M

VOLUME (24H) ⓘ

UAH 9.8M

CIRCULATING SUPPLY ⓘ

87.6M DIMO

TYPICAL HOLD TIME ⓘ

Not enough data

POPULARITY ⓘ

#193

ALL TIME HIGH ⓘ

UAH 11.35

PRICE CHANGE (1H) ⓘ

-3.03%

PRICE CHANGE (24H) ⓘ

-1.56%

PRICE CHANGE (7D) ⓘ

-11.35%

Рис. 1.3. Основні характеристики криптовалюти DIMO з сайту Coinbase.com

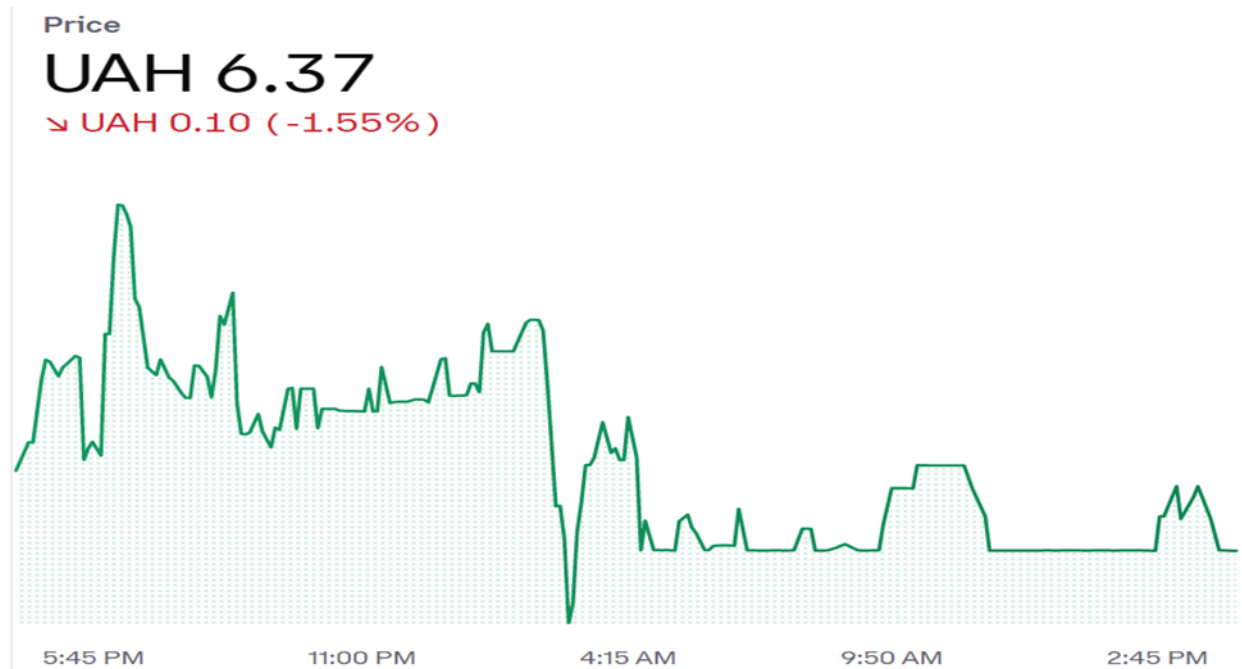


Рис. 1.4. Часовий ряд ціни криптовалюти DIMO з сайту Coinbase.com за добу

Розглянувши графік змін ціни за весь час існування DIMO (рисунок 1.5) можна зробити висновок, що дана криптовалюта все ж має перспективи щодо свого розвитку, однак, в період з 24 квітня і дотепер не демонструє позитивної динаміки росту. Отримані дані свідчать про те, що наразі DIMO не є найкращим вибором для інвестицій, оскільки існує значний ризик, що вартість даного активу продовжуватиме спадати.

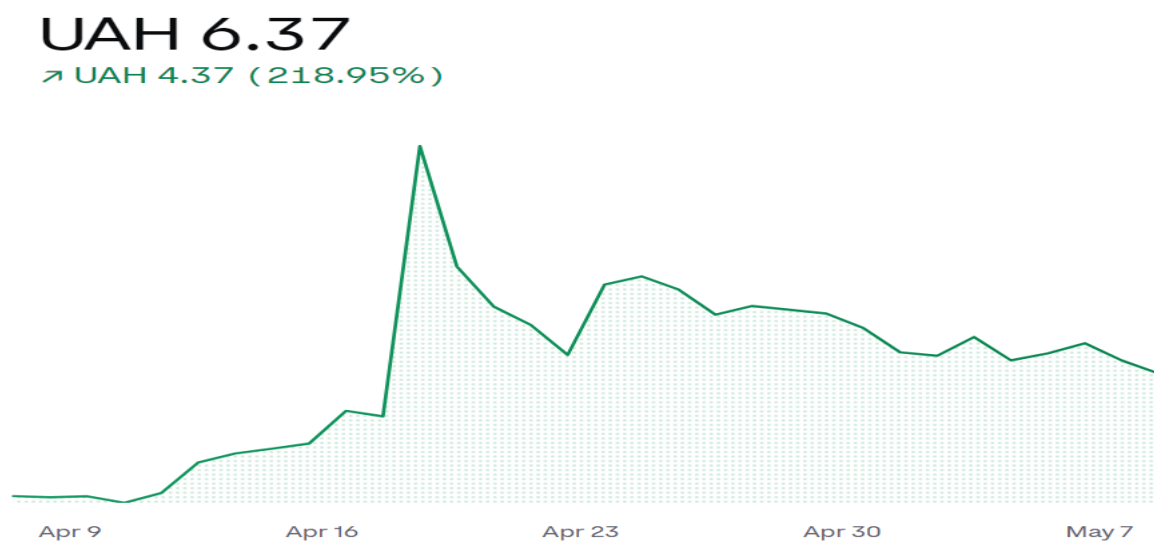


Рисунок 1.5. Часовий ряд ціни на криптовалюту DIMO з сайту Coinbase.com за весь час існування криптовалюти

З даного прикладу видно основний недолік комбінації цих двох методів – основний фокус йде на дослідження ціни криптовалюти і залежність її від різних факторів, при цьому отримаємо стратегію інвестицій у активи, перспективи зростання ціни яких будуть найкращі. Для інвестора це означатиме потенційну втрату вигідних активів, якщо на момент аналізу їхні технічні показники свідчатимуть про ймовірне падіння ціни, а також незахищеність власного гаманця від волатильності ринку криптовалют, отже шанс продати криптовалюту та не втратити значну кількість вкладених коштів досить низький.

З метою запобігання великим втратам інвестованих коштів, пропонується наступна комбінація методів – моделювання ризиків та оптимізація портфелю. За допомогою цієї комбінації можна зібрати максимально ефективний гаманець та отримати максимальний прибуток. Основним недоліком є те, що отримати повну картину ринку криптовалют користувач не зможе, оскільки буде досліджувати лише наявні у гаманці криптовалют та цифрові валюти, які можуть вигідно вписатися у портфель. В першу чергу це призводить до вимушеного зменшення кількості даних для аналізу, що з одного боку є позитивним фактором, оскільки призведе до зменшення затраченого часу на дослідження. З іншого боку, оскільки автоматизувати дану комбінацію майже неможливо, дана комбінація буде достатньо складною у застосуванні на етапі створення портфелю, особливо за відсутності досвіду роботи з криптовалютами. Також при застосуванні комбінації методів моделювання ризику та оптимізації гаманця, важливо враховувати також певні фундаментальні показники, що стосуються не лише криптовалют у гаманці, а й інших криптовалют на ринку. Це пов'язано з тим, що ціни більшості цифрових валют є корельованими випадковими величинами.

Розглянемо приклад застосування даної комбінації методів. Для збору максимально прибуткового портфелю розглядаються наступні криптовалюти – Bitcoin, Ethereum, Ethereum 2, Binance Coin, Stacks, Jupiter, DEXTools та

Kava. Перші чотири варіанти досить давно на ринку і вже мають певну репутацію, останні ж просто показують тенденцію зростання ціни. За допомогою сервісу Cryptowatch можна розглянути кореляційну матрицю цін даних криптовалют за останній місяць (рисунок 1.6).



Рис. 1.6. Кореляційна матриця криптовалют з сайту Cryptowatch.com

Бачимо, що ціни більшості валют корелюють між собою і мають досить високе значення коваріації. Наразі, інвестору краще обирати до гаманця криптовалюту з найменшою кореляцією, оскільки це допоможе запобігти ризику банкрутства при падінні ціни одного з активів, а також віртуальні валюти з від'ємною кореляцією, оскільки вклавши гроші одночасно в обидві, є шанс дещо захистити свої кошти та компенсувати втрати при падінні ціни однієї з них. Отже, серед розглянутих криптовалют варто звернути увагу на Stacks, Jupiter, DEXTools і Kava. Жодна з цих валют не спричинить сильних збитків для інших у гаманці, якщо зазнає падіння у ціні.

В даному прикладі, серед аналізу технічних показників, увага була сконцентрована на кореляційній матриці, для того щоб зрозуміти ризики від купівлі тієї чи іншої комбінації криптовалют, однак врахувати залежність обраних криптовалют від усіх інших представлених на ринку вкрай важко, тож потрібно залучати й інші засоби аналізу, зокрема попередньо визначитися з

набором криптовалют на основі, наприклад, фундаментальних даних. В комбінації з методом оптимізації гаманця це є доволі затратним, оскільки ціни на криптовалюту швидко змінюються, тож продавши актив при падінні ціни дослідження варто повторити. Крім того, виникає питання за якою логікою краще відбирати криптовалюту для дослідження, адже можна отримати програшну комбінацію, навіть обравши криптовалюту, що показують позитивну динаміку на момент дослідження. Крім того, стратегія, яку ми отримаємо в результаті дослідження є досить ненадійною, оскільки не дозволяє максимізувати прибуток від швидкого продажу активів та потребує оперування великою кількістю даних, що є технічно складним у реалізації.

Досить вигірною є комбінація методів управління ризиками та аналізу технічних показників. Дана комбінація дозволяє захистити гаманець від волатильності ринку криптовалют та урахувавши спосіб захисту коштів обрати найбільш вигідну торгівельну стратегію. На відміну від попередньої, комбінація цих методів дослідження ринку не вимагає попереднього аналізу даних про криптовалюту, а також дозволяє отримати повну картину ринку без дослідження фундаментальних показників на відміну від першої комбінації. До того ж дана комбінація дозволяє захистити від втрат вже інвестовані кошти шляхом застосування методів управління ризиків, наприклад стоп-лосу – автоматичного продажу криптовалюти, якщо ціна падає нижче певного рівня, який можна визначити за допомогою технічного аналізу показників. Розглянемо приклад застосування даної комбінації.

Розглядається графік зміни ціни за квітень 2023 року криптовалюти Ethereum2 (рисунок 1.7).

UAH 68,120.06

↗ UAH 1,030.41 (1.54%)



Рис. 1.7. Часовий ряд ціни криптовалюти Ethereum2 з сайту Coinbase.com за квітень 2023

За даним графіком, можна визначити показник підтримки – значення ціни, при досягненні якої припиняється спад та починається зростання та опірний рівень – значення ціни, при досягненні якої припиняється зростання та починається спад. За даними сайту Coinbase, показник підтримки цього місяця становить 66930 грн, а показник опору 74613 грн. Важливо розуміти, що ціна може підійматися вище вказаного показника опору та опускатися нижче за показник підтримки. Для інвестора важливо, щоб при падінні ціни, грошові втрати були помірними. Рекомендовано встановлювати значення стоп-лосу у вигляді 50% від очікуемого прибутку, оскільки це допоможе уникнути ситуації передчасного продажу криптовалюти, а також серйозних збитків[5]. Отже, за дохід, що очікується краще взяти середнє арифметичне показника зростання та опору – 70771,5 грн, тоді стоп-лос буде встановлено на межі 35385.75 грн. Однак, важливо зазначити, що останній раз, коли Ethereum 2 настільки падав у ціні був листопад 2022 року, з того часу ціна тримає тенденцію на зростання, отже наразі дана криптовалюта є безпечним рішенням для інвестицій.

Комбінація методів управління ризиками та технічного аналізу показників є доволі вдалою, хоча має певні недоліки у застосуванні. В першу

чергу, наразі не існує єдиного оптимізованого програмного забезпечення, що дозволить швидко знаходити оптимальні рішення для роботи на ринку криптовалют. Другою причиною є високе розмаїття засобів та підходів, що можуть застосовуватися методами, частина з яких не має математичного обґрунтування в контексті криптовалют, оскільки застосовувалась лише для більш стабільних фінансових ринків. Надалі, розглядатиметься комбінація глибокого хеджування та статистичного арбітражу.

1.4.3. Обґрунтування комбінації глибокого хеджування та статистичного арбітражу

Хеджування – метод управління ризиками шляхом додавання до портфелю хеджів, тобто, позицій, яка компенсуватиме ризики ціни іншої позиції. В контексті криптовалют, хеджування є доволі важливим з огляду на високу волатильність ринку. Найпоширеніші способи реалізації хеджування – ф'ючерси, опціони та свопи. Ф'ючерс – фактично, угода між продавцем та покупцем на продаж активу у конкретну дату за фіксовану ціну, а опціон – угода, згідно якої покупець має право викупити актив за фіксованою ціною протягом певного періоду часу. Свопом є контракт між двома сторонами на обмін платежів згідно до певної формули [20].

Глибоке хеджування базується на методах глибокого навчання та навчання з підкріпленням, що дозволяє швидше обробити великі масиви даних. Цей тип хеджування може слугувати методом для вибору хеджувальної стратегії або слугувати етапом аналізу гаманця перед побудовою арбітражної стратегії. На ринку криптовалют другий варіант застосування є більш поширеним в силу складності пошуку на ринку ф'ючерсів та опціонів. Крім того, щоб знайти оптимальний хедж для криптовалют серед інших активів

потрібно витратити досить багато ресурсів, дослідити велику кількість бірж та пропозицій на них.

Глибоке хеджування є досить ефективним засобом захисту дослідження ринку криптовалют, оскільки допомагає урегулювати існуючі ризики на ринку шляхом створення оптимальної комбінації криптовалют для торгівлі спреду. Крім того, оскільки глибоке хеджування має на увазі застосування методів машинного навчання, даний метод управління ризиками можна ефективно застосувати у поєднанні з методами технічного аналізу та швидше коригувати обрану стратегію роботи, що збільшить її ефективність.

Арбітраж є типом торгівлі, коли трейдер купляє актив на ринку за низькою ціною та перепродає його на іншому ринку за вищою ціною. Статистичний арбітраж є класом короткострокових стратегій, що застосовують кількісний та обчислювальний підхід до торгівлі активами [20]. Зазвичай даний підхід до торгівлі застосовується наступним чином – досліджуються показники кореляції та коінтеграції акцій, а також стаціонарність їхніх часових рядів та, в результаті приймається рішення про купівлю відповідних пар цінних паперів. Далі, після короткотривалого зберігання, папери продаються на біржах, де ціна на них на даний момент найвищі. Даний спосіб застосування потребує велику кількість ресурсів та наближений до методів оптимізації гаманця, що робить його фактично непридатним для криптовалют.

Оскільки статистичний арбітраж являє собою цілий клас торгівельних стратегій, у контексті криптовалют його можна застосувати для визначення торгових можливостей на ринку [10]. Проблемою є лише вибір стратегії за допомогою якої буде проводитися дослідження, оскільки не кожна з арбітражних стратегій може бути реалізована на ринку криптовалют. Крім того, деякі стратегії, як наприклад торгівля парами, для ринку, що розглядається мають очевидно низьку ефективність.

Отже, застосування на ринку криптовалют статистичного арбітражу відбувається наступним чином: спершу проводиться попереднє дослідження

даних, далі на основі вхідних даних відбувається глибоке хеджування, в ході якого знаходяться вагові коефіцієнти хеджування та будується часовий ряд спреда. Далі, обирається стратегія генерації торгових сигналів, на ринку криптовалют найкраще себе показують стратегії з урахуванням значень стоп лосу. Проводиться, власне, генерація сигналів та розглядається ефективність стратегії за допомогою певних метрик.

Основними перевагами даної комбінації методів статистичного арбітражу та глибокого хеджування є:

- Контроль ризиків – за допомогою глибокого хеджування враховується вплив волатильності ринку криптовалют на ціни активів, та регулюється за допомогою похідних засобів хеджування.
- Додатковий прибуток – статистичний арбітраж є класом короткострокових торгівельних стратегій, що, як правило, є більш ефективними на криптовалютному ринку. Недоліком є те, що ринок за той самий короткий період часу може впасти у ціні, однак в такому випадку спрацює хедж, що дозволить заробити на арбітражних можливостях.
- Оптимізація затрачених ресурсів – комбінація статистичного арбітражу та глибокого хеджування дає можливість визначити стратегію для роботи на ринку без досліджень фундаментальних показників та технічно складних обчислень.
- Врахування тренду ринку – дана комбінація методів застосовує статистичні моделі прогнозування цін на ринку, на основі отриманих даних відбувається побудова стратегії, що економить затрачений дослідником час на попереднє дослідження цін.

Отже, завдання комбінації є наступним – з допомогою глибокого хеджування засобів глибокого хеджування виконати прогнозування цін на криптовалюту, та отримати коефіцієнт хеджування, який в подальшому буде використаний для статистичного арбітражу. На основі результатів глибокого

хеджування створюється арбітражна стратегія та проводиться оцінка її якості для визначення ефективності застосування комбінації на ринку криптовалют.

Висновки

У першому розділі були розглянуті основні поняття ринку криптовалют, перспективи його розвитку та ризики, пов'язані з інвестиціями у даний ринок. Основна проблема, яка є на ринку – волатильність, внаслідок чого, криптовалюти можуть різко впасти або зрости в ціні, через що важко визначитись з торгівельною стратегією. З огляду на це постає питання ефективності торгівельних стратегій, що застосовуються для традиційних ринків, на крипторинку. Для дослідження цього питання застосовується безліч стратегій дослідження ринку криптовалют, які мають свої переваги та недоліки. Однак, використання кожної з цих стратегій окремо не приносить значного результату, оскільки лише частково вирішує питання та потребує додаткових досліджень. Тому, зазвичай застосовують комбінації даних методів, що дозволяє комплексно підійти до вирішення проблеми. Різні комбінації методів пропонують різний підхід та застосовують різні засоби для аналізу, однак з математичної точки зору найбільш ефективним є поєднання методів управління ризиками та технічного аналізу.

Оскільки ринок криптовалют містить у собі активи, що більшою мірою пов'язані між собою то найкращими методами для аналізу будуть ті, що застосовують ці зв'язки, тобто враховують коваріацію. Однак, в одному з розглянутих прикладів було показано, що лише коваріації не достатньо для розробки ефективної стратегії трейду, та відповідно недостатньо для висновків про можливості ринку. Тому, було вирішено серед засобів технічного аналізу зупинитися на статистичному арбітражі, оскільки в даному випадку будуть враховані зв'язки між криптовалютами, та проведені певні

коінтеграційні тести запропонованої стратегії, що допоможе визначити чи є вона ефективною для даного набору активів. Методи управління ризиком, власне, вирішують питання впливу волатильності ринку на гаманець. Було обрано глибоке хеджування, оскільки з його допомогою можна швидко проаналізувати наявний гаманець та визначити оптимальні значення коефіцієнтів хеджування, що необхідні для подальшої роботи зі статистичним арбітражем.

Наступний розділ буде присвячено розгляду математичного апарату, стратегії реалізації глибокого хеджування та статистичного арбітражу та огляд основних метрик, що будуть застосовані для перевірки ефективності обраної комбінації методів дослідження на ринку криптовалют.

2 ПРИНЦИП РЕАЛІЗАЦІЇ СТАТИСТИЧНОГО АРБІТРАЖУ ТА ГЛИБОКОГО ХЕДЖУВАННЯ НА РИНКУ КРИПТОВАЛЮТ

2.1. Основні математичні визначення

- Випадковим процесом $(X(t), t \in T) = (X(t, \omega), t \in T, \omega \in \Omega)$, називають сімейство випадкових величин, що задане на ймовірнісному просторі (Ω, A, P) та параметризоване параметром $t \in T$, $\omega \in \Omega$ – елементарна подія. Випадковою функцією $\zeta(\omega, t), t \in T$, називають вимірне відображення $\zeta: \Omega \rightarrow \mathbb{R}^n$ простору елементарних подій, що залежить від параметру t . T є відрізком числової осі, параметр t при цьому інтерпретується як час
- Часовий ряд – це ряд точок даних, проіндексованих у хронологічному порядку, що характеризує поведінку об’єкта (або процесу) протягом певного періоду часу. Позначається як $x_t = \{x_1, \dots, x_n\} = \{x_\tau \mid \tau \in T\}$. Інтервальним часовим рядом називають такий ряд, де з даних випливає тенденція розвитку об’єкта за певний період часу.
- Введемо припущення, що часовий ряд x_t можна представити у вигляді авторегресійного процесу AR(p):

$$x_t = a_1 x_{t-1} + \dots + a_p x_{t-p} + e_p, \quad e_p \sim N(0, \sigma^2)$$

Характеристичне рівняння цього процесу запишемо у наступному вигляді:

$$m^p - a_1 m^{p-1} - \dots - a_p = 0$$

Якщо існує r одиничних коренів цього рівняння, то ряд є інтегрованим порядку r , $r \geq 1$. Записується це позначення як $I(r)$.

- Стационарність за коваріацією, або слабка стационарність, означає, що для стохастичного процесу, що має скінченне середнє та дисперсію, $\forall k, k-s$ виконуються наступні умови:

- Незалежність математичного сподівання від часу:

$$E\{x_k\} = E\{x_{k-s}\} = \mu = \text{const}$$

- Дисперсія є постійною для всього часового інтервалу, на якому розглядається процес:

$$E\{(x_k - \mu)^2\} = E\{(x_{k-s} - \mu)^2\} = \sigma_x^2 = \text{const}$$

- Автоковаріація залишається незмінною в часі для всього часового інтервалу:

$$E\{(x_k - \mu)(x_{k-s} - \mu)\} = E\{(x_{k-j} - \mu)(x_{k-j-s} - \mu)\} = \gamma(s) = \text{const}$$

- Коінтеграція - це властивість двох і більше нестационарних випадкових часових рядів, що полягає в існуванні їх стационарної лінійної комбінації. Нехай $y_t = (y_t^1, y_t^2, \dots, y_t^k)^T$ - сукупність часових рядів, кожен з яких представляє собою I(1) процес. Такі часові ряди називаються коінтегровними, якщо існує вектор $a = (a_1, a_2, \dots, a_k)^T$, такий що часовий ряд $\varepsilon_t = a^T y_t = \sum_{i=1}^k a_i y_t^i$ є стационарним процесом. Вектор a називається коінтеграційним вектором. Після множення коінтеграційного вектора на константу властивість коінтеграції не руйнується.

2.2. Реалізація глибокого хеджування

Процес глибокого хеджування полягатиме у тому, щоб спершу методами машинного навчання зробити прогнозування майбутніх цін на криптовалюту, а далі, застосовуючи лінійну регресію знайти коефіцієнти хеджування та дослідити комбінацію часових рядів криптовалют на

стаціонарність. Останнє є дослідженням контеграції, яке зазвичай проводиться безпосередньо під час розробки арбітражної стратегії, тому глибоке хеджування в деякому сенсі міститиме перехідний етап до подальшого арбітражу.

2.2.1. Структура і принцип роботи нейронної мережі CNN-LSTM

Нейронна мережа CNN-LSTM є комбінацією двох мереж – згорткової нейронної мережі (Convolutional Neural Network або CNN) та мережі з довгою короткочасною пам'яттю (Long Short-Term Memory або LSTM) [6, 17, 19]. Модель прогнозування, що будується на основі даної нейронної мережі має у собі властивості обох мереж з комбінації, що дозволяє на основі аналізу часових рядів визначити основні залежності та на їх основі, власне, створити прогноз.

Для того щоб детальніше розглянути можливості даної нейронної мережі потрібно розуміти характеристики, які вона має від своїх складових та її структуру. CNN-LSTM, завдяки функціоналу згорткової нейронної мережі, а саме її локальному персептрону та розподілу ваг, можна значно зменшити кількість параметрів, що розглядаються, що є досить важливим для нових криптовалют, про які платформи ще не мають достатню кількість даних. За допомогою функцій рекурентної моделі LSTM отримуємо можливість обробити послідовність даних за певний термін, виявити залежності, що є довготривалими, та визначити тенденції на основі яких побудувати прогноз.

Схема роботи даної нейронної мережі є наступною:

- Спершу вхідні дані нормалізуються за допомогою формули 2.1:

$$y_i = \frac{x_i - \bar{x}}{s} \quad (2.1)$$

Тут, y_i – вибіркові значення, x_i – вхідні значення, $\bar{x}$ – вибіркове середнє, s – середньоквадратичне відхилення

Оскільки дані подаються у вигляді дискретної вибірки, вибіркове середнє та середньоквадратичне відхилення знаходиться за формулами 2.2 та 2.3:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.2)$$

$$s = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.3)$$

- Ініціалізація моделі, та встановлення параметрів.
- Робота нейронної мережі CNN. Спершу відбувається згортка, замість матричного добутку, в результаті чого на кожному згортковому шарі, можна виявити одну спільну властивість вхідних даних, внаслідок чого вхідні дані «фільтруються» та залишається менша кількість параметрів, що пришвидшує подальшу роботу моделі. Далі, відбувається пулінг – фактично зменшення розмірності вхідних даних. Процеси згортки та пулінгу відбуваються за допомогою формул 2.4 та 2.5 відповідно:

$$y_j^l = f \left(\sum_{i \in M_j} x_i^{l-1} k_j + b_j^l \right) \quad (2.4)$$

Тут, f – активаційна функція, M – згорткове ядро, x_i^{l-1} – вхідна вибірка шару l , k_j – ваги матриці згорткового ядра, b_j^l – параметр зміщення шару l .

$$x_j^l = g(\beta_j^l \text{pooling}(x_j^{l-1}) + b_j^l) \quad (2.5)$$

Тут, g – активаційна функція, β_j^l – коефіцієнт каналу відповідного шару пулінгу l , x_i^{l-1} – вхідна вибірка, pooling – функція пулінгу, b_j^l – параметр зміщення.

- Робота нейронної мережі LSTM. На даному етапі отримані результати роботи мережі CNN через три гейти – структури, які регулюють потік інформації всередині нейронної мережі.

Спершу, отримані дані проходять через гейт забуття (forget gate) – шар, який визначає, яка інформація має бути забута мережею, а яка повинна залишитися, присвоюючи значення 0 або 1 відповідно. Даний процес описується за формулою 2.6:

$$f_t = \sigma(W_f x_t + W_f h_{t-1} + b_f) \quad (2.6)$$

У цій формулі σ – сигмоїдна функція, W_f – ваги гейту забуття, x_t – вхідні дані шару, h_{t-1} – вихідні дані шару, b_f – параметр зміщення гейту забуття.

Далі інформація надходить до гейту оновлення – цей шар вирішує, яку нову інформацію слід додати до виходу, тобто робить прогнозування. На даному шарі, спершу обчислюються вхідні дані процесу за формулою 2.7, потім за допомогою 2.8 розраховується вектор-кандидат значень, який за формулою 2.9 множиться на вектор значень i_t та додається до поточного стану.

$$i_t = \sigma(W_i x_t + W_i h_{t-1} + b_i) \quad (2.7)$$

У цій формулі σ – сигмоїдна функція, W_i – ваги гейту оновлення, x_t – вхідні дані шару, h_{t-1} – вихідні дані шару, b_i – параметр зміщення гейту оновлення.

$$\bar{C}_i = \tanh(W_c x_t + W_i h_{t-1} + b_c) \quad (2.8)$$

Тут W_c – ваги вектора-кандидата, x_t – вхідні дані шару, h_{t-1} – вихідні дані шару, b_c – параметр зміщення вектора-кандидата.

$$C_t = f_t C_{t-1} + \bar{C}_i i_t \quad (2.9)$$

Оновлений вектор містить значення від 0 до 1 та надходить до останнього гейту – вихідного. Всередину даного шару за допомогою формул 2.10 та 2.11 визначається, які саме зі значень C_t після обробки функцією гіперболічного тангенса будуть додані до фінального виводу.

$$o_t = \sigma(W_o x_t + W_i h_{t-1} + b_o), \quad (2.10)$$

де σ – сигмоїдна функція, W_o – ваги вихідного, x_t – вхідні дані шару, h_{t-1} – вихідні дані шару, b_p – параметр зміщення вихідного гейту.

$$h_t = o_t \tanh(C_t) \quad (2.11)$$

- Отриманий вектор h_t порівнюється з заданими нормованими значеннями та обчислюються відповідні похибки для перевірки роботи моделі. Якщо ці похибки вищі від заданого значення процес повторюється.
- Відновлення початкових даних – якщо процес роботи моделі завершено, то отриманий вектор даних відновлюється від нормованих до початкових даних за формулою 2.12:

$$x_i = h_i s + \bar{x} \quad (2.12)$$

Відновлений вектор i буде результатом прогнозування. Важливо розуміти, що цей вектор є представленням часового ряду, оскільки дані відображаються послідовно та протягом певного періоду часу.

Дана схема описана згідно рисунка 2.1, схеми нейронних мереж CNN та LSTM зображені на рисунках 2.2 та 2.3 відповідно.

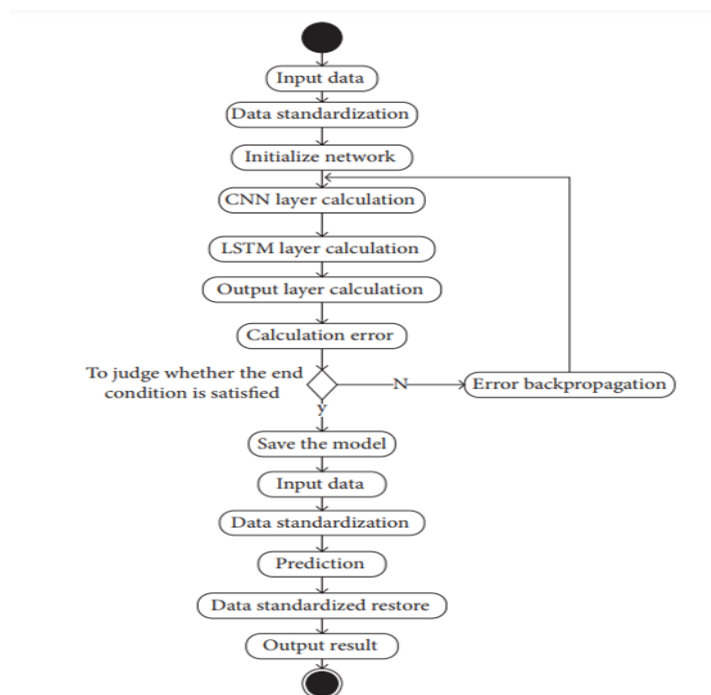


Рис. 2.1. Принцип роботи нейронної мережі CNN-LSTM

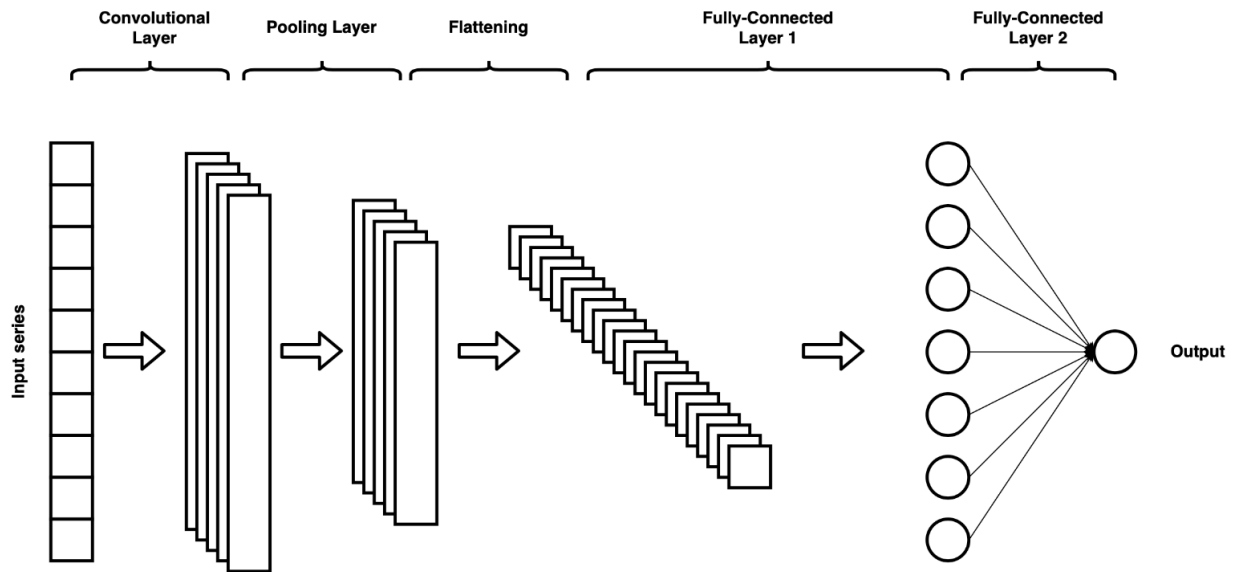


Рис. 2.2. Структура нейронної мережі CNN

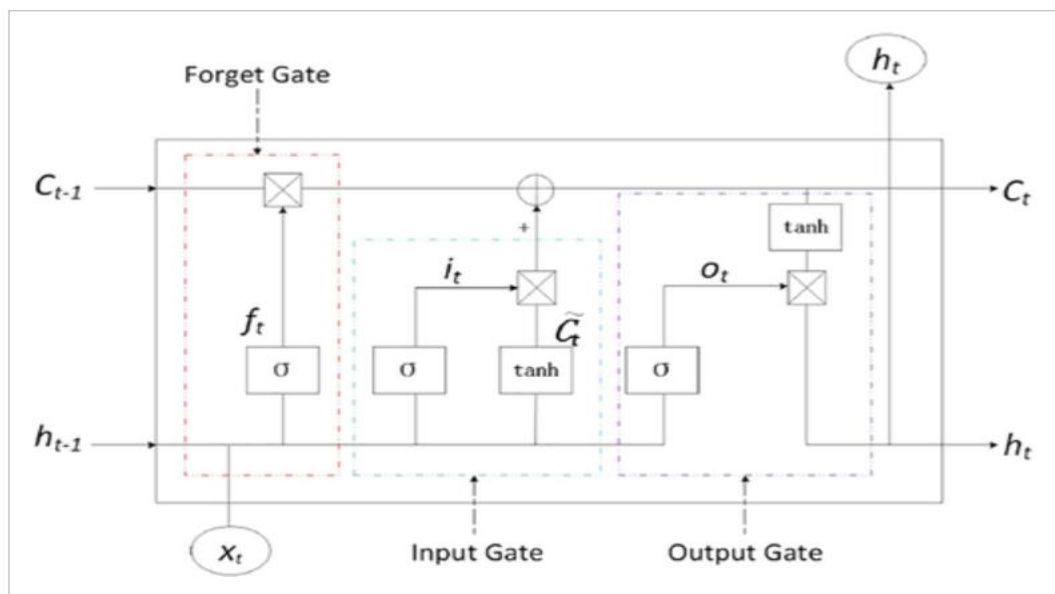


Рис. 2.3. Структура нейронної мережі LSTM

2.2.2. Дослідження коінтеграції

Для подальшого створення та дослідження арбітражної стратегії на ринку криптовалют, важливо сформувати лінійну комбінацію групи активів,

що досліджуються та встановити оптимальні вагові коефіцієнти хеджування для портфелю, а також визначити чи може такий набір активів взагалі бути використаний для торгівлі. Крім того, для подальшої роботи ми маємо визначити чи є обрана комбінація криптовалют стаціонарною, оскільки окремі часові ряди цифрових валют такої властивості не мають.

Для цього скористаємося поняттям коінтеграції – статистичного методу, який використовується для визначення довгострокового зв'язку між двома або більше фінансовими рядами. Дане поняття є навіть більш важливим ніж кореляція, оскільки кореляція не виявляє недостатньо узгоджений рух цін між активами, що, як вже було описано раніше заважає визначити вагові коефіцієнти хеджування [15]. Зазвичай тести, що досліджують кореляцію застосовують вже при побудові арбітражної стратегії, однак для роботи на ринку криптовалют коінтеграція є важливою для проектування портфелю, що досліджується, тому коінтеграційні тести проводяться в рамках глибокого хеджування[9, 11].

Для визначення коінтеграції між прогнозованими часовими рядами криптовалют застосуємо метод Енгла-Грейнджера. Даний підхід полягає у тому, щоб спершу до заданих часових рядів $I(1)$ застосувати множинну лінійну регресію для визначення коефіцієнтів хеджування β , а потім перевірити лишки побудованої комбінації на стаціонарність.

Для перевірки на стаціонарність застосуємо стаціонарні тести Квятковського-Філіпса-Шмідта-Шина та доповнений тест Дікі-Фуллера. Перший тест має суттєвий недолік – ризик похибки першого роду, тобто відхилення нульової гіпотези на заданому рівні значущості. При цьому, зміна рівня значущості перевірки гіпотез призводить до зменшення точності тесту, тому його часто поєднують з другим та порівнюють результати.

Отже, тест Квятковського-Філіпса-Шмідта-Шина та доповнений тест Дікі-Фуллера мають схожий принцип роботи, який полягає у представленні часових рядів, що розглядаються у вигляді суми детермінованого тренду,

процесу випадкового блукання та стаціонарного залишку. Дану залежність можна записати з використанням результатів лінійної регресії:

$$y_t = \alpha + \beta x_t + e \sim I(0)$$

Тут, β – вектор коефіцієнтів хеджування, y_t, x_t – часові ряди, що розглядаються, e – стохастичний член.

Для даної залежності потрібно знайти такі α, e щоб у результаті склався стаціонарний процес. Однак, пошук таких значень є можливим не для кожної групи часових рядів, що, власне, і перевіряється тестами.

Основною відмінністю є система гіпотез, що розглядаються. Для розширеного тесту Дікі-Фуллера за основну гіпотезу приймається наявність одиничного кореня і, як наслідок, відсутність стаціонарності. Для тесту Квятковського навпаки, нульовою гіпотезою буде відсутність одиничного кореня та наявність стаціонарності.

Після проведення тестів, наступні кроки дослідження залежатимуть від результатів. Нижче наведені основні варіанти [16]:

- Обидва тести вказують на стаціонарність. Отже, розглянуті часові ряди коінтегрують та їх лінійна комбінація є стаціонарною. Можна переходити до застосування трейдингових стратегій.
- Обидва тести вказують на відсутність стаціонарності. Комбінація даних часових рядів не є стаціонарною.
- Тест Квятковського вказує на стаціонарність, а доповнений тест Дікі-Фуллера на її відсутність. У такому випадку лінійна комбінація часових рядів вважається стаціонарною за трендом. Методами машинного навчання тренд прибирається і оновлена комбінація знову перевіряється на стаціонарність.
- Тест Дікі-Фуллера вказує на стаціонарність, а тест Квятковського на відсутність стаціонарності. У цьому випадку для подальших досліджень приймається стаціонарність комбінації.

Отже, по результатам проведених тестів, можемо зробити висновок щодо розподілу коефіцієнтів хеджування – чи є він ефективним для зменшення ризиків пов'язаних з портфелем та чи можна застосувати даний портфель для торгівлі.

2.3. Статистичний арбітраж як метод торгівлі криптовалютами

Найкраще на ринку криптовалют працюють арбітражні стратегії з використанням спреду [12-13]. Спредом, у випадку криптовалют, називають наступний стаціонарний процес

$$s_t = y_t - \beta x_t$$

Тут β – коефіцієнти хеджування, y_t – часовий ряд залежної криптовалюти, x_t – часові ряди всіх інших криптовалют у гаманці.

Застосування класичного підходу парної торгівлі на ринку криптовалют має доволі обмежені можливості, оскільки пріоритетним буде розгляд криптовалют з одного сімейства, тих що випускаються однією компанією, для таких криптовалют проведена попередньо робота з хеджування може нівелюватися непередбачуваними зовнішніми факторами, які вплинуть на все сімейство, відповідно максимізувати прибуток не вийде. Крім того, для дослідження недоцільно використовувати лише парні набори криптовалют, оскільки це не відображатиме реальну картину і не дасть змоги інтерпретувати результати дослідження на весь ринок.

Ефективною стратегією з застосуванням спреду є стратегія відкриття довгих та коротких торгових позицій, для максимізації прибутку [18]. Для генерації торгових сигналів у дослідженні скористаємось моделлю GARCH [14, 27].

GARCH (Generalized AutoRegressive Conditional Heteroscedasticity) – модель, що застосовується для аналізу часових рядів, умовна дисперсія яких залежить від попередніх значень ряду, а також від попередніх значень умовних дисперсій ряду. Для ринку криптовалют дана модель є доволі важливою, оскільки дозволяє враховувати періоди з більш високою та більш низькою волатильністю та їх вплив на торгівельну стратегію.

Модель GARCH описується за допомогою формули 2.13:

$$\sigma^2(t) = a\sigma^2(t-1) + be^2(t-1) + W, \quad (2.13)$$

Де W – початкова дисперсія процесу, a, b – коефіцієнти GARCH-процесу, їхня сума має бути строго менша за 1 інакше модель – нестабільна, $\sigma^2(t-1)$ – попередня дисперсія процесу, $e^2(t-1)$ – попередній стандартизований шок процесу.

При цьому, стандартизований шок процесу задається за допомогою 2.14:

$$e(t) = \sqrt{\sigma^2(t)}z_t \quad (2.14)$$

Тут, z_t – випадкова змінна з нульовим математичним сподіванням та одиничною дисперсією.

Цими рівняннями описується окремий випадок моделі GARCH(1,1), який є найчастіше використовується у фінансовому контексті. На основі умовної дисперсії у момент часу t розраховуються значення для стоп-лосу у вигляді:

$$mean_{spread} \pm 2\sigma_t$$

Більше значення – означає точку при перетині якої відкривається коротка позиція, менше – довга позиція. $mean_{spread}$ у цьому випадку є середнім значенням спреду.

Застосування GARCH дозволяє моделювати значення динамічно – у вигляді часового ряду, на відміну від іншого методу – процесу Орнштейна-Уленбека, який є часто вживаним для більш стандартних ринків. Саме це і дозволяє врахувати волатильність ринку на основі гаманця, що використовується під час дослідження та дозволяє розширити можливості для

арбітражу. Крім того це дозволяє провести дослідження асиметрії коливань ринку – модель здатна реагувати як на позитивні так і на негативні зміни на ринку і змінювати значення стоп-лосу в залежності від загальної волатильності гаманця в певний момент часу, що дозволяє максимізувати можливий прибуток.

2.4. Показники для оцінювання якості проведеного дослідження

Якість проведеного дослідження перевірятимемо у два етапи – спершу оцінимо якість роботи моделі CNN-LSTM, яку застосуємо для прогнозування цін на криптовалюти, а потім буде оцінено якість роботи арбітражної стратегії на ринку криптовалют. По результатам оцінювання якості прогнозування можна буде зробити висновки щодо надійності даного дослідження, а по результатам оцінювання роботи арбітражної стратегії зробити висновки щодо торгових можливостей на ринку криптовалют. Крім того, оцінювання стратегії також дозволить зробити певні висновки про можливості глибокого хеджування на ринку криптовалют.

2.4.1. Оцінювання роботи нейронної мережі CNN-LSTM

Для оцінювання якості роботи нейронних мереж застосовується досить багато показників. Ми скористаємось лише основними – середньоквадратичною помилкою, середньою абсолютною помилкою та коефіцієнтом детермінації [22-24].

Середньоквадратична помилка (Mean Squared Error, MSE) – цей показник застосовується для вимірювання відхилень між прогнозованими

значеннями моделі та фактичними значеннями даних. Чим меншим є значення середньоквадратичної помилки, тим кращою вважається робота моделі. MSE обчислюється за формулою 2.15:

$$MSE = \frac{\sum (y_i - \hat{y}_i)^2}{n}, \quad (2.15)$$

Де y_i – значення i -того спостереження, $\hat{y}_i$ – прогнозоване значення i -того спостереження, n – кількість спостережень.

Дана похибка, фактично, оцінює дисперсію між фактичними та прогнозованими значеннями. Це дозволяє впевнитися в тому, що значення даної похибки завжди буде більшою за 0. Нульова середньоквадратична похибка свідчить про ідеальність побудованої моделі, однак на практиці це трапляється вкрай рідко.

Середня абсолютна помилка (Mean Absolute Error, MAE) – даний показник вимірює середню абсолютну різницю між прогнозованими та фактичними значеннями моделі. Обчислюється дана похибка за формулою 2.16:

$$MAE = \frac{\sum |y_i - \hat{y}_i|}{n} \quad (2.16)$$

Очевидно, обидві помилки є доволі схожими за своїм апаратом обчислення, однак їхні результати вказують на різні проблеми, що може мати побудована модель прогнозування. При високих значеннях MSE можна зробити висновок, що модель робить прогнози, розбіжності яких з реальними даними є досить великими, та вплив цих розбіжностей на якість прогнозів є доволі суттєвим. MAE допомагає зрозуміти якою є середня абсолютна різниця між прогнозованими та фактичними значеннями, крім того її легше інтерпретувати для конкретного датасету. Як і у випадку з середньоквадратичною помилкою, нижчі значення MAE свідчать про кращу роботу моделі, однак важливо спиратися і на середнє значення прогнозування по датасету, наприклад, якщо середнє значення даних у датасеті 170, а значення середньої абсолютної помилки 7.71 це свідчить про високу точність

моделі, тому для розуміння результатів даної похибки потрібно також обчислювати і середнє арифметичне вхідних даних.

Коефіцієнт детермінції (R-Squared) – даний показник вимірює якість навчання моделі та показує наскільки частки дисперсії між значеннями залежних змінних пояснюється дисперсією між значеннями незалежних змінних у загальній дисперсії моделі, тобто фактично пояснює наскільки добре модель працює на даних. Коефіцієнт детермінації обчислюється за формулою 2.17:

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - y_{mean})^2} \quad (2.17)$$

У цій формулі y_{mean} – вибіркове середнє датасету.

Значення коефіцієнту детермінації знаходиться в межах від 0 до 1, при цьому чим вище значення цього показника, тим краще модель працює на даних. Мінімальним значенням коефіцієнту детермінації, при якому ми можемо вважати, що модель працює добре є 0.5.

Отже, за допомогою цих показників можна оцінити якість роботи моделі та зробити висновки про надійність проведеного дослідження, оскільки нейронна мережа CNN-LSTM застосовується для прогнозування цін на криптовалюту, тобто для першого кроку глибокого хеджування, відповідно, якщо якість роботи моделі буде низькою, подальше дослідження також можна вважати ненадійним. Для того щоб уникнути такої проблеми варто провести аналіз обраного датасету та визначити, чи всі фактори потрібно використовувати для прогнозування цін та зробити певні зміни у даних, наприклад, прибрати непотрібні стовпці або нормувати датасет [21].

2.4.2 Оцінювання якості арбітражної стратегії

Оцінювання якості алгоритмічної стратегії базуватиметься на коефіцієнті Шарпа, побудові кумулятивного прибутку та просідання гаманця [25]. Дані методи є найбільш часто застосованими при оцінюванні торгової стратегії та дозволяють зробити висновки щодо прибутковості стратегії протягом короткого періоду часу, що є доволі важливим для криптовалют, зберігання яких у гаманці протягом довгого часу є ненадійним з огляду на високий вплив зовнішніх факторів на ціни.

Коефіцієнт Шарпа вимірює середню прибутковість, яку отримує інвестор за кожен одиницю ризику(волатильності) з урахуванням безризикової ставки повернення. Даний показник розраховується за допомогою формули 2.18:

$$S = \frac{E[R - R_f]}{\sigma} = \frac{E[R - R_f]}{\sqrt{Var[R - R_f]}} \quad (2.18)$$

Тут, R – це прибутковість портфеля, R_f – це дохідність від альтернативного вкладення.

За альтернативне вкладення приймається портфель активів з нульовим ризиком, так звана безризикова процентна ставка, це можуть бути покупка векселів, для яких очікуваною дохідність є безризикова процентна ставка. Важливо зазначити, що на ринку криптовалют знаходження такого портфелю є досить складною задачею, оскільки через непередбачуваність ризиків неможливо стверджувати напевне, що обраний хедж-фонд дійсно знизить ризики пов'язані з портфелем. Однак, глибоке хеджування в деякому сенсі допомагає вирішити цю проблему, оскільки дозволяє знайти оптимальний розподіл акцій у гаманці та динамічно його змінити враховуючи зміни.

Оптимальним значенням коефіцієнту Шарпа, що означатиме прибутковість стратегії, має бути значення що перевищує 1. Значення 2

говорить про те, що стратегія є прибутковою щомісяця, 3 – щодня. Застосування коефіцієнта Шарпа для оцінювання ефективності торгівельної стратегії на ринку криптовалют має певні переваги, основною з яких є врахування загальної волатильності гаманця, тобто не розрізняє окремо коливання активів вгору або вниз, що дозволяє оцінювати саме прибутковість, а не ризик.

Кумулятивний прибуток – метрика, що відображає зростання прибутку(або його втрати) з часом, в результаті використання торгівельної стратегії. Дана метрика відображає загальну накопичену суму прибутку з урахуванням усіх попередніх операцій. Для обчислення кумулятивного прибутку необхідно мати дані про доходність або вартість портфеля на різних часових періодах.

Розрахунок кумулятивного прибутку проводиться за допомогою формули 2.19:

$$R_c = \frac{P_{current} - P_{initial}}{P_{initial}} \quad (2.19)$$

Тут, $P_{current}$ – поточна вартість гаманця, $P_{initial}$ – початкова вартість гаманця. Оскільки поточна вартість гаманця обчислюється за результатами прогнозів цін на криптовалюту, важливо спершу оцінити якість роботи моделі прогнозування.

Останньою метрикою, яка буде розглядатися під час оцінювання ефективності арбітражної стратегії є просідання портфеля. Дана метрика відображає втрати, які виникають в результаті падіння ринку, а зниження ціни окремих активів у гаманці. Просідання портфеля вимірює максимальні втрати від піку до мінімальної ціни портфелю перед наступним зростанням.

Просідання портфеля зазвичай моделюється наступним чином:

- На основі історичних даних про вартість гаманця обчислюються піки та мінімуми гаманця та обчислюються втрати для кожної пари піка та мінімуму у відсотковому вигляді.

- Пошук максимального просідання портфеля, тобто максимально можливу втрату, яку може зазнати інвестор.
- Аналіз тривалості просідання. Тривалість просідання показує скільки знадобиться часу, щоб відшкодувати збитки, тобто перевищити глобальний максимум кривої просідання

За допомогою аналізу останніх двох метрик ми можемо зробити висновок не лише про ефективність арбітражної стратегії, а й про якість проведеного глибокого хеджування гаманця. Високий кумулятивний прибуток та низький відсоток просідання свідчатимуть про якісне глибоке хеджування, що дійсно мінімізує ризики пов'язані з гаманцем.

Висновки

У даному розділі було розглянуто математичний апарат комбінації глибокого хеджування та статистичного арбітражу. Прогнозування цін на криптовалюту було вирішено проводити з використанням нейронної мережі CNN-LSTM, тож було описано принцип роботи цієї мережі та розглянуто структуру кожного з її компонентів. Далі, було описано поняття коінтеграції та запропоновано апарат дослідження даної властивості. Для цього були описані коінтеграційні тести Квятковського та Дікі-Фуллера, які базуються на одному принципі, однак мають різну систему гіпотез, внаслідок чого дають кращі результати при застосуванні їх разом.

Також, було розглянуто арбітражні можливості на ринку криптовалют. Було вирішено обрати стратегію торгівлі спреду з використанням значень стоп-лосу. Дані значення знаходитимуться за допомогою генерації торгових сигналів методом GARCH, що є більш ефективним для ринку криптовалют на відміну від інших процесів. У розділі було описано модель GARCH та пояснено її застосування для пошуку значень стоп-лосу.

Наприкінці було розглянуто основні метрики якості дослідження. Було вирішено проводити оцінювання у два етапи – спершу оцінити якість роботи моделі прогнозування CNN-LSTM для висновків щодо надійності проведеного дослідження. Другим етапом є оцінка якості арбітражної стратегії за допомогою коефіцієнта Шарпа, кумулятивного прибутку та просідання гаманця. На основі останніх двох метрик можна зробити висновки щодо ефективності глибокого хеджування, оскільки арбітражна стратегія буде побудована на основі спреду з використанням вагових коефіцієнтів хеджування, отриманих на відповідному етапі.

У наступному розділі будуть представлені два датасети з історичними даними про криптовалюти - датасеті з криптовалютами, що очевидно коінтегруватимуть та на реальному користувачькому гаманці, що міститиме нові криптовалюти. Ми проведемо статистичний аналіз цих даних на основі якого ми визначимо можливі проблеми, що можуть виникнути при торгівлі, та опишемо шляхи їх вирішення з застосуванням функціоналу обраної комбінації методів. Далі розглянемо реалізацію моделі, що базуватиметься на методах статистичного арбітражу та глибокого хеджування, яка зможе вирішити виявлені раніше питання та максимізувати можливий прибуток портфелів. На основі аналізу та порівняння результатів роботи комбінації методів будуть зроблені висновки щодо ефективності та можливостей застосування даної комбінації методів на ринку криптовалют.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

3.1. Опис розробленого програмного продукту

Для розробки програмного продукту було вирішено використати мову програмування Python. Дана мова програмування є кросплатформеною, що дозволяє програмі працювати на будь-яких операційних системах. Крім того, Python має широке розмаїття бібліотек та вбудованих засобів, що полегшують роботу з машинним навчанням та дозволяють швидше реалізовувати математичні операції.

Реалізація моделі прогнозування CNN-LSTM була створена за допомогою бібліотеки tensorflow, GARCH-модель для обчислення значень стоп-лосу реалізована за допомогою бібліотеки arch, зокрема засобу arch_model. Робота з вхідними даними здійснюється за допомогою бібліотек pandas та numpy, а також методом MinMaxScaler з бібліотеки sklearn. Також, засобами останньої бібліотеки реалізований механізм множинної лінійної регресії для пошуку коефіцієнтів хеджування.

Обчислення метрик для перевірки якості роботи моделі відбувається за допомогою бібліотек numpy та спеціальної бібліотеки з фінансовими функціями – ffn, реалізація тестів Квятковського та Дікі-Фуллера реалізується відповідними засобами бібліотеки statsmodels. Така реалізація програмного продукту дозволяє зменшити час затрачений виконання програмних блоків та отримати оптимізований код для дослідження.

Для того щоб досягти кращих результатів прогнозування на основі даних, було вирішено кожен датасет розбити на окремі набори даних – для кожної криптовалюти та навчати модель CNN-LSTM на тренувальній вибірці з кожного набору даних перед прогнозуванням на тестовій вибірці. Дане рішення зумовлюється тим, що ціни на криптовалюти поводять себе досить

по-різному, хоча й можуть мати схожий тренд, крім того значення цін можуть суттєво відрізнятися, що може призвести до гіршого прогнозування. Наприклад, навчивши модель на цінах Bitcoin, можемо отримати неадекватні результати прогнозування цін на Dagecoin, оскільки останні є у 1000 разів нижчими.

Створений програмний продукт складається з файлу jupyter notebook, в якому покроково виконуються наступні дії:

- Прогнозування для кожної криптовалюти та оцінка його якості.
- Застосування лінійної регресії на основі прогнозованих даних.
- Коінтеграційні тести.
- Реалізація та оцінка арбітражної стратегії з використанням значень стоп-лосу.

З використанням платформи jupyter та обраної мови програмування, створений продукт працює на персональних комп'ютерах під управлінням різних операційних систем, зокрема Windows, Linux та MacOS.

3.2. Аналіз вхідних даних

Вхідні дані являють собою два датасети з історичними даними про криптовалюту за період з 01.12.2022 по 01.05.2023, зібраних за допомогою сервісу Investopedia. Для першого датасету були обрані криптовалюти Bitcoin, Bitcoin Cash, Wrapped Bitcoin, Ethereum, Ethereum Classic. Згідно до [29] Bitcoin та Ethereum коінтегрують, а інші криптовалюти були створені з них шляхом форків, отже мають подібну поведінку, чим пояснюється висновок про коінтеграцію і між ними. Для створення другого датасету було розглянуто реальний користувацький гаманець до якого всодять криптовалюти DEXTools, Suku, ApeCoin, Drep (new), Multichain, Sui та XMOL. Про коінтеграцію цих

криптовалют заздалегідь нічого не відомо, крім того Multichain, Sui на момент дослідження існують на ринку лише декілька днів.

3.2.1 Аналіз першого датасету

Перший датасет містить дані про криптовалюти, які існують на ринку довгий період часу і є широковідомими серед користувачів, що позитивно впливає на їхній тренд. Bitcoin та Ethereum тут є основними криптовалютами, шляхом форків від яких утворились:

- Bitcoin Cash. Дана криптовалюта з'явилась внаслідок хардфорку від Bitcoin у 2017 році.
- Ethereum Classic. Ця цифрова валюта виникла шляхом хардфорку від Ethereum у 2016 році.

Wrapped Bitcoin в певному сенсі також є форком від біткоїна, однак функціонує за іншими принципами. Фактично, дана криптовалюта створює зв'язок між Bitcoin та Ethereum та дозволяє привнести ліквідність біткоїну у децентралізовані фінансові дотатки на блокчейні Ethereum. Концепція Wrapped Bitcoin є наступною – користувачі можуть блокувати свої Bitcoin на контракті на блокчейні Ethereum, тобто отримати еквівалентну Wrapped-валюту у співвідношенні 1:1. Це означає, що за кожний Wrapped Bitcoin у обігу є один біткоїн, який був блокований та збережений у централізованому кастодіальному сховищі. Важливо розуміти, що попри близькі значення цін на Bitcoin Wrapped Bitcoin та концепцію останньої, дана криптовалюта все ж є окремим цифровим активом.

Як вже було зазначено раніше, коінтеграція Bitcoin, Ethereum та Bitcoin Cash була досліджена у[29]. Оскільки, згідно до концепції, Wrapped Bitcoin поводить себе ідентично до самого біткоїна можна говорити про те, що дана цифрова валюта також коінтегруватиме з іншими. Ethereum Classic, як

хардфорк від Ethereum, також має схожу поведінку до свого «оригіналу», що теж говорить про наявність коінтеграції.

Хоча ціни на криптовалюти, що розглядаються мають доволі схожий тренд, значення, яких вони набувають є доволі далекими. На рисунку 3.1 можемо побачити, що криптовалюти Wrapped Bitcoin та Bitcoin поведуть себе однаково, та їх ціни знаходяться в одному діапазоні.

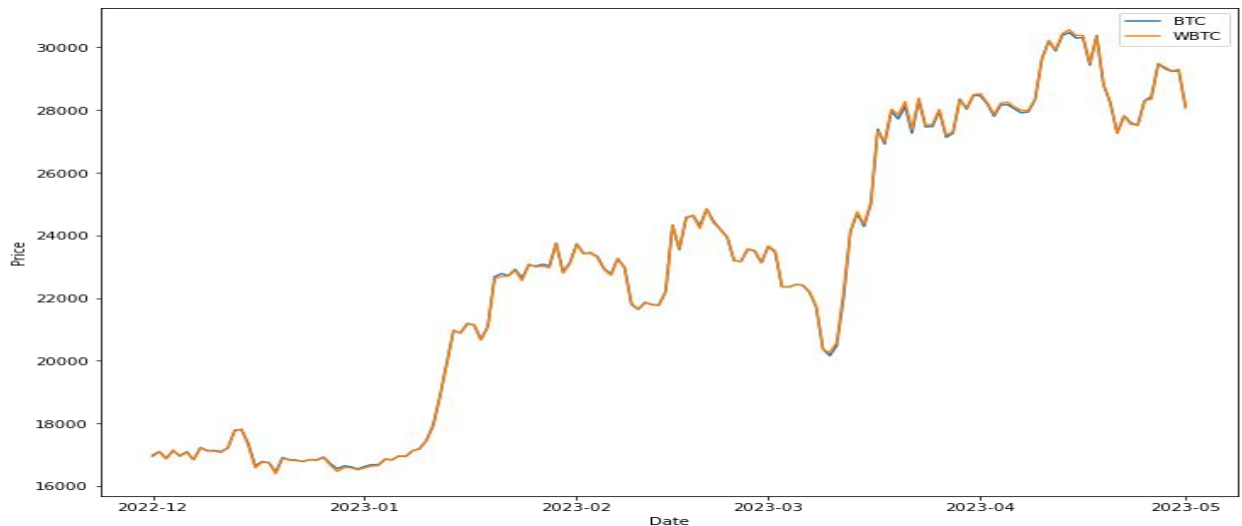


Рис. 3.1. Часові ряди цін криптовалют Bitcoin та Wrapped Bitcoin

В той самий час, розглянувши часові ряди цін на криптовалюти Ethereum (рисунок 3.2), Bitcoin Cash(рисунок 3.3) та Ethereum Classic(рисунок 3.4), можна помітити, що вони також мають тренд, подібний до перших двох криптовалют, однак значення, що приймають ціни є значно нижчими.

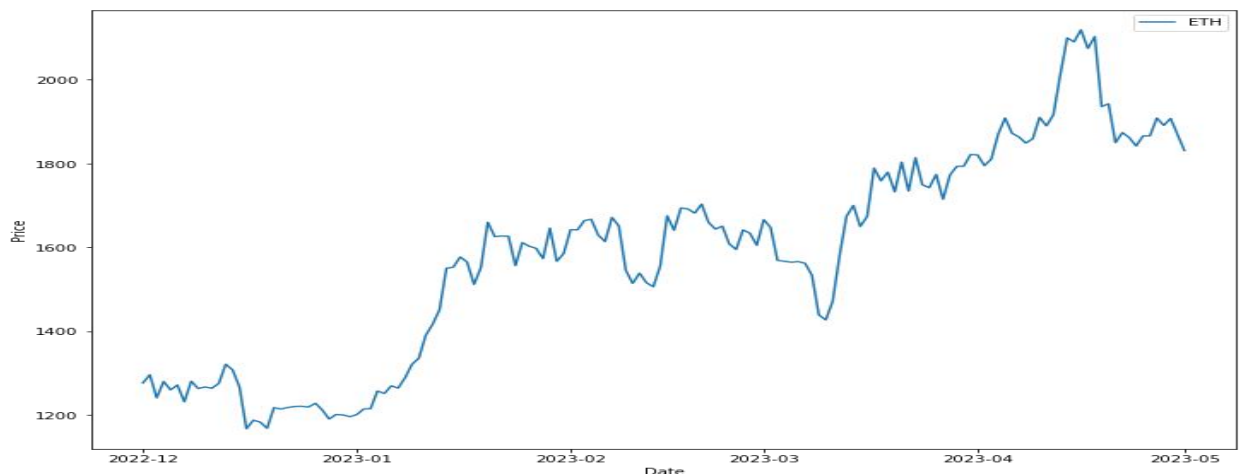


Рис. 3.2. Часовий ряд ціни криптовалюти Ethereum

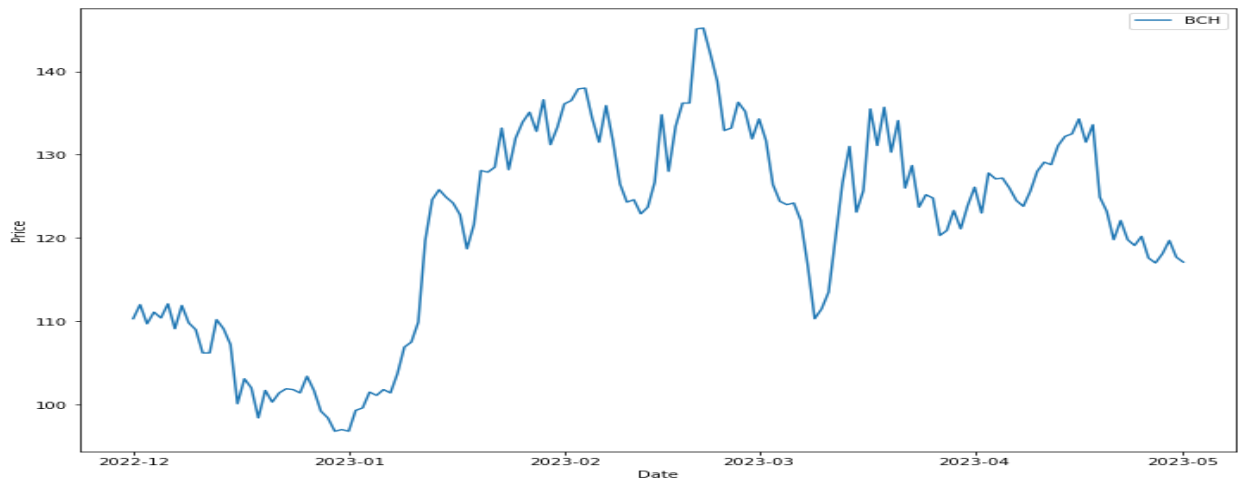


Рис. 3.3. Часовий ряд ціни криптовалюти Bitcoin Cash

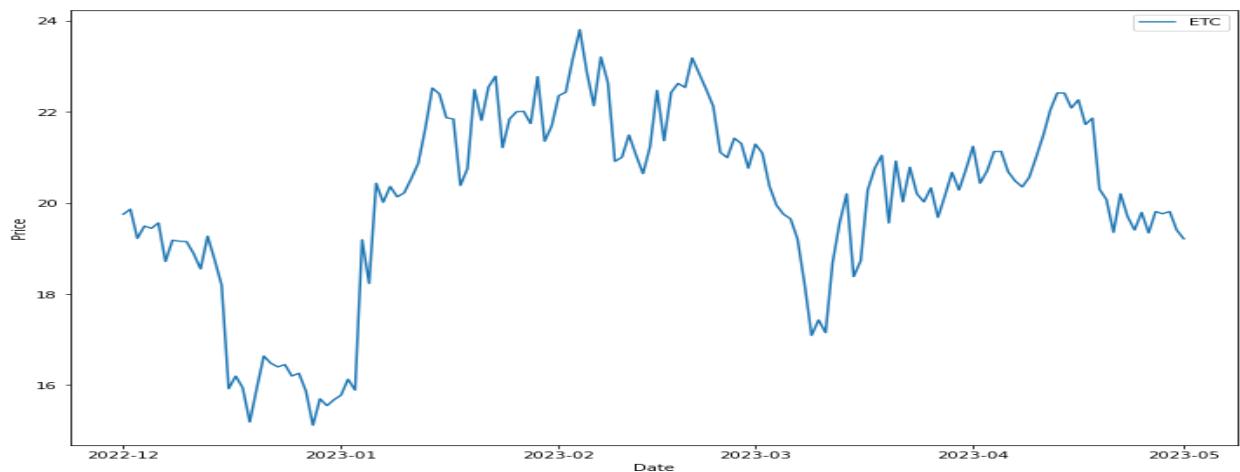


Рис. 3.4. Часовий ряд ціни криптовалюти Ethereum Classic

Крім даних про ціни на криптовалюти, датасет також містить значення про ціну на початку доби, найнижчу та найвищу ціну протягом дня, а також інформацію про кількість криптовалюти, що перебувала у обігу протягом дня. З рисунків видно, що криптовалюти у датасеті зростають у ціні, що робить їх доволі привабливими для торгівлі. Як вже зазначалось, ціни знаходяться у різному діапазоні, що може призвести до складностей під час прогнозування даних на довгий період часу, оскільки побудована модель може призводити до значних відхилень між прогнозованими та фактичними значеннями цін, що також вплине на дослідження арбітражних можливостей на ринку, однак загалом результати отримані на даному датасеті можна вважати еталонними в

рамках дослідження, хоча в реальності зібрати гаманець з таким набором криптовалют може бути досить складною задачею з фінансової точки зору.

3.2.2. Аналіз другого датасету

Дані у другому датасеті зібрані дані про 7 криптовалют на основі реального користувацького гаманця. До нього входять такі цифрові валюти як DEXTools, SUKU, Multichain, Sui, Drep, ApeCoin та XMON. Більшість з криптовалют є досить новими, а Multichain та Sui існують на біржах всього декілька днів на момент дослідження. Оскільки даний датасет збирався з метою оцінки арбітражної стратегії в реальних умовах, про коінтеграцію криптовалют заздалегідь нічого невідомо.

Найдорожчою криптовалютою у наборі є XMON. Дані про неї зібрані у період з 01.12.2022 по 05.02.2023. Потім криптовалюта тимчасово була виведена з торгового циклу у зв'язку з її різким падінням, повернули її на біржі у травні 2023 року. Часовий ряд ціни на дану криптовалюту зображено на рисунку 3.5.

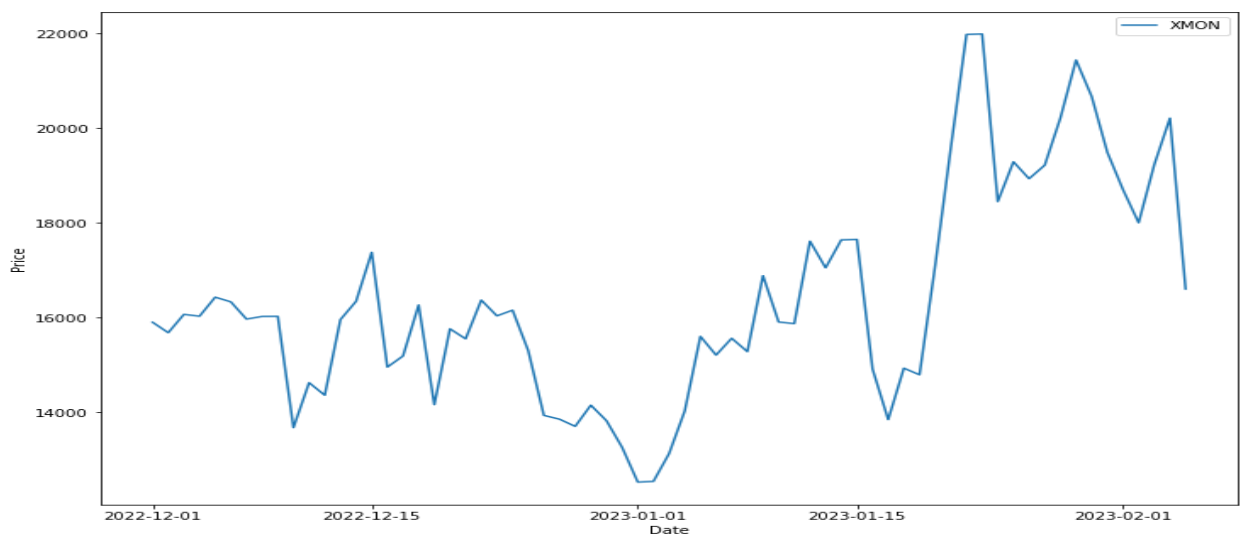


Рис. 3.5. Часовий ряд ціни криптовалюти XMON

Криптовалюта MULTI є новою на ринку та існує на криптобіржах з 12.04.2023 року. Наразі дана криптовалюта падає у вартості та безпосередньо залежить від зовнішніх факторів, як і будь-яка криптовалюта на початку свого існування. Подібну картину демонструє й криптовалюта SUI, яка представлена на біржах з 09.05.2023 року. Часові ряди цін на ці криптовалюти представлені на рисунках 3.6 та 3.7 відповідно.

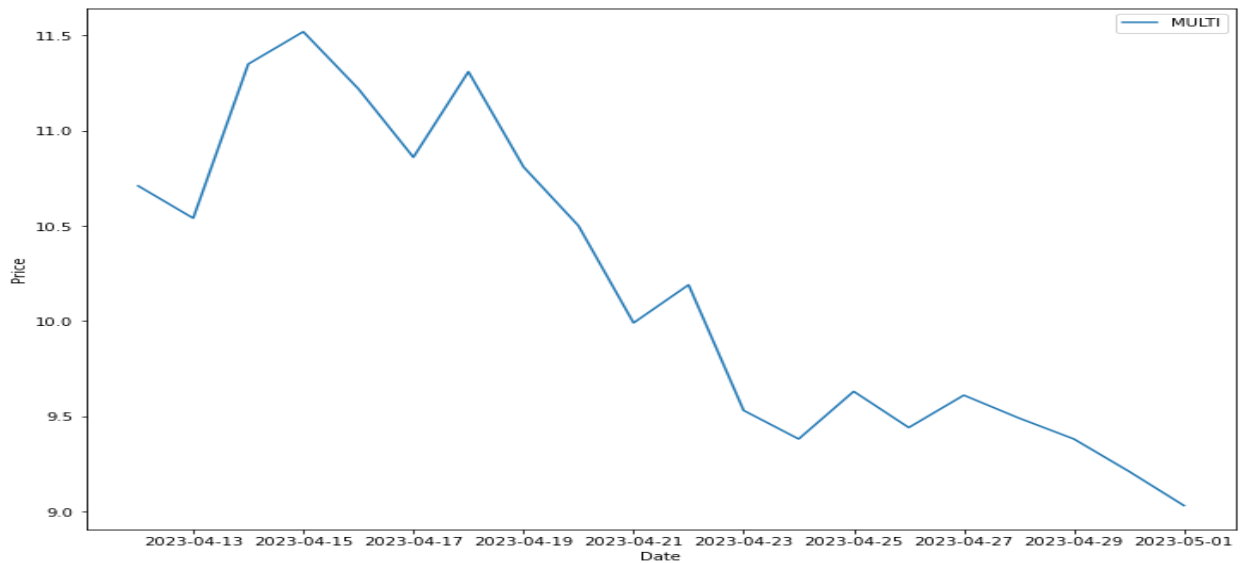


Рис. 3.6. Часовий ряд ціни криптовалюти Multichain

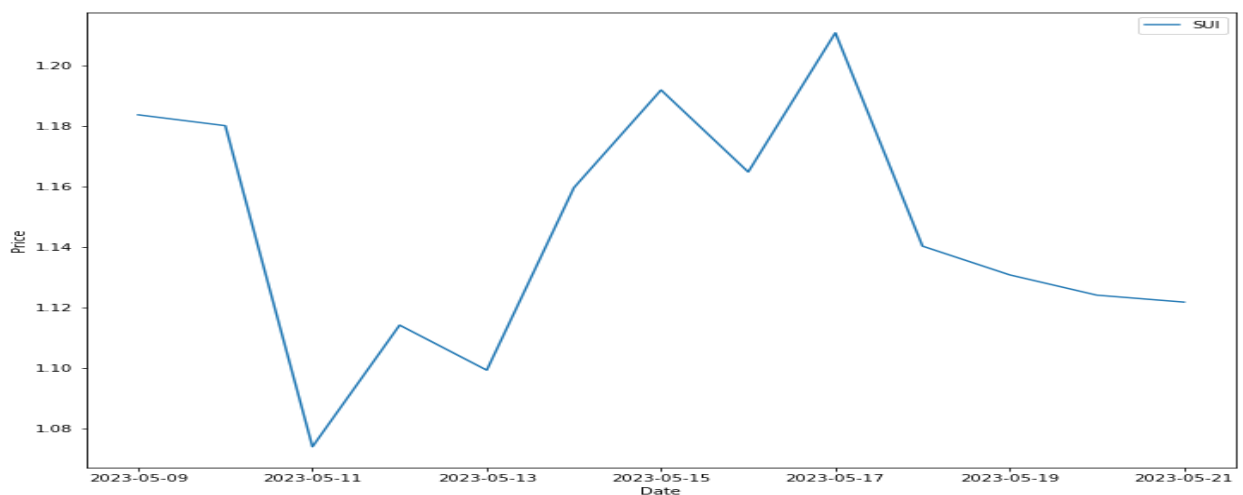


Рис. 3.7. Часовий ряд ціни криптовалюти SUI

Дані про інші криптовалюти зібрані у період з 01.12.2023 по 01.05.2023. Всі вони вже існують на ринку принаймні 6 місяців, мають певну репутацію

та тренд. На рисунку 3.8 представлені графіки цін на DEXTools та Drep, вони знаходяться в приблизно однаковому діапазоні та наразі зростають у вартості.



Рис. 3.8. Часові ряди цін криптовалют DEXTools та DREP

На рисунку 3.9 зображено часовий ряд криптовалюти ApeCoin, ціна якої наразі впала до початкової у період часу, що розглядається. З погляду на рисунок, можна зробити висновок, що дане падіння сталося внаслідок памп-енд-дампу, що відбувся у період з лютого по березень 2023 року.

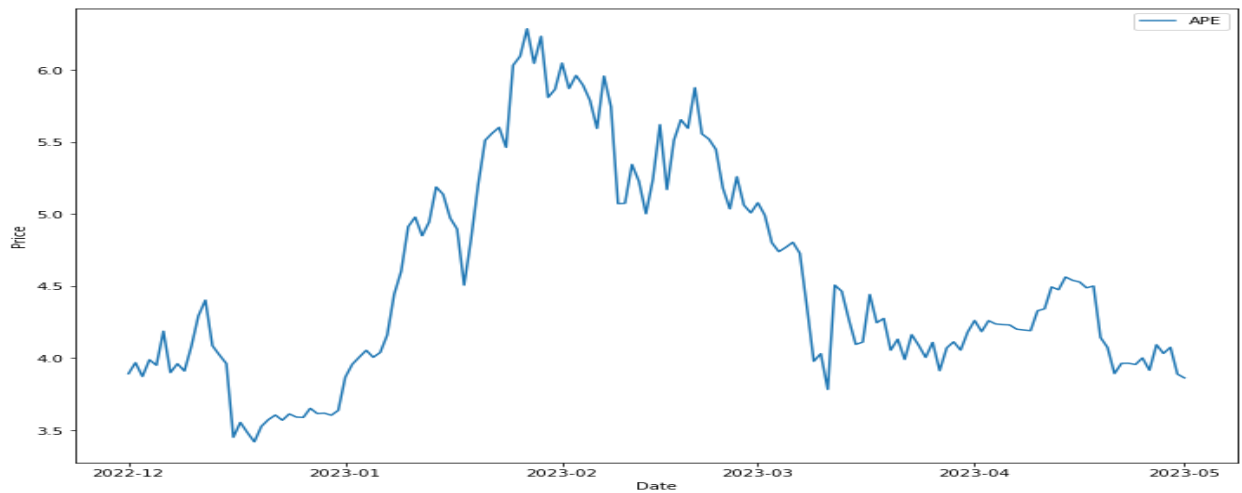


Рис. 3.9. Часовий ряд ціни криптовалюти ApeCoin

На рисунку 3.10 представлено графік ціни Suku. Дана криптовалюта є відносно стабільною та зберігає свою ціну протягом довгого часу на одному рівні без значних коливань. Крім того, її часовий ряд є доволі нетиповим для криптовалют, що може свідчити про відносну незалежність даного токена від зовнішніх обставин, що робить його привабливим для торгівлі.

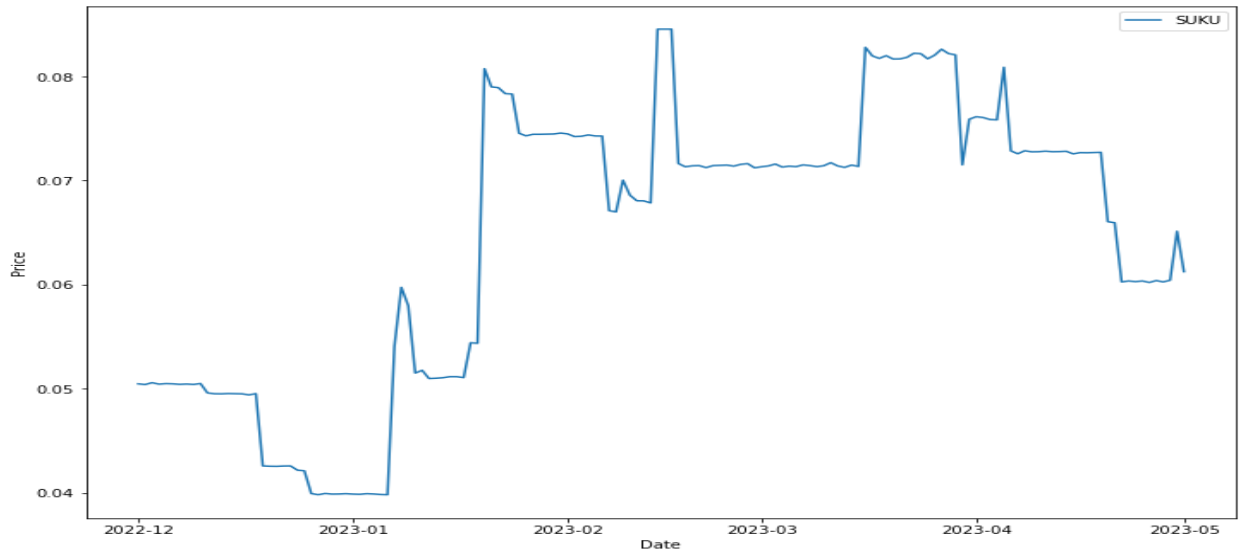


Рис. 3.10. Часовий ряд ціни криптовалюти Suku

З розглянутих рисунків очевидно, що криптовалюти у наборі не мають спільного тренду. Це може стати як плюсом для торгових стратегій, так і мінусом, якщо з використанням глибокого хеджування не вдасться створити лінійну комбінацію, яка буде стаціонарною, що одразу перекреслить можливості застосування і глибокого хеджування і статистичного арбітражу на ринку криптовалют для будь-яких гаманців, оскільки у реальних умовах користувачі підбирають криптовалюти виходячи з фінансових можливостей, а не статистичних аспектів.

Результати отримані на даному датасеті дозволяють узагальнити висновки отримані під час початкового дослідження на ринок криптовалют у цілому та зробити висновок щодо можливостей застосування глибокого хеджування та статистичного арбітражу для гаманців, що містять

криптовалюти прогнози щодо яких зробити неможливо з огляду на час їх існування, зовнішні фактори тощо.

3.3. Архітектура програмного продукту

Програма реалізована у середовищі jupyter notebook та складається з двох частин – застосування глибокого хеджування та статистичного арбітражу для першого та другого датасету відповідно. Реалізація дослідження для другого датасету, в свою чергу, також поділена на дві частини – дослідження на всьому датасеті та на зміненому відповідно до отриманих початкових результатів.

Алгоритм роботи програми є наступним:

1. Завантаження та попередня обробка даних.
2. Прогнозування майбутніх цін на криптовалюти, на основі яких буде побудовано спред та оцінка якості цих прогнозів.
3. Застосування лінійної регресії, як етапу глибокого хеджування для пошуку вагових коефіцієнтів.
4. Визначення спреда, як лінійної комбінації криптовалют та перевірка його на стаціонарність за допомогою доповненого тесту Дікі-Фуллера та тесту Квятковського.
5. Реалізація арбітражної стратегії з динамічними значеннями стоп лосу та генерація торгових сигналів.
6. Бектестинг отриманої стратегії, розрахунок та графічне представлення кумулятивного прибутку та просідання гаманця.
7. Розрахунок коефіцієнта Шарпа.

В ході реалізації механізму дослідження було вирішено відійти від класичної стандартизації даних та застосувати функцію `MinMaxScaler()` з

бібліотеки `sklearn` для оптимізації моделі. Дана функція застосовує формулу 3.1.

$$x_{sc} = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (3.1)$$

Тут, x_i – значення, що нормалізується, x_{min} – мінімальне значення вибірки, x_{max} – максимальне значення вибірки.

Прогнозування цін на криптовалюту було вирішено робити на основі факторів ‘Open’, ‘High’, ‘Low’. Фактор ‘Vol.’, що також представлений у зібраному датасеті, хоча й є доволі важливим при аналізі ринку та формуванні ціни, на роботу моделі прогнозування впливає досить негативно, оскільки сам по собі залежить від зовнішніх факторів і часто не корелює з ціною, що мала криптовалюта у певний день.

Також, було вирішено змінити формули для обчислення значень стоп-лосу та застосувати формулу 3.2, оскільки вона є більш вживаною саме на крипторинку.

$$mean_{spread} \pm \sigma_t \quad (3.2)$$

Генерація торгових сигналів на основі спреду відбувається наступним чином:

1. За формулою 3.2 обчислюються значення стоп-лосу у момент часу i .
2. Відбувається порівняння обчислених значень з поточною вартістю спреду.
3. Якщо x_i вище за верхнє значення стоп-лосу, то відкривається короткострокова позиція ‘SELL’.
4. Якщо x_i менше за нижнє значення стоп-лосу, то відкривається довгострокова позиція ‘BUY’.
5. У випадку, якщо поточна ціна знаходиться між двома границями стоп-лосу вважається, що наразі опцій для отримання прибутку від торгівлі немає та генерується сигнал ‘HOLD’.

Подальший бектестинг стратегії відбувається за допомогою апарату описаного у розділі 2.

3.4. Аналіз результатів дослідження

Введемо наступні припущення:

- Застосування арбітражної стратегії з використанням динамічного стоп-лосу на ринку криптовалют є ефективним у контексті максимізації прибутку.
- Стандартний механізм глибокого хеджування на ринку криптовалют дозволяє отримати оптимальні вагові коефіцієнти хеджування для моделювання стаціонарного спреда та є універсальним для будь-якого набору криптовалют.
- Комбінацію глибокого хеджування та статистичного арбітражу можна застосувати до набору криптовалют без попереднього їх аналізу та отримати прийнятні результати.

Висновки щодо ефективності застосування глибокого хеджування та статистичного арбітражу на ринку криптовалют по результатам дослідження будемо робити спираючись на наведені припущення.

3.4.1. Аналіз результатів для першого датасету

В першу чергу, як і було зазначено раніше, відбулось прогнозування цін на криптовалюту у датасеті. У таблиці 3.1 зведені значення метрик, що використовувалися для аналізу результатів роботи моделі прогнозування. Як ми можемо помітити, значення коефіцієнту детермінації є доволі високим, тож отримані результати ми можемо вважати точними і використовувати у подальшому дослідженні.

Таблиця 3.1. Оцінка якості результатів прогнозування для кожної криптовалюти

Crypto	MSE	MAE	R^2
ETH	643.05861	23.65901	0.92
BTC	41781.83064	167.66552	0.96
BCH	2.54470	1.50524	0.90
ETC	0.06325	0.21384	0.93
WBTC	34705.31284	147.09506	0.96

Щодо похибок MSE та MAE, їх значення є доволі високими для криптовалют Ethereum, Bitcoin та Wrapped Bitcoin, однак, оскільки коефіцієнт детермінації вказує на високу точність роботи моделі на відповідних даних, є сенс співставити графіки прогнозів та реальних цін на ці криптовалюти, оскільки високе значення цих похибок може бути спричинене в тому числі й великими значеннями у початковому датасеті. Графіки цін та прогнозів на криптовалюту Ethereum, Bitcoin та Wrapped Bitcoin представлені на рисунках 3.11, 3.12 та 3.13 відповідно.

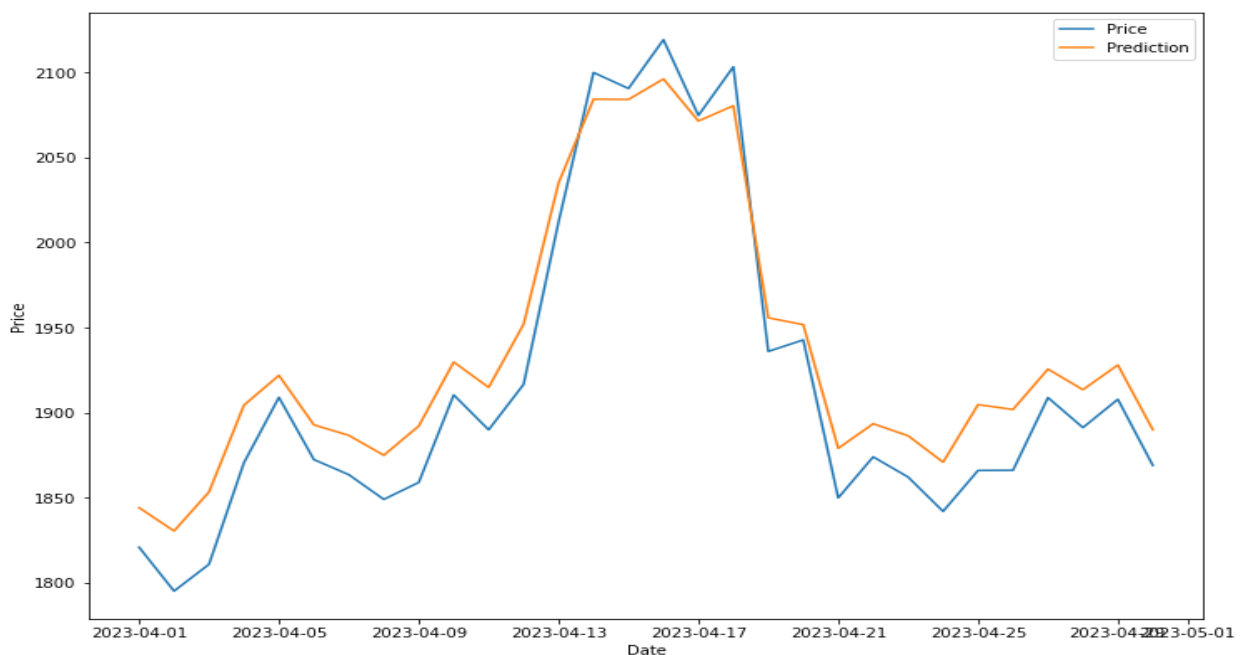


Рис. 3.11. Графічне представлення реальної та прогнозованої цін криптовалюти Ethereum

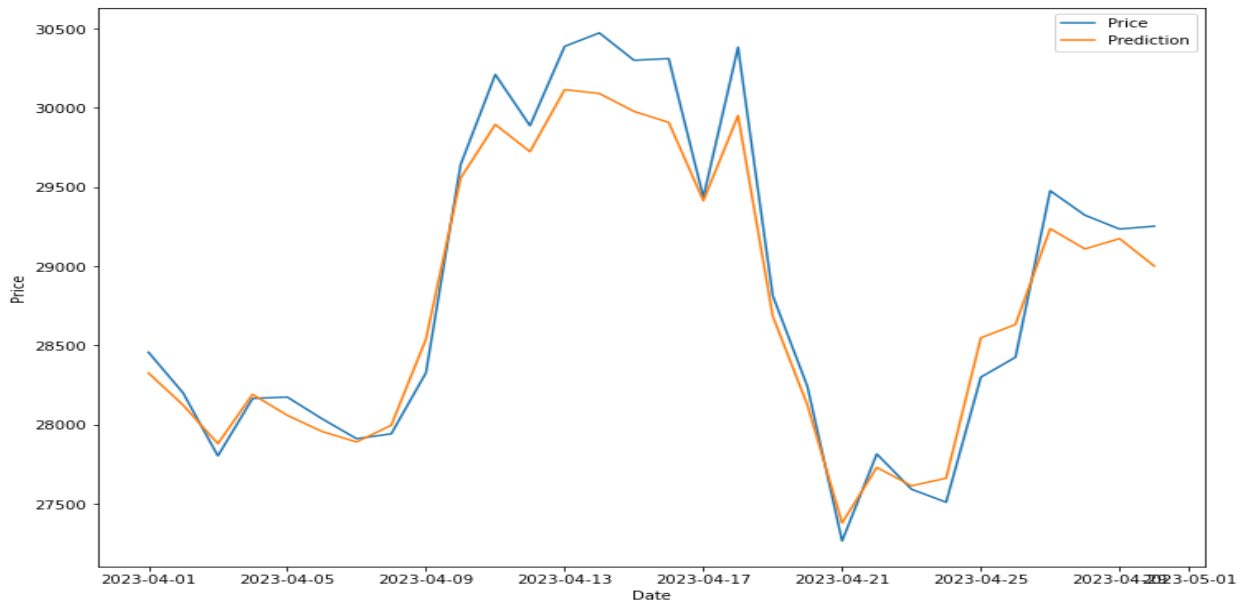


Рис. 3.12. Графічне представлення реальної та прогнозованої цін криптовалюти Bitcoin

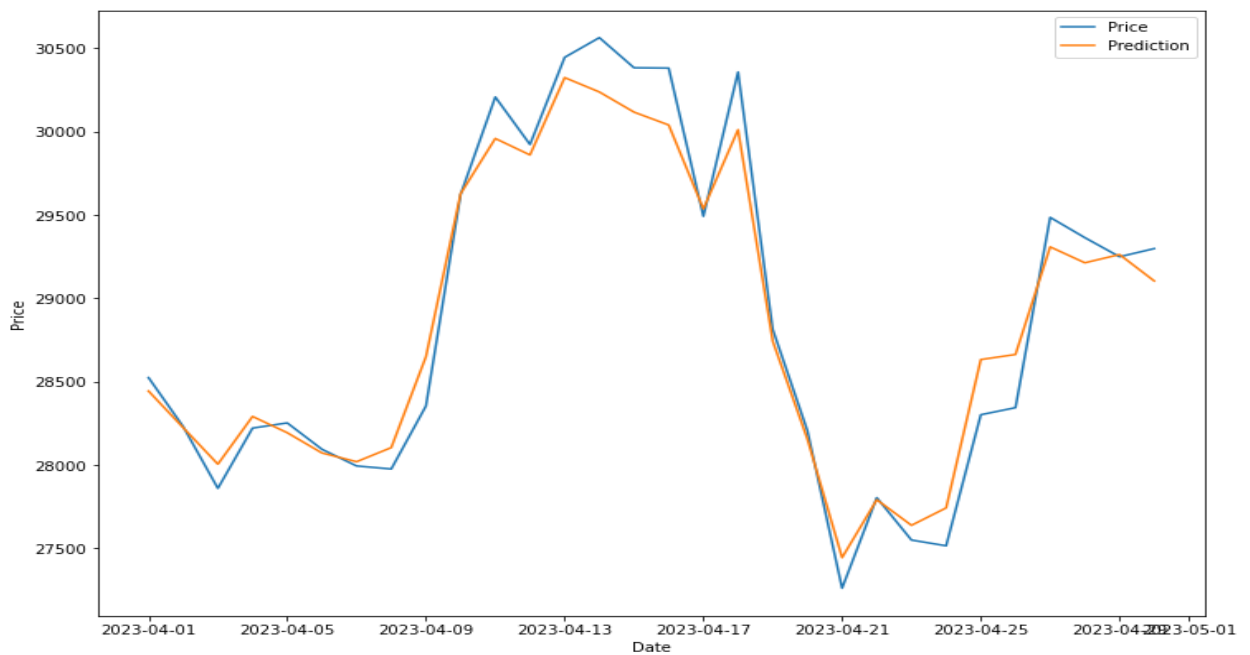


Рис. 3.13. Графічне представлення реальної та прогнозованої цін криптовалюти Wrapped Bitcoin

Бачимо, що результати прогнозування є досить близькими до реальних значень ціни, тож, як і було сказано раніше, великі похибки зумовлені високими цінами на дані криптовалюти.

Провівши глибоке хеджування отримали спред, часовий ряд якого представлений на рисунку 3.14.

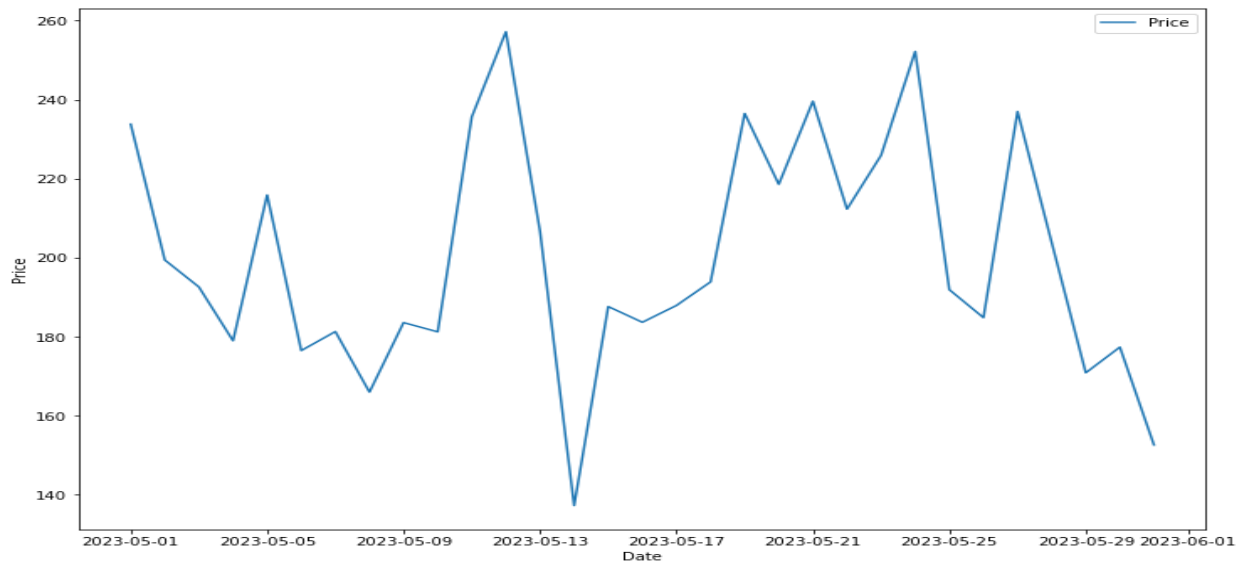


Рис. 3.14. Часовий ряд спреда побудованого на основі датасету 1

Оскільки криптовалюти мають тенденцію до швидких та різких змін у ціні, доцільно розглядати певну торгову стратегію протягом короткого терміну, у нашому випадку шукатимемо торгівельні можливості за травень 2023 року.

Даний спред є стаціонарним часовим рядом, що підтверджується відповідними тестами, результати яких зібрані у таблицю 3.2.

Таблиця 3.2. P_value отримане в результаті тестів на стаціонарність

ADF	KPSS
0.006	0.1

Як ми бачимо з тесту Дікі-Фулера, значення p_value є меншим за 0.5, тото приймаємо основну гіпотезу даного тесту про відсутність одиничних коренів та стаціонарність спреда. За результатами тесту Квятковського видно, що значення p_value є більшим за 0.05, тобто гіпотеза про існування одиничних коренів відкидається, За результатами обох тестів часовий ряд є

стаціонарним, тож тепер можна переходити до реалізації арбітражної стратегії.

Спершу, генеруємо торгові сигнали за допомогою описаного раніше алгоритму. Графічне представлення торгових сигналів зображено на рисунку 3.15. Бачимо, що для спреда, що базується на обраних криптовалютах вигідно частіше відкривати короткі позиції ніж довгі. Ці результати є доволі очікуваними, оскільки жодна з криптовалют за останній час не зазнавала різких падінь у ціні тож відкривати довгі позиції є доволі невигідним, принаймні в рамках даної стратегії.

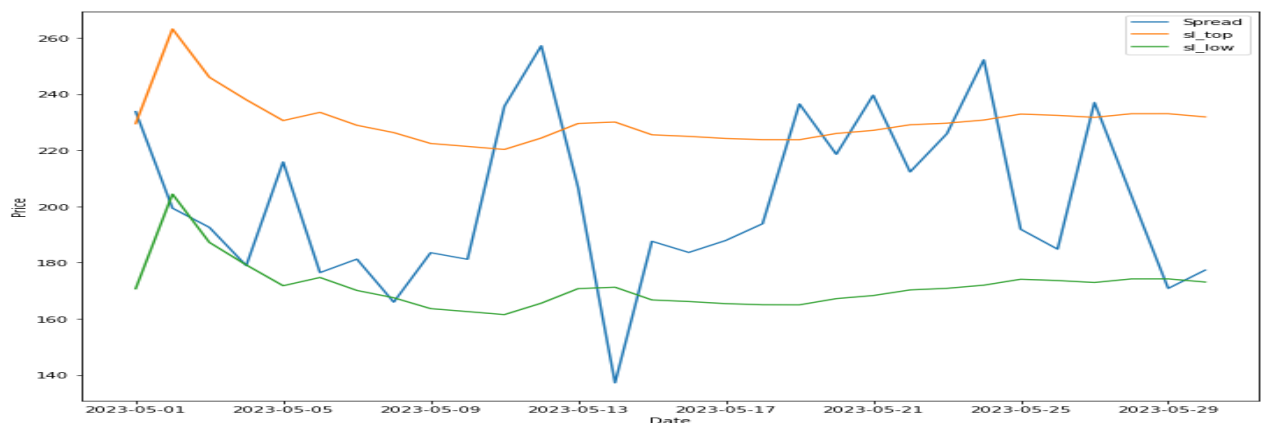


Рис. 3.15. Генерація торгових сигналів на основі спреда

Останнім етапом дослідження є оцінка якості роботи стратегії. На рисунках 3.16 та 3.17 зображений графік кумулятивного прибутку на даному датасеті та загальне просідання гаманця.

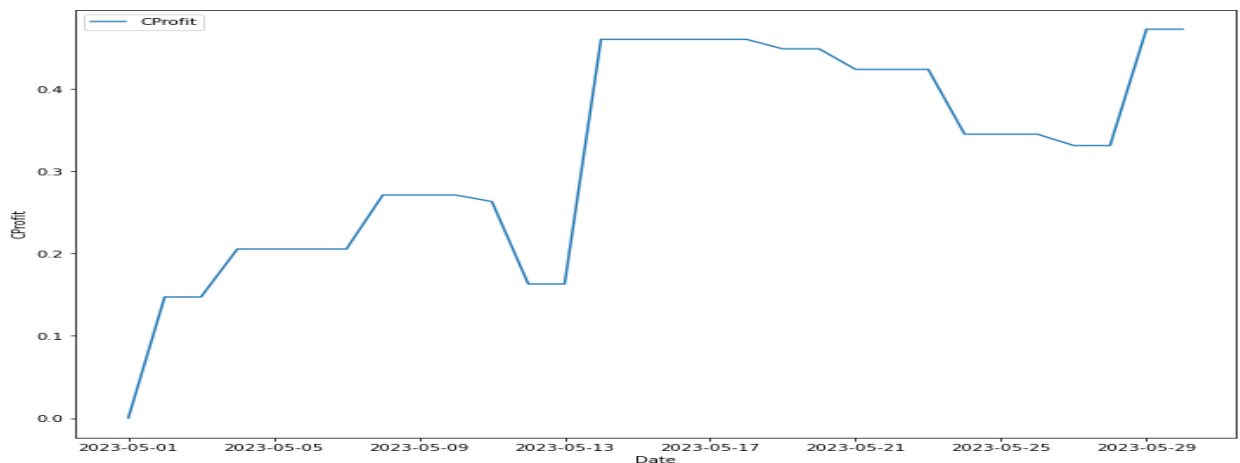


Рис. 3.16. Кумулятивний прибуток арбітражної стратегії для датасету 1

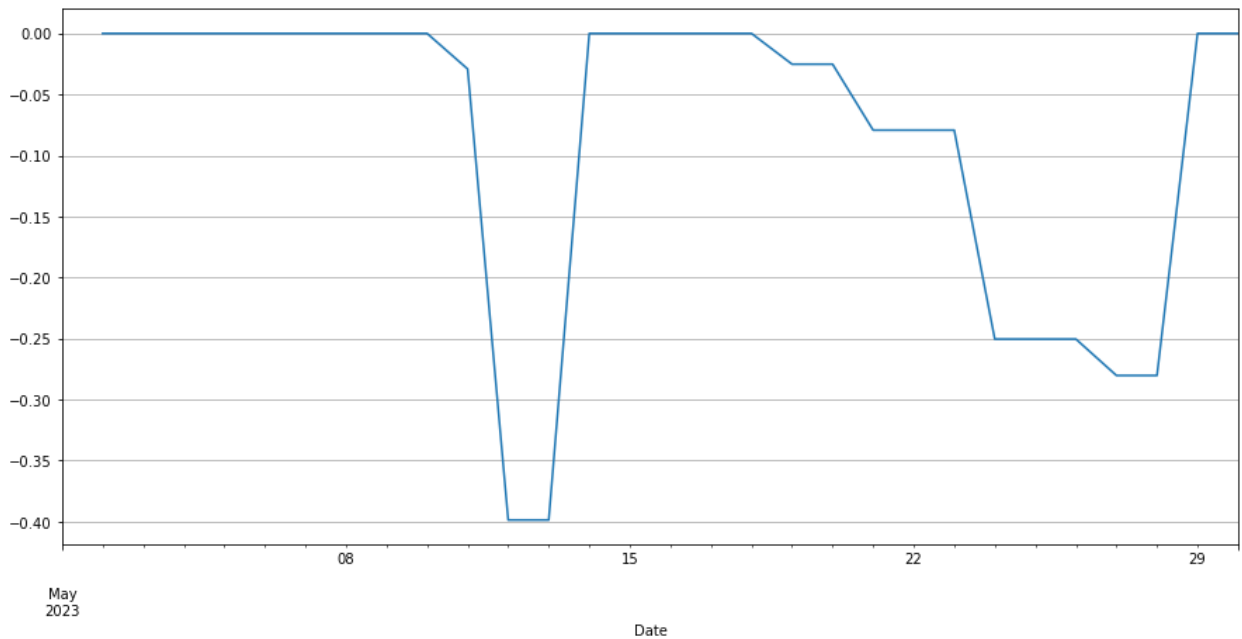


Рис. 3.17. Просідання портфеля для арбітражної стратегії на датасеті 1

З даних рисунків видно, що прибуток з використанням даної стратегії в загальному виріс на 45%, при цьому максимальне просідання становить 39%. Для ринку криптовалют це досить непогані результати, враховуючи що даний датасет складається з доволі дорогих цифрових валют з високою кореляцією, тож будь-які коливання в їх ціні можуть призвести до значних збитків. Однак, з використанням глибокого хеджування та побудувавши з його допомогою арбітражну стратегію отримали захищений гаманець, торгівля з використанням якого дійсно приносить прибуток перекриваючи просідання.

Значення коефіцієнта Шарпа для даної стратегії становить 3.4, що є оптимальним для ринку криптовалют, оскільки якщо стратегія є прибутковою щодня, то ризики втрат капіталу через волатильність ринку мінімізується. Крім того, у такому випадку в деякому сенсі зменшується вплив і зовнішніх факторів на торгівлю.

Отже, по результатам дослідження ринку криптовалют на першому датасеті можна зробити наступні висновки:

- Комбінація глибокого хеджування та статистичного арбітражу дійсно має місце на ринку криптовалют у контексті розробки прибуткової стратегії.
- Застосування глибокого хеджування на ринку криптовалют допомагає визначити чи є криптовалюти у гаманці коінтегрованими та мінімізує ризики, що пов'язані з торгівлею.
- На ринку криптовалют наявні арбітражні можливості, які, за допомогою статистичного арбітражу можна використати для максимізації прибутку на ринку.

Орієнтуючись на дані висновки, можна сказати, що використовуючи дану комбінацію методів на ринку користувачі дійсно можуть збільшувати свій капітал та захистити його від волатильності. З часом це призведе до збільшення довіри до ринку та, відповідно, до його популяризації, оскільки за допомогою комбінації цих методів ми отримуємо прогнозовані результати торгівлі, що є важливим для молодих ринків.

Однак, як вже було зазначено раніше, даний датасет був додатково проаналізований з математичної точки зору та зібраний на основі інших досліджень, тож на даному етапі можна сказати, що отримані результати є релевантними для криптовалют, що існують довгий час на ринку, оскільки про них є достатньо інформації для попереднього аналізу перед розробкою арбітражної стратегії.

3.4.2. Аналіз результатів для другого датасету

Повторне дослідження має мету узагальнити зроблені висновки на весь ринок криптовалют, враховуючи нові екземпляри цифрових валют, що можуть бути у гаманці. В першу чергу, слід зазначити, що прогнозовані значення для криптовалюти SUI не були обчислені у зв'язку з тим, що вона була створена

за 13 днів до дослідження, тож даних для навчання складної моделі виявилось недостатньо. Також, мною була виконана спроба побудувати більш просту модель прогнозування, з меншою кількістю шарів та параметрів, а також застосування лінійної регресії, однак досягти адекватних результатів не вдалося. Було вирішено, при реалізації механізму глибокого хеджування застосувати поточні значення ціни на дану криптовалюту, оскільки час її появи співпадає з періодом торгівлі.

Оцінка якості прогнозування наведена у таблиці 3.3. Зазначимо, що точність прогнозів дещо нижча, ніж отримана на першому датасеті, однак, оскільки значення коефіцієнта детермінації все ж перевищує вказаний у розділі 2 поріг, результати прогнозів є прийнятними для подальшого дослідження.

Таблиця 3.3. Оцінка якості результатів прогнозування для криптовалют у датасеті 2

Crypto	MSE	MAE	R^2
SUKU	8.41036e-06	0.00175	0.78
DEXT	0.00027	0.01276	0.90
DREP	0.00022	0.01119	0.52
APE	0.01108	0.08922	0.74
XMON	262335.77119	456.37063	0.79
MULTI	0.00582	0.05723	0.56

Цікаво розглянути графіки реальної та прогнозованої ціни на криптовалюту Drep (рисунок 3.17) та Multichain (рисунок 3.18), оскільки точність прогнозування для них є доволі низькою, хоча й вищою за порогове значення, яке має становити коефіцієнт детермінації, щоб вважати результати прогнозування адекватними для подальшого дослідження. Також розглянемо

відповідний графік для криптовалюти XMON (рисунок 3.19), оскільки значення MSE та MAE для неї є значно вищим ніж для інших криптовалют.

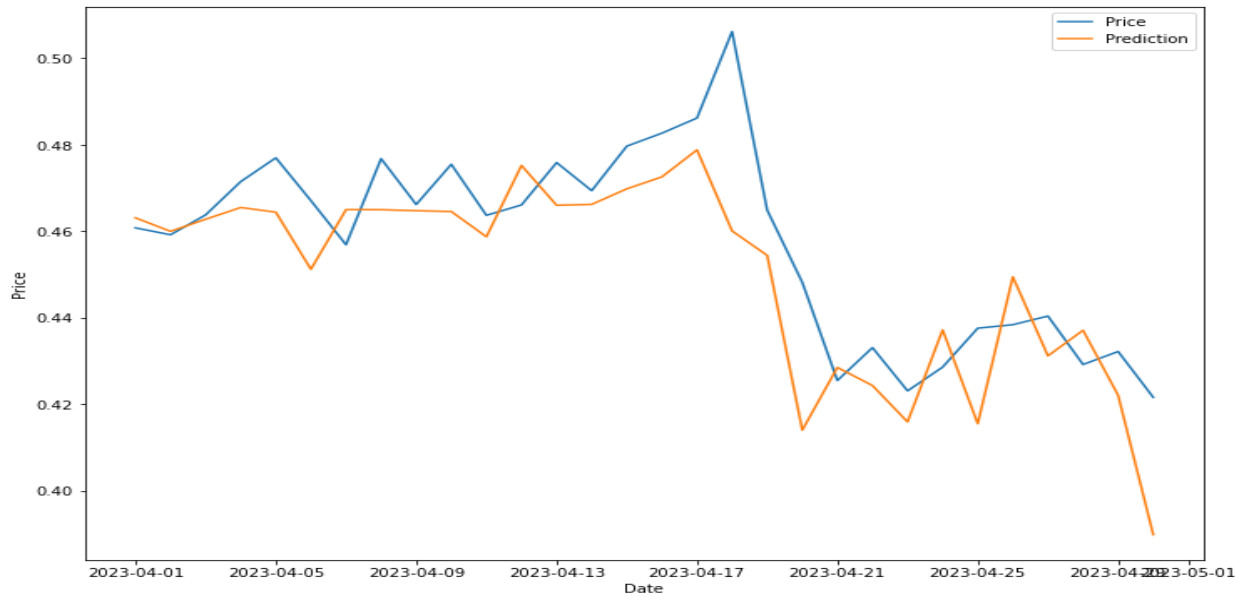


Рис. 3.17. Графічне зображення реальної та прогнозованої цін криптовалюти DREP

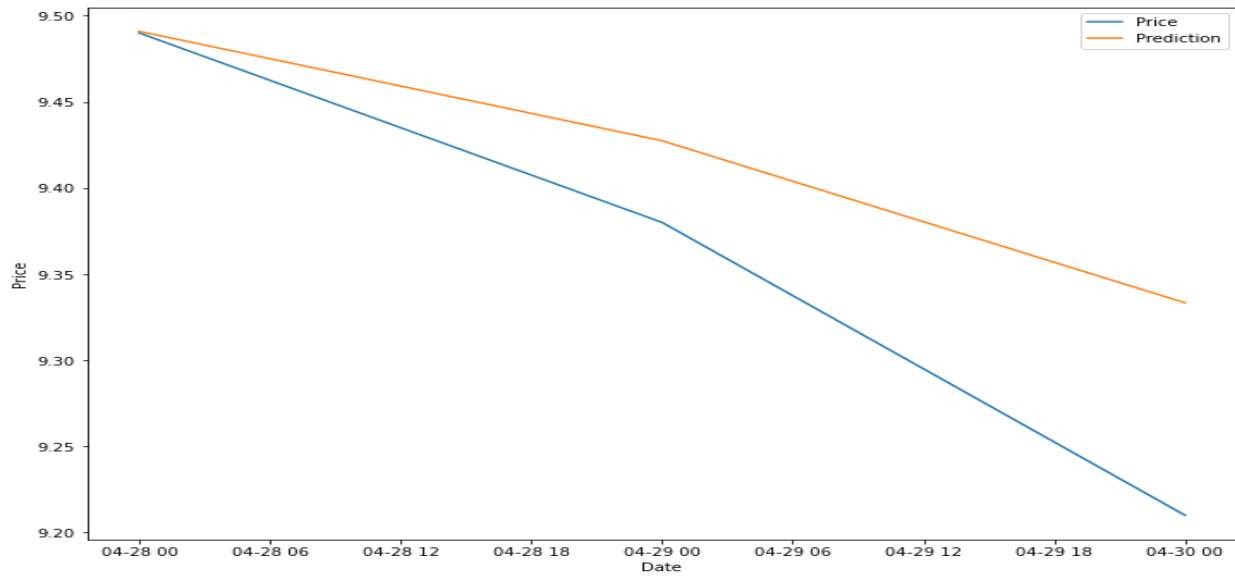


Рис. 3.18. Графічне зображення реальної та прогнозованої цін криптовалюти Multichain

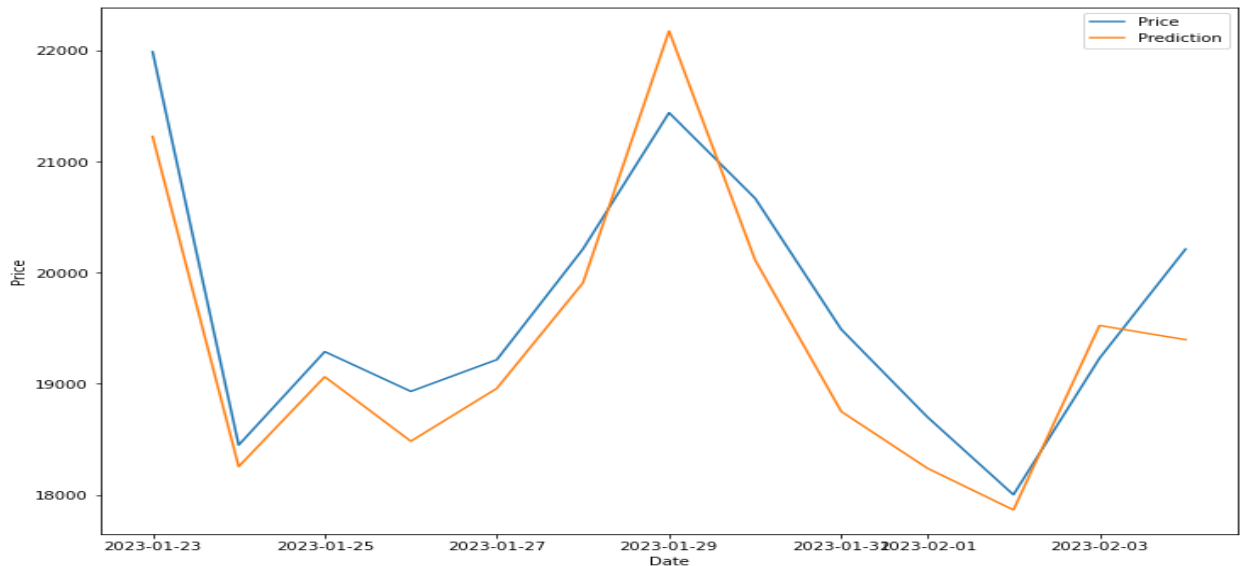


Рис. 3.19. Графічне зображення реальної та прогнозованої цін криптовалюти XMON

Бачимо, що прогнозовані значення для Dger та Multichain дійсно є менш точними ніж для інших криптовалют, однак значення отриманих прогнозів є доволі близькими до реальних. Дана проблема може бути пов'язана з тим, що Multichain існує на ринку порівняно небагато часу, тож можливо для досягнення вищої точності було замало тренувальних даних. Для криптовалюти Dger, проблема полягає у різкому зростанні ціни наприкінці періоду, тож імовірно саме це збільшило похибки та зменшило точність прогнозів. Що стосується криптовалюти XMON – як і у розглянутих раніше прикладах, високі значення похибок пов'язані з вхідними даними, прогнози є доволі точними.

Застосувавши глибоке хеджування був побудований спред, однак при перевірці його на стаціонарність отримали досить високе значення p_value при виконанні доповненого тесту Дікі-Фуллера. Результати обох тестів зібрано у таблицю 3.4.

Таблиця 3.4. P_value тестів на стаціонарність для спреду побудованого на датасеті 2

ADF	KPSS
7.07	0.1

За даними результатами очевидно, що на стаціонарність вказує лише тест Квятковського, тобто дані криптовалюти вважаються стаціонарними за трендом. В даному випадку потрібно прибрати тренд та перевірити отриманий часовий ряд на стаціонарність знову. Результати перевірки на стаціонарність після проведення процедури детренду занесені у таблицю 3.5.

Таблиця 3.5. P_value тестів на стаціонарність для спреду після детрендингу

ADF	KPSS
2.04	0.04

Тепер, обидва тести говорять про відсутність стаціонарності спреду. Отже, можемо зробити проміжний висновок, що застосування комбінації глибокого хеджування і статистичного арбітражу все ж вимагає попереднього дослідження гаманця та визначення певних зв'язків. В даному випадку проблема може полягати також у наявності у датасеті криптовалют SUI та XMON. SUI може швидко та неочікувано змінити свій тренд, а так як даних для прогнозування наразі недостатньо, це може суттєво впливати на результати нашого дослідження. Криптовалюта XMON, як вже зазначалося раніше, була виведена з торгових процесів на певний час, тому, хоча вона й зберігалася у гаманці користувача і її можливо застосувати у торгівлі на момент дослідження, прогнозування близьких значень на поточний місяць є складною задачею. Виключимо ці криптовалюти з дослідження та спробуємо виконати етап глибокого хеджування ще раз.

За допомогою лінійної регресії знову обчислюємо вагові коефіцієнти хеджування та з їх використанням будуємо спред. Часовий ряд спреду зображений на рисунку 3.20.

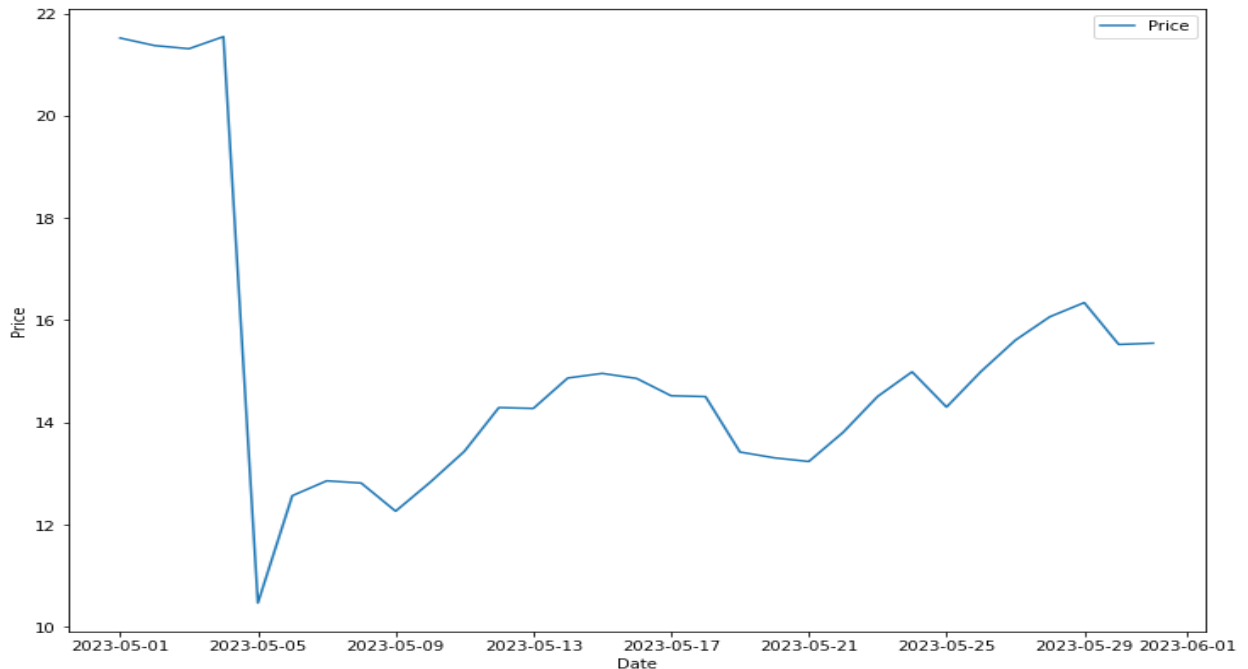


Рис. 3.20. Часовий ряд спреду побудованого на зміненому датасеті 2.

Для побудованого спреду знову застосуємо тести на стаціонарність, результати яких зберемо у таблицю 3.6.

Таблиця 3.6. P_value тестів на стаціонарність для спреду побудованого на зміненому датасеті 2.

ADF	KPSS
0.03	0.06

Результати обох тестів вказують на стаціонарність спреду, тож можемо переходити до реалізації арбітражної стратегії. Графічне зображення спреду та меж стоп-лосу зображено на рисунку 3.21. Бачимо, що на даному наборі датасеті переважають сигнали відкриття довгострокових позицій, що є досить зрозумілим, оскільки криптовалюти є досить новими та часто падають у ціні.

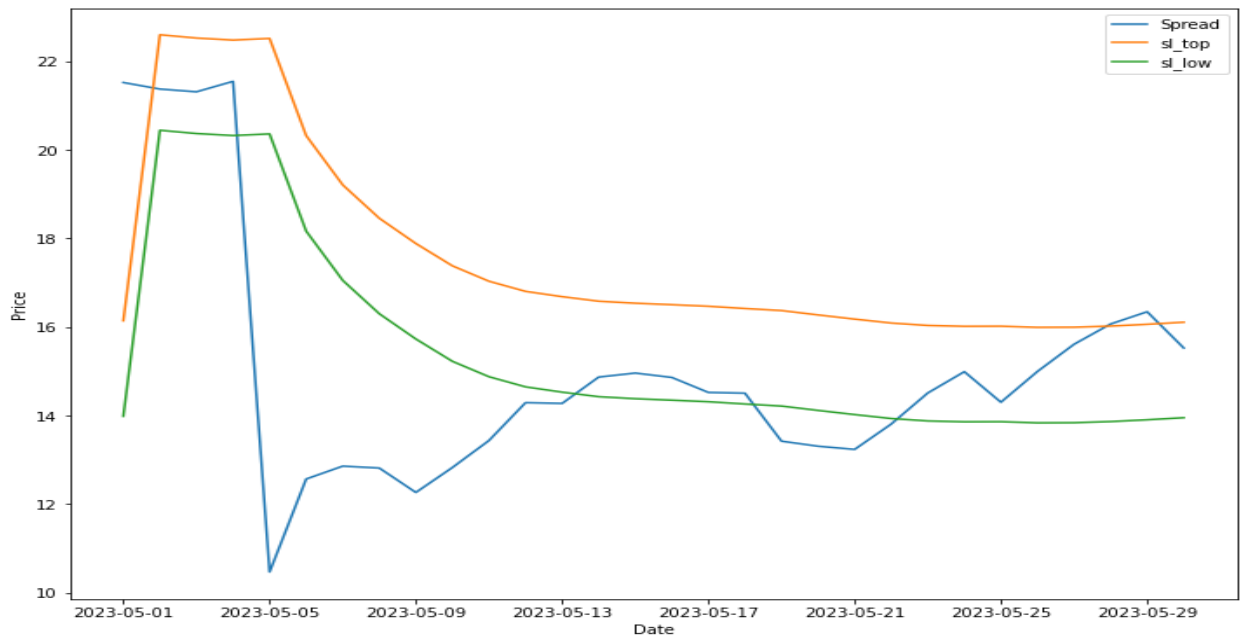


Рисунок 3.21. Генерація торгових сигналів на основі спреда побудованому на зміненому датасеті 2

Проведемо бектестинг побудованої стратегії. На рисунках 3.22 та 3.23 зображений графік кумулятивного прибутку та просідання портфеля відповідно.

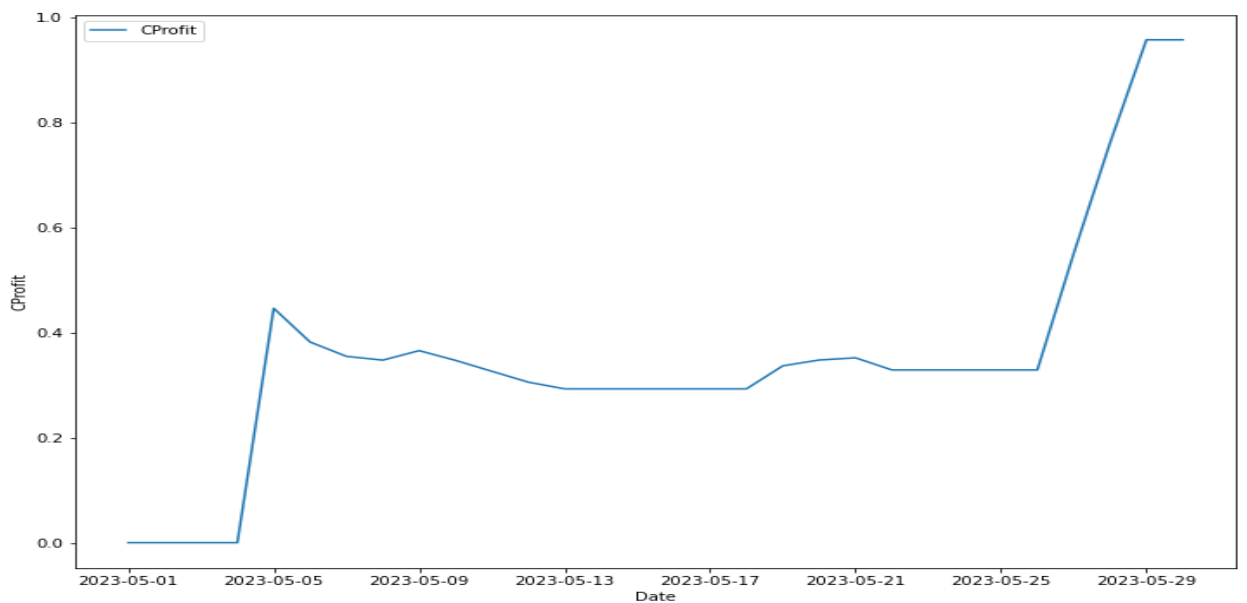


Рис. 3.22. Графічне представлення кумулятивного прибутку стратегії на зміненому датасеті 2.

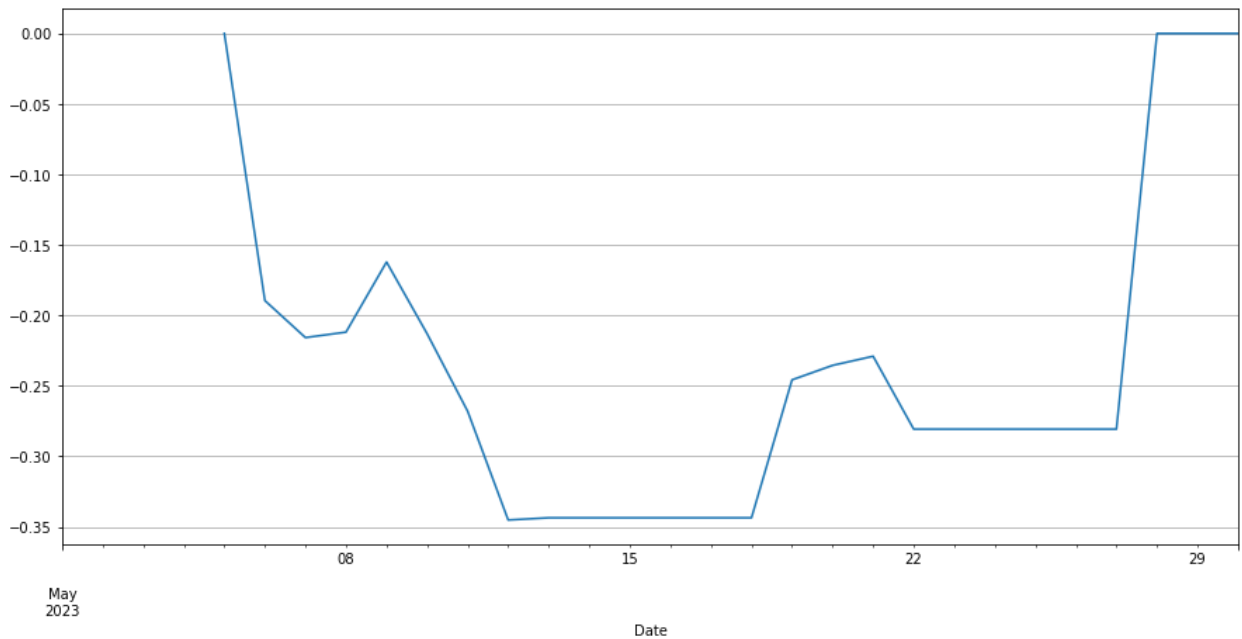


Рис. 3.23. Графічне представлення просідання портфелю для арбітражної стратегії побудованої на зміненому датасеті 2

Отже, кумулятивний прибуток зріс на 47%, різке зростання наприкінці періоду скоріше є ситуативним ніж прогнозованим. Просідання портфеля становить 34% і перекривається кумулятивним прибутком, що говорить про те, що торгові можливості для гаманця все ж є, хоча й не вдалось застосувати всі криптовалюти, що у ньому наявні.

Коефіцієнт Шарпа становить 2.15, що свідчить про прибутковість стратегії щотижня. Такий результат є зрозумілим, оскільки для отриманого спреда переважають довгострокові торгові позиції, період від їхнього відкриття до закриття зазвичай і займає близько тижня.

На основі результатів дослідження на другому датасеті можна зробити наступні висновки:

- Комбінація глибокого хеджування та статистичного арбітражу дозволяє виявляти арбітражні можливості на ринку, однак без попереднього аналізу гаманця не завжди можна отримати стаціонарний спред для арбітражної стратегії.

- Для найновіших криптовалют застосувати дану комбінацію методів досить складно, оскільки даних для прогнозування досить мало і якщо ціна різко змінюється, то виконати цей етап не вдасться.
- Для розробки арбітражної стратегії не варто застосовувати криптовалюту, яких не було в обігу протягом певного часу, оскільки їхні часові ряди є нестабільними та важкими для прогнозування.

Висновки

Цей розділ було присвячено опису механізму реалізації дослідження та аналізу його результатів. Було детально розглянуто схему реалізації дослідження та описано кроки глибокого хеджування та статистичного арбітражу, що виконуються програмним продуктом. Арбітражна стратегія, яку було реалізовано, використовує динамічне значення стоп-лосу, що робить її ефективною для ринків з високою волатильністю.

Дослідження було проведено на двох датасетах та було описано отримані результати, по кожному датасету було зроблено проміжні висновки. Отримані результати свідчать про те, що застосування глибокого хеджування та статистичного арбітражу на ринку криптовалют беззаперечно має місце та допомагає максимізувати прибуток та захистити гаманець від волатильності ринку. Однак, при проведенні дослідження на другому датасеті, було виявлено, що для довільного користувачього гаманця, без аналізу вхідних даних застосування обраних методів не дає очікуваних результатів. Це може бути пов'язано як з недостатньою кількістю даних про криптовалюту для аналізу, так і з відсутністю цифрової валюти у обігу на момент розробки торгової

стратегії. Ці фактори не дозволяють якісно хеджувати гаманець та досягти стаціонарності часового ряду спреда. Важливо зазначити, що наявність нових криптовалют у гаманці не означає те, що застосувати комбінацію методів дослідження для розробки торгівельної стратегії, проте їх наявність може ускладнити даний процес, тож краще перший час після купівлі «молодої» криптовалюти не застосовувати її у стратегіях, а зачекати та зібрати більше даних.

З огляду на волатильність ринку та велику кількість ризиків, результати дослідження можуть бути корисними для подальшого розвитку ринку криптовалют. Хоча застосування комбінації глибокого хеджування та статистичного арбітражу й вимагає певної підготовки, результати отримані з її допомогою дійсно допомагають захистити гаманець та отримати прибуток, що зменшує ризики пов'язані з цими факторами. Це допомагає збільшити довіру до ринку і тим самим збільшити його популярність, відповідно й вартість криптовалют.

4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ

У даному розділі проводиться оцінка основних характеристик програмного продукту, розробленого для проведення дослідження ринку криптовалют за допомогою статистичного арбітражу та глибокого хеджування.

Нижче наведено аналіз різних варіантів реалізації модулю з метою вибору оптимальної, з огляду при цьому як на економічні фактори, так і на характеристики продукту, що впливають на продуктивність роботи і на його сумісність з апаратним забезпеченням. Для цього було використано апарат функціонально-вартісного аналізу.

Функціонально-вартісний аналіз (ФВА) – це технологія, яка дозволяє оцінити реальну вартість продукту або послуги незалежно від організаційної структури компанії. ФВА проводиться з метою виявлення резервів зниження витрат за рахунок ефективніших варіантів виробництва, кращого співвідношення між споживчою вартістю виробу та витратами на його виготовлення. Для проведення аналізу використовується економічна, технічна та конструкторська інформація.

Алгоритм функціонально-вартісного аналізу включає в себе визначення послідовності етапів розробки продукту, визначення повних витрат (річних) та кількості робочих часів, визначення джерел витрат та кінцевий розрахунок вартості програмного продукту.

4.1. Постановка задачі проектування

У роботі застосовується метод ФВА для проведення техніко-економічного аналізу розробки системи дослідження ринку криптовалют. Оскільки рішення стосовно проектування та реалізації компонентів, що розробляються, впливають на всю систему, кожна окрема підсистема має її задовольняти. Тому фактичний аналіз представляє собою аналіз функцій програмного продукту, призначеного для обробки та проведення аналізу даних по білковим послідовностям.

Технічні вимоги до програмного продукту є наступні:

- функціонування на персональних комп'ютерах із стандартним набором компонентів;
- зручність та зрозумілість для користувача;
- швидкість обробки даних та доступ до інформації в реальному часі;
- можливість зручного масштабування та обслуговування;
- мінімальні витрати на впровадження програмного продукту.

4.2. Обґрунтування функцій програмного продукту

Головна функція F_0 – розробка програмного продукту, який реалізує механізм дослідження ринку криптовалют методами глибокого хеджування та статистичного арбітражу. Беручи за основу цю функцію, можна виділити наступні:

F_1 – вибір мови програмування;

F_2 – вибір бібліотеки для роботи з вхідними даними;

F_3 – вибір бібліотеки для реалізації алгоритмів машинного навчання.

Кожна з цих функцій має декілька варіантів реалізації:

Функція F_1 :

а) Python

б) C++

Функція F_2 :

а) pandas;

б) zipline

Функція F_3 :

а) tensorflow;

б) theano.

Варіанти реалізації основних функцій наведені у морфологічній карті системи (рис. 4.1).

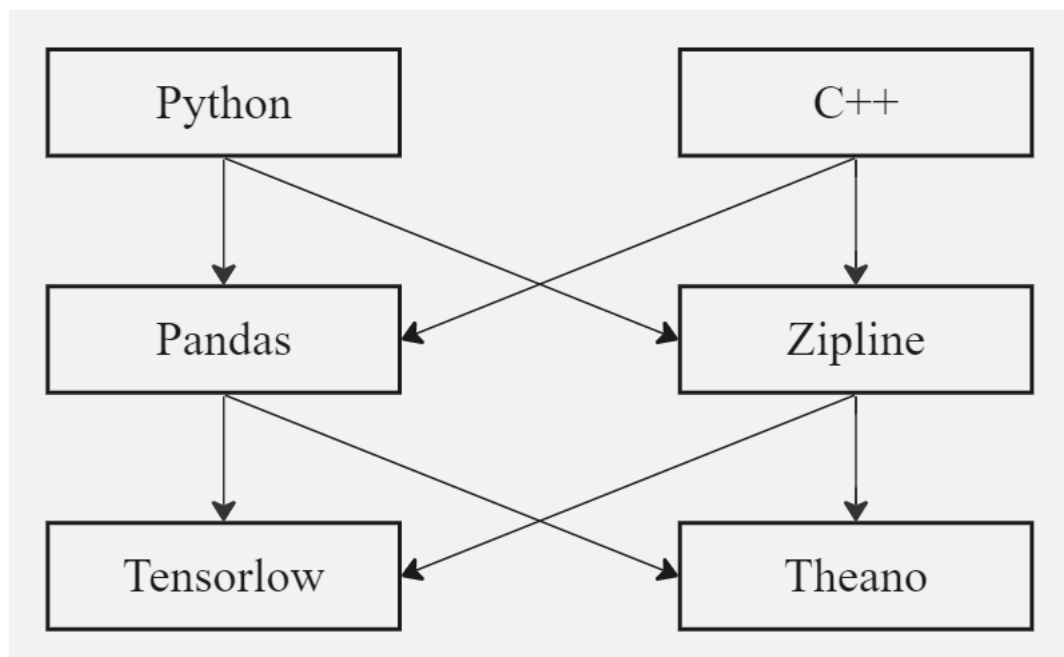


Рисунок 4.1 – Морфологічна карта

Морфологічна карта відображає множину всіх можливих варіанти основних функцій. Позитивно-негативна матриця наведена у таблиці 4.1

Таблиця 4.1. Позитивно-негативна матриця

Функції	Варіанти реалізації	Переваги	Недоліки
F_1	A	Велика кількість вбудованих методів, зручна та швидка реалізація програми. Доступність бібліотек та кросплатформеність.	Висока вартість розробки. Повільне виконання написаного коду, якщо мова йде про велику кількість даних.
	B	Швидка робота програмного продукту.	Збільшення часу затраченого на розробку програми, мала кількість вбудованих алгоритмів. Об'ємність написаного коду.
F_2	A	Широкий функціонал, кросплатформеність.	Складність при тестуванні розробленого продукту.
	B	Більша швидкість розробки.	Вузький функціонал.
F_3	A	Більша кількість вбудованих методів, можливість створення складних моделей.	Ускладнений підбір гіперпараметрів, більший час затрачений на розробку.
	B	Дозволяє оптимізувати час роботи програми.	Вужчий функціонал, обмежені можливості застосування при розробці.

На основі побудованої позитивно-негативної матриці робимо висновок, що при розробці програмного продукту деякі варіанти реалізації функцій варто відкинути, тому що вони не відповідають поставленим перед програмним продуктом задачам. Ці варіанти відзначені у морфологічній карті.

Функція F_1 :

Перевагу віддаємо швидкості розробки продукту, легкості у застосуванні та наявності вбудованих методів необхідних для реалізації програми. Для спрощення роботи по написанню коду варіант Б має бути відкинутий.

Функція F_2 :

Програма допускає вибір обох варіантів, тож можемо розглядати як варіант А так і варіант Б.

Функція F_3 :

Слід віддати перевагу варіанту А, оскільки програмний продукт передбачає використання досить складних моделей машинного навчання.

Таким чином, будемо розглядати такий варіанти реалізації ПП:

$$F_1a - F_2a - F_3a$$

$$F_1a - F_2b - F_3a$$

Для оцінювання якості розглянутих функцій обрана система параметрів, описана нижче.

4.3. Обґрунтування системи параметрів ПП

На основі даних, розглянутих вище, визначаються основні параметри вибору, які будуть використані для розрахунку коефіцієнта технічного рівня.

Для того, щоб охарактеризувати програмний продукт, будемо використовувати наступні параметри:

- X_1 – швидкодія мови програмування;
- X_2 – об’єм пам’яті для обчислень та збереження даних;
- X_3 – час навчання даних;
- X_4 – потенційний об’єм програмного коду.

Гірші, середні і кращі значення параметрів вибираються на основі вимог замовника й умов, що характеризують експлуатацію ПП як показано у таблиці 4.2.

Таблиця 4.2 – Основні параметри ПП

Назва Параметра	Умовні позначення	Одиниці виміру	Значення параметра		
			гірші	середні	кращі
Швидкодія мови програмування	X_1	оп/мс	800	450	300
Об’єм пам’яті	X_2	Мб	512	256	128
Час попередньої обробки даних	X_3	мс	30	17	13
Потенційний об’єм програмного коду	X_4	кількість рядків коду	3000	1500	1000

За даними таблиці 4.1 будуються графічні характеристики параметрів – рис. 4.2 – рис. 4.5.

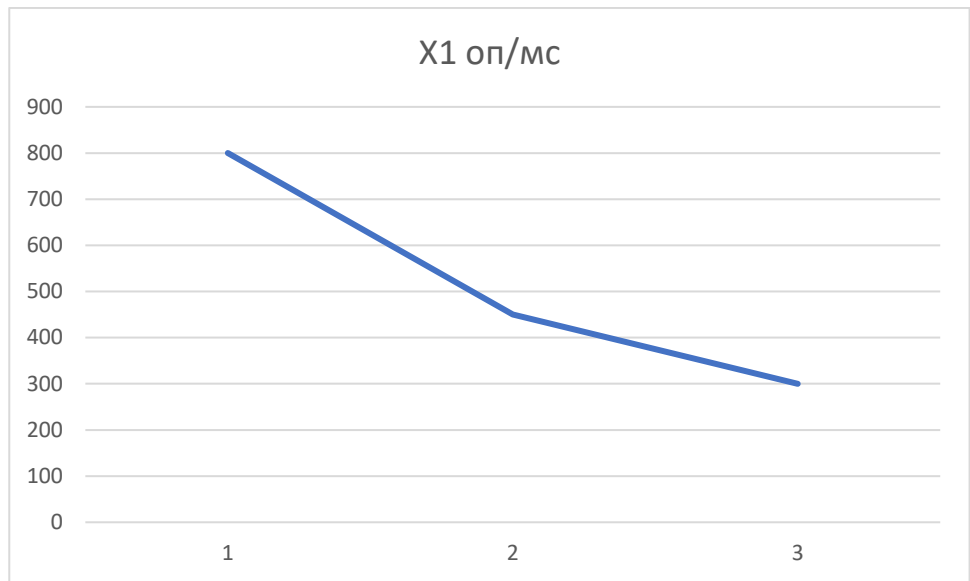


Рисунок 4.2 – X1, швидкодія мови програмування

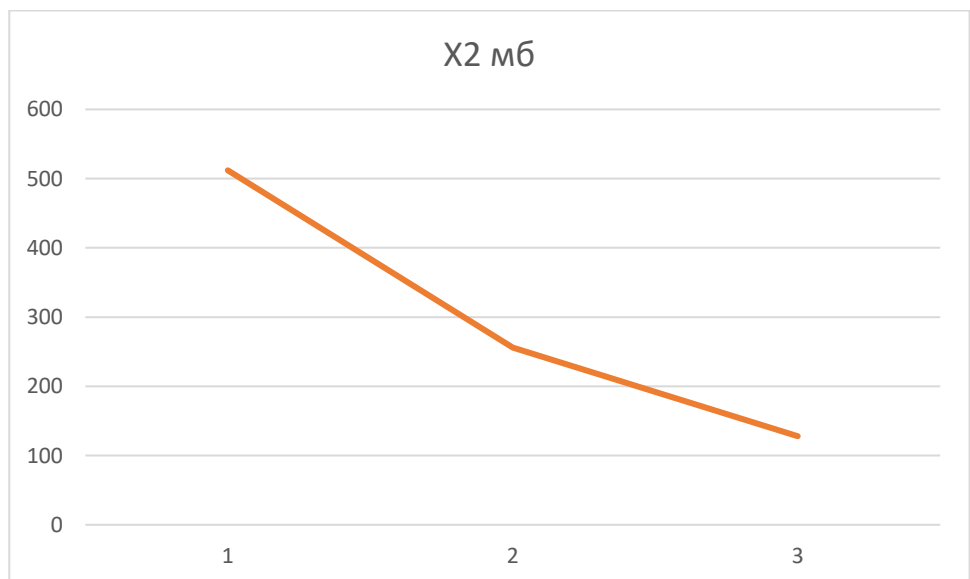


Рисунок 4.3 – X2, об'єм пам'яті

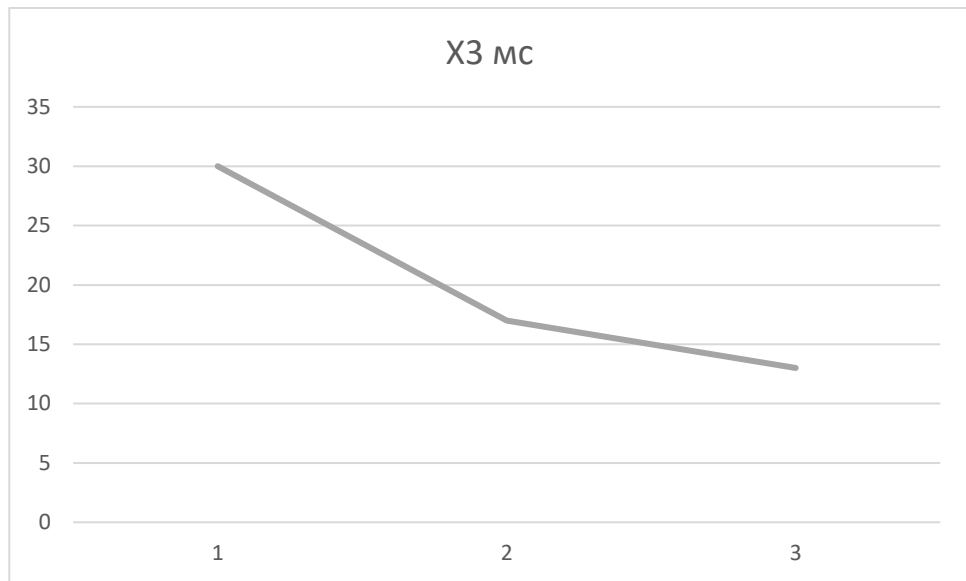


Рисунок 4.4 – X3, час попередньої обробки даних

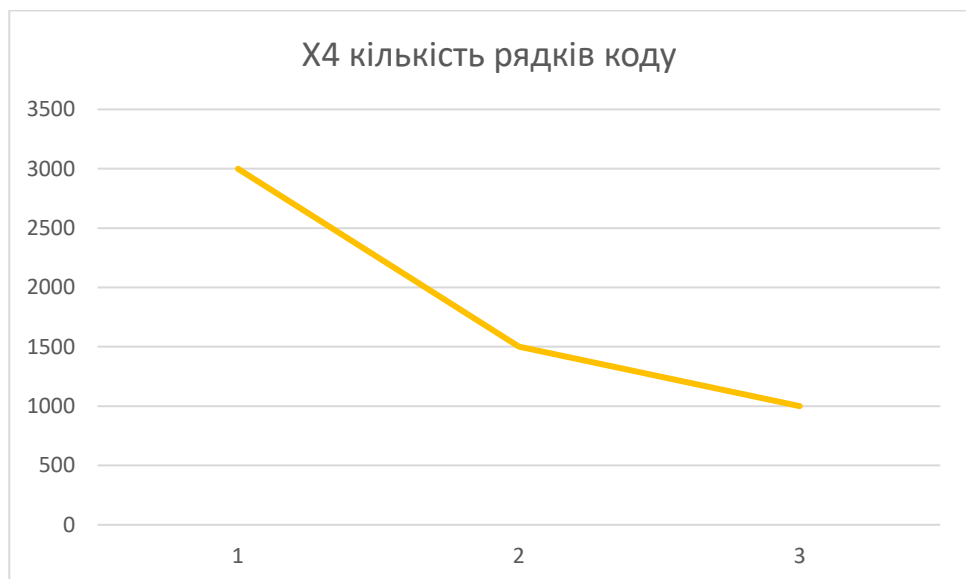


Рисунок 4.5 – X4, потенційний об'єм програмного коду

4.4. Аналіз експертного оцінювання параметрів

Після детального обговорення й аналізу кожний експерт оцінює ступінь важливості кожного параметру для конкретно поставленої цілі – розробка програмного продукту, який дає найбільш точні результати при знаходженні параметрів моделей адаптивного прогнозування і обчислення прогнозних значень.

Значимість кожного параметра визначається методом попарного порівняння. Оцінку проводить експертна комісія із 7 людей. Визначення коефіцієнтів значимості передбачає:

- визначення рівня значимості параметра шляхом присвоєння різних рангів;
- перевірку придатності експертних оцінок для подальшого використання;
- визначення оцінки попарного пріоритету параметрів;
- обробку результатів та визначення коефіцієнту значимості.

Результати експертного ранжування наведені у таблиці 4.3.

Таблиця 4.3 – Результати ранжування параметрів

Позначення параметра	Назва параметра	Одиниці виміру	Ранг параметра за оцінкою експерта							Сума рангів R_i	Відхилення Δ_i	Δ_i^2
			1	2	3	4	5	6	7			
X_1	Швидкодія мови програмування	Оп/мс	2	1	1	2	1	2	1	10	-7,5	56,25
X_2	Об'єм пам'яті	Мб	4	3	2	5	4	4	3	25	7,5	56,25
X_3	Час попередньої обробки даних	мс	1	1	2	1	2	2	2	11	-6,5	42,25
X_4	Потенційний об'єм програмного коду	Кількість рядків коду	2	4	4	3	3	3	4	23	5,5	30,25
	Разом		9	9	10	11	11	9	10	70	0	185

Для перевірки степені достовірності експертних оцінок, визначимо наступні параметри:

а) сума рангів кожного з параметрів і загальна сума рангів:

$$R_i = \sum_{j=1}^N r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 70, \quad (4.1)$$

де N – число експертів,

n – кількість параметрів;

б) середня сума рангів:

$$T = \frac{1}{n} R_{ij} = 17,5 \quad (4.2)$$

в) відхилення суми рангів кожного параметра від середньої суми рангів:

$$\Delta_i = R_i - T. \quad (4.3)$$

Сума відхилень по всіх параметрах повинна дорівнювати 0;

г) загальна сума квадратів відхилення:

$$S = \sum_{i=1}^N \Delta_i^2 = 185. \quad (4.4)$$

Порахуємо коефіцієнт узгодженості:

$$W = \frac{12S}{N^2(n^3 - n)} = \frac{12 \cdot 185}{7^2(4^3 - 4)} = 0,75 > W_k = 0,67. \quad (4.5)$$

Ранжування можна вважати достовірним, тому що знайдений коефіцієнт узгодженості перевищує нормативний, котрий дорівнює 0,67.

Скориставшись результатами ранжування, проведемо попарне порівняння всіх параметрів і результати занесемо у таблицю 4.4.

Таблиця 4.4 – Попарне порівняння параметрів

Параметри	Експерти							Кінцева оцінка	Числове значення
	1	2	3	4	5	6	7		
X1 і X2	<	<	<	<	<	<	<	<	0,5
X1 і X3	>	=	<	>	<	=	<	<	0,5
X1 і X4	=	<	<	<	<	<	<	<	0,5
X2 і X3	>	>	=	>	>	>	>	>	1,5
X2 і X4	>	<	<	>	>	>	<	>	1,5
X3 і X4	<	<	<	<	<	<	<	<	0,5

Числове значення, що визначає ступінь переваги i -го параметра над j -тим, a_{ij} визначається по формулі:

$$a_{ij} = \begin{cases} 1.5 \text{ при } X_i > X_j \\ 1.0 \text{ при } X_i = X_j \\ 0.5 \text{ при } X_i < X_j \end{cases} \quad (4.6)$$

З отриманих числових оцінок переваги складемо матрицю $A = \| a_{ij} \|$.

Для кожного параметра зробимо розрахунок вагомості K_{bi} за наступними формулами:

$$K_{bi} = \frac{b_i}{\sum_{i=1}^n b_i} \quad (4.7)$$

$$b_i = \sum_{j=1}^N a_{ij} \quad (4.8)$$

Відносні оцінки розраховуються декілька разів доти, поки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%). На другому і наступних кроках відносні оцінки розраховуються за наступними формулами:

$$K_{Bi} = \frac{b'_i}{\sum_{i=1}^n b'_i}, \quad (4.9)$$

$$b'_i = \sum_{j=1}^N a_{ij} b_j \quad (4.10)$$

Як видно з таблиці 4.5, різниця значень коефіцієнтів вагомості не перевищує 2%, тому більшої кількості ітерацій не потрібно.

Таблиця 4.5 – Розрахунок вагомості параметрів

Параметри x_i	Параметри x_j				Перша ітер.		Друга ітер.		Третя ітер	
	X1	X2	X3	X4	b_i	K_{Bi}	b_i^1	K_{Bi}^1	b_i^2	K_{Bi}^2
X1	1	0,5	0,5	0,5	2,5	0,16	9,25	0,16	34,125	0,16
X2	1,5	1	1,5	1,5	5,5	0,34	21,25	0,36	77,875	0,36
X3	1,5	0,5	1	0,5	3,5	0,22	12,25	0,21	41,875	0,20
X4	1,5	0,5	1,5	1	4,5	0,28	16,25	0,27	59,125	0,28
Всього:					16	1	59	1	213	1

4.5 Аналіз рівня якості варіантів реалізації функцій

Визначаємо рівень якості кожного варіанту виконання основних функцій окремо.

Абсолютні значення параметрів X_2 (Об'єм пам'яті), X_3 (час попередньої обробки даних) та X_4 (потенційний об'єм програмного коду) відповідають технічним вимогам умов функціонування даного ПП.

Абсолютне значення параметра X_1 (швидкість роботи мови програмування) обрано не найгіршим.

Коефіцієнт технічного рівня для кожного варіанта реалізації ПП розраховується так (таблиця 5.7):

$$K_K(j) = \sum_{i=1}^n K_{ei,j} B_i, \quad (5.11)$$

де n – кількість параметрів;

K_{ei} – коефіцієнт вагомості i -го параметра;

B_i – оцінка i -го параметра в балах.

Таблиця 4.6 - Розрахунок показників рівня якості варіантів реалізації основних функцій ПП

Основні функції	Варіант реалізації функції	Параметри	Абсолютне значення параметра	Бальна оцінка параметра	Коефіцієнт вагомості параметра	Коефіцієнт рівня якості
F1	A	X1	800	33	0,16	5,28
F2	A	X2	256	28	0,36	10,08
	Б	X3	128	16	0,20	3,2
F3	A	X4	100	25	0,28	7

За даними з таблиці 4.6 за формулою:

$$K_K = K_{\text{ТУ}}[F_{1k}] + K_{\text{ТУ}}[F_{2k}] + \dots + K_{\text{ТУ}}[F_{zk}], \quad (4.12)$$

визначаємо рівень якості кожного з варіантів:

$$K_{K1} = 5,28 + 10,08 + 7 = 22,36,$$

$$K_{K2} = 5,28 + 3,2 + 7 = 15,48.$$

Як видно з розрахунків, кращим є перший варіант, для якого коефіцієнт технічного рівня має найбільше значення.

4.6 Економічний аналіз варіантів розробки ПП

Для визначення вартості розробки ПП спочатку проведемо розрахунок трудомісткості.

Всі варіанти включають в себе два окремих завдання:

1. Розробка проекту програмного продукту;
2. Розробка програмної оболонки;

Завдання 1 за ступенем новизни відноситься до групи А, завдання 2 – до групи Б. За складністю алгоритми, які використовуються в завданні 1 належать до групи 1; а в завданні 2 – до групи 3.

Для реалізації завдання 1 використовується довідкова інформація, а завдання 2 використовує інформацію у вигляді даних.

Проведемо розрахунок норм часу на розробку та програмування для кожного з завдань.

Загальна трудомісткість обчислюється як

$$T_0 = T_p \cdot K_{\Pi} \cdot K_{СК} \cdot K_M \cdot K_{СТ} \cdot K_{СТ.М},$$

де T_p – трудомісткість розробки ПП;

K_{Π} – поправочний коефіцієнт;

$K_{СК}$ – коефіцієнт на складність вхідної інформації;

K_M – коефіцієнт рівня мови програмування;

$K_{СТ}$ – коефіцієнт використання стандартних модулів і прикладних програм;

$K_{СТ.М}$ – коефіцієнт стандартного математичного забезпечення

Для першого завдання, виходячи із норм часу для завдань розрахункового характеру степеню новизни А та групи складності алгоритму 1, трудомісткість дорівнює: $T_p = 90$ людино-днів. Поправочний коефіцієнт, який враховує вид нормативно-довідкової інформації для першого завдання: $K_{\Pi} = 1.6$. Поправочний коефіцієнт, який враховує складність контролю вхідної та вихідної інформації для всіх семи завдань рівний 1: $K_{СК} = 1$. Оскільки при розробці першого завдання використовуються стандартні модулі, врахуємо це за допомогою коефіцієнта $K_{СТ} = 0.8$. Тоді загальна трудомісткість програмування першого завдання дорівнює:

$$T_1 = 90 \cdot 1.6 \cdot 0.8 = 115.2 \text{ людино-днів.}$$

Проведемо аналогічні розрахунки для подальших завдань.

Для другого завдання (використовується алгоритм третьої групи складності, степені новизни Б), тобто $T_p = 27$ людино-днів, $K_{\Pi} = 0.9$, $K_{СК} = 1$, $K_{СТ} = 0.8$:

$$T_2 = 27 \cdot 0.9 \cdot 0.8 = 19.44 \text{ людино-днів.}$$

Складаємо трудомісткість відповідних завдань для кожного з обраних варіантів реалізації програми, щоб отримати їх трудомісткість:

$$T_I = (115.2 + 19.44 + 4.8 + 19.44) \cdot 8 = 1271,04 \text{ людино-годин.}$$

$$T_{II} = (115.2 + 19.44 + 6.91 + 19.44) \cdot 8 = 1287,92 \text{ людино-годин.}$$

Найбільш високу трудомісткість має варіант II.

В розробці беруть участь два програмісти з окладом 18 500 грн., один аналітик в області даних з окладом 16 300. Визначимо середню зарплату за годину за формулою:

$$C_q = \frac{M}{T_m \cdot t} \text{ грн.,} \quad (4.14)$$

де M – місячний оклад працівників;

T_m – кількість робочих днів тиждень;

t – кількість робочих годин в день.

$$C_q = \frac{18500 + 18500 + 16300}{5 \cdot 21 \cdot 8} = 63,45 \text{ грн.} \quad (4.15)$$

Тоді, розрахуємо заробітну плату за формулою:

$$C_{зп} = C_q \cdot T_i \cdot K_d, \quad (4.16)$$

де C_q – величина погодинної оплати праці програміста;

T_i – трудомісткість відповідного завдання;

K_d – норматив, який враховує додаткову заробітну плату.

Зарплата розробників за варіантами становить:

$$\text{I. } C_{\text{ЗП}} = 63,45 \cdot 1271,04 \cdot 1,2 = 96776,99 \text{ грн.}$$

$$\text{II. } C_{\text{ЗП}} = 63,45 \cdot 1287,92 \cdot 1,2 = 98062,23 \text{ грн.}$$

Відрахування на єдиний соціальний внесок становить 22%:

$$\text{I. } C_{\text{ВД}} = C_{\text{ЗП}} \cdot 0,22 = 96776,99 \cdot 0,22 = 21290,94 \text{ грн.}$$

$$\text{II. } C_{\text{ВД}} = C_{\text{ЗП}} \cdot 0,22 = 98062,23 \cdot 0,22 = 21573,69 \text{ грн.}$$

Тепер визначимо витрати на оплату однієї машино-години. (C_M)

Так як одна ЕОМ обслуговує одного програміста з окладом 13500 грн., з коефіцієнтом зайнятості 0,2 то для однієї машини отримаємо:

$$C_M = 12 \cdot M \cdot K_3 = 12 \cdot 13500 \cdot 0,2 = 32400 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$C_{\text{ЗП}} = C_M \cdot (1 + K_3) = 32400 \cdot (1 + 0,2) = 38880 \text{ грн.}$$

Відрахування на соціальний внесок:

$$C_{\text{ВД}} = C_{\text{ЗП}} \cdot 0,22 = 38880 \cdot 0,22 = 8553,6 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 24000 грн.

$$C_A = K_{TM} \cdot K_A \cdot C_{ПР} = 1.15 \cdot 0.25 \cdot 24000 = 6900 \text{ грн.},$$

де K_{TM} — коефіцієнт, який враховує витрати на транспортування та монтаж приладу у користувача;
 K_A — річна норма амортизації;
 $C_{ПР}$ — договірна ціна приладу.

Витрати на ремонт та профілактику розраховуємо як:

$$C_P = K_{TM} \cdot C_{ПР} \cdot K_P = 1.15 \cdot 24000 \cdot 0.05 = 1380 \text{ грн.},$$

де K_P — відсоток витрат на поточні ремонти.

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$$T_{\text{ЕФ}} = (D_K - D_B - D_C - D_P) \cdot t_3 \cdot K_B = (365 - 104 - 12 - 9) \cdot 8 \cdot 0.9 = \\ = 1728 \text{ годин},$$

де D_K — календарна кількість днів у році;
 D_B, D_C — відповідно кількість вихідних та святкових днів;
 D_P — кількість днів планових ремонтів устаткування;
 t — кількість робочих годин в день;
 K_B — коефіцієнт використання приладу у часі протягом зміни.

Витрати на оплату електроенергії розраховуємо за формулою:

$$C_{\text{ЕЛ}} = T_{\text{ЕФ}} \cdot N_C \cdot K_3 \cdot C_{\text{ЕН}} = 1728 \cdot 0,3 \cdot 4,87 \cdot 1,2 = 3029,53 \text{ грн.},$$

де N_C — середньо-споживча потужність приладу;

K_3 – коефіцієнтом зайнятості приладу;

$\Pi_{ЕН}$ – тариф за 1 КВт-годин електроенергії.

Накладні витрати розраховуємо за формулою:

$$C_H = \Pi_{ПР} \cdot 0.67 = 24000 \cdot 0,67 = 16080 \text{ грн.}$$

Тоді, річні експлуатаційні витрати будуть:

$$C_{ЕКС} = C_{ЗП} + C_{ВІД} + C_A + C_P + C_{ЕЛ} + C_H, \quad (4.17)$$

$$C_{ЕКС} = 38880 + 8553.6 + 6900 + 1380 + 3029,53 + 16080 = 74823,13 \text{ грн.}$$

Собівартість однієї машино-години ЕОМ дорівнюватиме:

$$C_{М-Г} = C_{ЕКС} / T_{ЕФ} = 74823,13 / 1728 = 43,3 \text{ грн/год.}$$

Оскільки в даному випадку всі роботи, які пов'язані з розробкою програмного продукту ведуться на ЕОМ, витрати на оплату машинного часу, в залежності від обраного варіанта реалізації, складає:

$$C_M = C_{М-Г} \cdot T, \quad (4.18)$$

$$\text{I. } C_M = 43,3 \cdot 1271,04 = 55036,032 \text{ грн.}$$

$$\text{II. } C_M = 43,3 \cdot 1287,92 = 55766,936 \text{ грн.}$$

Накладні витрати складають 67% від заробітної плати:

$$C_H = C_{3П} \cdot 0,67, \quad (4.19)$$

$$\text{I. } C_H = 96776,99 \cdot 0,67 = 64840,58 \text{ грн.}$$

$$\text{II. } C_H = 98062,23 \cdot 0,67 = 65701,69 \text{ грн.}$$

Отже, вартість розробки ПП за варіантами становить:

$$C_{ПП} = C_{3П} + C_{ВІД} + C_M + C_H, \quad (4.20)$$

$$\text{I. } C_{ПП} = 96776,99 + 21290,94 + 55036,032 + 64840,58 = 237944,542 \text{ грн.}$$

$$\text{II. } C_{ПП} = 98062,23 + 21573,69 + 55766,936 + 65701,69 = 241104,546 \text{ грн.}$$

4.7 Вибір кращого варіанту ПП техніко-економічного рівня

Розрахуємо коефіцієнт техніко-економічного рівня за формулою:

$$K_{\text{ТЕР}j} = K_{kj} / C_{\Phi j}, \quad (4.21)$$

$$K_{\text{ТЕР}1} = 22,36 / 237944,542 = 9,39 \cdot 10^{-5},$$

$$K_{\text{TEP}2} = 15,48 / 241104,546 = 6,42 \cdot 10^{-5}.$$

Як бачимо, найбільш ефективним є перший варіант реалізації програми з коефіцієнтом техніко-економічного рівня $K_{\text{TEP}1} = 9,39 \cdot 10^{-5}$.

Після виконання функціонально-вартісного аналізу програмного комплексу що розроблюється, можна зробити висновок, що з альтернатив, що залишились після першого відбору двох варіантів виконання програмного комплексу оптимальним є перший варіант реалізації програмного продукту.

Висновки

У даному розділі було проведено функціонально-вартісний аналіз розробленого програмного продукту. Процес аналізу складався з двох частин – дослідження продукту з технічної точки зору, визначення основних характеристик та формування множини варіантів їх реалізації. Спираючись на обчислені значення параметрів, а також експертні оцінки їх важливості було обчислено коефіцієнт технічного рівня, який і дав змогу визначити оптимальну з технічної точки зору альтернативу реалізації функцій ПП.

Друга частина аналізу була присвячена вибору із альтернативних варіантів реалізації найбільш економічно обґрунтованого. Порівняння запропонованих варіантів реалізації в рамках даної частини виконувалось за коефіцієнтом ефективності, для обчислення якого були обчислені такі допоміжні параметри, як трудомісткість, витрати на заробітну плату, накладні витрати.

В цілому, за результатами обох частин функціонально-вартісного аналізу було зроблено висновок про те, що оптимальним є перший варіант

програмної реалізації, що розглядається. У нього виявився найкращий показник техніко-економічного рівня якості $K_{\text{ТЕР}} = 9,39 \cdot 10^{-5}$.

Цей варіант реалізації програмного продукту має такі параметри:

- мова програмування – Python;
- використання бібліотеки Pandas;
- використання бібліотеки Tensorflow для побудови складних моделей;
- використання стандартного інтерфейсу візуалізації, швидкість розробки .

Даний варіант виконання програмного комплексу дає досліднику зручний інтерфейс, широкий функціонал і швидкодію.

ВИСНОВКИ

У даній дипломній роботі було проведено дослідження ринку криптовалют з використанням статистичного арбітражу та глибокого хеджування. Ми детально розглянули механізм роботи даних стратегій та визначити можливості їх застосування на крипторинку.

Дослідження проводилося на основі двох наборів даних – датасету з дослідженими раніше криптовалютами та реальному користувачькому гаманці. На першому датасеті, як і очікувалось, ми підтвердили наявність арбітражних можливостей на крипторинку та показали, що з використанням глибокого хеджування дійсно можна захистити навіть такі нестабільні активи та зробити їх придатними для подальшої стратегії. Проводячи дослідження на другому наборі даних, було виявлено, що все ж дана комбінація методів не є на 100 відсотків універсальною, та вимагає хоча б мінімального аналізу криптовалют. Так, для занадто нових та нестабільних цифрових валют застосувати комбінацію глибокого хеджування та статистичного арбітражу ефективно не вдалося.

Отримані результати дозволяють зробити припущення про перспективи розвитку ринку криптовалют, як світового фінансового ринку, оскільки можливість прохеджувати активи і максимізувати прибуток, робить даний тип валют привабливим для широкого застосування. У подальших дослідженнях планується покращувати механізм глибокого хеджування та зробити його придатним для будь-яких криптовалют у гаманці, а також розширити зроблені висновки на більшу кількість арбітражних стратегій.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

- 1) Cryptocurrency vs. Gold: Which Is the Better Hedge Against Inflation? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.smartfinancefreedom.com/post/cryptocurrency-vs-gold-which-is-the-better-hedge-against-inflation>
- 2) Crypto Is Still a Mess. A Crackdown Would Do It Good. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.bloomberg.com/news/articles/2023-05-02/crypto-is-still-a-mess-a-crackdown-would-do-it-good>
- 3) Crypto Crackdown: Here Are All The Major Crypto Firms Facing Charges From Regulators This Year. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.forbes.com/sites/tylerroush/2023/03/29/crypto-crackdown-here-are-all-the-major-crypto-firms-facing-charges-from-regulators-this-year/?sh=72c8c1ee4727>
- 4) May 2023 Crypto Market Forecast. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.forbes.com/advisor/investing/cryptocurrency/crypto-market-outlook-forecast/>
- 5) Stop-loss strategies: Where to put a stop loss. [Електронний ресурс] – Режим доступу до ресурсу: <https://capital.com/stop-loss-strategies>
- 6) Yan Li, Wei Dai. Bitcoin price forecasting method based on CNN-LSTM hybrid neural network model / Yan Li, Wei Dai, 2020
- 7) H. Buehler, L. Gonon, J. Teichmann & B. Wood. Deep Hedging / H. Buehler, L. Gonon, J. Teichmann & B. Wood, 2019
- 8) H. Buehler, P. Murray, B. Wood. Deep Bellman Hedging / H. Buehler, P. Murray, B. Wood, 2022
- 9) Tim Leung, Hung Nguen. Constructing cointegrated cryptocurrencies portfolio for statistical arbitrage. / Tim Leung, Hung Nguen, 2019

- 10) T. Fischer, C. Krauss, A. Deinert. Statistical arbitrage in cryptocurrency markets. / T. Fischer, C. Krauss, A. Deinert, 2019
- 11) H. Ko, B. Son, J. Lee. Portfolio insurance strategy in the cryptocurrency market. / H.Ko, B.Son, J. Lee, 2023
- 12) J. Guijarro-Ordóñez, Markus Pelger, Greg Zanotti. Deep Learning Statistical arbitrage. / J. Guijarro-Ordóñez, Markus Pelger, Greg Zanotti, 2022
- 13) I. Makarov, A. Schoar. Trading and arbitrage in cryptocurrency market. / I.Makarov, A. Schoar, 2018
- 14) J. Wang, J. Fang. The application of Ornstein-Uhlenbeck Process model and ARCH/GARCH model in statistical arbitrage. / J. Wang, J. Fang, 2017
- 15) G. Keilbar, Y. Zhang. On cointegration and cryptocurrency dynamics. / G. Keilbar, Y. Zhang, 2021
- 16) Stationarity and detrending [Электронный ресурс] – Режим доступа до ресурсу:
https://www.statsmodels.org/dev/examples/notebooks/generated/stationarity_detrending_adf_kpss.html
- 17) W. Lu, J. Li, Y. Li, A Sun, J. Wang. A CNN-LSTM Model to Forecast Stock Price. / W. Lu, J. Li, Y. Li, A Sun, J. Wang, 2020
- 18) D. Vezeris, C. Schinas, T. Kyrgos. Take Profit and Stop-Loss Trading Strategies Comparison in Combination with an MACD Trading System. / D. Vezeris, C. Schinas, T. Kyrgos, 2018
- 19) V. Kochliaridis, E. Koluloumpris, I. Vlahavas. Combining deep reinforcement learning with technical analysis and trend monitoring on cryptocurrency markets. / V. Kochliaridis, E. Koluloumpris, I. Vlahavas, 2023
- 20) Yuh-Dauh Lyuu. Financial Engineering and Computation. Principles, Mathematics, Algorithms. / Yuh-Dauh Lyuu – Cambridge: CAMBRIDGE UNIVEERSITY PRESS, 2021. – 738 с. – (4th).
- 21) W. McKinney. Python for Data Analysis. / W. McKinney – Sebastopol: O'Reilly Media Inc., 2015 – 316-380 с.

- 22) I. Goodfellow, Y. Bengio, A. Courville. Deep Learning / I. Goodfellow, Y. Bengio, A. Courville – Cambridge: THE MIT PRESS, 2018 – 653 с.
- 23) Sebastian Raschka, Vahid Mirjalili. Python Machine Learning. / S. Raschka, V. Mirjalili – Birmingham: Packt, 2020 – 377 – 417 с. – (3rd)
- 24) A. Mishra. Metrics to Evaluate your Machine Learning Algorithm. [Электронный ресурс] – Режим доступа до ресурсу: <https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234>
- 25) T. Segal. Measuring Portfolio Performance. [Электронный ресурс] – Режим доступа до ресурсу: <https://www.investopedia.com/articles/08/performance-measure.asp>
- 26) Y. Xia, C. Sang, L. He, Z. Wang. The role of uncertainty index in forecasting volatility of Bitcoin: Fresh evidence from GARCH-MIDAS approach / Y. Xia, C. Sang, L. He, Z. Wang, 2023
- 27) S.F. Ramli, Z.A. Adzhar, S. Bashar, M.F. Abdullah. Analysis of Ethereum versus Bitcoin: The GARCH approach. [Электронный ресурс] – Режим доступа до ресурсу: <https://pubs.aip.org/aip/acp/article/2500/1/020049/2875243/Analysis-of-Ethereum-versus-Bitcoin-The-GARCH>
- 28) Crypto Winter ‘Finally Over’—Bank Crisis A Catalyst For Huge \$2 Trillion Bitcoin Prediction As Ethereum Price Suddenly Storms Back. [Электронный ресурс] – Режим доступа до ресурсу: <https://www.forbes.com/sites/digital-assets/2023/04/26/crypto-winter-finally-over-bank-crisis-a-catalyst-for-huge-2-trillion-bitcoin-prediction-as-ethereum-price-suddenly-storms-back/?sh=2046138346e3>
- 29) Cointegration among cryptocurrencies. [Электронный ресурс] – Режим доступа до ресурсу: <https://www.diva-portal.org/smash/get/diva2:1331682/FULLTEXT01.pdf>

ДОДАТОК А

Лістинг програмного продукту:

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import ffn
import matplotlib
from sklearn import linear_model
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_error,
mean_absolute_percentage_error, f1_score
import tensorflow as tf
from sklearn.preprocessing import StandardScaler, MinMaxScaler
from statsmodels.tsa.stattools import adfuller, kpss
from arch import arch_model
data = pd.read_csv('data.csv')
data
#CNN-LSTM
eth = pd.DataFrame()

df = data[data['Name']=='ETH']
df = df[::-1]

eth['Date'] = pd.to_datetime(df['Date']).values
eth['Price'] = [float(value.replace(',', '')) for value in df['Price']]
eth['Open'] = [float(value.replace(',', '')) for value in df['Open']]
eth['High'] = [float(value.replace(',', '')) for value in df['High']]
eth['Low'] = [float(value.replace(',', '')) for value in df['Low']]

eth
y = eth['Price']
X1 = eth.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)
X
scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)
X.head()
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)
X_train.shape
from tensorflow.keras import Sequential,utils

```

```

from tensorflow.keras.layers import Flatten, Dense, Conv1D, MaxPool1D, Dropout,
LSTM, TimeDistributed, Reshape

def reg():

    model = Sequential()

    model.add(LSTM(128, return_sequences=True, activation = 'tanh', input_shape
= (3,1)))
    model.add(TimeDistributed(Reshape((128, 1))))
    model.add(Conv1D(128, kernel_size=(3,), padding='same', activation='relu'))
    model.add(Dropout(0.5))
    model.add(Conv1D(64, kernel_size=(3,), padding='same', activation='relu'))
    model.add(Dropout(0.5))
    model.add(Conv1D(32, kernel_size=(5,), padding='same', activation='relu'))
    model.add(Dropout(0.5))

    model.add(Flatten())
    model.add(Dense(128, activation='relu'))
    model.add(Dense(32, activation='relu'))
    model.add(Dense(1))

    model.compile(loss='mean_squared_error', optimizer='adam')

    return model
model_3 = reg()
model_3.fit(X_train, y_train, epochs=200, validation_split=0.1)
predictions_eth = model_3.predict(X_test)
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error
predictions_eth21 = predictions_eth[1:]
y_test1 = y_test[:-1]
print("MSE: {0}".format(mean_squared_error(y_test1, predictions_eth21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_eth21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_eth21)))
btc = pd.DataFrame()

df = data[data['Name']=='BTC']
df = df[:-1]

btc['Date'] = pd.to_datetime(df['Date']).values
btc['Price'] = [float(value.replace(',',' ')) for value in df['Price']]
btc['Open'] = [float(value.replace(',',' ')) for value in df['Open']]
btc['High'] = [float(value.replace(',',' ')) for value in df['High']]
btc['Low'] = [float(value.replace(',',' ')) for value in df['Low']]

y = btc['Price']

```

```

X1 = btc.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=200, validation_split=0.1)

predictions_btc = model_3.predict(X_test)
predictions_btc21 = predictions_btc[1:]
y_test1 = y_test[:-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_btc21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_btc21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_btc21)))
bch = pd.DataFrame()

df = data[data['Name']=='BCH']
df = df[:-1]

bch['Date'] = pd.to_datetime(df['Date']).values
bch['Price'] = [float(value.replace(',', '')) for value in df['Price']]
bch['Open'] = [float(value.replace(',', '')) for value in df['Open']]
bch['High'] = [float(value.replace(',', '')) for value in df['High']]
bch['Low'] = [float(value.replace(',', '')) for value in df['Low']]

y = bch['Price']
X1 = bch.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=200, validation_split=0.1)

predictions_bch = model_3.predict(X_test)
predictions_bch21 = predictions_bch[1:]
y_test1 = y_test[:-1]

```

```

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_bch21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_bch21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_bch21)))
etc = pd.DataFrame()

df = data[data['Name']=='ETC']
df = df[::-1]

etc['Date'] = pd.to_datetime(df['Date']).values
etc['Price'] = [float(value.replace(',','')) for value in df['Price']]
etc['Open'] = [float(value.replace(',','')) for value in df['Open']]
etc['High'] = [float(value.replace(',','')) for value in df['High']]
etc['Low'] = [float(value.replace(',','')) for value in df['Low']]

y = etc['Price']
X1 = etc.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=200, validation_split=0.1)

predictions_etc = model_3.predict(X_test)
predictions_etc21 = predictions_etc[1:]
y_test1 = y_test[::-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_etc21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_etc21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_etc21)))
wbtc = pd.DataFrame()

df = data[data['Name']=='WBTC']
df = df[::-1]

wbtc['Date'] = pd.to_datetime(df['Date']).values
wbtc['Price'] = [float(value.replace(',','')) for value in df['Price']]
wbtc['Open'] = [float(value.replace(',','')) for value in df['Open']]
wbtc['High'] = [float(value.replace(',','')) for value in df['High']]

```

```

wbtc['Low'] = [float(value.replace(',','')) for value in df['Low']]

y = wbtc['Price']
X1 = wbtc.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=200, validation_split=0.1)

predictions_wbtc = model_3.predict(X_test)
predictions_wbtc21 = predictions_wbtc[1:]
y_test1 = y_test[:-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_wbtc21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_wbtc21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_wbtc21)))
#Regression
predicted_set0 = np.stack((predictions_wbtc, predictions_eth))
predicted_set1 = np.stack((predictions_bch, predictions_etc))
predicted_set = np.concatenate((predicted_set0, predicted_set1))
X = np.transpose(predicted_set)
X = X.reshape(31,4)
y = np.transpose(predictions_btc)
y = y.reshape(31,1)
newX = pd.DataFrame(X, columns = ['WBTC','BTC','BCH', 'ETC'])
newy = pd.DataFrame(y, columns =['ETH'])
xTrain, xTest, yTrain, yTest = train_test_split(newX, newy, test_size = 0.2,
random_state = 0)
regression = LinearRegression().fit(xTrain, yTrain)
yPred = regression.predict(xTest)
coefs = regression.coef_.flatten()
spread_parts=X*coefs
newsread = pd.DataFrame(spread_parts, columns = ['WBTC','BTC','BCH', 'ETC'])
spread = newy['ETH'] - newsread['WBTC'] - newsread['BTC'] - newsread['BCH']
- newsread['ETC']
adfuller(spread)
kpss(spread, regression = 'ct', nlags = 'auto')
start_date = pd.to_datetime('2023-05-01')
trading = pd.DataFrame()

```

```

trading['Spread'] = spread
trading['Date'] = pd.date_range(start=start_date, periods=len(trading))
plt.figure(figsize=(12, 9))
plt.plot(trading['Date'], trading['Spread'], label='Price')
plt.xlabel('Date')
plt.ylabel('Price')
plt.legend()
plt.show()

# Генерація торгових сигналів на основі спреду та GARCH моделі з динамічним
стоп-лосом

def generate_signals(spread):
    signals = []
    stop_loss_values_top = []
    stop_loss_values_low = []

    model = arch_model(spread, vol='Garch', p=1, q=1, rescale = False)
    result = model.fit(dispatch='off')
    mean = np.mean(spread)

    for i in range(len(spread)-1):
        # Прогнозування волатильності та обчислення верхньої межі (стоп-лосу)
на кожному кроці
        forecast_volatility = result.forecast(start=i, horizon=1)
        stop_loss_top = mean + forecast_volatility.variance.values[-1]**0.5
        stop_loss_low = mean - forecast_volatility.variance.values[-1]**0.5
        stop_loss_values_top.append(stop_loss_top)
        stop_loss_values_low.append(stop_loss_low)

        if spread.iloc[i] > stop_loss_top:
            signals.append('SELL, short')
        elif spread.iloc[i] < stop_loss_low:
            signals.append('BUY, Long')
        else:
            signals.append('HOLD')

        mean = (mean * i + spread.iloc[i]) / (i + 1)

    return signals, stop_loss_values_top, stop_loss_values_low
signals, stop_loss_values_top, stop_loss_values_low = generate_signals(spread)
plt.figure(figsize=(12, 9))
plt.plot(df['Date'], df['Spread'], label='Spread')
plt.plot(df['Date'], df['Stop Loss (Upper Bound)'], label='sl_top')
plt.plot(df['Date'], df['Stop Loss (Lower Bound)'], label='sl_low')
plt.xlabel('Date')
plt.ylabel('Price')
plt.legend()

```

```

plt.show()
cumulative_profit = [0] # Кумулятивний прибуток (початкове значення 0)
drawdown = [0] # Процент просідання (початкове значення 0)
peak = 0 # Пікове значення прибутку (початкове значення 0)
drawdown_duration = 0 # Тривалість просідання (початкове значення 0)

for i in range(1, len(signals)):
    if signals[i] == 'SELL, short':
        profit = (spread.iloc[i] - spread.iloc[1-1])/spread.iloc[0] #
        Розрахунок прибутку при продажу (short)
        cumulative_profit.append(cumulative_profit[-1]-profit)
    elif signals[i] == 'BUY, Long':
        profit = (spread.iloc[i-1] - spread.iloc[i])/spread.iloc[0] #
        Розрахунок прибутку при покупці (long)
        cumulative_profit.append(cumulative_profit[-1]+profit)
    else:
        cumulative_profit.append(cumulative_profit[-1]) # Якщо сигнал HOLD,
        прибуток залишається незмінним

profit = pd.DataFrame({'Cumulative_profit': cumulative_profit})
profit.insert(0, 'Date', trading['Date'])
plt.figure(figsize=(12, 9))
plt.plot(profit['Date'], profit['Cumulative_profit'], label='CProfit')
plt.xlabel('Date')
plt.ylabel('CProfit')
plt.legend()
plt.show()
profit['Date'] = pd.to_datetime(profit['Date'])
profit.set_index('Date', inplace=True)
ffn.to_drawdown_series(profit['Cumulative_profit']).plot(figsize=(15,7),grid=True)

perf = profit['Cumulative_profit'].calc_stats()

perf.display()
sharpe = (profit['Cumulative_profit'].mean() - 0.05) /
(profit['Cumulative_profit'].std()-0.05)
sharpe
data = pd.read_csv('data1.csv')
data
suku = pd.DataFrame()

df = data[data['Name']=='SUKU']
df = df[:-1]

suku['Date'] = pd.to_datetime(df['Date']).values
suku['Price'] = [float(value.replace(',',' ')) for value in df['Price']]

```

```

suku['Open'] = [float(value.replace(',', ' ')) for value in df['Open']]
suku['High'] = [float(value.replace(',', ' ')) for value in df['High']]
suku['Low'] = [float(value.replace(',', ' ')) for value in df['Low']]

y = suku['Price']
X1 = suku.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=1000, validation_split=0.1)

predictions_suku = model_3.predict(X_test)
predictions_suku21 = predictions_suku[1:]
y_test1 = y_test[:-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_suku21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_suku21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_suku21)))
dext = pd.DataFrame()

df = data[data['Name']=='DEXT']
df = df[:-1]

dext['Date'] = pd.to_datetime(df['Date']).values
dext['Price'] = [float(value.replace(',', ' ')) for value in df['Price']]
dext['Open'] = [float(value.replace(',', ' ')) for value in df['Open']]
dext['High'] = [float(value.replace(',', ' ')) for value in df['High']]
dext['Low'] = [float(value.replace(',', ' ')) for value in df['Low']]

y = dext['Price']
X1 = dext.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)
#X_train = np.array(X_train).reshape(X_train.shape[0], X_train.shape[1], 1)

```

```

#X_test = np.array(X_test).reshape(X_test.shape[0], X_test.shape[1], 1)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=500, validation_split=0.1)

predictions_dext = model_3.predict(X_test)
predictions_dext21 = predictions_dext[1:]
y_test1 = y_test[:-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_dext21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_dext21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_dext21))) drep = pd.DataFrame()

df = data[data['Name']=='DREP']
df = df[:-1]

drep['Date'] = pd.to_datetime(df['Date']).values
drep['Price'] = [float(value.replace(',','')) for value in df['Price']]
drep['Open'] = [float(value.replace(',','')) for value in df['Open']]
drep['High'] = [float(value.replace(',','')) for value in df['High']]
drep['Low'] = [float(value.replace(',','')) for value in df['Low']]

y = drep['Price']
X1 = drep.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=1500, validation_split=0.1)

predictions_drep = model_3.predict(X_test)
predictions_drep21 = predictions_drep[1:]
y_test1 = y_test[:-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_drep21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_drep21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_drep21)))
ape = pd.DataFrame()

```

```

df = data[data['Name']=='APE']
df = df[::-1]

ape['Date'] = pd.to_datetime(df['Date']).values
ape['Price'] = [float(value.replace(',','')) for value in df['Price']]
ape['Open'] = [float(value.replace(',','')) for value in df['Open']]
ape['High'] = [float(value.replace(',','')) for value in df['High']]
ape['Low'] = [float(value.replace(',','')) for value in df['Low']]

y = ape['Price']
X1 = ape.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=800, validation_split=0.1)

predictions_ape = model_3.predict(X_test)

predictions_ape21 = predictions_ape[1:]
y_test1 = y_test[:-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_ape21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_ape21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_ape21)))
xmon = pd.DataFrame()

df = data[data['Name']=='XMON']
df = df[::-1]

xmon['Date'] = pd.to_datetime(df['Date']).values
xmon['Price'] = [float(value.replace(',','')) for value in df['Price']]
xmon['Open'] = [float(value.replace(',','')) for value in df['Open']]
xmon['High'] = [float(value.replace(',','')) for value in df['High']]
xmon['Low'] = [float(value.replace(',','')) for value in df['Low']]

y = xmon['Price']
X1 = xmon.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()

```

```

X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=200, validation_split=0.1)

predictions_xmon = model_3.predict(X_test)

predictions_xmon21 = predictions_xmon[1:]
y_test1 = y_test[:-1]

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_xmon21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_xmon21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_xmon21)))
mlt = pd.DataFrame()

df = data[data['Name']=='MULTI']
df = df[:-1]

mlt['Date'] = pd.to_datetime(df['Date']).values
mlt['Price'] = [float(value.replace(',',' ')) for value in df['Price']]
mlt['Open'] = [float(value.replace(',',' ')) for value in df['Open']]
mlt['High'] = [float(value.replace(',',' ')) for value in df['High']]
mlt['Low'] = [float(value.replace(',',' ')) for value in df['Low']]

y = mlt['Price']
X1 = mlt.drop(['Price'], axis = 1)
X = X1.drop(['Date'], axis = 1)

scaler = MinMaxScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
shuffle=False)

model_3 = reg()
model_3.fit(X_train, y_train, epochs=2000, validation_split=0.1)

predictions_mlt = model_3.predict(X_test)

predictions_mlt21 = predictions_mlt[1:]
y_test1 = y_test[:-1]

```

```

print("MSE: {0}".format(mean_squared_error(y_test1, predictions_mlt21)))
print("MAE: {0}".format(mean_absolute_error(y_test1, predictions_mlt21)))
print("Accuracy score of the predictions: {0}".format(r2_score(y_test1,
predictions_mlt21)))

predictions_ape = predictions_ape.reshape(31)
predictions_suku = predictions_suku.reshape(31)
predictions_dext = predictions_dext.reshape(31)
predictions_drep = predictions_drep.reshape(31)
predicted_set0 = np.stack((predictions_mlt, predictions_xmon))
predicted_set1 = np.stack((predictions_ape, predictions_suku))
predicted_set2 = np.stack((predictions_dext, predictions_drep))
predicted_set = np.concatenate((predicted_set0, predicted_set1,
predicted_set2))

X = np.transpose(predicted_set)
X = X.reshape(31,6)
y = np.transpose(predictions_sui)
y = y.reshape(31,1)
newX = pd.DataFrame(X, columns = ['MULTI','XMON','APE', 'SUKU','DEXT','DREP'])
newy = pd.DataFrame(y, columns =['SUI'])
xTrain, xTest, yTrain, yTest = train_test_split(newX, newy, test_size = 0.2,
random_state = 0)
regression = LinearRegression().fit(xTrain, yTrain)
yPred = regression.predict(xTest)
regression.coef_
coefs = regression.coef_.flatten()
spread_parts=X*coefs
newsread = pd.DataFrame(spread_parts, columns = ['MULTI','XMON','APE',
'SUKU','DEXT','DREP'])
spread = newy['SUI'] - newsread['MULTI'] - newsread['XMON'] - newsread['APE']
- newsread['SUKU'] - newsread['DEXT'] - newsread['DREP']
adfuller(spread)
kpss(spread, regression = 'ct', nlags = 'auto')
diff = []
for i in range(1, len(spread)):
    value = spread[i] - spread[i - 1]
    diff.append(value)
plt.plot(diff)
plt.show()
adfuller(diff)
kpss(diff, regression = 'ct')
predicted_set1 = np.stack((predictions_ape, predictions_suku))
predicted_set2 = np.stack((predictions_dext, predictions_drep))
predicted_set = np.concatenate((predicted_set1, predicted_set2))

X = np.transpose(predicted_set)
X = X.reshape(31,4)

```

```

y = np.transpose(predictions_mlt)
y = y.reshape(31,1)
newX = pd.DataFrame(X, columns = ['APE', 'SUKU', 'DEXT', 'DREP'])
newy = pd.DataFrame(y, columns =['MULTI'])
xTrain, xTest, yTrain, yTest = train_test_split(newX, newy, test_size = 0.2,
random_state = 0)

regression = LinearRegression().fit(xTrain, yTrain)
yPred = regression.predict(xTest)
regression.coef_
coefs = regression.coef_.flatten()
spread_parts=X*coefs
newsread = pd.DataFrame(spread_parts, columns = ['APE', 'SUKU', 'DEXT', 'DREP'])
spread = newy['MULTI'] - newsread['APE'] - newsread['SUKU'] -
newsread['DEXT'] - newsread['DREP']

adfuller(spread)
kpss(spread, regression = 'ct', nlags = 'auto')
start_date = pd.to_datetime('2023-05-01')
trading = pd.DataFrame()
trading['Spread'] = spread
trading['Date'] = pd.date_range(start=start_date, periods=len(trading))
plt.figure(figsize=(12, 9))
plt.plot(trading['Date'], trading['Spread'], label='Price')
plt.xlabel('Date')
plt.ylabel('Price')
plt.legend()
plt.show()

signals, stop_loss_values_top, stop_loss_values_low = generate_signals(spread)
plt.figure(figsize=(12, 9))
plt.plot(df['Date'], df['Spread'], label='Spread')
plt.plot(df['Date'], df['Stop Loss (Upper Bound)'], label='sl_top')
plt.plot(df['Date'], df['Stop Loss (Lower Bound)'], label='sl_low')
plt.xlabel('Date')
plt.ylabel('Price')
plt.legend()
plt.show()

cumulative_profit = [0] # Кумулятивний прибуток (початкове значення 0)
drawdown = [0] # Процент просідання (початкове значення 0)
peak = 0 # Пікове значення прибутку (початкове значення 0)
drawdown_duration = 0 # Тривалість просідання (початкове значення 0)

for i in range(1, len(signals)):
    if signals[i] == 'SELL, short':
        profit = (spread.iloc[i] - spread.iloc[1-1])/spread.iloc[0] #
Розрахунок прибутку при продажу (short)
        cumulative_profit.append(cumulative_profit[-1]-profit)
    elif signals[i] == 'BUY, Long':

```

```

        profit = (spread.iloc[i-1] - spread.iloc[i])/spread.iloc[0]    #
Розрахунок прибутку при покупці (long)
        cumulative_profit.append(cumulative_profit[-1]+profit)
    else:
        cumulative_profit.append(cumulative_profit[-1])    # Якщо сигнал HOLD,
прибуток залишається незмінним

profit = pd.DataFrame({'Cumulative_profit': cumulative_profit})
profit.insert(0, 'Date', trading['Date'])

profit
plt.figure(figsize=(12, 9))
plt.plot(profit['Date'], profit['Cumulative_profit'], label='CProfit')
plt.xlabel('Date')
plt.ylabel('CProfit')
plt.legend()
plt.show()
trading['Date'] = pd.to_datetime(trading['Date'])
trading.set_index('Date', inplace=True)
stats = trading.calc_stats()
stats.display()
profit['Date'] = pd.to_datetime(profit['Date'])
profit.set_index('Date', inplace=True)
ffn.to_drawdown_series(profit['Cumulative_profit']).plot(figsize=(15,7),grid=T
rue)

perf = profit['Cumulative_profit'].calc_stats()

perf.display()
sharpe

```

ДОДАТОК Б

презентація