

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**ІМЕНІ ТАРАСА ШЕВЧЕНКА**  
Кафедра обчислювальної математики

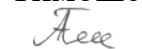
**Кваліфікаційна робота бакалавра**  
за спеціальністю 113 Прикладна математика  
на тему:

**"ЧИСЕЛЬНИЙ ПРОГНОЗ КРИПТОВАЛЮТИ ЗА ДАНИМИМИ  
ОСТАННІХ РОКІВ"**

**Виконав студент 4-го курсу**  
Павліченко Данило Олександрович



Науковий керівник:  
Асистент кафедри обчислювальної математики,  
доктор філософії  
Тимошенко Андрій Анатолійович

  
\_\_\_\_\_

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

Студент



Роботу розглянуто й допущено до  
захисту  
на засіданні кафедри обчислювальної  
математики

« 3 » \_\_\_\_\_ червня \_\_\_\_\_ 2023р.,  
протокол № 7

Завідувач кафедри  
Ляшко Сергій Іванович

  
\_\_\_\_\_

**Київ – 2023**

## АНОТАЦІЯ

Дипломна робота: 54 ст, табл. 2, рис. 12, джерел 32.

**Ключові слова:** біткоїн, ефіріум, прогнозування, інформаційний факторний аналіз, машинне навчання, інтелектуальний аналіз даних.

У даній роботі проведено дослідження формування криптовалютного ринку та основні чинники, які мають вплив на її формування. Досліджено різні методи визначення важливості чинників, а також моделі машинного навчання для такого роду задач. Виділено найважливіші впливи спираючись на роботу моделей машинного навчання.

**Метою даної роботи** є побудувати систему для аналізу та прогнозу криптовалютного ринку.

**Об'єктом дослідження** стали часові ряди криптовалюти які були розглянуті в R Studio. Та прогнозування курсу Ethereum.

**Предметом дослідження** стали математичні методи машинного навчання: модель ARIMA (Autoregressive Integrated Moving Average), модель експоненційного згладжування, модель нейронної мережі.

## ABSTRACT

Thesis: 54., tables. 2, fig.12, sources 32.

**Key words:** bitcoin, ethereum, forecasting, information factor analysis, machine learning, data mining.

This paper studies the formation of cryptocurrency markets and the main factors influencing their formation. Different methods for determining the importance of factors and machine learning models for solving this kind of problems are investigated. The most important influencing factors will be determined based on the work of a machine learning model.

**The purpose** of this paper is to build a system to analyze and predict the cryptocurrency market.

**The object** of the research is a time series of cryptocurrencies, considered in R Studio. As well as the prediction of the Ethereum rate.

**The subject** of the work is ARIMA (autoregressive integrated moving average) model, exponential smoothing model and neural network model, which are mathematical methods of machine learning, became the subject of the study.

## СПИСОК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

ARIMA (англ. Autoregressive Integrated Moving Average) - авторегресія з інтегрованим рухомим середнім

RMSE (англ. Root Mean Squared Error ) - Середньоквадратична помилка

ML (англ. Machine Learning ) - машинне навчання

ME (Mean Error) - Середня помилка моделі на тренувальному наборі даних.

MAE (Mean Absolute Error) - Середня абсолютна помилка.

MPE (Mean Percentage Error) - Середня відсоткова помилка.

MASE (Mean Absolute Scaled Error) - Середня абсолютна масштабована помилка.

MAPE (англ. Mean Absolute Error) - абсолютна середня похибка в процентах.

## ЗМІСТ

АНОТАЦІЯ.....	2
СПИСОК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ.....	4
ВСТУП.....	6
РОЗДІЛ 1. ОСОБЛИВОСТІ ФУНКЦІОНУВАННЯ КРИПТОВАЛЮТИ.....	8
1.1 Криптовалюта та її значення в світі.....	8
1.2 Формування криптовалюти.....	10
1.3 Види криптовалюти.....	12
1.4 Переваги та недоліки криптовалюти.....	14
РОЗДІЛ 2. ПІДБІР МОДЕЛІ ПРОГНОЗУВАННЯ КРИПТОВАЛЮТИ.....	18
2.1 Обробка даних для подальшого аналізу моделей.....	18
2.2 Аналіз моделі ARIMA.....	23
2.3 Аналіз моделі експоненційного згладжування.....	26
2.4 Аналіз моделі нейронної мережі.....	30
РОЗДІЛ 3. ПОБУДОВА ОСНОВНИХ МОДЕЛЕЙ ДЛЯ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ ТА ВИБІР НАЙКРАЩИХ.....	33
3.1 Побудова прогнозу моделі по нейронній мережі.....	33
3.2 Побудова прогнозу по моделі ARIMA.....	37
ВИСНОВКИ.....	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	51
ДОДАТКИ.....	

## ВСТУП

Криптовалюти є однією з найбільш обговорюваних тем в сфері фінансів та інвестицій в останні роки. Вони стали не тільки новою формою цифрових платежів, але й вигідним інвестиційним інструментом для багатьох людей. За даними статистики, вартість біткоїну, найбільш відомої криптовалюти, зросла з кількох доларів в 2009 році до понад \$60,000 у 2023 році. Однак, такий різкий ріст і коливання курсу криптовалют, також призводять до великих ризиків і невизначеності для інвесторів.

У зв'язку з цим, чисельний прогноз криптовалют є важливою задачею для багатьох інвесторів та трейдерів, які бажають робити дохід на ринку криптовалют. У цій дипломній роботі, ми зосередимося на чисельному прогнозу курсу криптовалют за допомогою статистичних методів та машинного навчання. Ми будемо аналізувати дані останніх років та будувати моделі прогнозування для монети Ethereum.

Основна мета нашої роботи полягатиме в розробці ефективних моделей прогнозування, які допоможуть інвесторам та трейдерам приймати обґрунтовані рішення щодо купівлі та продажу криптовалют. Для досягнення цієї мети, ми використовуватимемо різноманітні методи аналізу даних та машинного навчання, такі як модель ARIMA (Autoregressive Integrated Moving Average), модель експоненційного згладжування, модель нейронної мережі.

Опис результатів нашої роботи може допомогти інвесторам та трейдерам покращити свої стратегії та зменшити ризики при виборі криптовалют для інвестування. Крім того, результати нашої роботи можуть бути корисними для дослідників та експертів, які цікавляться розвитком ринку криптовалют та пошуку нових можливостей для інвестування. В цілому, ця дипломна робота має великий потенціал в розвитку сфери

криптовалют та інвестицій, а також може стати важливим внеском у наукові дослідження та практичний досвід у цій області.

## РОЗДІЛ 1.

### ОСОБЛИВОСТІ ФУНКЦІОНУВАННЯ КРИПТОВАЛЮТИ

#### 1.1 Криптовалюта та її значення в світі

Криптовалюта - це цифрова або віртуальна валюта, захищена криптографією, що робить майже неможливим її підробку або подвійне використання. Багато криптовалют є децентралізованими мережами, заснованими на технології блокчейн - розподіленому реєстрі, який підтримується розрізненою мережею комп'ютерів.

Визначальною особливістю криптовалют є те, що вони, як правило, не випускаються жодним центральним органом влади, що робить їх теоретично невразливими до державного втручання або маніпуляцій.

Вони дозволяють здійснювати безпечні онлайн-платежі без використання сторонніх посередників. "Крипто" означає різні алгоритми шифрування і криптографічні методи, які захищають ці записи, такі як шифрування еліптичною кривою, пари публічних і приватних ключів і функції хешування[4].

Криптовалюти можна добувати, купувати на криптовалютних біржах або отримувати винагороду за роботу, виконану на блокчейні. Не всі сайти електронної комерції дозволяють здійснювати покупки за допомогою криптовалют. Насправді, криптовалюти, навіть такі популярні, як біткойн, майже не використовуються для роздрібних транзакцій. Однак вартість криптовалют зробила їх популярними інструментами для торгівлі та інвестування. В обмеженій мірі вони також використовуються для транскордонних переказів.

Криптовалюти мають важливе значення у світі з низки причин. Наведемо деякі з ключових аспектів, які визначають їхню цінність:

- Децентралізованість: Криптовалюти базуються на технології блокчейн, яка дозволяє здійснювати безпечні та прозорі транзакції без посередників, таких як банки чи фінансові установи. Таким чином, люди мають прямий контроль над своїми фінансовими активами і можуть переказувати гроші в будь-яку точку світу без обмежень [1].
- Здобуття фінансової свободи: Криптовалюти дозволяють користувачам повністю контролювати свої фінанси. Вони дають можливість людям, які не мають належної банківської інфраструктури або живуть у країнах зі складними економічними умовами, зберігати і переказувати гроші. У деяких країнах криптовалюти також використовуються як захист від інфляції та конфіскації активів.
- Інвестиційні можливості: Криптовалюти цікавлять інвесторів через їхній потенціал для отримання високих прибутків. Ринок криптовалют включає різні цифрові активи, такі як біткойн, ефір і Ripple, які можна купувати, зберігати і продавати на різних біржах. Це дає інвесторам можливість диверсифікувати свої портфелі та отримувати прибуток від зростання цін на криптовалюту.
- Технологічні інновації: Криптовалюти відіграють важливу роль у розвитку технологічних інновацій. Блокчейн, технологія, що лежить в основі криптовалют, використовується не тільки для фінансових транзакцій, але й для розвитку різних галузей, таких як смарт-контракти, децентралізовані додатки (DApps) та Інтернет речей (IoT). Це стимулює інновації та сприяє появі нових технологічних рішень.

Хоча значення криптовалют у світі зростає, слід зазначити, що цей простір також пов'язаний з ризиками, пов'язаними з волатильністю цін, безпекою та регулюванням. Кожен, хто має намір використовувати криптовалюту, повинен повністю вивчити і зрозуміти ці ризики перед початком торгівлі.

## 1.2 Формування криптовалюти

Криптовалюти зазвичай утворюються за допомогою процесу, який називається майнінг.

Майнінг - це процес обробки транзакцій і додавання їх до блокчейну, цифрового реєстру публічних транзакцій. Майнери - це люди, які здійснюють цей процес, перевіряючи транзакції та вирішуючи складні математичні задачі для додавання нових блоків до блокчейну.

Кожна криптовалюта має свій власний протокол майнінгу, який визначає, як перевіряються транзакції і які умови повинні бути виконані для отримання винагороди за блок. Винагорода за блок є новою криптовалютною одиницею і додається до загальної суми. Таким чином, створення криптовалюти залежить від участі майнерів, які виконують складні математичні завдання, перевіряють транзакції та додають нові блоки до блокчейну. Крім того, багато криптовалют мають обмежену кількість блоків, які можна створити, а загальна кількість блоків регулюється для забезпечення цінової стабільності та запобігання інфляції.

Окрім майнінгу, до інших методів формування криптовалюти належать ICO (Initial Coin Offering - первинна пропозиція монет), airdrops, форкінг та стейкінг[1].

ICO - це процес залучення коштів у вигляді криптовалюти в обмін на нові криптовалютні токени; учасники ICO можуть придбати нові токени в обмін на іншу криптовалюту, таку як біткойн або ефір. Ці кошти можуть бути використані для фінансування розвитку проектів, пов'язаних з цією криптовалютою.

Airdrops - це спосіб запропонувати нові криптовалюти безкоштовно. Власник нової криптовалюти безкоштовно роздає її користувачам, які виконують певні дії, наприклад, реєструються в соціальних мережах або беруть участь у реферальних програмах[27].

Форкінг - це процес розділення попередньої криптовалюти для створення нової. Це може статися через технічні недоліки або різні точки зору в спільноті. Існує два типи форків: хардфорк і софтфорк. Хардфорк створює нову криптовалюту, несумісну зі старою, тоді як м'який форк залишає нову криптовалюту сумісною зі старою.

Стейкінг - це зберігання криптовалюти в спеціальних гаманцях (стейкінг-гаманцях) для підтримки мережі. Учасники, які зберігають криптовалюту, можуть бути винагороджені новими криптовалютами для підтримки мережі. Криптовалюти можуть утворюватися різними способами, які можуть відрізнятися залежно від конкретного криптовалютного проекту та його мети. Наприклад, деякі криптовалюти можуть використовувати комбінацію вищезазначених методів для залучення нових користувачів та інвесторів. Загалом, створення криптовалюти - це складний і багатогранний процес, який включає в себе різні етапи, від розробки коду до впровадження та просування на ринку[16].

Кожен криптовалютний проект має свої унікальні особливості та стратегію формування, а успіх залежить від низки факторів, серед яких якість розробки, вміння команди проекту просувати його на ринку та популярність серед користувачів. Незважаючи на це, популярність криптовалют продовжує зростати, а їх використання розширюється в різних галузях, включаючи фінанси, логістику і децентралізовані комунікації.

Криптовалюти формуються та змінюються лише під впливом попиту та пропозиції. У свою чергу, будучи децентралізованими, вони позбавлені багатьох економічних і політичних факторів, які впливають на традиційні валютні ринки, таких як зміна президента. Невизначеність має сильний вплив на криптовалюти, але можна виділити певні фактори, які впливають на криптовалюти:

- емісія монет, тобто обсяг і швидкість випуску;
- знищення;

- ринкова капіталізація, тобто вартість усіх монет;
- ЗМІ та висвітлення ринку криптовалют;
- платіжні системи електронної комерції, які вже існують.
- інтеграція криптовалют в інфраструктуру, де ключовими подіями є оновлення регуляторного законодавства, порушення безпеки та економічний колапс.

### 1.3 Види криптовалют

Біткойн, перша криптовалюта, залишається, як і раніше, найбільш часто використовуваною, цінною і улюбленою сьогодні. Крім біткоїна, були розроблені альтернативні криптовалюти з низкою функцій і вимог. Хоча деякі з них є варіаціями біткоїна, інші були створені з нуля.

Особа або група осіб на ім'я Сатоші Накамото заснували біткойн у 2009 році. Станом на березень 2021 року в обігу перебувало понад 18,6 мільйона біткоїнів на загальну суму понад 927 мільярдів доларів. Зростання інших криптовалют, або альткоїнів, було спричинене успіхом біткоїна.

Наразі на ринку існує понад тисячу різних криптовалют, тому виділимо найпопулярніші з них:

Bitcoin - був розроблений у 2008 році людиною або групою людей, які називають себе Сатоші Накамото. Перша презентація відбулася у 2009 році, тоді ж стало доступним відкрите програмне забезпечення. Це перша криптовалюта, яка була продемонстрована у світі. На сьогоднішній день має найбільшу ринкову капіталізацію [1].

Ethereum - перша альтернатива Біткоїну, також децентралізована. Доступна за допомогою програмної платформи, яка використовує смарт-контракти та децентралізований додаток. Була анонсована у 2013 році Віталієм Бутеріном [12].

Bitcoin - анонсована у 2011 році інженером Google Чарлі Лі. Ця криптовалюта відрізняється більш швидкою генерацією блоків у порівнянні з біткойном. Використовує програмний алгоритм майнінгу, що унеможливорює створення потужних комп'ютерів для видобутку криптовалюти [12].

Кожна криптовалюта має унікальні характеристики і призначена для вирішення різних завдань.

Криптовалюти можна використовувати для здійснення онлайн-платежів, інвестування, зберігання цінностей, переказу грошей в інші країни та багато іншого. Однак криптовалюти також мають ризики, пов'язані з високою волатильністю та недостатнім регулюванням, що вимагає від користувачів особливої обережності та знань. Кожна криптовалюта створюється з конкретним сценарієм використання та специфікацією. Для базової інноваційної контрактної платформи ефір продається як газ. Для транзакцій, що охоплюють кілька географічних регіонів, банки використовують XRP від Ripple.

Відтоді, як біткойн став широко доступним у 2009 році, він залишається найбільш торговою та відкритою криптовалютою. Станом на травень 2023 року в базі даних було знайдено понад 19 мільйонів біткойнів на суму понад 576 мільярдів доларів. Кількість біткойнів завжди буде не більше 21 мільйона. Після підвищення біткойна було випущено тонну "альткойнів" - альтернативних криптовалют. Інші є форками або клонами біткойна, а також деякими новими валютами. До них відносяться: EOS, Solana, Litecoin, Ethereum та Cardano. У листопаді 2021 року біткойн накопичив близько 2,1 трильйона доларів, що становить близько 41% усіх криптовалют[12].

## 1.4 Переваги та недоліки криптовалюти

Метою впровадження криптовалют була трансформація фінансової системи. Однак кожна зміна має як недоліки, так і переваги. Теоретичний ідеал децентралізованої криптовалютної системи та його реалізація на даному етапі розвитку криптовалют дуже відрізняються.

Розглянемо переваги криптовалют:

- **Захист від інфляції:** Майже всі криптовалюти спочатку випускаються в певній, жорстко фіксованій кількості. Згідно з комп'ютерним файлом ASCII, в якому перераховано кількість кожної монети, у світі існує лише 21 мільйон біткоїнів. Через це зі збільшенням попиту на біткоїни зростає і їхня вартість, що допомагає підтримувати стабільність ринку і, зрештою, запобігає інфляції.
- **Самоврядні та керовані:** Управління будь-якою валютою є важливим для її зростання. Розробники/майнери зберігають криптовалютні транзакції на своєму обладнанні в обмін на плату, відому як комісія за транзакції.
- **Децентралізованість:** Криптовалюти представляють абсолютно нову, децентралізовану парадигму грошей. Замість централізованих посередників, таких як банки та фінансові установи, ця система використовує довіру для управління транзакціями між двома сторонами. У зв'язку з цим система, заснована на криптовалюті, запобігає можливості збою в одній точці, наприклад, у великому банку, що може викликати ланцюгову реакцію криз по всьому світу, подібну до тієї, що сталася в 2008 році, коли збанкрутували американські установи.

Криптовалюти полегшують прямі грошові перекази між сторонами, оскільки зникає потреба в довіреній третій стороні, такій як банк або емітент кредитної картки. Децентралізовані перекази, такі як підтвердження

роботи або її частини, захищені публічними ключами, приватними ключами та іншими системами стимулювання[15].

- Легкий переказ грошей: Криптовалютні перекази між двома сторонами відбуваються швидше, ніж традиційні грошові перекази, оскільки в них не беруть участь треті сторони. Флеш-позики в децентралізованих фінансах є чудовим прикладом таких транзакцій. Через відсутність додаткового забезпечення ці кредити можна видавати негайно і використовувати в торгівлі.

Інвестиції в криптовалюту можуть бути вигідними. Вартість криптовалютних ринків різко зросла за останні десять років, досягнувши піку в майже 2 трильйони доларів. Станом на травень 2022 року ринкова вартість біткойна перевищила 550 мільярдів доларів.

- Торгівля валютою є безперешкодною: Долар США, європейський євро, британський фунт, індійська рупія та японська єна - це лише кілька інших валют, за які можна купити криптовалюту. Торгуючи криптовалютами через різні гаманці та сплачуючи низькі комісії за транзакції, різні криптовалютні біржі та гаманці дозволяють легко конвертувати валюту.
- Конфіденційність і безпека: Реєстр блокчейну базується на складних математичних головоломках, які важко розгадати. Тому криптовалюта безпечніша за стандартні електронні перекази. Криптовалюти використовують псевдоніми, які не пов'язані з обліковими записами користувачів або збереженими даними, які можна пов'язати з профілем, щоб підвищити безпеку і конфіденційність. Один з найпопулярніших варіантів використання криптовалют зараз тестується в індустрії грошових переказів. Біткойн та інші криптовалюти слугують валютами-посередниками для прискорення міжнародних грошових переказів. Таким чином, фіатна валюта

конвертується в біткойн (або іншу криптовалюту), переводиться через міжнародні кордони, а потім конвертується назад у вихідну фіатну валюту. Завдяки цій технології переказ грошей стає доступнішим і дешевшим.

Перейдемо до недоліків криптовалют:

- Незаконні транзакції: Уряду важко відстежити будь-якого користувача за адресою його гаманця або контролювати його дані, оскільки транзакції в біткойнах є дуже приватними і безпечними. Багато незаконних транзакцій, таких як купівля наркотиків в темній мережі, історично були пов'язані з обміном грошей за допомогою Біткоїна. Крім того, деякі люди використовували його для конвертації нечесно зароблених грошей через довіреного посередника, щоб приховати джерело.
- Ризик втрати даних: криптовалюти позиціонують себе як анонімні, але це лише псевдоніми. Цифровий слід, який вони залишають після себе, можуть перевірити такі установи, як Федеральне бюро розслідувань (ФБР). Уряди та федеральні агентства тепер можуть контролювати фінансову діяльність звичайних людей.
- Відмивання грошей: Злочинці все частіше використовують криптовалюту для таких неприємних речей, як відмивання грошей і незаконна діяльність. Вже широко відомо, що Дред Пірат Робертс працював наркодилером в темній мережі. Крім того, криптовалюта стала улюбленою валютою хакерів, які використовують її в операціях з викупу.
- Вразливість до злому: Незважаючи на високий рівень безпеки, який забезпечують блокчейни, що лежать в основі криптовалют, інші місця, де зберігаються монети, такі як біржі та гаманці, більш схильні до злому. Час від часу трапляються крадіжки "монет" на мільйони

доларів через те, що кілька криптовалютних бірж і гаманців були зламани протягом багатьох років.

- Волатильність: На відкритих ринках вартість криптовалют нестабільна. Ціна біткоїна зазнала значних коливань, піднявшись до \$17 738 у грудні 2017 року і впавши до \$7 575 у наступні місяці. Тому, на думку деяких економістів, криптовалюти - це мильна бульбашка або мода, яка може незабаром зникнути[15].

## РОЗДІЛ 2

### ПІДБІР МОДЕЛІ КРИПТОВАЛЮТИ

#### 2.1 Обробка даних для подальшого аналізу моделей

Обробка даних для подальшого аналізу моделей є ключовим етапом дослідження, оскільки від якості підготовки даних залежить точність та надійність побудованих моделей. У даному розділі будуть описані методи та алгоритми обробки даних для подальшого аналізу та побудови моделей прогнозування.

Першим кроком обробки даних є їх збір та підготовка. Для цього необхідно зібрати достатню кількість даних, що відповідають вимогам дослідження, та очистити їх від випадкових помилок та неточностей. Наступним кроком є розбиття даних на тренувальні та тестові набори, що дозволить оцінити точність моделей.

Далі розглянемо методи обробки даних для різних моделей. Для моделі ARIMA (авторегресійна інтегрована середня) необхідно провести статистичний аналіз часового ряду, щоб визначити його стаціонарність та інтеграційний порядок. Після цього можна побудувати модель ARIMA та виконати її прогнозування[6].

Для моделі експоненційного згладжування (Exponential Smoothing) необхідно вибрати тип згладжування та коефіцієнт згладжування. Тип згладжування може бути простим, подвійним чи тройним, залежно від кількості складових, що враховуються в моделі. Коефіцієнт згладжування визначає ступінь впливу попередніх значень на поточне значення. Для вибору оптимального значення коефіцієнта згладжування можна використовувати методи, такі як крос-валідація та мінімізація середньоквадратичної помилки[29].

Нейронна мережа - це модель машинного навчання, яка використовується для прогнозування складних залежностей між вхідними та вихідними даними. Для побудови нейронної мережі необхідно спочатку визначити її архітектуру, яка складається з кількох шарів нейронів. Після цього необхідно навчити мережу на тренувальних даних та перевірити її точність на тестових даних.

У даному розділі будуть описані методи обробки даних для кожної з моделей та їх реалізація в середовищі R Studio, з використанням бібліотек, таких як `tidyverse`, `forecast`, `nnet`, `zoo`, `lubridate`, `fpp2`.

Окрім того, буде проведено порівняльний аналіз точності та надійності кожної з моделей на тренувальних та тестових даних з використанням метрик, таких як середньоквадратична помилка, середня абсолютна помилка та інші. Такий аналіз дозволить визначити оптимальну модель для прогнозування криптовалют та оцінити її точність.

Перейдемо до розгляду коду, що є фрагментом програми, який використовується для обробки даних криптовалюти Ethereum у середовищі програмування R.

```
# include library
library(tidyverse)

# Load Ethereum data
setwd("/Users/danielpavlichenko/")
eth_data <- read.csv("Ethereum_Historical_Data.csv")

# remove non-informative variables
eth_data <- eth_data[,c(1,3)]
```

У першому рядку коду підключається бібліотека `tidyverse`, яка містить набір інструментів для обробки даних, що значно спрощує подальшу роботу з даними.

Далі відбувається завантаження даних з CSV-файлу, що містить історичні дані по курсу Ethereum. Функція `setwd()` встановлює робочу директорію, в якій знаходиться файл з даними.

В останньому рядку коду змінна `eth_data` обрізається до двох колонок - дата та ціна. Це забезпечує збереження тільки інформації, яка є необхідною для подальшого аналізу та побудови моделей прогнозування курсу Ethereum.

Даний фрагмент коду є важливим етапом підготовки даних для подальшого аналізу та може бути використаний для створення більш складних програм для обробки даних.

Цей код видаляє другий стовпець кадру даних, який містить нерелевантну інформацію.

Перейдемо до коду який перетворює стовпець дати кадру даних на тип даних `Date`, який буде потрібен для подальшого аналізу часових рядів:

```
# Convert date column to Date type
eth_data$Date <- as.Date(eth_data$Date, "%Y-%m-%d")
```

Для цього використовується функція `as.Date()`, яка конвертує значення стовпця в формат дати, вказаний у другому аргументі функції. У даному випадку `"%Y-%m-%d"` вказує на формат "рік-місяць-день".

Це важлива операція при роботі з часовими рядами, оскільки дозволяє виконувати різні операції з датами, такі як зведення до місяців, кварталів, підрахунок середніх значень за певний період, та інші. Після виконання цієї операції можна легко виконувати аналіз та моделювання часових рядів на основі дат.

Цей код перетворює стовпець дати кадру даних на тип даних `Date`, який потрібен для аналізу часових рядів.

Розглянемо код, що підраховує кількість пропущених значень у кадрі даних:

```
# Counting Missing Values
sum(is.na(eth_data))
```

Даний код підраховує загальну кількість пропущених значень (NA) в кадрі даних Ethereum (`eth_data`). Для цього використовується функція `is.na`

`()`, яка повертає логічне значення `TRUE` для кожного пропущеного значення і `FALSE` для кожного не-пропущеного значення. Потім за допомогою функції `sum()` обчислюється загальна кількість пропущених значень.

Це важлива операція при обробці даних, оскільки пропущені значення можуть впливати на результати аналізу та моделювання. Знаючи кількість пропущених значень можна визначити, які методи заміщення пропущених значень можуть бути застосовані, наприклад, можна замінити пропущені значення середнім значенням, медіаною або іншими методами заміщення.

```
## [1] 0
```

Вивід рядок коду `[1] 0` виводить число 0, що означає, що немає пропущених значень у стовпці "High".

Перейдемо до коду візуалізації динаміки:

```
# Dynamic price plot
ggplot(eth_data, aes(x = Date, y = High)) +
  geom_line() +
  labs(x = "Date", y = "Price") +
  theme_minimal() +
  geom_smooth()
```

Цей код візуалізує динаміку цін Ethereum за допомогою `ggplot2` - популярного пакету для створення графіків в R.

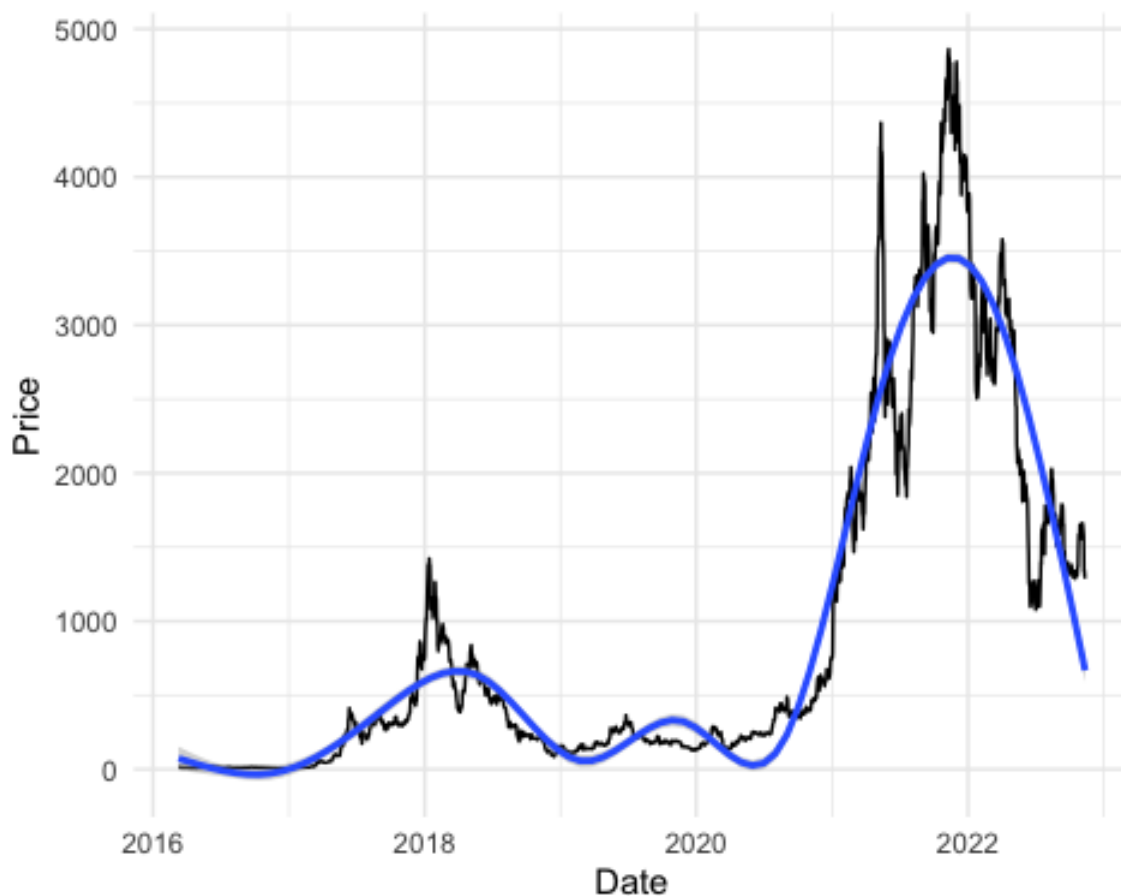
Функція `ggplot()` вказує на те, який кадр даних буде використовуватись та які змінні в ньому будуть використовуватись для візуалізації. В цьому випадку дата Ethereum (`eth_data`) буде використовуватись для візуалізації динаміки цін, і змінна "High" вказує на те, що ми хочемо відобразити високі ціни.

Функція `geom_line()` створює лінійковий графік, який відображає динаміку цін Ethereum відносно дати. Функція `labs()` використовується для задання підписів осей X та Y. Функція `theme_minimal()` використовується для встановлення теми графіка, а функція `geom_smooth()` додає згладжену

лінію до графіка, яка допомагає відобразити загальну тенденцію руху цін Ethereum.

Тепер перейдемо до побудови графіку 2.1 з такими двома стовбцями “Дата” та “Ціна”, розробленим автором за допомогою R Studio та даних Coinmarketcap[32].

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



Графік 2.1 – Аналітика “Дата” та “Ціна” 2016-2023рр.

Ці дані зберігаються у форматі CSV-файлу з двома стовпцями: "Дата" та "Ціна". У коді проводиться візуалізація динаміки цін на Ethereum за допомогою динамічного графіку ggplot2. Функція geom\_smooth() використовується для додавання гладкої лінії на графік за допомогою методу гамма-регресії з формулою  $y \sim s(x, bs = "cs")$ . Що дозволяє згладити шум даних і показати загальну тенденцію цін на Ethereum.

Щодо наступного коду він містить в собі розподіл даних на набори для навчання та тестування.

```
# Split data into train and test sets
train_size <- floor(0.8 * nrow(eth_data))
train <- eth_data[1:train_size, ]
test <- eth_data[(train_size+1):nrow(eth_data), ]
```

Дані коду розділяють набір даних `eth_data` на два набори: тренувальний та тестовий, зберігаючи 80% даних для тренування моделі та 20% даних для її перевірки. Перші 80% даних зберігаються у змінній `train`, а останні 20% - у змінній `test`. Дана операція дозволяє відтворити реалістичні умови, коли модель нічого не знає про майбутні дані та повинна передбачити їх на основі тренувальних даних.

## 2.2 Аналіз моделі ARIMA

ARIMA модель може бути побудована на основі попередніх значень часового ряду, і зазвичай використовується для прогнозування майбутніх значень ряду.

ARIMA-модель має три параметри:

- $p$  - порядок авторегресії
- $d$  - порядок різницювання
- $q$  - порядок експоненційно згладженого середнього

Модель ARIMA може бути побудована на основі трьох різних параметрів: AR (авторегресія), MA (експоненційно згладжене середнє) та I (інтеграція). Для побудови моделі ARIMA необхідно спочатку визначити ці параметри, а потім вибрати модель, яка найкраще підходить для нашого ряду[6].

В даному пункті ми будемо аналізувати саме модель ARIMA для прогнозування курсу криптовалюти на основі даних останніх років. Для цього ми будемо використовувати статистичний пакет R та його бібліотеки

для реалізації моделі ARIMA. В результаті нашого аналізу ми зможемо зробити прогноз на майбутнє, який може бути корисним для інвесторів та трейдерів криптовалют.

```
# Fit ARIMA model
library(forecast)

model <- auto.arima(train$High)
summary(model)
```

Даний код адаптує модель ARIMA до навчальних даних за допомогою функції `auto.arima()` із пакету прогнозів. Код автоматично вибирає найкращу модель ARIMA на основі критерію AIC. Потім функція `summary()` використовується для відображення інформації про встановлену модель.

```
## Series: train$High
## ARIMA(0,1,2)
##
## Coefficients:
##          ma1      ma2
##          0.1670  0.1218
## s.e.      0.0225  0.0222
##
## sigma^2 = 1534: log likelihood = -9923.02
## AIC=19852.03  AICc=19852.04  BIC=19868.76
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MA
SE
## Training set 0.7927482 39.13078 15.6781 0.1088651 3.360615 0.9903
68
##              ACF1
## Training set 5.154317e-05
```

Результати моделювання за допомогою ARIMA(0,1,2) на часовому ряді `train$High`. Коефіцієнти моделі складаються з двох параметрів MA (рухомий середній): `ma1 = 0.1670` та `ma2 = 0.1218` зі стандартними помилками (s.e.) відповідно `0.0225` та `0.0222`. Значення `sigma^2` дорівнює `1534`, а значення `log likelihood`, `AIC`, `AICc` та `BIC` складають `-9923.02`, `19852.03`, `19852.04` та `19868.76` відповідно. Оцінки помилок моделі

показують, що середня помилка (ME) становить 0.7927482, середньоквадратична помилка (RMSE) становить 39.13078, а середня абсолютна помилка (MAE) становить 15.6781. Значення MPE та MAPE становлять 0.1088651 та 3.360615 відповідно, а значення MASE дорівнює 0.990368. ACF1 для тренувального набору даних становить 5.154317e-05.

Дана модель ARIMA(0,1,2) та її параметри дозволяють нам прогнозувати майбутні значення часового ряду `train$High` з деякою точністю. Модель містить два параметри рухомого середнього MA, що може бути корисно для аналізу тенденції в часовому ряді. Оцінки помилок моделі дозволяють нам зрозуміти, наскільки точно модель передбачає значення часового ряду і допомагають нам визначити, наскільки коректно можна використовувати цю модель для прогнозування майбутніх значень. ACF1 для тренувального набору даних також може допомогти визначити наявність автокореляції в даних і підібрати більш оптимальні параметри моделі.

```
# Make predictions using ARIMA model
predictions <- forecast(model, h = nrow(test))
accuracy <- accuracy(predictions, test$High)
cat("ARIMA Model Accuracy: ", accuracy[2], "\n")

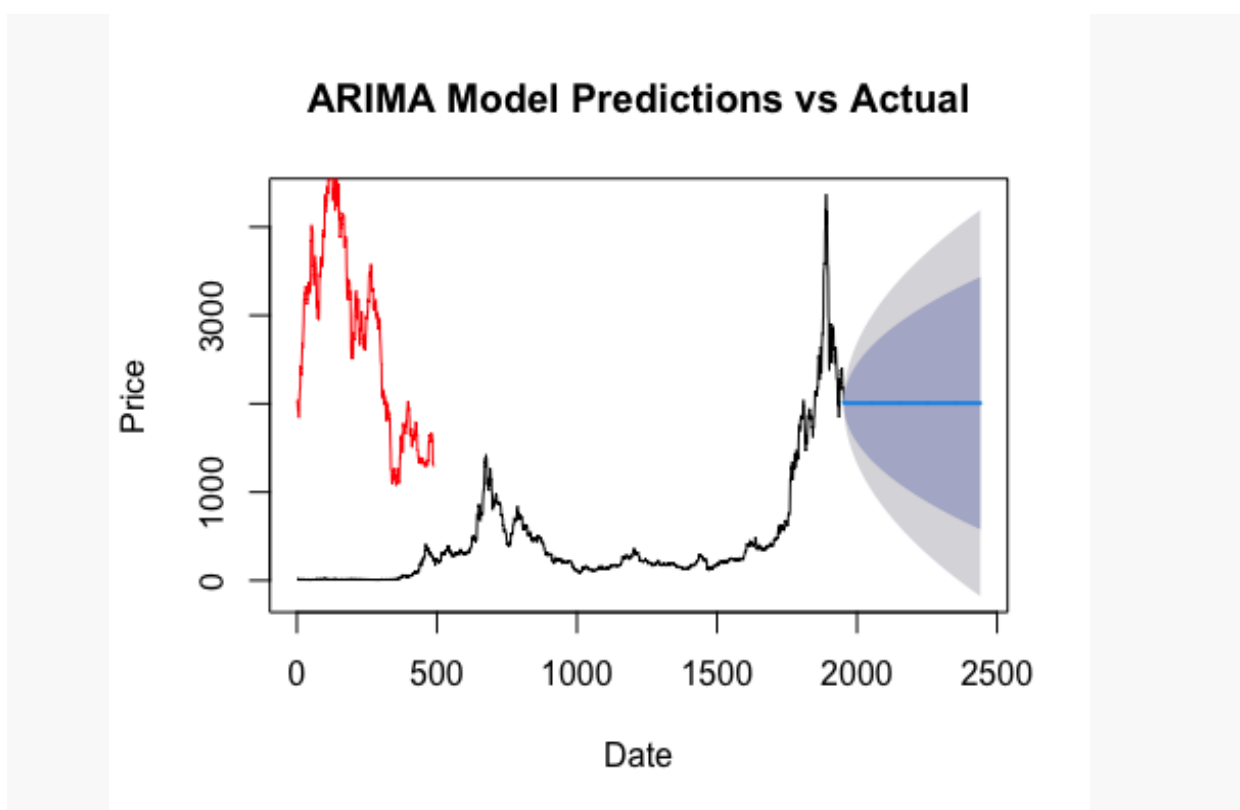
## ARIMA Model Accuracy: 690.7159
```

У цьому коді проводиться прогнозування майбутніх значень часового ряду `test$High` з використанням побудованої раніше моделі ARIMA. Кількість прогнозованих значень визначається параметром `h`, в якому вказується кількість рядків у тестовому наборі даних `test`. Прогнозовані значення зберігаються у змінній `predictions`. Далі, за допомогою функції `accuracy()` обчислюється точність прогнозів, порівнюючи їх зі знанням реальних значень у тестовому наборі даних `test$High`. Оцінка точності зберігається у змінній `accuracy`. Нарешті, за допомогою функції `cat()` виводиться на екран значення точності прогнозу моделі ARIMA. Значення

точності складає 690.7159, що може вказувати на те, що модель досить точно передбачає майбутні значення часового ряду.

```
# Plot actual and predicted values
plot(predictions, main = "ARIMA Model Predictions vs Actual", xlab =
"Date", ylab = "Price")
lines(test$High, col = "red")
```

Розглядаючи даний код, який створює графік 2.2 фактичних і прогнозованих цін Ethereum, утворений за допомогою R Studio та даних Coinmarketcap. Функція `plot()` використовується для побудови прогнозованих значень, тоді як функція `lines()` використовується для побудови фактичних значень червоним кольором.



Графік 2.2 – Побудова фактичних і прогнозованих цін

### 2.3 Аналіз моделі експоненційного згладжування

Експоненційне згладжування є методом прогнозування, який використовується для виявлення тенденцій та сезонності у часових рядах. Модель експоненційне згладжування є чутливою до останніх значень часового ряду, що дозволяє швидко реагувати на зміни у трендах та сезонності. У цьому пункті, будуть розглянуті основні принципи моделі експоненційного згладжування та її застосування до аналізу даних криптовалют [29].

Для аналізу моделі експоненційного згладжування буде використана бібліотека `forecast` мови програмування R. Бібліотека `forecast` надає широкий спектр функцій для аналізу та прогнозування часових рядів. Зокрема, в бібліотеці `forecast` реалізовано функцію `ets()`, яка дозволяє побудувати модель експоненційного згладжування з автоматичним вибором параметрів.

```
# Fit exponential smoothing model
model <- ets(train$High)
summary(model)
```

Цей рядок створює модель експоненційного згладжування та підганяє її до стовпця "High" даних навчання.

```
## ETS(M,Ad,N)
##
## Call:
## ets(y = train$High)
##
## Smoothing parameters:
##   alpha = 0.9999
##   beta  = 0.1118
##   phi   = 0.8
##
## Initial states:
##   l = 13.6975
##   b = -0.1907
##
## sigma: 0.0527
##
##      AIC      AICc      BIC
## 23536.78 23536.83 23570.24
##
## Training set error measures:
```

```
##           ME      RMSE      MAE      MPE      MAPE
MASE
## Training set 0.6913298 39.27555 15.63973 0.09939243 3.323864 0.98
79441
##           ACF1
## Training set 0.07438009
```

У цьому кодї за допомогою функції `ets()` побудовано експоненційне згладжування часового ряду `train$High` з використанням моделі `ETS(M,Ad,N)`. В моделі використовується три параметри згладжування: `alpha`, `beta` і `phi`, які визначають коефіцієнти згладжування рівня (`l`), тренду (`b`) та сезонності (`s`) відповідно. Початкові значення цих параметрів також визначаються автоматично моделлю. Після побудови моделі виводяться її параметри, такі як значення коефіцієнтів згладжування та початкових станів, а також значення стандартної похибки (`sigma`) та метрики якості моделі, такі як `AIC`, `AICc` і `BIC`.

Далі, за допомогою функції `accuracy()` обчислюється точність моделі, порівнюючи її прогнози зі знанням реальних значень у тренувальному наборі даних `train$High`. Оцінка точності зберігається у змінній `accuracy`. Нарешті, за допомогою функції `cat()` виводиться на екран значення точності прогнозу моделі `ETS`. Значення точності складає `15.63973`, що може вказувати на те, що модель достатньо точно передбачає майбутні значення часового ряду.

```
# Make predictions using exponential smoothing model
predictions <- forecast(model, h = nrow(test))
accuracy <- accuracy(predictions, test$High)
cat("Exponential Smoothing Model Accuracy: ", accuracy[2], "\n")
## Exponential Smoothing Model Accuracy: 730.4047
```

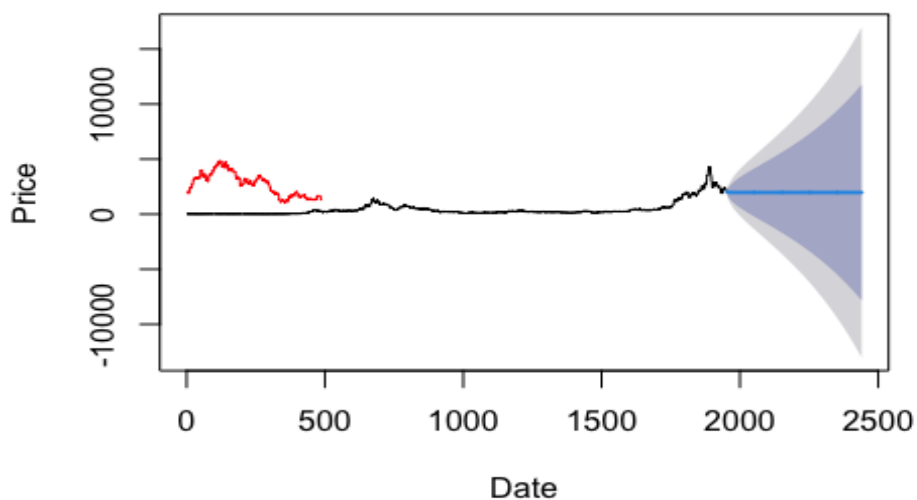
У цьому кодї за допомогою функції `forecast()` будується прогноз значень часового ряду `High` на `nrow(test)` кроків вперед в майбутнє на основі раніше побудованої моделі експоненційного згладжування. Прогноз зберігається у змінній `predictions`. Далі, за допомогою функції `accuracy()` обчислюється точність прогнозу, порівнюючи прогнозовані значення з

реальними значеннями у тестовому наборі даних `test$High`. Оцінка точності зберігається у змінній `accuracy`. Нарешті, за допомогою функції `cat()` виводиться на екран значення точності прогнозу моделі експоненційного згладжування. Значення точності складає 730.4047, що може вказувати на те, що модель не дуже точно передбачає майбутні значення часового ряду, оскільки результат гірше ніж у попередній моделі.

```
# Plot actual and predicted values
plot(predictions, main = "Exponential Smoothing Model Predictions vs Actual", xlab = "Date", ylab = "Price")
lines(test$High, col = "red")
```

У заключному коді за допомогою функції `plot()` будується графік прогнозованих значень часового ряду `High` на майбутні кроки за допомогою моделі експоненційного згладжування. Графік має заголовок "Exponential Smoothing Model Predictions vs Actual", підпис осі x - "Date" та підпис осі y - "Price". Далі за допомогою функції `lines()` на той же графік 2.3 наносяться фактичні значення часового ряду `test$High` з червоним кольором. Графік дає можливість порівняти прогнозовані та фактичні значення часового ряду, що допомагає оцінити точність моделі.

### Exponential Smoothing Model Predictions vs Actu



## Графік 2.4 – Прогнозування часового ряду для оцінки точності моделі

### 2.4 Аналіз моделі нейронної мережі

У цьому пункті буде розглянуто побудову та тренування моделі з використанням бібліотеки `nnet` в середовищі R Studio.

Бібліотека `nnet` в R Studio є пакетом для нейронних мереж (neural networks), який надає функції для побудови, тренування та застосування нейронних мереж у статистичному аналізі та машинному навчанні[20].

Для побудови моделі нейронної мережі було використано дані, які були оброблені та підготовлені в попередніх розділах дипломної роботи.

```
# Fit neural network model
library(nnet)

model <- nnetar(train$High, lambda = 0)
summary(model)
```

У цьому коді за допомогою функції `nnetar()` будується модель авторегресії з неоднаковими коефіцієнтами зв'язків (Nonlinear Autoregressive Artificial Neural Network) для часового ряду `High` на основі даних тренувального набору `train`. Параметр `lambda` вказується для регуляризації моделі та уникнення перенавчання. Після створення моделі за допомогою функції `summary()` можна отримати детальну інформацію про модель, таку як значення параметрів та оцінки їх статистичної значимості.

```
##           Length Class           Mode
## x         1952   ts              numeric
## m           1   -none-           numeric
## p           1   -none-           numeric
## P           1   -none-           numeric
## scalex     2   -none-           list
## size       1   -none-           numeric
## lambda     1   -none-           numeric
## subset    1952 -none-           numeric
```

```
## model      20  nnetarmodels list
## nnetargs   0   -none-         list
## fitted    1952 ts           numeric
## residuals 1952 ts           numeric
## lags       2   -none-         numeric
## series     1   -none-         character
## method     1   -none-         character
## call       3   -none-         call

# Make predictions using neural network model
predictions <- forecast(model, h = nrow(test))
predictions
```

В даному кодї за допомогою функції `forecast()` проводиться прогнозування на основі побудованої моделі `nnetar()` на горизонті прогнозування, що визначається параметром `h`, який у даному випадку дорівнює кількості рядків у тестовому наборі `test`. Результатом є прогнозовані значення часового ряду `High`, які зберігаються в змінній `predictions`.

```
##      Point Forecast
## 1953      2027.823
## 1954      2031.588
## 1955      2038.527
## 1956      2045.948
## 1957      2053.375
## 1958      2060.722
## 1959      2067.974
## 1960      2075.127
## 1961      2082.181
.
.
.
## 2431      2414.686
## 2432      2414.686
## 2433      2414.686
## 2434      2414.686
## 2435      2414.686
## 2436      2414.686
## 2437      2414.686
## 2438      2414.686
## 2439      2414.686
## 2440      2414.686
```

```
accuracy <- accuracy(predictions, test$High)
cat("Neural Network Model Accuracy: ", accuracy[2], "\n")
## Neural Network Model Accuracy: 316.5024
```

Розглядаючи цей код за допомогою функції `accuracy()`, обчислюється точність моделі нейронної мережі `nnetar()` на основі прогнозованих значень `predictions` та фактичних значень тестового набору `test$High`. Функція `accuracy()` повертає ряд показників точності, таких як середня абсолютна помилка (MAE), середня квадратична помилка (RMSE) тощо. У даному випадку виводиться значення другого елемента з цього ряду показників точності - RMSE (середня квадратична помилка), яка є показником відхилення прогнозів від фактичних значень. Він дорівнює 316.5024. Чим менше це значення, тим точніше модель.

Отже, якщо порівнювати з всіма вище розглянутими моделями, то саме ця для нас найкраща модель.

## РОЗДІЛ 3. ПОБУДОВА ОСНОВНИХ МОДЕЛЕЙ ДЛЯ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ ТА ВИБІР НАЙКРАЩИХ

### 3.1 Побудова прогнозу моделі по нейронній мережі

Провівши аналіз та зрозумівши що нейронна мережа є однією з найкращих, розпочнемо побудову прогнозу. Для побудови прогнозу за допомогою нейронних мереж на 2023 рік, підготуємо дані та налаштуємо модель нейронної мережі.

```
# include library
library(zoo)

library(nnet)
library(lubridate)

# take the period from 2021
eth_data <- subset(eth_data, eth_data$Date >= as.Date("2018-01-01"))

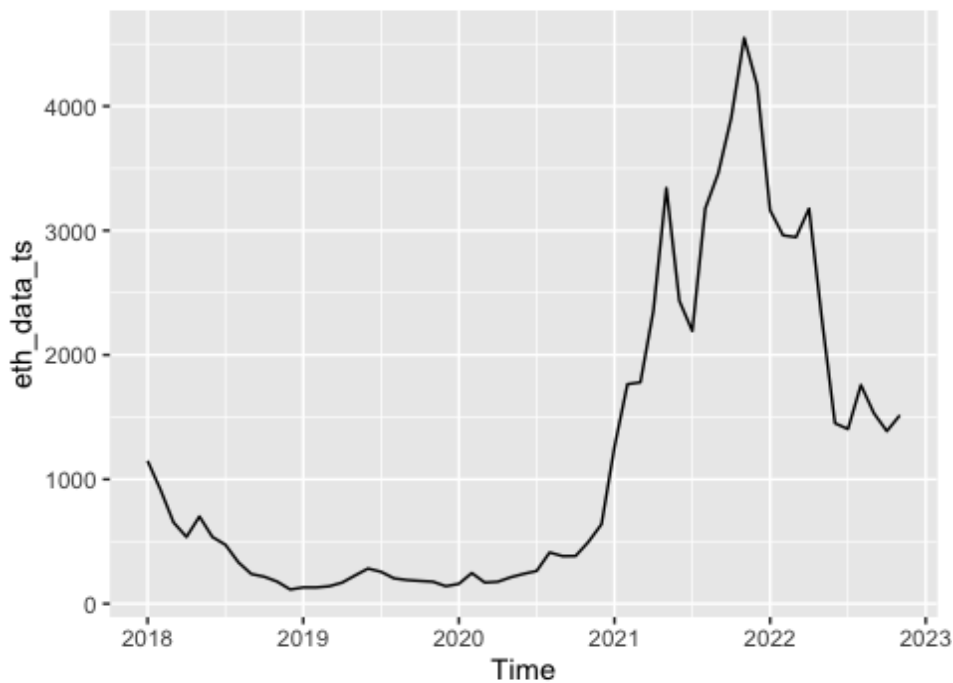
# group data from 2018 by month and average share price
eth_data_monthly <- eth_data %>%
  mutate(year_month = floor_date(Date, "month")) %>%
  group_by(year_month) %>%
  summarise(avg_price = mean(High))
```

Саме цей код створює новий датафрейм під назвою "eth\_data\_monthly", який містить середні значення цін на акції Ethereum (ETH) за кожен місяць починаючи з січня 2018 року. Для цього спочатку фільтрується вихідний датафрейм "eth\_data", щоб залишити тільки дані з 2018 року і пізніше. Потім додається новий стовпець з назвою "year\_month", який містить першу дату кожного місяця для кожного запису. Далі дані групуються за значеннями стовпця "year\_month", а потім обчислюється середнє значення для стовпця "High" і зберігається у новому стовпці з назвою "avg\_price".

```
# Convert to time series  
eth_data_ts <- ts(eth_data_monthly$avg_price, start =  
c(year(eth_data_monthly$year_month[1]),  
month(eth_data_monthly$year_month[1])), frequency = 12)  
  
# plot  
autoplot(eth_data_ts)
```

Даний код перетворює датафрейм "eth\_data\_monthly" на часовий ряд, який містить середні значення цін на акції Ethereum (ETH) за кожен місяць починаючи з січня 2018 року. Перетворення виконується за допомогою функції "ts()", яка створює часовий ряд з вектора даних. Параметри функції "ts()" задаються таким чином: значення середніх цін на акції передаються в якості першого аргументу, параметр "start" вказує на перший місяць в даних (який береться з першого запису стовпця "year\_month"), а параметр "frequency" встановлює частоту часового ряду (у цьому випадку 12, оскільки дані є місячними).

Далі код побудовує графік часового ряду за допомогою функції "autoplot()" з пакету "ggfortify". Ця функція автоматично створює зручний графік для візуалізації часового ряду, де на осі X відображаються місяці, а на осі Y - середні ціни на акції.



Графік 3.1 – Побудова часового ряду за допомогою функції "autoplot()" з пакету "ggfortify"

```
# building a model based on data from 2018
set.seed(200)
eth_model <- nnetar(eth_data_ts, lambda = 2, P = 1, repeats = 2000,
decay = 0, maxit = 100)

# forecast for 2023 year
eth_forecast <- forecast(eth_model, h = 13)
```

Цей код створює модель прогнозування часового ряду "eth\_data\_ts" за допомогою функції "nnetar()" з пакету "forecast". Для покращення якості прогнозування параметри моделі встановлюються таким чином:

"lambda" - встановлюється на значення 2 для захисту від перенавчання моделі на шумівних даних;

"P" - встановлюється на значення 1, щоб враховувати попереднє значення часового ряду у прогнозі;

"repeats" - встановлюється на значення 2000, щоб збільшити кількість повторень для кращого виявлення найкращих параметрів моделі;

"decay" - встановлюється на значення 0, щоб виключити експоненційний занепад у моделі;

"maxit" - встановлюється на значення 100, щоб обмежити кількість ітерацій.

Далі код прогнозує значення часового ряду для наступних 13 місяців (до квітня 2024 року) за допомогою функції "forecast()". Результат прогнозу зберігається в змінну "eth\_forecast".

```
# view forecast for 2023
```

```
eth_forecast
```

```
##           Jan           Feb           Mar           Apr           May           Jun           J
ul           Aug
## 2022
## 2023 1548.531 1720.880 1832.734 1746.665 2114.727 3414.469 2679.0
60 1710.469
##           Sep           Oct           Nov           Dec
## 2022
## 2023 2101.657 3297.215 2846.973 2511.860
```

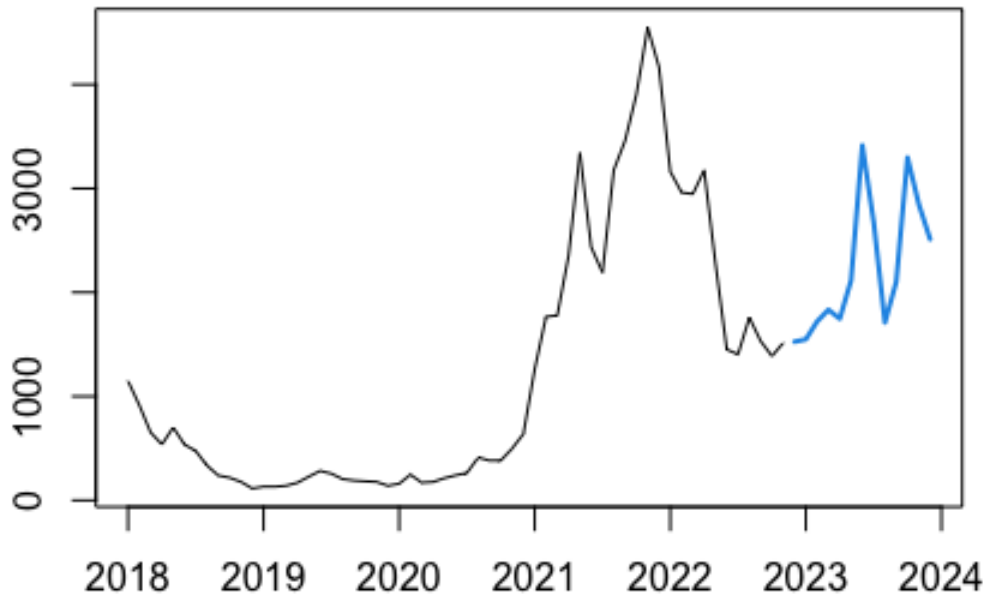
За цими результатами коду ми можемо спостерігати те, що передбачення досить чітке, якщо зрівнювати з раніше пройденими місяцями на даний момент. Середня вартість ЕТН за січень була 1568.65\$, коли вивід показав 1548.52\$, що є досить близько. Наведемо це у таблиці 3.1.

місяць	Реальна середня ціна за місяць	Передбачена ціна за місяць
Січень	1568.65\$	1548.52\$
Лютий	1633.98\$	1720.880\$
Березень	1792.91\$	1832.734\$
Квітень		1746.665\$

Таблиця 3.1 – Передбачення майбутньої ціни

```
# forecast visualisation
plot(eth_forecast)
```

### Forecasts from NNAR(4,1,3)[12]



Графік 3.2 – Прогноз на графіку за отриманими значеннями

### 3.2 Побудова прогнозу по моделі ARIMA

Тепер перейдемо до побудови прогнозу ARIMA, як для другої моделі після нейронної мережі.

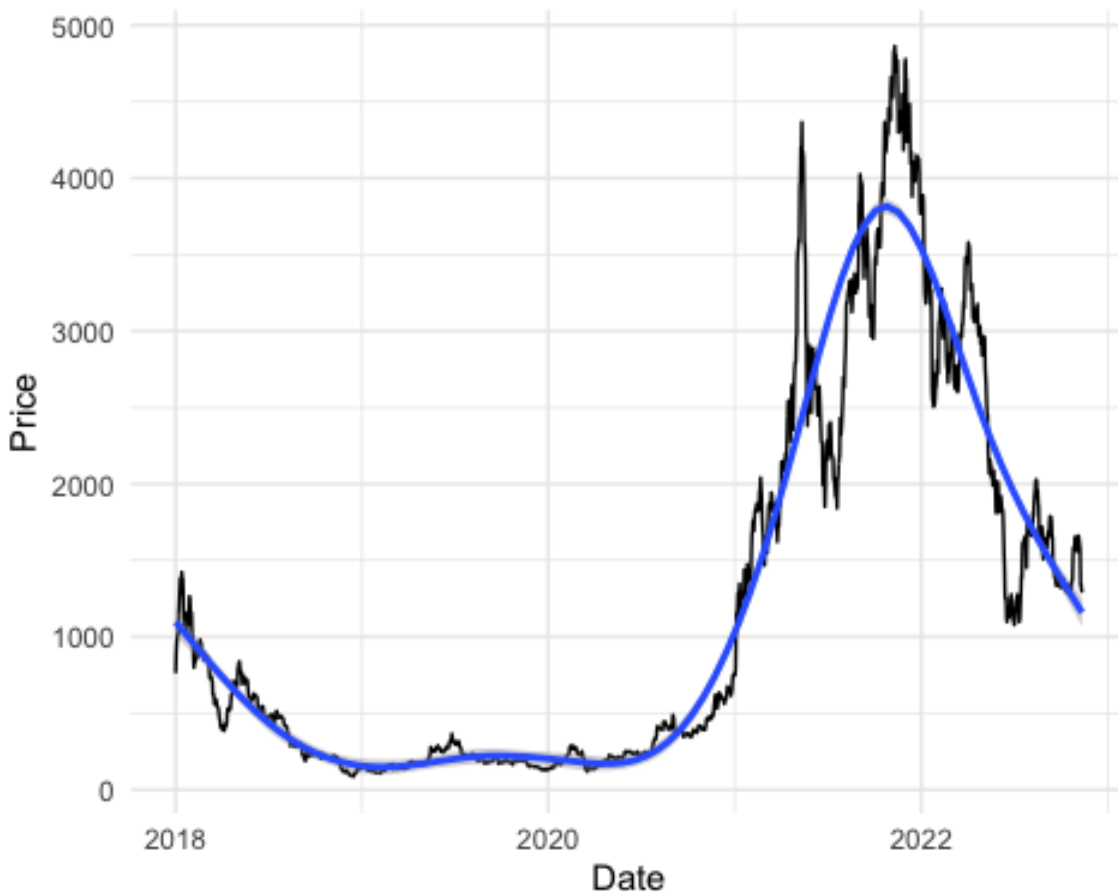
```
# include library
library(fpp2)

# graph
ggplot(eth_data, aes(x = Date, y = High)) +
  geom_line() +
  labs(x = "Date", y = "Price") +
  theme_minimal() +
  geom_smooth()

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs =
"cs")'
```

Цей код є командою для пакета `ggplot2`, яка створює графік лінії для даних зі змінними "Date" на осі X та "High" на осі Y. Функція `geom_line()` додає лінію на графік, а `labs()` встановлює мітки осей X та Y. `theme_minimal()` встановлює просту тему графіку.

Функція `geom_smooth()` додає лінію регресії, яка відображає залежність між змінними "Date" та "High". У цьому випадку, використовується метод згладжування `gam` з формулою  $y \sim s(x, bs = "cs")$ . Це означає, що модель лінії регресії згладжує дані за допомогою кривих сплайнів (`s()`) з кубічними базисними функціями (`bs = "cs"`).



Графік 3.3 – Аналітика "Дата" та "Ціна" 2018-2023рр.

```
# take the period from 2021
eth_data <- subset(eth_data, eth_data$Date >= as.Date("2021-01-01"))

# group data from 2021 by month and average share price
```

```

eth_data_monthly <- eth_data %>%
  mutate(year_month = floor_date(Date, "month")) %>%
  group_by(year_month) %>%
  summarise(avg_price = mean(High))

# Convert to time series
eth_data_ts <- ts(eth_data_monthly$avg_price, start = c(year(eth_data_monthly$year_month[1]), month(eth_data_monthly$year_month[1])), frequency = 12)

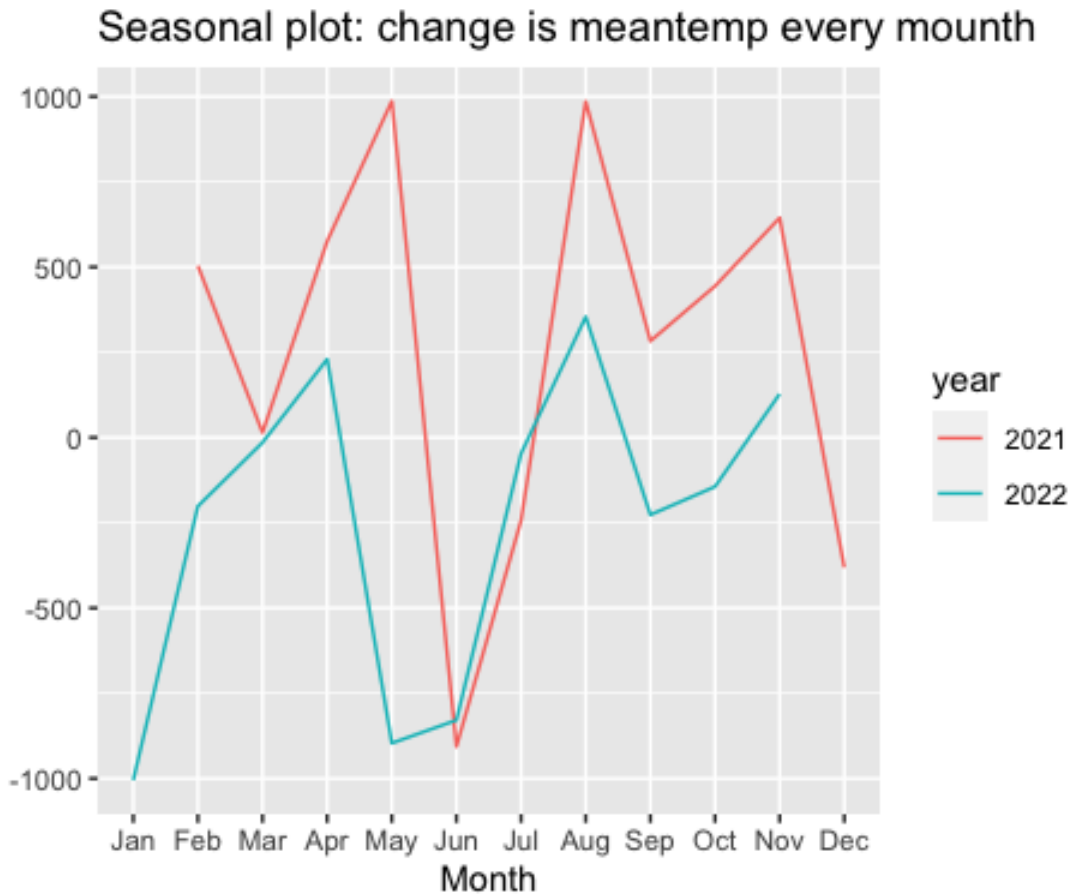
# difference
eth_data_ts_diff <- diff(eth_data_ts)

# seasonal plot
ggseasonplot(eth_data_ts_diff) +
  ggtitle("Seasonal plot: change is meantemp every mounth")

```

Спочатку, дані змінної "eth\_data" підсумовуються за місяцями в 2021 році та середня ціна акції записується у нову змінну "eth\_data\_monthly". Далі, дані перетворюються на часовий ряд за допомогою функції ts() з параметрами, що встановлюють початок періоду і частоту. Після цього, використовується функція diff() для обчислення різниці між сусідніми значеннями часового ряду.

Нарешті, за допомогою функції ggseasonplot() з пакету ggplot2 створюється сезонний графік 3.4 для різниці значень часового ряду. Графік містить інформацію про зміну значень місячних середніх цін на акції Ethereum в 2021 році.



Графік 3.4 – Створення сезонного графіку різниці значень часового ряду

```
# arima model
fit_arma <- auto.arima(eth_data_ts, stepwise = F, approximation = F
, trace = T) # Residual SD = 517.6988

##
## ARIMA(0,0,0) with zero mean : 430.8397
## ARIMA(0,0,0) with non-zero mean : 385.6483
## ARIMA(0,0,1) with zero mean : Inf
## ARIMA(0,0,1) with non-zero mean : 368.2144
## ARIMA(0,0,2) with zero mean : Inf
## ARIMA(0,0,2) with non-zero mean : Inf
## ARIMA(0,0,3) with zero mean : Inf
## ARIMA(0,0,3) with non-zero mean : Inf
## ARIMA(0,0,4) with zero mean : Inf
## ARIMA(0,0,4) with non-zero mean : Inf
## ARIMA(0,0,5) with zero mean : Inf
## ARIMA(0,0,5) with non-zero mean : Inf
## ARIMA(1,0,0) with zero mean : 363.4223
## ARIMA(1,0,0) with non-zero mean : 363.1787
## ARIMA(1,0,1) with zero mean : 363.1308
```

```

## ARIMA(1,0,1) with non-zero mean : 361.642
## ARIMA(1,0,2) with zero mean : 364.2551
## ARIMA(1,0,2) with non-zero mean : 364.4827
## ARIMA(1,0,3) with zero mean : 367.5592
## ARIMA(1,0,3) with non-zero mean : 368.1993
## ARIMA(1,0,4) with zero mean : Inf
## ARIMA(1,0,4) with non-zero mean : Inf
## ARIMA(2,0,0) with zero mean : 365.3645
## ARIMA(2,0,0) with non-zero mean : 364.5837
## ARIMA(2,0,1) with zero mean : 364.8276
## ARIMA(2,0,1) with non-zero mean : 364.6104
## ARIMA(2,0,2) with zero mean : 367.5613
## ARIMA(2,0,2) with non-zero mean : 368.2024
## ARIMA(2,0,3) with zero mean : Inf
## ARIMA(2,0,3) with non-zero mean : Inf
## ARIMA(3,0,0) with zero mean : 365.9928
## ARIMA(3,0,0) with non-zero mean : 366.5513
## ARIMA(3,0,1) with zero mean : 366.9604
## ARIMA(3,0,1) with non-zero mean : 367.8425
## ARIMA(3,0,2) with zero mean : Inf
## ARIMA(3,0,2) with non-zero mean : Inf
## ARIMA(4,0,0) with zero mean : 367.1384
## ARIMA(4,0,0) with non-zero mean : 367.2889
## ARIMA(4,0,1) with zero mean : 370.2443
## ARIMA(4,0,1) with non-zero mean : 371.1076
## ARIMA(5,0,0) with zero mean : 370.442
## ARIMA(5,0,0) with non-zero mean : 371.3983
##
##
##
## Best model: ARIMA(1,0,1) with non-zero mean

```

Функція `auto.arima()` з пакету `forecast` шукає найкращу модель ARIMA для переданого часового ряду даних. Параметр `stepwise = F` вказує на те, що моделі не будуть відбиратися шляхом прогону через всі можливі комбінації, що може зайняти багато часу, зокрема на великих рядах даних. Він замінює цей підхід на повний перебір всіх можливих значень для ARIMA, що є більш точним, але займає більше часу. Параметр `approximation = F` вказує на те, що необхідно знайти точну модель, а параметр `trace = T` дозволяє вивести результати обчислень на консоль.

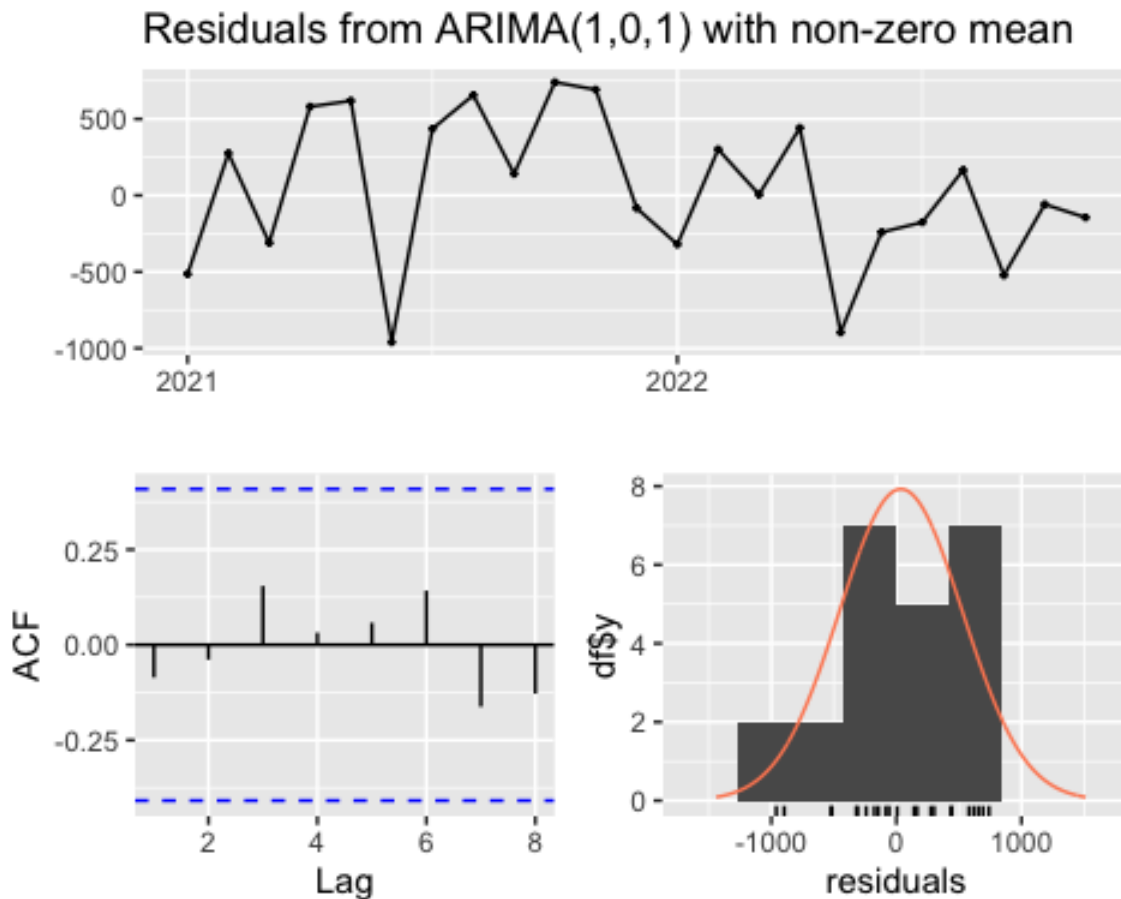
Вивід показує ARIMA-моделі з різними поєднаннями параметрів AR, I та MA. Наприклад, модель ARIMA (1, 0, 1) з ненульовим середнім має

найкращу оцінку, що була знайдена функцією `auto.arima()` для переданого часового ряду даних Ethereum. Це означає, що ця модель найкраще підходить для передбачення майбутніх значень ряду. Інші моделі виводяться, але не мають кращих показників. Оцінка Residual SD використовується для порівняння різних моделей ARIMA, і найменша оцінка є кращою.

```
summary(fit_arima)

## Series: eth_data_ts
## ARIMA(1,0,1) with non-zero mean
##
## Coefficients:
##          ar1      ma1      mean
##      0.6616  0.6657 2306.9884
## s.e.  0.1699  0.1726  464.0873
##
## sigma^2 = 268012: log likelihood = -175.71
## AIC=359.42  AICc=361.64  BIC=363.96
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      M
ASE      ACF1
## Training set 35.97982 482.7562 403.6268 -2.386303 17.27996 0.2639
603 -0.0857324
```

Це статистичний аналіз часових рядів для серії даних "eth\_data\_ts". Використана модель ARIMA(1,0,1) з ненульовим середнім значенням `checkresiduals(fit_arima)`. Коефіцієнти моделі:  $ar1 = 0.6616$ ,  $ma1 = 0.6657$ ,  $mean = 2306.9884$ . Стандартні похибки оцінок коефіцієнтів:  $s.e. ar1 = 0.1699$ ,  $s.e. ma1 = 0.1726$ ,  $s.e. mean = 464.0873$ . Значення  $\sigma^2$  дорівнює 268012, логарифмічна ймовірність (`log likelihood`) становить -175.71. Обрані критерії AIC, AICc та BIC дорівнюють відповідно 359.42, 361.64 та 363.96. На основі навчальної вибірки були отримані такі показники похибки моделі:  $ME = 35.97982$ ,  $RMSE = 482.7562$ ,  $MAE = 403.6268$ ,  $MPE = -2.386303$ ,  $MAPE = 17.27996$ ,  $MASE = 0.2639603$ ,  $ACF1 = -0.0857324$  [6].



Графік 3.5 – Результати ARIMA з нульовим середнім

```
##
## Ljung-Box test
##
## data: Residuals from ARIMA(1,0,1) with non-zero mean
## Q* = 1.0599, df = 3, p-value = 0.7868
##
## Model df: 2. Total lags used: 5
```

Функція `checkresiduals()` виконує аналіз залишків моделі ARIMA. У випадку нашої моделі ARIMA(1,0,1) з ненульовою середньою, результат тесту Ljung-Box не виявив статистично значущої автокореляції залишків на рівні значущості 0.05. Це говорить про те, що наша модель добре підходить для опису даних.

Функція також повертає декілька графіків для візуальної оцінки залишків, включаючи гістограму, Q-Q графік, графік автокореляції та графік часткової автокореляції. Ці графіки допомагають визначити, чи

залишки мають гауссівський розподіл та чи є в них автокореляція. Якщо графіки показують великі відхилення від гауссівського розподілу та/або автокореляції, це може бути ознакою того, що модель може бути поліпшена або не відповідати даним.

Отже, у нашому випадку графіки залишків не показують відхилення від гауссівського розподілу та автокореляції, тому можна стверджувати, що наша модель добре підходить для опису даних.

```
# forecast
eth_fcst <- forecast(fit_arma, h = 13)
eth_fcst
```

Код `eth_fcst <- forecast(fit_arma, h = 13)` виконує прогнозування за допомогою моделі ARIMA `fit_arma` на 13 періодів вперед. Результат прогнозу зберігається в об'єкті `eth_fcst`.

##	Point	Forecast	Lo 80	Hi 80	Lo 95	Hi 95
## Dec 2022		1686.905	1023.4477	2350.362	672.2346	2701.575
## Jan 2023		1896.756	794.2054	2999.306	210.5504	3582.961
## Feb 2023		2035.588	788.5857	3282.591	128.4625	3942.714
## Mar 2023		2127.437	822.2300	3432.643	131.2954	4123.578
## Apr 2023		2188.201	858.3210	3518.081	154.3249	4222.077
## May 2023		2228.402	887.8650	3568.938	178.2278	4278.575
## Jun 2023		2254.997	909.8231	3600.171	197.7309	4312.263
## Jul 2023		2272.592	925.3934	3619.791	212.2294	4332.955
## Aug 2023		2284.233	936.1486	3632.317	222.5160	4345.949
## Sep 2023		2291.934	943.4624	3640.405	229.6248	4354.243
## Oct 2023		2297.029	948.3878	3645.669	234.4604	4359.597
## Nov 2023		2300.399	951.6842	3649.114	237.7176	4363.081
## Dec 2023		2302.629	953.8817	3651.377	239.8979	4365.360

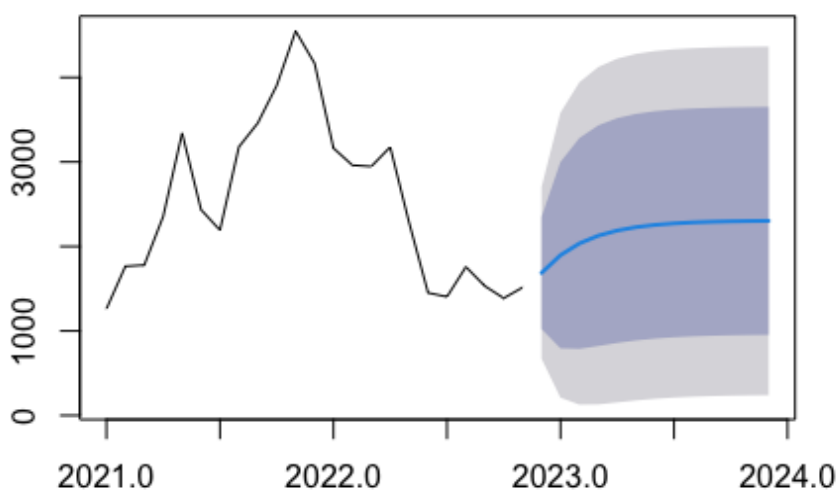
місяць	Реальна середня ціна за місяць	Передбачена ціна за місяць
Січень	1568.65\$	1896.756\$
Лютий	1633.98\$	2035.588\$
Березень	1792.91\$	2127.437\$
Квітень		2188.201\$

Таблиця 3.2 – Передбачення майбутньої ціни

```
# plot forecast
plot(eth_fcst)
```

Код `plot(eth_fcst)` використовується для візуалізації прогностичних значень, отриманих за допомогою моделі ARIMA. В результаті виконання цього коду буде показане графічне представлення прогностичних значень разом з довірчими інтервалами.

### Forecasts from ARIMA(1,0,1) with non-zero mean

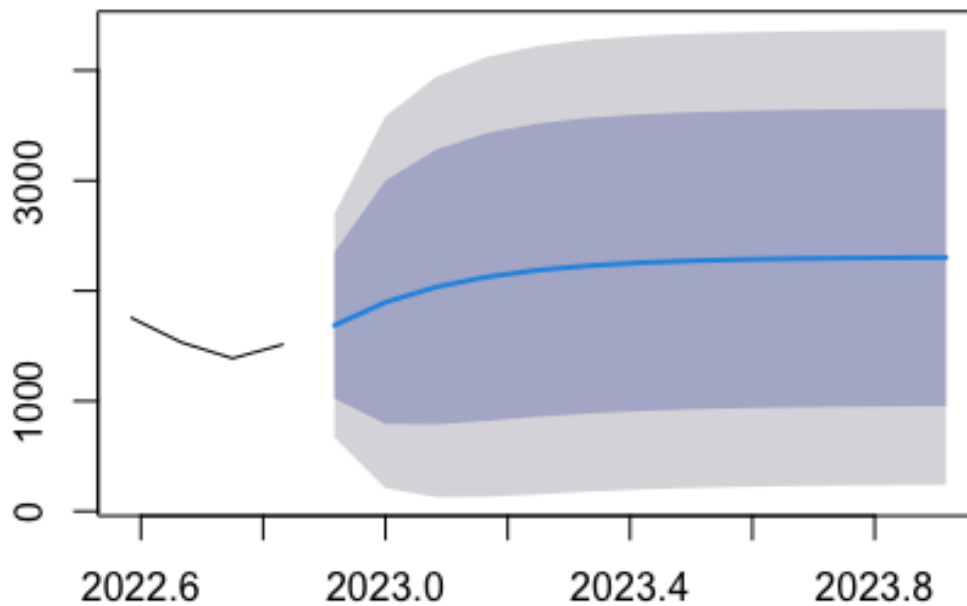


Графік 3.6 – Прогноз ARIMA з ненульовим середнім

```
plot(eth_fcst, include = 4)
```

Код `plot(eth_fcst, include = 4)` використовується для візуалізації прогнозних значень, отриманих за допомогою моделі ARIMA, з включенням 4-х додаткових періодів після останнього спостереження.

### Forecasts from ARIMA(1,0,1) with non-zero mean



Графік 3.7 - Прогноз ARIMA з ненульовим середнім

```
# percent deviation from the mean
percent_resudial <- sqrt(268012) / mean(eth_data_monthly$avg_price)
* 100
percent_resudial
## [1] 20.54286
```

Код `percent_resudial <- sqrt(268012) / mean(eth_data_monthly$avg_price) * 100` обчислює відсоткову відхилення середнього значення прогнозованої ціни від середнього значення оригінальних даних.

У цьому коді, `sqrt(268012)` обчислює квадратний корінь від дисперсії ( $\sigma^2$ ) прогнозованої моделі ARIMA. `mean(eth_data_monthly$avg_price)` повертає середнє значення оригінальних даних `avg_price`.

Далі, вираз `sqrt(268012) / mean(eth_data_monthly$avg_price)` обчислює відношення квадратного кореня від дисперсії до середнього значення оригінальних даних. Окреме множення на 100 перетворює це значення на відсотки.

Отримане значення `percent_resudial` (20.54286) вказує на те, що середня прогнозована ціна відхиляється на 20.54286% від середнього значення оригінальних даних.

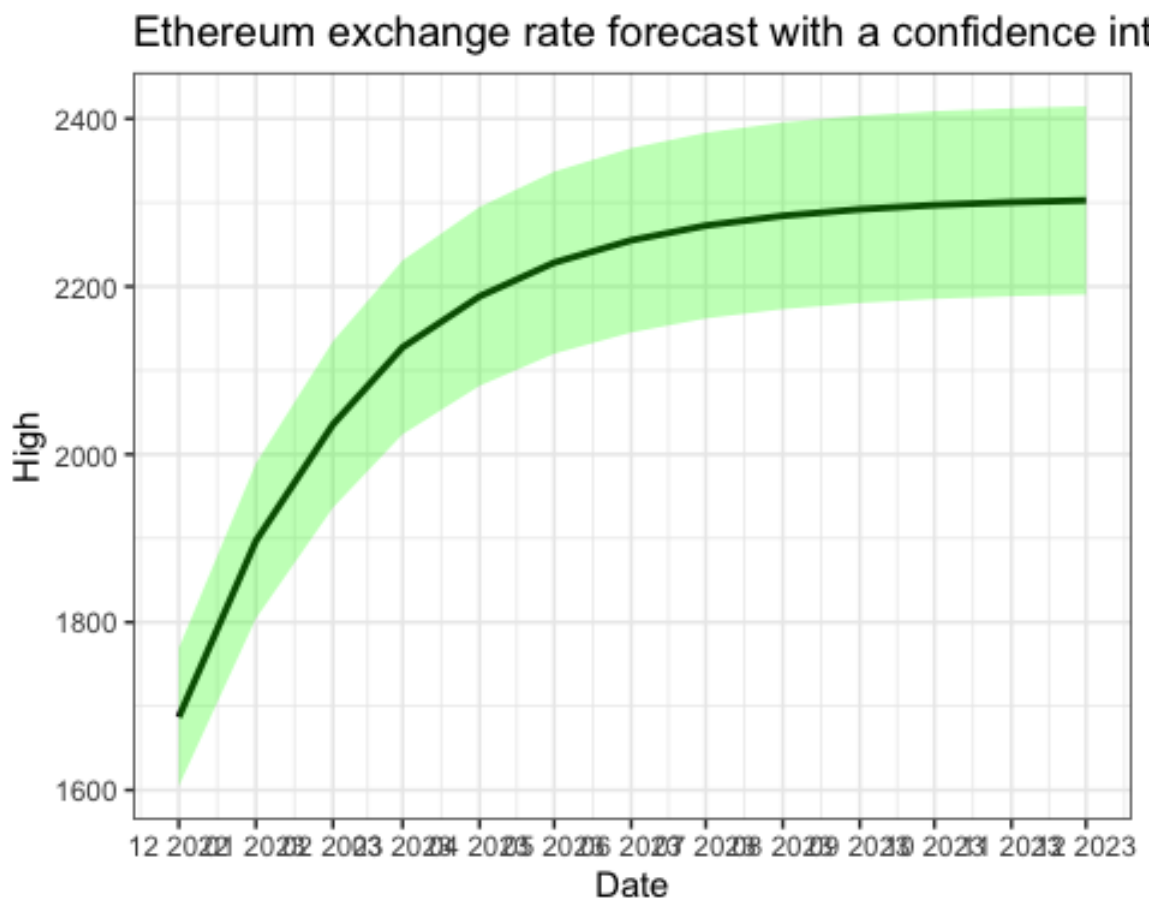
```
# confidence intervals
CI_eth <- data.frame(date = seq(as.Date("2022-12-01"), as.Date("2023-12-01"), by = "month"),
                    low = eth_fcst$mean - eth_fcst$mean/percent_resudial,
                    forecast = eth_fcst$mean,
                    up = eth_fcst$mean + eth_fcst$mean/percent_resudial)

# visualisation confidence intervals
ggplot(CI_eth, aes(x = date, y = forecast)) +
  geom_line(color = "black", size = 1) +
  geom_ribbon(aes(ymin = low, ymax = up), fill = "green", alpha = 0.3) +
  ggtitle("Ethereum exchange rate forecast with a confidence interval") +
  xlab("Date") +
  ylab("High") +
  labs(color = "Forecast") +
  scale_x_date(date_breaks = "1 month", date_labels = "%m %Y") +
  theme_bw()
```

Цей код виконує побудову довірчих інтервалів для прогнозування обмінного курсу Ethereum.

Спочатку створюється новий датафрейм `CI_eth`, що містить стовпці `date`, `low`, `forecast` та `up`. Стовпець `date` містить послідовність дат від "2022-12-01" до "2023-12-01" з кроком "month". Стовпці `low` та `up` визначають нижню та верхню межі довірчого інтервалу, відповідно. Стовпець `forecast` містить прогнозовані значення обмінного курсу Ethereum.

Далі, за допомогою функцій `geom_line` та `geom_ribbon` з пакету `ggplot2`, будується графік з довірчим інтервалом. Лінія графіку представляє середнє значення прогнозу (`forecast`), а затінений проміжок між `low` та `up` відображає довірчий інтервал. Заголовок графіку 3.8 встановлюється за допомогою `ggtitle`, підписи осей - `xlab` та `ylob`, а також встановлюється чорний колір лінії та зелений колір для затіненого проміжку.



Графік 3.8 – Прогноз курсу Ethereum

Таким чином, отримується візуалізація прогнозу обмінного курсу Ethereum з довірчим інтервалом.

## ВИСНОВКИ

У цій дипломній роботі було проведено чисельний прогноз криптовалюти на основі даних останніх років, використовуючи різні моделі, такі як ARIMA, нейронні мережі та експоненційне згладжування. Метою дослідження було визначити, наскільки точно можна передбачити майбутню динаміку криптовалюти та порівняти ефективність різних моделей.

При проведенні дослідження були використані статистичні методи ARIMA, які ґрунтуються на аналізі часових рядів та дозволяють врахувати тренди та сезонність у даних. Крім того, була застосована нейронна мережа, здатна виявляти складні нелінійні залежності в даних, та метод експоненційного згладжування, який приділяє особливу увагу останнім спостереженням.

Результати дослідження показали, що кожна з використаних моделей має свої переваги та обмеження. ARIMA-модель показала гарну точність прогнозу за наявності явних трендів та сезонності у даних, але мала деякі труднощі із передбаченням нелінійних змін. Нейронна мережа впоралася з цим завданням найкраще, дозволяючи врахувати складні залежності, але вимагала більшого обсягу даних та обчислювальних ресурсів. Метод експоненційного згладжування виявився простим, але неефективним, особливо при прогнозуванні короткострокових трендів.

В цілому, проведене дослідження дозволило отримати корисні висновки щодо застосування різних моделей для прогнозування криптовалюти. Рекомендується використовувати комбінацію різних моделей та методів для досягнення найточніших результатів прогнозування. Більш широкі та детальні дослідження в цій галузі можуть

призвести до покращення прогностичних моделей та більш точних передбачень для криптовалютних ринків та інвестицій.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. How may people Own Bitcoin? URL <https://www.buybitcoinworldwide.com/how-many-bitcoin-users/#:~:text=With%20one%20study%20suggesting%2025,the%20world's%20population%20owns%20bitcoin>. (дата звернення: 15.04.2023).
2. Robert H. Sgumway, David S. Stoffer. Time Series Analysis and Its Applications. 2021. Springer. С. 316-339.
3. Демківський Є.О., Демківська Т.1. Прогнозувани нестационарних фінансово-економічних процесів. IV- Міжнародна науково-практична конференція «Мехатронні системи: інновації та інжиніринг»: Київ, КНУТД. 22 жовтня 2022 року, с. 39-40
4. URL: <https://en.wikipedia.org/wiki/Cryptocurrency>
5. Principles of Predictive Abalytics and the Path to Time-Series Predictions URL <https://medium.com/@ODSC/principles-of-predictive-analytics-and-the-path-to-time-series-predictions-a61fb547efa6> (дата звернення: 24.03.2023).
6. Rolling Window Regression: a Simple Approach for Time Series Next value Predictoins URL: <https://medium.com/making-sense-of-data/time-series-next-value-prediction-using-regression-over-a-rolling-window-228f0aca363> (датазвернення: 31.03.2023).
7. Financial, Economic and Alternative Data. Quandl URL: <https://www.quandl.com/> (дата звернення: 10.04.2023)
8. Бідюк П.І., Коршевніук Л.О. Проектування комп'ютерних інформаційних систем підтримки прийняття рішень. Київ, 2022. С. 115-136.
9. Probabilisic data analysis in forecasting nonstationary processes. Kuznietsova N.V., Polozhayenko S.A., Levenchuk LB.. System Analysis

- and Information Technologies: IEEE-Xplore, SAIC - 2020; Kyiv, October, 2021, pp. 297 - 302.
10. Spirtes, P., Glymour, C. Schines, R., Causation, prediction and search. Adaptive computation and machine learning. MIT press, 2001. P. 540-600
  11. Бідюк П.І., Меняйленко О.С., Половцев О. В. Методи прогнозування. Луганськ: Альма-Матер, 2008. С. 583-607.
  12. URL: <https://www.coingecko.com/> (дата звернення: 01.04.2023).
  13. Facebook Inc Stock Predictions URL: <https://financhill.com/stock-forecast/fb-stock-prediction> (дата звернення: 20.04.2023).
  14. Financial Forecasting Preparing a prediction of the future URL: <https://corporatefinanceinstitute.com/resources/knowledge/modeling/financial-forecasting-guide/> (дата звернення: 23.02.2021).
  15. URL: <https://www.investopedia.com/terms/c/cryptocurrency.asp#:~:text=The%20advantages%20of%20cryptocurrencies%20include,and%20use%20in%20criminal%20activities.> (дата звернення: 20.04.2023).
  16. URL: <https://www.coinbase.com/learn/crypto-basics/what-is-staking> (дата звернення: 18.04.2023)
  17. Breiman L., Friedman J.H., Olshen R. A., Stone C. J. Classification and regression trees. Monterey, CA: Wadsworth & Brooks/Cole Advanced Books & Software. 1984. P. 312.
  18. A new correlation coefficient between categorical, ordinal and interval variables with Pearson characteristics URL: <https://arxiv.org/pdf/1811.11440.pdf> (дата звернення: 23.03.2023)
  19. Financial Time Series Forecasting with Deep Learning:A Systematic Literature Review: 2005-2019 URL: <https://arxiv.org/pdf/1911.13288.pdf> (дата звернення: 05.03.2023)
  20. Stochastic Recurrent Neural Network for Multistep Time Series Forecasting. URL: <https://arxiv.org/pdf/2104.12311.pdf> (дата звернення: 21.04.2021)

21. Multivariate time series forecasting URL:  
<https://towardsdatascience.com/multivariate-time-series-forecasting-65337263db36> (дата звернення: 10.03.2023)
22. Reconstruct Google Trends Daily Data for Extended Period URL:  
<https://towardsdatascience.com/reconstruct-google-trends-daily-data-for-extended-period-75b6cald3420> (дата звернення: 11.03.2023)
23. Introduction to interactive Time Series Visualizations with Plotly in Python URL:  
<https://towardsdatascience.com/introduction-to-interactive-time-series-visualizations-with-plotly-in-python-d3219eb7a7af> (дата звернення: 12.03.2023)
24. Machine Learning models for 100% better return in Algo-trading URL:  
<https://medium.datadriveninvestor.com/machine-learning-models-for-market-beating-trading-strategies-c773ba46db66> (дата звернення: 13.03.2023)
25. How powerful can an ensemble of linear models be? URL:  
<https://towardsdatascience.com/how-powerful-can-an-ensemble-of-linear-models-be-231824de50e1> (дата звернення: 14.03.2023)
26. How-To Guide on Exploratory Data Analysis for Time Series Data URL:  
<https://medium.com/analytics-vidhya/how-to-guide-on-exploratory-data-analysis-for-time-series-data-34250ff1d04f> (дата звернення: 15.03.2023)
27. URL: <https://supraoracles.com/academy/what-are-crypto-airdrops-and-how-do-they-work/> (дата звернення: 16.03.2023)
28. Feature Selection Techniques URL: <https://medium.com/analytics-vidhya/feature-selection-techniques-26146367efed> (дата звернення: 17.03.2021)
29. URL: <https://machinelearningmastery.com/exponential-smoothing-for-time-series-forecasting-in-python/> (дата звернення: 17.03.2023)

30. Calibration in Machine Learning URL: <https://medium.com/analytics-vidhya/calibration-in-machine-learning-e7972ac93555> (дата звернення: 18.03.2023)
31. Hyper parameter optimization Python URL: <https://towardsdatascience.com/hyperparameter-optimization-in-python-part-2-hyperopt-5f661db91324> (дата звернення: 19.03.2023)
32. URL: <https://coinmarketcap.com>

# КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

## СИСТЕМА ЗАПОБІГАННЯ ТА ВИЯВЛЕННЯ АКАДЕМІЧНОГО ПЛАГІАТУ

Довідка про оригінальність кваліфікаційної роботи за освітнім рівнем бакалавр



Ім'я користувача:  
Оноцький В'ячеслав ФКомпНаук

ID перевірки:  
1015303615

Дата перевірки:  
29.05.2023 18:52:09 EEST

Тип перевірки:  
Doc vs Internet + Library

Дата звіту:  
29.05.2023 18:55:31 EEST

ID користувача:  
100002816

Назва документа: ПавліченкоДанилоОлександрович

Кількість сторінок: 49 Кількість слів: 7833 Кількість символів: 60115 Розмір файлу: 574.50 KB ID файлу: 1014975185

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

**6.7%**  
**Схожість**

Найбільша схожість: 1.84% з Інтернет-джерелом (<https://www.fws.gov/sites/default/files/documents/adaptive-harvest-r>)

5.96% Джерела з Інтернету 190 ..... Сторінка 51

2.4% Джерела з Бібліотеки 67 ..... Сторінка 52

**0% Цитат**

Вилучення цитат вимкнено

Вилучення списку бібліографічних посилань вимкнено

**0%**  
**Вилучень**

Немає вилучених джерел

**Модифікації**

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 81

Підозріле форматування 14 сторінок

Експертна оцінка роботи науковим керівником :

Робота студента 4-го курсу Павліченка Данила Олександровича «Чисельний прогноз криптовалюти за даними останніх років» виконана самостійно, при цьому обсяг цитувань та запозичень становить 6.7% та відповідає належному рівню.

Науковий керівник:

(підпис)

Тимошенко А.А.

(ПБ)

Оператор:

(підпис)

Оноцький В.В.

(ПБ)

**ВІДГУК**  
**на випускню кваліфікаційну роботу бакалавра**  
**« Чисельний прогноз криптовалюти за даними останніх років. »**  
**студента 4-го курсу кафедри обчислювальної математики**  
**факультету комп'ютерних наук та кібернетики**  
**Київського національного університету імені Тараса Шевченка**  
**Павліченка Данила Олександровича**

В даній роботі проведено аналіз курсу криптовалюти Ethereum та факторів, що впливають на нього, використовуючи підходи: метод аналізу даних ARIMA (Autoregressive Integrated Moving Average), модель експоненційного згладжування та модель нейронної мережі. Автор роботи описує характерні риси криптовалюти та основні операції з нею, фактори що впливають на курс, переваги та недоліки. Далі використовуючи дані з відкритих джерел, виконується побудова моделі та здійснено порівняльний аналіз точності за трьома вище зазначеними підходами. Обчислювальний експеримент та побудовані графіки дозволяють наочно продемонструвати переваги та недоліки кожного з підходів.

Розглянуті методи та результати наведені у роботі є корисними для аналізу коливань курсу валют та мають опис параметрів та функцій для здійснення короткострокового прогнозу. Такий підхід кількох альтернативних способів прогнозу дозволяє бути більш впевненим у ризиках та прийнятті рішень при роботі з криптовалютами.

Студент самостійно виконав роботу та досяг необхідних результатів. Поєднавши математичну параметризацію із програмуванням, він продемонстрував належний рівень кваліфікації. Роботу можу оцінити на «відмінно».

Асистент кафедри обчислювальної  
математики факультету комп'ютерних  
наук та кібернетики  
Київського національного  
університету  
імені Тараса Шевченка,  
доктор філософії



Андрій  
ТИМОШЕНКО

## **РЕЦЕНЗІЯ**

**на кваліфікаційну роботу бакалавра:**

**«Чисельний прогноз криптовалюти за даними останніх років»**

**студента 4-го курсу кафедри обчислювальної математики**

**факультету комп'ютерних наук та кібернетики**


**Павліченка Данила Олександровича**

У роботі пропонується системний підхід до чисельного прогнозування криптовалют. Автор використовує різні статистичні методи, аналіз часових рядів і алгоритми машинного навчання, щоб зробити важливі висновки з даних. Розглянуто три підходи до моделювання - ARIMA, нейронні мережі та експоненційне згладжування. Включення даних за останні роки підвищить актуальність і точність прогнозів, сприяючи глибшому розумінню поведінки ринку.

Робота відображає підхід чисельного аналізу та моделювання для ринку криптовалют. За допомогою цієї методики та використання останніх даних, робота поєднує теоретичний підхід із його практичним застосуванням.

Вважаю, що кваліфікаційна робота студента відповідає вимогам до бакалаврських робіт і заслуговує на оцінку «відмінно», а її автор заслуговує на присвоєння кваліфікації бакалавра.

Професор кафедри обчислювальної  
математики факультету комп'ютерних  
наук та кібернетики  
Київського національного університету  
імені Тараса Шевченка,  
доктор фізико-математичних наук



**Дмитро КЛЮШИН**