

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

САБАТ АНДРІЙ ОЛЕКСАНДРОВИЧ

Допускається до захисту:
завідувач кафедри
інформаційних технологій,
д. т. н., доцент

_____ Т. В. Нескородева
« _____ » _____ 2022р.

**ДОСЛІДЖЕННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ РИНКУ
КРИПТОВАЛЮТ**

Спеціальність 122 Комп'ютерні науки

Кваліфікаційна (магістерська) робота

Науковий керівник:
Нескородева Т. В., завідувач кафедри
інформаційних технологій,
д.т.н., доцент

(підпис)

Оцінка: _____ / _____ / _____
(бали за шкалою ЄКТС/за національною шкалою)

Голова ЕК: _____
(підпис)

Вінниця 2022

АНОТАЦІЯ

Сабат А.О. Дослідження методів прогнозування часових рядів криптовалют. Спеціальність 122 «Комп'ютерні науки», Освітня програма «Комп'ютерні технології обробки даних (Data Science)». Донецький національний університет імені Василя Стуса, Вінниця, 2022.

У кваліфікаційній (магістерській) роботі досліджено методи прогнозування часових рядів ринку криптовалют, за допомогою розробки ПЗ для вирішення поставленої задачі.

У вступі окреслено ключові теми дослідження, визначено предмет, мету і завдання дослідження.

У першому розділі розглянуто теоретичні відомості про ринок та біржову торгівлю, а також проведений поверхневий огляд існуючих методів до прогнозу часових рядів.

Другий розділ присвячений більш детальному огляду існуючих методів прогнозу часових рядів та їх властивостей, а також визначені ключові особливості часових рядів і чим задачі часових рядів відрізняються від звичайних задач прогнозу.

Третій розділ присвячений реалізації методів прогнозу, їх застосуванню до показників на ринку криптовалют і порівнянню результатів.

У висновку підведено підсумок виконаної роботи, а також виконаний загальний аналіз та рекомендації щодо застосування методів прогнозу часових рядів.

Ключові слова: біржа, криптовалюта, волатильність, методи прогнозу часових рядів, нейромережі, тренувальні моделі.

91 с., 1 табл., 27 рис., 1 дод., 59 джерел.

ABSTRACT

Sabat A. Research of cryptocurrency market time series forecasting methods.

Specialty 122 "Computer science", Educational program "Computer data processing technologies (Data Science)". Vasyl Stus Donetsk National University, Vinnytsia, 2022.

In the qualification (master's) work, the methods of forecasting the time series of the cryptocurrency market were investigated, with the help of software development to solve the given problem.

In the introduction, the key topics of the research are outlined, the subject, purpose and tasks of the research are defined.

In the first chapter, theoretical information about the market and stock trading is considered, as well as a superficial review of existing methods for forecasting time series.

The second section is devoted to a more detailed review of existing methods of time series forecasting and their properties, as well as defining the key features of time series and how time series tasks differ from ordinary forecasting tasks.

The third section is devoted to the implementation of forecasting methods, their application to cryptocurrency and comparison of results.

The conclusion summarizes the work performed, as well as general analysis and recommendations for the application of time series forecasting methods.

Keywords: stock exchange, cryptocurrency, volatility, time series forecasting methods, neural networks, training models.

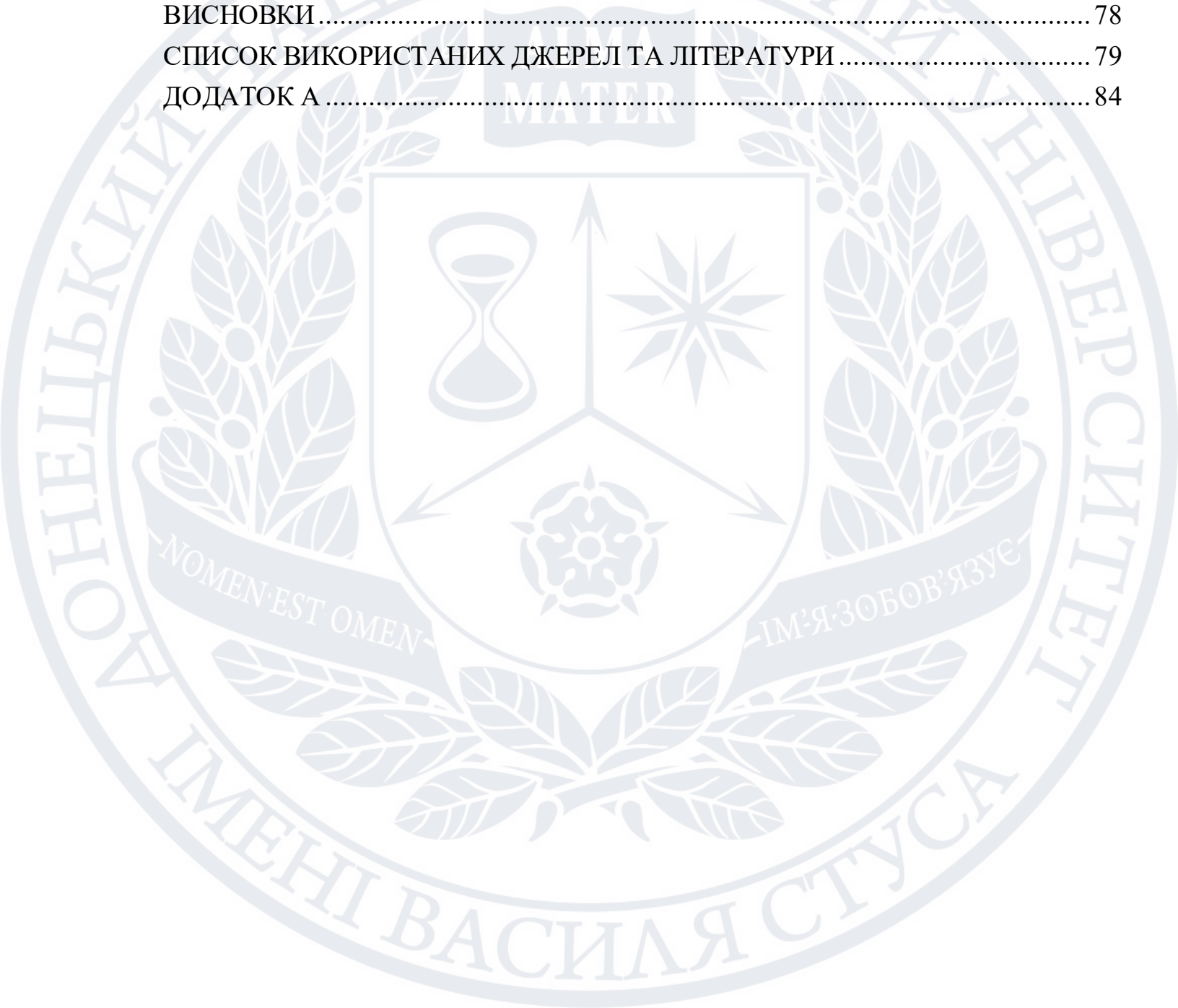
91 p., 27 figures., 1 tab., 59 ref.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 ПОНЯТТЯ РИНКУ ТА ОГЛЯД ІСНУЮЧИХ МЕТОДІВ АНАЛІЗУ І ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ	9
1.1 Поняття ринку.....	9
1.1.1 Структура ринку.....	9
1.1.1.1 Досконала конкуренція.....	10
1.1.1.2 Монополістична конкуренція.....	11
1.1.1.3 Олігополія.....	11
1.1.1.4 Монополія.....	13
1.1.2 Фондовий ринок.....	14
1.1.2.1 Індокси фондового ринку.....	18
1.1.2.2 Бичачий та ведмежі ринки.....	19
1.1.2.3 Аналіз акцій – ринкова капіталізація, прибутки на акцію та фінансові коефіцієнти.....	20
1.1.3 Криптовалютний ринок.....	22
1.1.3.1 Визначення криптовалюти.....	22
1.1.3.2 Блокчейн.....	23
1.1.3.3 Типи криптовалют.....	24
1.2 Поняття біржі.....	25
1.2.1 Основні відмінності між фондовою та криптовалотною біржами.....	25
1.2.1.1 Об’єкти торгівлі.....	25
1.2.1.2 Зрілість ринків.....	26
1.2.1.3 Волатильність.....	27
1.2.1.3 Охоплення ринку.....	28
1.2.1.4 Комісії та регулювання.....	28
1.3 Існуючі методи аналізу і прогнозування часових рядів.....	29
1.3.1 Авторегресивна інтегрована ковзна середня (ARIMA).....	30
1.3.2 Узагальнена авторегресивна умовна гетероскедастичність (GARCH)..	30
1.3.3 TBATS.....	31
1.3.4 LSTM.....	32

1.4 Висновки до розділу	32
РОЗДІЛ 2 ДЕТАЛЬНИЙ ОГЛЯД МОДЕЛЕЙ І ОСНОВНИХ ПІДХОДІВ У МЕТОДАХ АНАЛІЗУ І ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ.....	33
2.1 Особливості у прогнозуванні часових рядів	33
2.1.1 Зони аналізу та прогнозування часових рядів	33
2.1.2 Одновимірні та багатовимірні моделі часових рядів.....	36
2.1.3 Декомпозиція часових рядів.....	37
2.1.4 Автокореляція	40
2.1.4.1 ACF – автокореляційна функція	41
2.1.4.2 PACF – часткова автокореляційна функція	42
2.1.5 Стаціонарність.....	44
2.1.5.1 Тест Дікі-Фуллера.....	44
2.1.5.2 Розширений тест Дікі-Фуллера.....	46
2.1.5.3 Метод різниць	48
2.1.6 Однокрокові та багатокрокові моделі прогнозування часових рядів	50
2.2 Детальний огляд базових методів прогнозування часових рядів.....	51
2.2.3 Сімейство методів прогнозування ARIMA	51
2.2.3.1 Авторегресія (AR).....	51
2.2.3.2 Ковзне середнє (MA)	52
2.2.3.3 Авторегресивне ковзне середнє (ARMA)	53
2.2.3.4 Авторегресивне інтегроване ковзне середнє (ARIMA).....	54
2.2.3.5 Сезонне авторегресивне інтегроване ковзне середнє (SARIMA)	55
2.2.3.6 Сезонне авторегресивне інтегроване ковзне середнє з екзогенними регресорами (SARIMAX).....	56
2.2.4 Сімейство методів прогнозування GARCH.....	57
2.2.4.1 Авторегресивна умовна гетероскедастичність (ARCH)	57
2.2.4.1 Узагальнена авторегресивна умовна гетероскедастичність (GARCH)	58
2.2.4 Модель TBATS.....	59
2.2.5 Довга короткочасна пам'ять (LTSM)	61
2.3 Висновки до розділу	67

РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ПОРІВНЯННЯ ЕФЕКТИВНОСТІ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ.....	68
3.1 Опис інструментів для програмної реалізації.....	68
3.2 Практичне застосування методів на часовому ряді криптовалюти.....	69
3.3 Висновки до розділу	77
ВИСНОВКИ.....	78
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ.....	79
ДОДАТОК А.....	84



ВСТУП

За останні десять років сфера криптовалютного трейдингу набула неабиякого резонансу. Зараз дуже важко зустріти людину, яка б не чула нічого про криптовалютний ринок. Таке підвищення популярності пов'язано у першу чергу з тим, що вперше, за довгий час, з'явилась альтернатива фондовому ринку (або ринку цінних паперів). Разом з тим відбувався розвиток методів прогнозування часових рядів, що викликало прямий вплив і на підходи до прогнозування на ринках.

Сьогодні, серед найпопулярніших методів та технік для прогнозування, можна виділити наступні:

1. Сезонна декомпозиція

Очевидно, що деякі ряди даних можуть складатися не лише з певної сталої функції, але і з періодичної, в яких міститься так звана «сезонна» складова. Сезонна декомпозиція допомагає виявити цю складову і зробити проноз базуючись на отриманому значені.

2. Авторегресивна інтегрована ковзна середня (ARIMA)

Форма регресійного аналізу, що вимірює силу однієї залежної змінної відносно інших змінних. Метою моделі є прогнозування майбутніх рухів цінних паперів або фінансового ринку шляхом вивчення різниці між значеннями в серії замість фактичних значень.

3. Узагальнена авторегресивна умовна гетероскедастичність (GARCH).

Попередні моделі передбачали, що помилки в випадкових процесах, які породжують часові ряди, були гомоскедастичними, тобто з постійною дисперсією.

Натомість модель GARCH припускає, що дисперсія умов помилки слідує за процесом авторегресивного ковзного середнього (ARMA), отже, дозволяє їй змінюватися в часі. Це особливо корисно для моделювання фінансових часових рядів, волатильність яких змінюється з часом.

4. Динамічні лінійні моделі

Динамічні лінійні моделі представляють інший клас моделей для прогнозування часових рядів. Ідея полягає в тому, що в кожен момент часу t ці моделі відповідають лінійній моделі, але коефіцієнти регресії змінюються з часом.

5. TBATS

Модель TBATS – це модель прогнозування, заснована на експоненційному згладжуванні. Назва є акронімом для:

Trigonometric

Box-Cox transform

ARMA errors

Trend

Seasonal components

6. LSTM

Моделі LSTM, так само, як і інші рекурентні нейронні мережі можна використовувати для прогнозування часових рядів. LSTM – це аббревіатура для «Long-Short Term Memories» що означає «Довга короткочасна пам'ять».

Стан мережі LSTM представляється через вектор простору станів. Ця методика дозволяє відстежувати залежності нових спостережень з минулими (навіть дуже далекими).

Отже метою даної роботи стане дослідження ефективності усіх вище приведених методів прогнозування на прикладі застосування їх для прогнозування ціни на різноманітні криптовалютні монети.

РОЗДІЛ 1

ПОНЯТТЯ РИНКУ ТА ОГЛЯД ІСНУЮЧИХ МЕТОДІВ АНАЛІЗУ І ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

1.1 Поняття ринку

Ринком можна вважати місце, де дві групи можуть взаємодіяти певним чином, де одна група відповідає за продаж товару або надання послуг, а ніша – купує товар, або користується послугам [1]. Зазвичай групи, які приймають участь у цих процесах, прийнято називати продавцями або покупцями відповідно. Сам ринок – це фізична річ, на кшталт торгової точки, де продавці та покупці можуть фізично зустрітись один з одним, або взаємодіяти віртуально, якщо це онлайнова форма ринку. Тобто контакт між покупцями і продавцями визначається типом ринку, до якого ці групи належать.

Отож «ринок» – це термін, який може використовуватись у різних випадках і зазвичай назва ринку визначається самим об'єктом взаємодії на ринку [2]. Наприклад, якщо взаємодія між групами відбувається за допомогою цінних паперів, то сама площадка для виконання операції обміну буде мати назву «ринок цінних паперів», якщо взаємодія відбувається за допомогою криптовалют, то площадка буде мати назву «криптовалютний ринок» і т.д.

1.1.1 Структура ринку

Ринок може мати різні структури [3] і різноманітність цих структур обумовлена елементами економіки. Зазвичай ринкові структури, по суті, визначаються рівнями конкуренції на ринку, але також існують й інші визначники, які мають вплив на структуру ринку:

- Характер товарів та послуг
- Кількість покупців та продавців

- Економічний масштаб

Отже, розглянемо основні типи ринкових структур.

1.1.1.1 Досконала конкуренція.

У такому типі ринковій структурі, як досконала конкуренція, мається велика кількість покупців та продавців, де кожна група конкурує між собою. Також, у цьому типі ринковій структури, як правило, відсутній основний, або великий за ступеню свого впливу, продавець. Отже продавців на цьому ринку зазвичай називають «отримувачами цін» [4]. Тобто ціна на товар або послуги визначається найбільш оптимальною ціною, або іншими словами, такою ціною, яка задовольняє усіх в межах цього ринку [5].

Основні характеристики ринку досконалої конкуренції:

- Нескінченна кількість покупців і продавців. Нескінченно багато покупців з бажанням та можливістю придбати товар або послугу за певною ціною та нескінченна кількість продавців з бажанням та можливістю надавати товар або послуги за певною ціною.
- Нульовий поріг входу та виходу. Відсутність порогів для входу і виходу надає можливість зайти і вийти на ринок без зайвих витрати за зусиль.
- Ідеальна мобільність виробництв. У довгостроковій перспективі виробництва є дуже мобільними, що дозволяє дуже легко пристосовуватись до мінливих умов ринку.
- Ідеальне знання інформації про продукт. Усі покупці та продавці чудово усвідомлюють ціну, якість і методи виробництва товару або послуги.
- Нульові витрати на транзакції. Покупці та продавці не витрачають додаткові кошти при обміні товарами та послугами.

- Максимізація прибутків. Припускається що продавці надають товар або послуги таким чином, що граничні витрати на товар або послугу відповідають граничному доходу за продаж.
- Однорідність. Якості та характеристики товарів чи послуг не відрізняються між постачальниками.
- Права власності. Чіткі права на власність визначають, які товари та послуги можуть бути продані, а також, які права та зобов'язання покладені на покупця.

1.1.1.2 Монополістична конкуренція

Цей тип ринкової структури є цілком реалістичним. В монополістичній конкуренції також присутня велика кількість покупців та продавців, але на відміну від досконалої конкуренції, продавці продають не однорідні товари, а схожі, або ж цілком різні [6]. При такій структурі продавці можуть змінювати ціну на товари або послуги за їхнім бажанням, оскільки вони мають домінуючу позицію в цій формі ринкової структури [5].

Основні характеристики ринку монополістичної конкуренції:

- Багато покупців та продавців. На ринку присутня велика кількість покупців та продавців, але жоден продавець не має контролю над загальною ринковою ціною.
- Покупці спостерігають нецінову відмінність між товарами продавців.
- Наявні бар'єри для входу і виходу з ринку.
- Продавці мають певний контроль над ціною.

1.1.1.3 Олігополія

В цій ринковій структурі, тільки декілька продавців присутні на ринку. Кількість покупців значно перевищує кількість продавців. У випадку олігополії,

продавці вже можуть не тільки конкурувати, а й співпрацювати використовуючи їхній вплив на ринок і виставляти ціни за власним бажанням максимізуючи таким чином прибуток [7]. Отож у цьому випадках вже покупці становляться «отримувачами цін». В олігополії присутні деякі бар'єри для входу на ринок, і новим продавцям дуже важко закріпитись у цьому типі ринкової структури [5].

Основні характеристики ринку олігополістичної конкуренції:

- У олігополії максимізація прибутку відбувається тоді, коли граничний дохід відповідає граничним витратам.
- Здатність встановлювати ціну. Продавці встановлюють ціни, а не отримують ціни.
- Вхід і вихід: високі бар'єри для входу. Найважливішими перешкодами є масштабування, патенти, доступ до дорогих та комплексних технологій і стратегічні дії існуючих продавців спрямовані на те, щоб перешкоджати та знищувати нових. Також часто додатковим джерелом бар'єрів на вхід виступає державне регулювання, яку надає різні переваги існуючим продавцям і цим самим ускладнює вхід на ринок новим продавцям.
- Кількість продавців настільки невелика, що це призводить до того, що дії одного продавця часто впливають на інших.
- Довгострокові прибутки. Продавці можуть утримувати аномально-високі ціни у довготривалій перспективі за рахунок того, що високі бар'єри заважають входу стороннім продавцям для створення конкуренції ціноутворення.
- Ідеальні знання продавців інформації про продукт. Продавці добре знають свої власні витрати та попит, але покупці мають недосконалі знання що до цих метрик.
- Взаємозалежність. Відмінною рисою олігополії є взаємозалежність продавців. Олігополії зазвичай складаються з декількох продавців і кожен

продавець настільки великий, що їх дії швидко впливають на ринкові умови. Це означає, що виконуючи або обдумуючи будь-яку дію на ринкові, продавець має брати до уваги можливі дії та контр ходи усіх інших конкуруючих продавців. Це дуже сходиться з грою у шахи або більярд, де гравець має передбачити цілу низку можливих ходів або контр ходів щоб визначити, як досягти своїх цілей. Наприклад, якщо один з продавців вирішить понизити ціну на товар чи послугу, він має брати до уваги те, що конкуренти також можуть понизити свої ціни, що може призвести до руйнівної цінової війни. Такий високий ступінь взаємозалежності та необхідність усвідомлення того, що роблять або можуть зробити інші продавці конкуренти, знову ж таки, створює додаткові бар'єри для нових фірм у цьому ринку.

- Нецінова конкуренція. Як правило, продавці в даній ринковій структурі надають перевагу конкуруванню на умовах існування, на відміну від цінової конкуренції. Схеми лояльності, реклама та різноманіття товарів та послуг – усе це є прикладом нецінової конкуренції.

1.1.1.4 Монополія

У монопольному типі ринкової структури є тільки один продавець, де під цим визначенням розуміють повний контроль ринкової структури в одинці. Продавець у цьому випадку може встановлювати будь-який рівень ціни за власним бажанням, оскільки вся сила ринку знаходиться під його домінуванням [8]. Через це, покупці не мають жодних альтернатив сплати цін, окрім встановленої одним продавцем.

Монополії є найбільш не бажаним типом ринкової структури, оскільки покупці втрачають свою силу і ринкові сили стають неактуальними. Однак, монополія у чистому вигляді зустрічається у реальності дуже рідко.

Основні характеристики монополістичного ринку [5]:

- Єдиний постачальник. Монополістичний ринок регулюється єдиним продавцем, отож ринковий попит на товар чи послугу – це попит на товар чи послугу з точки зору одного продавця.
- Бар'єри на вхід чи вихід. Патенти, авторські права, володіння ресурсами, зменшення загальних середніх витрат і значні початкові витрати є значущими бар'єрами для входу на монополістичний ринок. Коли один продавець контролює виробництво та постачання певного товару чи послуги, інші компанії не можуть увійти на ринок. Якщо влада вважає, що товар чи послуга яка надається продавцем є необхідною для покупця, то продавцю можуть заборонити покидати ринок.
- Максимальний прибуток. В монополістичному ринку продавець максимізує прибуток. Він може виставити ціни набагато вищими, ніж вони могли би бути при конкурентному ринку. Отож через те, що конкуренції немає, ринковою ціною на товар чи послугу вважається ціна продавця.
- Унікальність товару. В монополістичному ринкові, товар чи послуга яка надається продавцем – є унікальною. Немає жодних схожих товарів чи послуг на усьому ринкові.
- Цінова дискримінація. Продавець, який знаходиться у монополістичному ринкові, може вільно змінювати ціну або кількість товарів чи послуг. Цінова дискримінація виникає тоді, коли продавець виставляє один і той самий товар чи послугу за різними цінами для різних покупців.

1.1.2 Фондовий ринок

Фондовий, або як його ще часто називають, стоковий ринок – це вид публічного ринку який існує для того, щоб виконувати операції випуску, купівлі та продажу акцій, якими торгують на фондовій біржі або поза нею [10]. Акція у цьому

випадку представляє з себе частку власності в компанії, а сам фондовий ринок – є місцем, де інвестори можуть купувати та продавати власність на такі активи. Ефективно функціонуючий ринок, вважається критично важливим для економічного розвитку, так як він надає компаніям можливості до швидкого нарощування капіталу з публіки.

Сам фондовий ринок служить для двох дуже важливих цілей. Перша – це надавати можливість набрати капітал компаніям які вони зможуть використовувати для фінансування та розширення свого бізнесу. Якщо компанія випускає один мільйон акцій, які для прикладу можуть продаватися за 10 доларів за акцію, то це забезпечує компанію капіталом у розмірі 10 мільйонів доларів, який вона може використати для розвитку свого бізнесу. Таким чином пропонуючи акції замість запозичення капіталу, необхідного для розширення, компанія уникає боргів і сплати відсотків за цими боргами.

Друга ціль фондового ринку – це надавати інвесторам, які купують акції, можливість отримувати частину від прибутків компанії. Інвестори можуть отримати прибуток від купівлі акцій одним із двох способів.

- Деякі акції виплачують регулярні дивіденди (певна сума грошей за акцію, якою хтось володіє).
- Інвестори можуть отримати прибуток від акцій через їх продаж, якщо ціна акцій зросла від ціни на момент їхньої покупки.

Наприклад, якщо інвестор купує акції компанії по 10 доларів за акцію, а ціна акції згодом зростає до 15 доларів за акцію, інвестор може отримати 50% прибутку від своїх інвестицій, продавши свої акції.

Перші акції компаній випускалися на папері, що дозволяло інвесторам торгувати акціями з іншими інвесторами, але регульованих бірж не існувало до створення Лондонської фондової біржі у 1773 році [10]. І не дивлячись на те, що значні фінансові потрясіння негайно послідували після заснування Лондонської та

Нью-Йоркської фондових бірж – біржова торгівля змогла вижити та активно розвивалася упродовж 1800-х.

Загалом на внутрішньому ринку Нью-Йоркська фондова біржа спостерігала мізерну конкуренцію протягом більше двох століть, і її зростання в основному сприяло постійному зростанню американської економіки. У цей же час, Лондонська біржа продовжувала домінувати на європейському ринку торгівлі акціями, але Нью-Йоркська біржа перетворилася у дім для постійно зростаючої кількості великих компаній. Інші великі країни, такі як Франція та Німеччина, зрештою розробили власні фондові біржі, хоча вони часто розглядалися в першу чергу як сходинки для компаній на шляху до лістингу на Лондонській або Нью-Йоркській фондових бірж.

Наприкінці 20-го століття, торгівля акціями поширилася на багатьох інших біржах, у тому числі і на NASDAQ [11], яка стала улюбленою домівкою технологічних компаній що розвиваються, і набула особливого значення під час буму технологічного сектора в 1980-х і 1990-х роках. NASDAQ з'явилася як перша біржа, що працює у мережі комп'ютерів, які здійснюють угоди в електронному вигляді. Електронна торгівля [12] зробила весь процес торгівлі більш ефективним за часом і рентабельним. На додаток до зростання NASDAQ, Нью-Йоркська фондова біржа зіткнулася з посиленням конкуренції з боку фондових бірж Австралії та Гонконгу – фінансового центру Азії.

На теперішній час, більшість акцій торгуються на таких біржах, як Нью-Йоркська фондова біржа або NASDAQ. Фондові біржі по суті підтримують ринок шляхом полегшення купівлі та продажу акцій серед інвесторів. Фондові біржі регулюються державними установами, такими як Комісія з цінних паперів і бірж США [13], які спостерігають за ринком, щоб захистити інвесторів від фінансового шахрайства та забезпечити безперебійне функціонування обмінного ринку. Хоча переважна більшість акцій торгується на біржах, деякі акції торгуються поза

біржою (OTC), де покупці та продавці акцій зазвичай торгують через дилера, або «маркет-мейкера», який спеціалізується на роботі з акціями. Позабіржові акції – це акції, які не відповідають мінімальній ціні чи іншим вимогам для розміщення на біржах.

На позабіржові акції не поширюються ті ж правила публічної звітності, що й на акції, що котируються на біржах, тому інвесторам не так легко отримати достовірну інформацію про компанії, які випускають такі акції. Акції на позабіржовому ринку зазвичай торгуються набагато рідше, ніж акції, що торгуються на біржі, а це означає, що інвесторам часто доводиться мати справу з великими різницями між цінами пропозиції та попиту на позабіржові акції. На противагу, акції, що торгуються на біржі, є більш ліквідними, з відносно невеликими різницями між пропозицією та попитом.

Гравцями на фондовому ринку є – інвестиційні банки, біржові брокери та інвестори [14]. Інвестиційні банки здійснюють первинне публічне розміщення (IPO) акцій, яке відбувається, коли компанія вперше вирішує стати публічною компанією, пропонуючи акції.

Ось приклад того, як працює IPO. Компанія, яка бажає вийти на біржу та запропонувати свої акції, звертається до інвестиційного банку з проханням виступити посередником первинного розміщення акцій компанії. Інвестиційний банк після дослідження загальної вартості компанії та врахування того, від якого відсотка власності компанія бажає відмовитися у формі акцій, проводить початковий випуск акцій на ринку за плату, визначаючи мінімальну ціну за акцію встановлену для компанії. Також, в інтересах інвестиційного банку переконатися, що всі запропоновані акції будуть продані за найвищою можливою ціною.

Акції, які пропонуються під час IPO, найчастіше купують великі інституційні інвестори, такі як пенсійні фонди або компанії взаємних фондів.

Ринок IPO відомий як первинний або початковий ринок. Після випуску акцій на первинному ринку вся торгівля акціями відбувається через фондові біржі на так званому вторинному ринку. Термін «вторинний ринок» дещо вводиться в оману, оскільки це ринок, на якому щодня відбувається переважна більшість біржових торгів.

Біржові брокери, також можуть виступати в якості фінансових радників, купуючи і продаючи акції для своїх клієнтів, які можуть бути або інституційними інвесторами, або індивідуальними роздрібними інвесторами.

Аналітики з дослідження акцій можуть працювати в біржових брокерських компаніях, компаніях взаємних фондів або інвестиційних банках. Це особи, які досліджують публічні компанії та намагаються спрогнозувати, наскільки ймовірно зростання чи падіння ціни акцій тієї чи іншої компанії.

Менеджери фондів або менеджери портфелів, до яких належать менеджери взаємних фондів і менеджери біржових фондів, є важливими учасниками фондового ринку, оскільки вони купують і продають велику кількість акцій. Якщо популярний фонд вирішує інвестувати значні кошти в певну акцію, то попит на ці акції зростає що у наслідку є достатнім фактором для того, щоб ціна на акцію пішла угору.

1.1.2.1 Індeksi фондового ринку

Загальна продуктивність фондового ринку зазвичай відстежується та відображається в показниках різних індексів. Біржові індeksi складаються з вибірки акцій, яку розроблено для відображення загальної ефективності акцій [15]. Самі індeksi фондового ринку торгуються у формі опціонів і ф'ючерсних контрактів, які також торгуються на регульованих біржах.

Серед ключових індексів фондового ринку є:

- Dow Jones Industrial Average (DJIA) – індекс Нью-Йоркської фондової біржі. До нього входять 30 акцій промислових підприємств.
- NASDAQ Composite (IXIC) – індекс американської біржі NASDAQ. Він розраховується за котируваннями всіх акцій, що торгуються. В основному це інтернет-холдинги, виробники електроніки та програмного забезпечення. Майже третина ваги NASDAQ Composite припадає на п'ять компаній: Microsoft, Apple, Amazon, Facebook та Alphabet.
- FTSE 100 (FTSE) – індекс найбільшої в Європі Лондонської біржі, який розраховується компанією FTSE Group на основі ціни акцій 100 компаній з найбільшою капіталізацією, котируються на Лондонській фондовій біржі.
- DAX – фондовий індекс Німеччини, до якого входить 30 акцій найбільших німецьких компаній. Індекс розраховує компанія Deutsche Börse AG.
- CAC 40 – фондовий індекс Франції. До нього входять 40 найбільших компаній, які торгуються другою за величиною біржі Європи Euronext Paris.
- S&P 500 (SPX) розраховується агентством Standard & Poor's за котируваннями 500 акцій найбільших світових компаній.
- Nikkei 225 (N225) – індекс Токійської фондової біржі, розраховується як просте середнє арифметичне значення цін 225 найбільш ліквідних акцій.
- Shanghai Composite (SSEC) – індекс Шанхайської фондової біржі.

1.1.2.2 Бичачий та ведмежі ринки

Двома основними концепціями торгівлі на фондовому ринку є «бичачий» і «ведмежий» ринки [16]. Термін бичачий ринок використовується для позначення фондового ринку, на якому ціна акцій у загальному зростає. Це тип ринку, на якому процвітає більшість інвесторів, оскільки більшість інвесторів у акції є покупцями,

а не продавцями. Ведмежим ринок називають тоді, коли ціни на акції в цілому знижуються.

Інвестори все ще можуть отримати прибуток навіть на ведмежих ринках через короткі продажі. Короткий продаж – це практика запозичення акцій, які інвестор не має, у брокерської фірми, яка володіє акціям [17]. Потім інвестор продає позичені акції на вторинному ринку та отримує гроші від продажу цих акцій. Якщо ціна акцій знижується, як сподівається інвестор, тоді інвестор може отримати прибуток, придбавши ту саму кількість акцій яку продав до цього, щоб повернути брокеру кількість акцій, яку вони позичили за загальною ціною нижчою від тієї, яку вони отримали за продаж акцій раніше за вищою ціною.

Наприклад, якщо інвестор вважає, що акції компанії «А», ймовірно, впадуть із поточної ціни в 20 доларів за акцію, інвестор може внести так званий маржинальний депозит, щоб позичити 100 акцій у його брокера. Потім він продає ці акції по 20 доларів за кожну, що дає йому у результаті 2000 доларів. Якщо акція потім впаде до 10 доларів за акцію, інвестор може купити 100 акцій, щоб повернути своєму брокеру лише за 1000 доларів, залишаючи при цьому собі прибуток у 1000 доларів.

1.1.2.3 Аналіз акцій – ринкова капіталізація, прибутки на акцію та фінансові коефіцієнти

Аналітики фондового ринку та інвестори зазвичай розглядають низку факторів, для того, щоб вказати ймовірний майбутній напрямок ціни акцій – підвищення або зниження. Найбільш популярним індикатором для аналізу акцій є ринкова капіталізація акцій. Ринкова капіталізація – це загальна вартість усіх акцій, що знаходяться в обігу [18]. Висока ринкова капіталізація зазвичай вказує на те, що компанія є розвиненою та фінансово стабільною.

Регуляторні органи біржі зобов'язують публічні компанії регулярно надавати звіти про прибутки. Ці звіти, які публікуються щоквартально та щороку, ретельно перевіряються аналітиками ринку, і являються хорошим індикатором того, наскільки добре йде бізнес компанії. Серед ключових факторів, проаналізованих у звітах, є прибуток компанії на акцію (EPS), який відображає прибуток компанії, розподілений між усіма її акціями в обігу.

Аналітики та інвестори також часто використовують низку інших фінансових коефіцієнтів, які показують фінансову стабільність, прибутковість і потенціал зростання публічної компанії. Нижче наведено кілька ключових фінансових показників, якими користуються інвестори та аналітики:

- **Ціна до прибутку (P/E):** співвідношення ціни акцій компанії до її прибутку на акцію. Високий коефіцієнт (P/E) вказує на те, що інвестори готові платити вищу ціну за акцію за акції компанії, оскільки вони очікують зростання компанії та підвищення ціни акцій.
- **Дохід на акціонерний капітал (ROE):** Коефіцієнт рентабельності власного капіталу (ROE) вважається хорошим показником потенціалу зростання компанії, оскільки він показує чистий прибуток компанії по відношенню до загальних інвестицій у власний капітал компанії.
- **Рентабельність прибутку:** існує кілька коефіцієнтів маржинальності, які інвестори можуть враховувати, включаючи маржу операційного прибутку та маржу чистого прибутку. Перевага перегляду рентабельності прибутку, а не просто цифри прибутку в доларах, полягає в тому, що він показує відсоток прибутковості компанії. Наприклад, компанія може показати прибуток у два мільйони доларів, але якщо цей прибуток відображає лише 3% прибутку компанії, тоді будь-яке значне зниження доходів може поставити під загрозу прибутковість компанії.

- Співвідношення боргу до власного капіталу: це фундаментальний показник фінансової стабільності компанії, оскільки він показує, який відсоток операцій компанії фінансується за рахунок боргових зобов'язань порівняно з тим, який відсоток фінансується інвесторами акцій. Чим нижче співвідношення боргу до власного капіталу – тим краще, оскільки це вказує на те, що капітал в основному складається з інвесторів.

1.1.3 Криптовалютний ринок

На відміну від фондового ринку, операції на криптовалютному ринку відбуваються виключно онлайн. Тобто це означає, що фізично провести операцію обміну не можна. Якщо основною сутністю на фондовому ринку можна було вважати актив, то на криптовалютному ринку, як це вже можна зрозуміти з назви, є – криптовалюта.

1.1.3.1 Визначення криптовалюти

Криптовалюта – це цифрова або віртуальна валюта, захищена криптографією, що унеможливорює варіанти її підробки або подвійної витрати [19]. Багато криптовалют існують у децентралізованих мережах, заснованими на технології блокчейн – розподіленому реєстрі, який керується мережею комп'ютерів [20].

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

Слово «крипто» відноситься до різних алгоритмів шифрування та криптографічних методів, які захищають ці записи, наприклад шифрування за еліптичною кривою, пари відкритих і закритих ключів і функції хешування.

Криптовалюти можна «майнити», тобто добувати за допомогою певної кількості зробленої роботи, або купувати на біржах криптовалют. Не всі сайти

електронної комерції дозволяють покупки за допомогою криптовалют. Насправді криптовалюти, навіть такі популярні, як біткойн, майже не використовуються для роздрібних операцій. Однак стрімке зростання цінності криптовалют зробило їх популярними у вигляді інструментів для торгівлі. В обмеженій мірі вони також використовуються для транскордонних переказів.

1.1.3.2 Блокчейн

Центральне місце в привабливості та функціональності криптовалют займає технологія блокчейн. Блокчейн – це тип технології розподіленої бази даних, яка складається зі списку записів, які називаються блоками, що надійно пов'язані між собою за допомогою криптографії [20].

Найбільш відомою реалізацією технології блокчейну є криптовалюта Bitcoin, яка зберігає усі данні про транзакції у блоках і зберігає їх у вигляді послідовності. Тобто кожен блок містить часову позначку – хеш попереднього блоку та дані транзакцій, подані як хеш-дерево.

Кожен новий згенерований блок має бути перевірений кожним учасником мережі перед підтвердженням, що унеможлиблює підробити історію транзакцій. Вміст блокчейну погоджується усією мережею окремих вузлів або комп'ютером, який підтримує копію журналу.

Блокчейн має наступні властивості:

- Кожен блок даних у ланцюжку складається з трьох елементів: самих даних транзакції, 32-розрядного одноразового числа, яке генерується випадковим чином під час створення блоку, і вищезгаданого хешу.
- Коли блок даних отримує мітку часу, він підписується згенерованим одноразовим номером, зв'язується та перетворюється на криптографічний хеш.

- Учасники мережі, які тримають копії блокчейна, створюють ці блоки в процесі, який називається майнінг. Оскільки кожен блок не має хеш-штампів, окрім посилання на попередній блок у ланцюжку, майнінг стає складним завданням.

1.1.3.3 Типи криптовалют

Bitcoin (BTC) є найпопулярнішою та найдорожчою криптовалютою. Анонім на ім'я Сатоші Накамото винайшов її та представив світові у 2008 році [21]. Сьогодні на ринку існують тисячі криптовалют.

Коли справа доходить до розподілення криптовалют по типам, то більшість із них є так званими «гілками» від біткойна, але деякі інші були створені з нуля. Наразі існує лише 3 типи криптовалют:

- Bitcoin. Це перша криптовалюта, яка коли-небудь була представлена, і вважається «цифровим золотом». Наразі його ринкова капіталізація становить 367 мільярдів доларів, що становить найбільше значення серед усіх інших варіантів криптовалют.
- Altcoins. Ця категорія насамперед включає розгалуження та альтернативні версії біткойна. Однак деякі альткоїни експоненційно відрізняються від біткойна і використовують різні алгоритми. Наприклад альткойн Ethereum, є не валютою – а платформою, де організації можуть створювати свої програми на основі блокчейну.
- Tokens. Токени зазвичай створюються та розповсюджуються через первинну пропозицію монет (ICO), подібно до первинної публічної пропозиції (IPO) акцій. Як і американські долари, токени мають свою цінність, але цінність кожної їхньої окремої одиниці може відрізнятись, так само як вартість паперового долара може бути не один долар. Також, токени можна використовувати в транзакціях для інших речей. Токен відрізняється від монети тим, як він побудований у блокчейні існуючої монети, як, наприклад, у Bitcoin або Ethereum.

1.2 Поняття біржі

Біржа – це відкритий, організований ринок товарів, акцій, цінних паперів, деривативів та інших фінансових інструментів [22]. Терміни біржа та ринок часто використовуються як синоніми, оскільки вони обидва описують середовище, в якому товарами, що перебувають у списку, можна торгувати.

Ключовою функцією біржі є надання чесних, ефективних і впорядкованих торгових можливостей шляхом централізації купівлі та продажу певного типу активів. Замість того, щоб мати справу безпосередньо з біржами, більшість трейдерів надають перевагу у використанні посередника – брокера.

Біржі розташовані в більшості країн світу. Це може бути як фізичне місце, де трейдери збираються для ведення бізнесу, або електронна платформа.

Традиційно, біржа була фізичним місцем, яке використовувалося для торгівлі цінними паперами, яке функціонувало через систему відкритої ставки або подвійного аукціону. На теперішній час більшість бірж є електронними, хоча до недавнього часу деякі ще пропонували фізичну біржову торгівлю, зокрема Нью-Йоркська фондова біржа (NYSE) і Чиказька біржа опціонів (CBOE).

З швидкою цифровою трансформацією сучасної економіки, визначення біржі також значно змінилося. Оскільки традиційні біржі почали обробляти дедалі більший обсяг торгових потоків, більшість угод почали розміщуватись в електронному вигляді поза торговим майданчиком. Деякими прикладами суто цифрових бірж є фондова біржа NASDAQ або біржі криптовалют.

1.2.1 Основні відмінності між фондовою та криптовалютною біржами

1.2.1.1 Об'єкти торгівлі

Головна відмінність між біржами криптовалют і фондовими біржами полягає у типах активів, якими вони оперують. Фондова біржа торгує акціями або акціями

компанії, тоді як криптовалютна біржа торгує криптовалютами (цифровими валютами), такими як Bitcoin, Ethereum та багато інших.

Акції, якими торгують на фондових ринках, представляють собою капітал компанії. Коли ви купуєте акції компанії через фондову біржу, ви стаєте співвласником самої компанії. Те, наскільки є успішною компанія, також визначає вартість ваших акцій.

Купівля криптовалюти – будь то монета чи токен – не обов’язково означає часткове право власності на компанію, яка її випустила. Це цифрова валюта, тому її вартість суб’єктивна. Криптовалюту набагато легше придбати, ніж акції.

Відповідно до місцевих законів, публічна компанія може випускати акції за бажанням, щоб залучити гроші. На відміну від акцій, більшість криптовалют мають обмежену кількість монет або токенів. З цієї причини базова економічна теорія припускає, що вартість обмежених криптовалют зростатиме по мірі зростання попиту на них.

1.2.1.2 Зрілість ринків

Фондові біржі існують набагато довше, ніж біржі криптовалют, і тому є більш зрілими. Діяльність фондових бірж не тільки регулюються постановами та місцевими законами, а й також отримують державну підтримку [13]. Також, компанії на фондовій біржі повинні забезпечувати прозорість для акціонерів, публікуючи ринкову діяльність, включаючи щоквартальні фінансові оновлення та протоколи загальних зборів.

За рахунок зрілості, фондові біржі мають великі обсяги та різноманітність торгівлі. Проте ця особливість фондового ринку дала широкі можливості деяким трейдерам домінувати в торгових колах за рахунок менших комісійних зборів за торгівлю, що грає у певній ролі бар’єром для менших інвесторів.

З іншого боку, біржі криптовалют все ще молоді та перебувають у стані постійного розвитку. Незважаючи на те, що існують заходи щодо посилення регулювання бірж, щоб підвищити довіру інвесторів – значна частина їхньої діяльності наразі знаходиться поза регуляторною та політичною сферами. Враховуючи їх коротку історію, обсяг і різноманітність криптовалют, якими торгують, також набагато менші, ніж на фондових біржах.

1.2.1.3 Волатильність

Коли мова стосується ринків, волатильність [23] часто викликає особливу увагу та обережність. Насправді волатильність ринку можна розглядати як у позитивному, так і в негативному світлі.

Низька волатильність означає більш стабільний ринок, однак, це також часто означає довше очікування фінансової винагороди. На фондовій біржі низька волатильність можна зустріти досить часто.

Великі обсяги торгівлі підвищують стабільність фондового ринку та роблять його менш схильним до руху через дії трейдерів з великими капіталами. Проте, враховуючи зв'язки з урядами та корпораціями по всьому світу, фондова біржа часто зазнає впливу геополітичних подій.

Порівнюючи з фондовим ринком – ринок криптовалют відчуває більшу волатильність. Ринок є новим, тому значення його максимумів та мінімумів на графіку є дуже вираженими, що робить криптовалютний ринок вразливим до торгових рухів «китових» трейдерів. Китом називають трейдера, який володіє відносно великим об'ємом криптовалюти. Це означає, що весь ринок може бути вразливим до торговельних рішень тих, хто оперує на цьому ринку більшістю коштів. Наприклад, після того, як у січні 2021-го року інфлюенсер Ілон Маск інвестував 1,5 мільярда доларів у біткойн, ціна біткойна раптово підскочила на 17% до нового рекордного рівня.

Однак, оскільки криптовалюти знаходяться окремо від урядів та інших глобальних установ, вони значною мірою ізольовані від політичних впливів.

1.2.1.3 Охоплення ринку

Не дивлячись на зрілість фондової біржі та безліч правил і положень, які склалися навколо неї – процес початку торгівлі може зайняти багато часу та енергії.

По перше, потрібно знайти собі брокера, і коли він у вас буде, вам знадобиться дозвіл на купівлю та продаж. Але окрім цього, торгівля ще й обмежена робочим часом. Тому як можна бачити, доступ до фондового ринку є дуже контрольованим.

На противагу, торгувати криптовалютою можна в будь-який час і в будь-який день, незалежно від державних свят і великих подій. Будь-хто має можливість торгувати на криптовалютній біржі, що робить саму торгівлю набагато доступнішою для людей будь-якого соціального становища.

1.2.1.4 Комісії та регулювання

Цей пункт є найбільш-яскраво вираженою різницею між фондовими та криптовалютними біржами.

Фондові біржі набули того стану, при якому вони являються жорстко регульованими ринками. Існують правила для захисту торговців та інвесторів, щоб допомогти зберегти ігрове поле чесним.

На додаток до правил існують також комісії, а витрати, пов'язані з проходженням обміну на фондовій біржі, є відносно високими. Брокери стягують плату або комісію, банки стягуватимуть з вас плату за здійснення платежів, а приріст капіталу оподатковується.

Торгівля на криптовалютних біржах вимагає відносно менших витрат. Витрати, пов'язані з транзакціями в блокчейні – мізерні і складаються лише з комісії

за майнинг. Таким чином біржі потребують значно менших витрат при купівлі та продажу криптовалюти, ніж брокери для фондових бірж.

Незважаючи на те, що на сьогоднішній день криптовалютні біржі все ще відносно вільні від регулювання – існує підтримка думок за більше регулювання ринку криптовалют, але лише час покаже природу правил і норм, що будуть застосовуються до ринку криптовалют.

1.3 Існуючі методи аналізу і прогнозування часових рядів

Перш за все, для того щоб зрозуміти, які методи аналізу і прогнозування можуть підійти у випадку їх застосування на часовому ряді криптовалют – потрібно визначитись з наявною базою підходів. Звичайно, існують обмеження, коли доводиться мати справу з непередбачуваним і невідомим.

Прогнозування часових рядів не завжди є безпомилковим і також може не підходити для всіх ситуацій. Оскільки, насправді, немає чіткого набору правил щодо того, коли повинні використовуватись ті чи інші методи прогнозування – потрібно знати обмеження та те, які моделі слушно використовувати до якого набору даних.

Не кожна модель підходить до будь-якого набору даних та відповідатиме на всі запитання. Гарне прогнозування працює з чистими даними з мітками часу та може ідентифікувати справжні тенденції та закономірності в історичних даних. Для найбільш достовірних прогнозів, потрібно щоб модель відрізняла випадкові коливання та викиди і відокремлювала справжню інформацію від сезонних коливань. Аналіз часових рядів показує, як дані змінюються з часом, а якісне прогнозування може визначити напрямок, у якому змінюються дані.

1.3.1 Авторегресивна інтегрована ковзна середня (ARIMA)

Прогнозування – це процес передбачення розвитку подій у майбутньому на основі минулих і теперішніх даних. Одним із найвідоміших та найпоширеніших підходів для цього – є модель ARIMA, що є акронімом для AutoRegressive Integrated Moving Average (авторегресивна інтегрована ковзна середня) [24].

У моделі ARIMA є 3 параметри, які використовуються для моделювання основних параметрів часового ряду: сезонність, тенденція та шум. Ці параметри позначають p , d та q відповідно.

p – параметр, пов'язаний з авторегресійним аспектом моделі, яка включає в себе минулі значення. Наприклад, якщо відомі значення часового ряду за останній період включають в себе значення, які тільки зростають, то можна стверджувати, що в наступний момент часу $T+1$ найбільш ймовірно, що значення буде зростати, і навпаки, якщо за останній період часовий ряд мав тенденцію на спад – то і наступне значення найбільш ймовірно буде спадаючим.

d – параметр, що пов'язаний з інтегрованою частиною моделі. Він визначає величину різниць значень, що має часовий ряд. Наприклад, якщо дисперсія відомих значень часового ряду не велика, то можна стверджувати, що наступне значення часового ряду $T+1$ найбільш ймовірно буде знаходитись в діапазоні дисперсії.

q – параметр, пов'язаний із ковзною середньою частиною моделі.

Сама задача при використанні даного методу прогнозування полягає у тому, щоб підібрати такі параметри, які б найкраще описували модель.

1.3.2 Узагальнена авторегресивна умовна гетероскедастичність (GARCH)

Модель узагальненої авторегресійної умовної гетероскедастичності (GARCH) спрямована на моделювання умовної волатильності часового ряду [25].

Нехай r_t буде залежною змінною, яка визначається як прибутковість акцій за час t .

Ми можемо змоделювати цю послідовність як:

$$r_t = \sigma_t \epsilon_t, \quad (1.1)$$

Де σ_t – стандартне відхилення r_t за час t ;

ϵ_t – похибка за час t .

Отож у загальному, процес для моделювання GARCH включає в себе три етапи. Перший полягає в оцінці найкращої моделі авторегресії. Другий – обчислення автокореляції помилки. І третій крок – перевірка значущості.

1.3.3 TBATS

Метод TBATS [26] відомий тим, що здатен моделювати часові ряди з декількома сезонами.

Назва є аббревіатурою ключових характеристик моделі:

- T – Trigonometric seasonality (тригонометрична сезонність)
- B – Box-Cox transformation (перетворення Бокса-Кокса)
- A – ARMA errors (помилки ARMA)
- T – Trend component (трендова складова)
- S – Seasonal component (сезонна складова)

Кожна сезонність моделюється тригонометричним представленням на основі рядів Фур'є. Однією з головних переваг цього підходу є те, що він вимагає лише два початкових стану незалежно від тривалості періоду. Ще однією перевагою є можливість моделювати сезонні ефекти нецілочисельної довжини. Наприклад, враховуючи серію щоденних спостережень, можна змоделювати високосні роки з довжиною сезону 365,25.

1.3.4 LSTM

Однією з найдосконаліших моделей для прогнозування часових рядів є нейронна мережа довготривалої короткочасної пам'яті (LSTM). LSTM – це штучна нейронна мережа, яка використовується в галузі штучного інтелекту та глибокого навчання [27]. На відміну від стандартних нейронних мереж прямого зв'язку, LSTM має зворотні зв'язки. Така рекурентна нейронна [28] мережа може обробляти не лише окремі точки даних, але й цілі послідовності даних (наприклад, мову чи відео). Наприклад, LSTM можна застосувати до таких завдань, як несегментоване розпізнавання пов'язаного рукописного тексту, розпізнавання мови, машинний переклад, керування роботами та відеоігри. LSTM стала найбільш цитованою нейронною мережею 20-го століття.

Мережі LSTM добре підходять для класифікації, обробки та прогнозування на основі даних часових рядів, оскільки між важливими подіями в часовому ряді можуть бути затримки невідомої тривалості. LSTM були розроблені для вирішення проблеми зникнення градієнта, яка може виникнути під час навчання традиційних рекурентних нейронних мереж. Відносна нечутливість до довжини проміжків є перевагою LSTM перед рекурентними нейронними мережами, прихованими моделями Маркова та іншими методами навчання.

1.4 Висновки до розділу

У даному розділі було визначено основні поняття ринку, розглянуті різні типи структур ринку, а також проведене детальне порівняння різних типів ринку: фондового та криптовалютного. Описано поняття біржі і детально оглянуті її основні елементи, поняття і властивості. Проведений поверхневий огляд існуючих методів аналізу і прогнозування часових рядів з метою ознайомлення, визначення їх особливостей і основних компонентів.

РОЗДІЛ 2

ДЕТАЛЬНИЙ ОГЛЯД МОДЕЛЕЙ І ОСНОВНИХ ПІДХОДІВ У МЕТОДАХ АНАЛІЗУ І ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

2.1 Особливості у прогнозуванні часових рядів

Часовий ряд – це сукупність спостережень, отриманих шляхом повторних вимірювань протягом певного часу [29]. Особливістю графіку часового ряду полягає у тому, що однією з осей у ньому завжди буде час.

Через природу даних часових рядів існує ряд особливостей, пов'язаних з їх аналізом і прогнозуванням, які не стосуються інших типів наборів даних.

2.1.1 Зони аналізу та прогнозування часових рядів

Щоб з'ясувати, у чому полягають основні відмінності між підходами у прогнозуванні даних часових рядів від інших даних, розглянемо простий приклад. Припустимо, ми хочемо спрогнозувати продажі морозива у будь-який період часу в залежності від температури навколишнього середовища у певний день. Ми провели певний аналіз даних залежності кількості продажів від температури навколишнього середовища базуючись на інформації за минулі дні і отримали наступний графік (рис. 2.1):

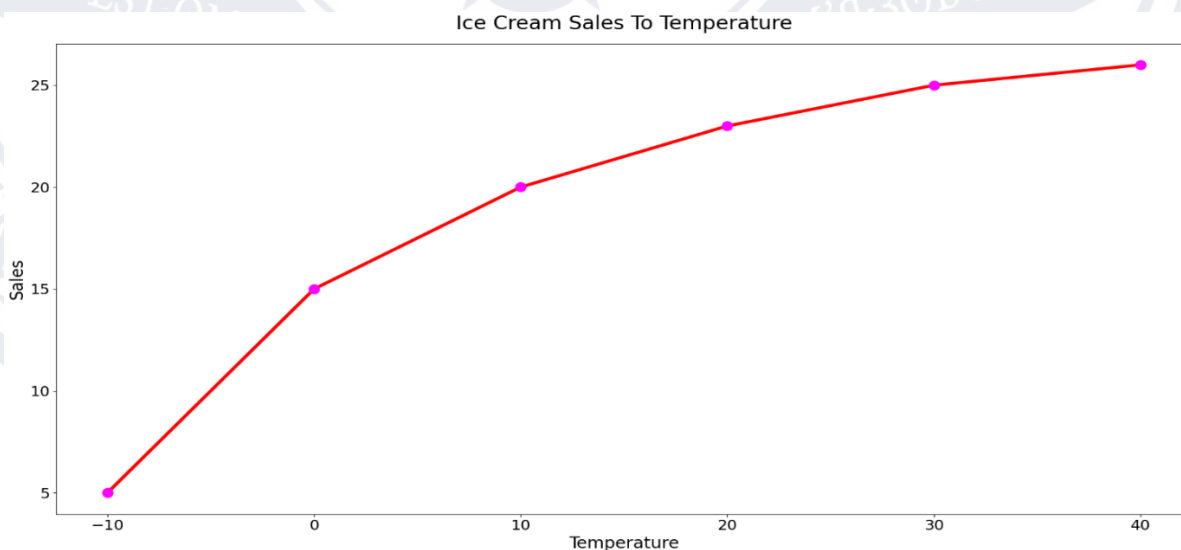


Рисунок 2.1 – Крива залежності кількості продаж морозива до температури навколишнього середовища

Виходячи з графіку (див. рис. 2.1), якщо, наприклад, ми знаємо, що температура на сьогоднішній день буде у діапазоні від 25 до 30 градусів, ми можемо сказати, що приблизна кількість продажів морозива буде складати 24 штуки (рис. 2.2).

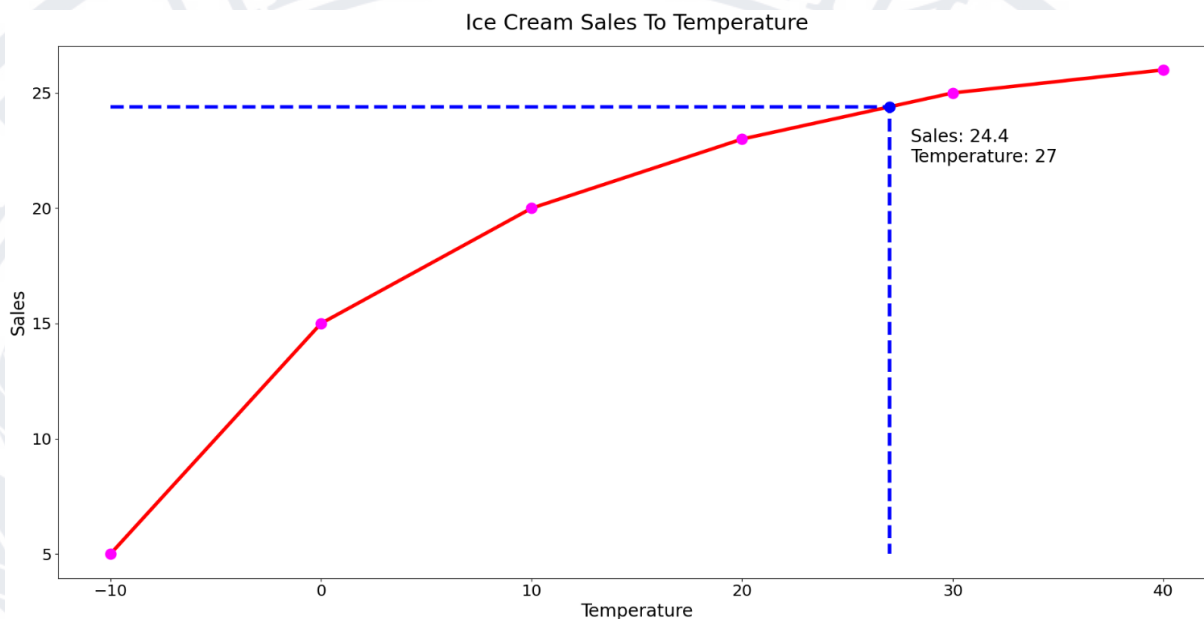


Рисунок 2.2 – Приблизна кількість продажів морозива

Здебільшого, використання даних залежностей такого типу не потребує спеціальних знань чи методологій для того, щоб зробити прогноз у будь який момент часу. Для цього лише необхідно у певний момент часу знати один із параметрів графіку залежності. Тобто якщо ми знаємо, що у певний день ми продали 24 морозива, ми можемо ствердити, що температура у цей день була приблизно 27 градусів. Також ця закономірність працює у зворотному напрямку – знаючи температуру можна визначити кількість продажів.

Розглянемо такий самий приклад з продажами морозива, але тепер представимо продажі у вигляді часового ряду (рис. 2.3).

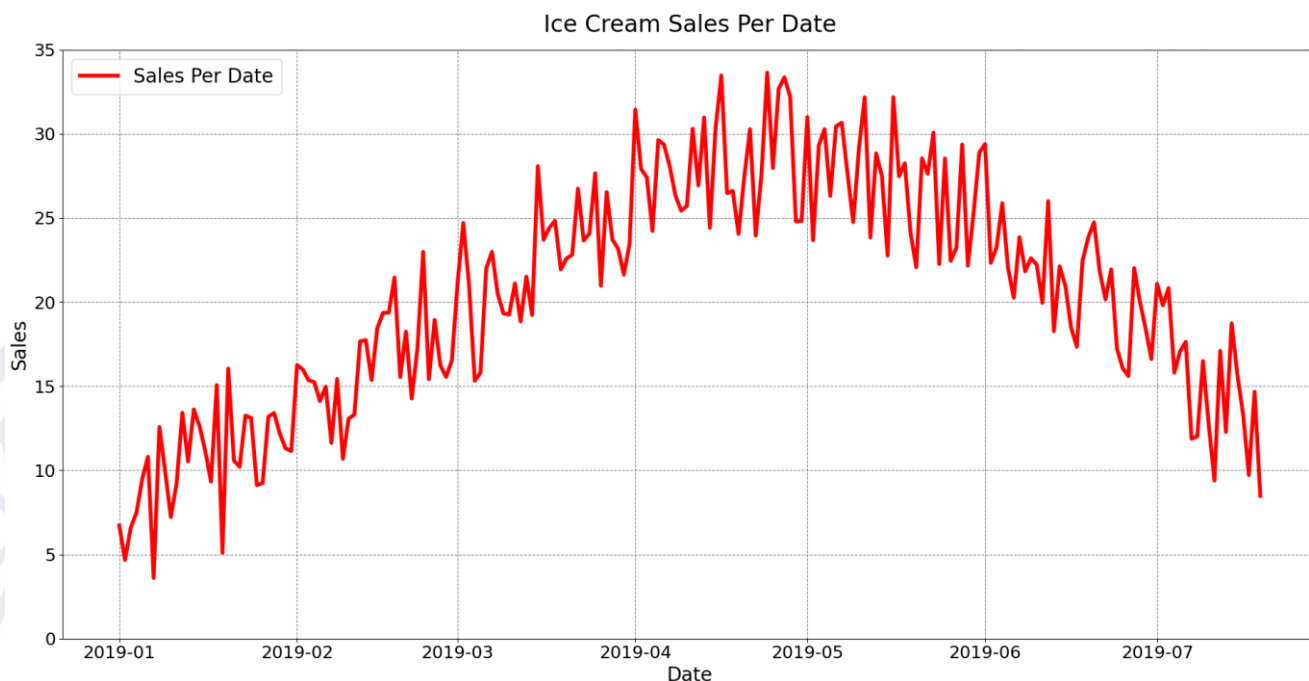


Рисунок 2.3 – Кількість продажів морозива по дням

Аналізуючи графік (див. рис. 2.3), можна побачити, що різниця продажів між сусідніми днями може сягати понад 12 пунктів. Припустимо, ми хочемо зробити прогноз на тиждень вперед (рис 2.4).

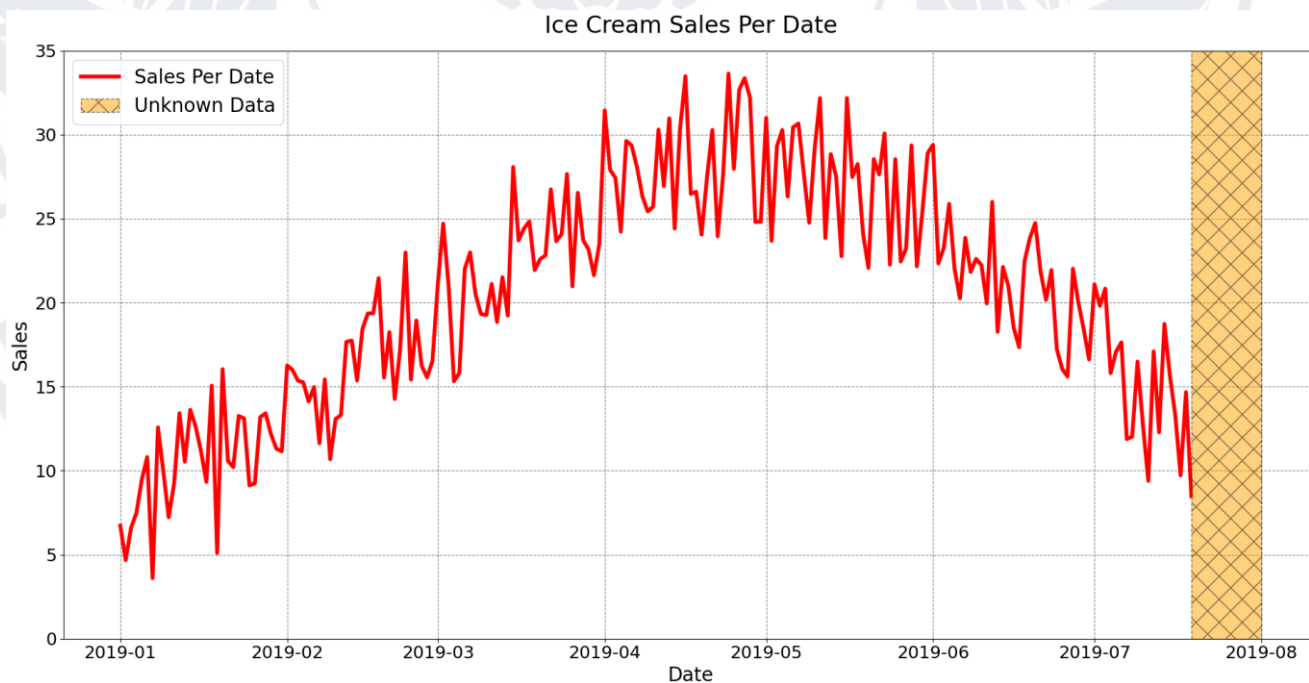


Рисунок 2.4 – Кількість продажів морозива по дням з невідомою зоною за тиждень

Таким чином, задача прогнозу буде полягати в тому, щоб знайти значення які знаходяться поза зоною графіку. І це є ключовою відмінністю між задачами прогнозування часових рядів і прогнозування будь-яких інших наборів даних. Задачі прогнозування часових рядів є задачами екстраполяції [30], коли нам потрібно зробити висновки що до певної частини системи базуючись на отриманих висновках з інших частин цієї ж системи. На відміну від задач інтерполяції [31] – задачі екстраполяції у більшості випадків є більш складними, оскільки зазвичай в таких задачах є дуже багато змінних які можуть дуже сильно впливати на прогнозуєчий результат і чим далі ми намагаємось зробити припущення від відомих даних – тим менш ми можемо бути впевнені, що наш прогноз є точним, чому що стає просто не відомо, чи з правдивими даними йде робота, чи вони ще до цього були передбачені не точно. Також в задачах інтерполяції зазвичай мається великий набір даних, який дозволяє певним чином підтримати припущення, що його результат буде знаходитись саме в конкретному інтервалі, тому знову ж таки, задачі такого типу є більш легкими у вирішенні.

2.1.2 Одновимірні та багатовимірні моделі часових рядів

Ще одна особливість у прогнозуванні часових рядів полягає в тому, що кількість їх змінних може варіюватись і відповідно до кількості можуть застосовуватись ті чи інші моделі. Одновимірні моделі часових рядів [32] – це моделі прогнозування, які використовують лише одну змінну (цільову змінну) та її часові зміни для прогнозування майбутнього. Одновимірні моделі є більш характерні для прогнозування часових рядів.

Але існують певні ситуації, коли у вас можуть бути і додаткові допоміжні дані для прогнозування майбутнього. Наприклад, в деяких ситуаціях доцільно враховувати дані про погоду в прогнозі попиту на продукцію, або дані про вік

цільової аудиторії продукту, або будь-яких інших даних, які є в наявності і які можуть впливати на результат прогнозів. У такому випадку можна використовувати багатовимірні моделі часових рядів [33]. Багатовимірні моделі часових рядів – це ті самі моделі одновимірних часових рядів, але адаптовані для інтеграції зовнішніх змінних. Також для цього завдання можна використовувати і машинне навчання з учителем. На таблиці 2.1 можна бачити коротке порівняння одновимірних та багатовимірних моделей часових рядів.

Таблиця 2.1 – порівняння одновимірних та багатовимірних моделей часових рядів.

Одновимірні моделі часових рядів	Багатовимірні моделі часових рядів
використовують лише одну змінну	використовують декілька змінних
не використовують додаткові зовнішні дані	можуть використовувати зовнішні дані
базуються лише на відношенні між минулими та теперішніми даними	базуються на відношенні між минулими та теперішніми даними, а також на відношенні між змінними

2.1.3 Декомпозиція часових рядів

Декомпозиція часових рядів [34] – це техніка для вилучення ключових властивостей, або так званих компонентів, із набору даних. У даних часового ряду є три важливі компоненти: сезонність, тренд та шум.

Сезонність [35] – це повторюваний рух (зазвичай щорічний), який присутній у часовому ряді. Наприклад, якщо ми розглядаємо набір даних, який представляє себе середню температуру повітря за день, то у такому випадку показники влітку будуть вище показників взимку і така закономірність буде повторюватись кожного

року. Тобто у цьому випадку можна використати щорічну сезонність як основу для прогнозування майбутніх значень.

Тренд [36] – це загальний напрямок, за яким рухається графік. На відміну від сезонності – тренд представляє з себе пряму лінію, яка може рухати загальні показники графіку вгору або вниз. Тренд може бути довгостроковим або короткостроковим. Якщо брати той самий приклад з температурою повітря, то графік скоріш за все буде мати позитивний тренд через те, що відбувається глобальне потепління.

Шум – це компонент змін в часовому ряді, який не може бути пояснений ані сезонністю, ані трендом. Під час створення моделей зазвичай усі компоненти об'єднуються в математичну формулу. Двома компонентами такої формули є сезонність і тренд. Модель, яка використовує лише ці дві компоненти, ніколи не покаже ідеальний прогноз, тому що у будь-якому часовому ряді присутня помилка. Ця помилка і є коефіцієнтом шуму.

Для того щоб краще описати ці поняття, скористуємось даними про вміст вуглекислого газу в повітрі за 1958-2001 роки (рис. 2.5).

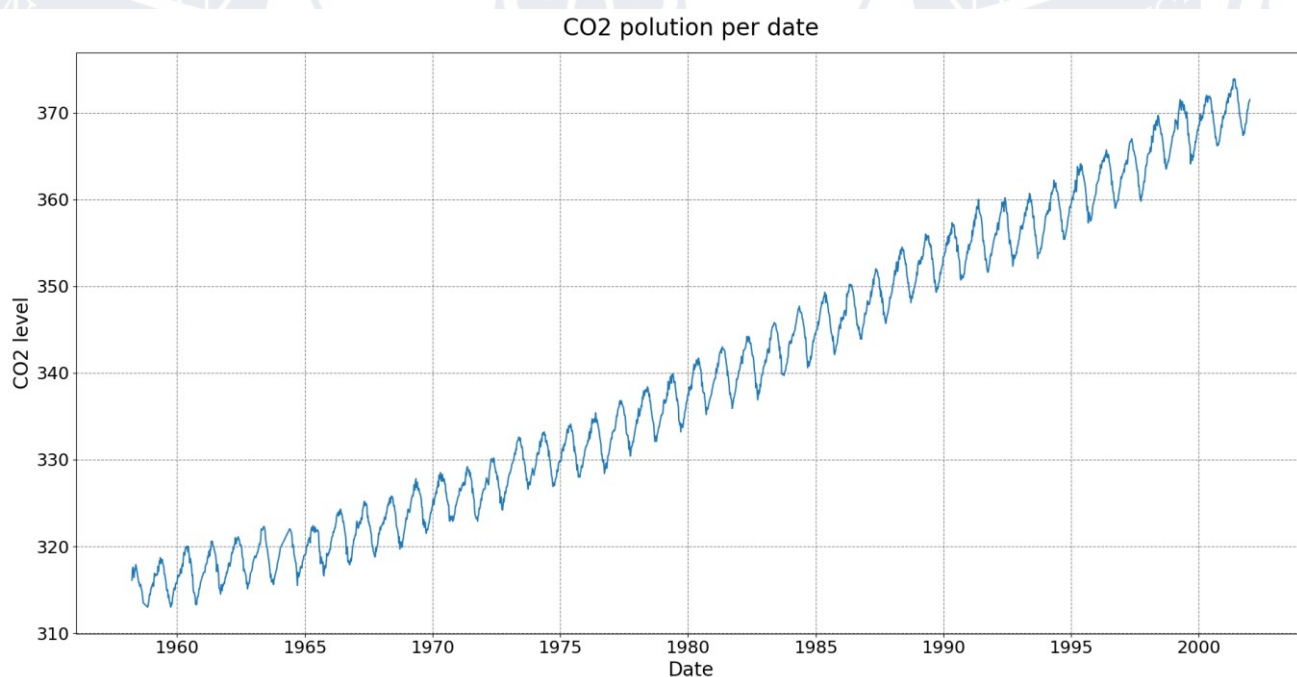


Рисунок 2.5 – Вміст вуглекислого газу в повітрі за 1958-2001 роки

Виконавши декомпозицію графіку (див. рис. 2.5), отримаємо такі результати (рис. 2.6):

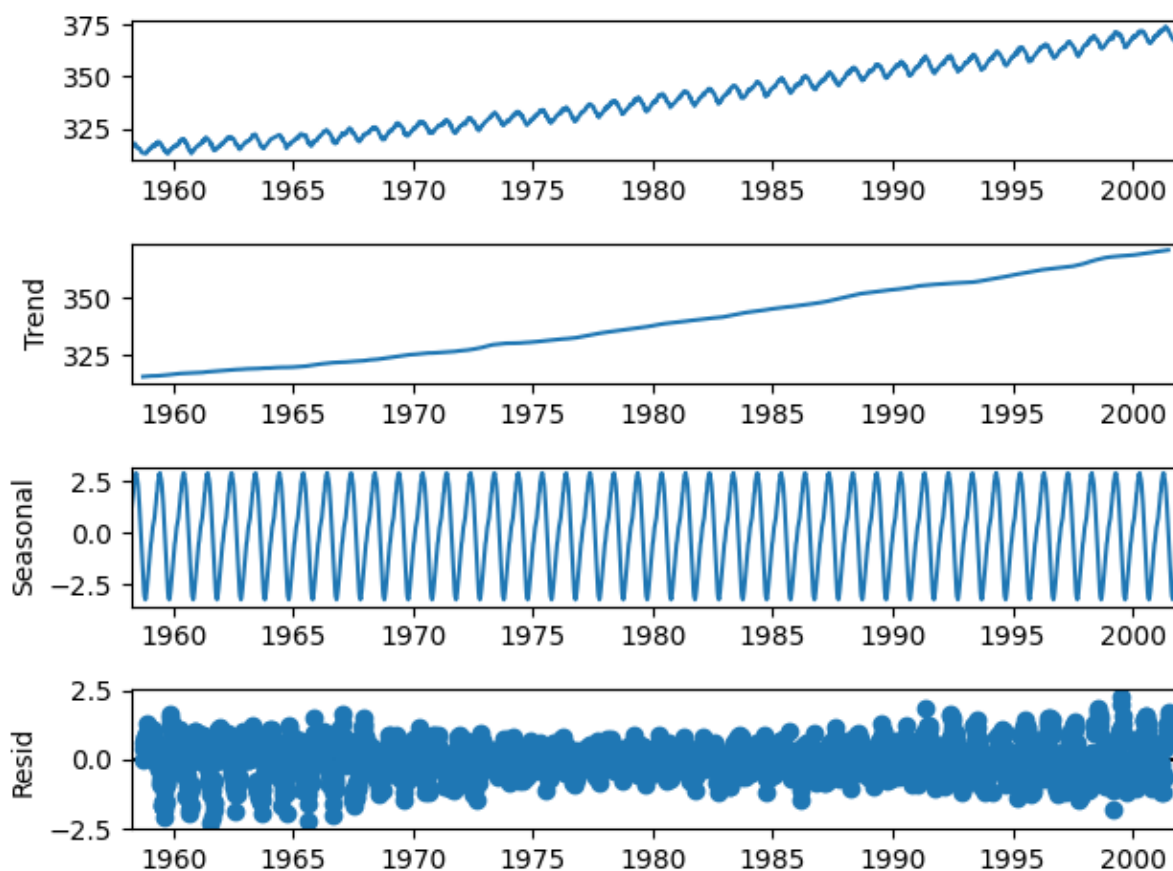


Рисунок 2.6 – Декомпозиція графіку вмісту вуглекислого газу в повітрі за 1958-2001 роки

На рисунку 2.6 зверху – представлений графік вхідних даних, а також:

- Trend – тренд, який має висхідний напрямок.
- Seasonal – сезонність.
- Resid – шум.

Тобто, якщо ми скомбінуємо усі три компоненти разом – ми отримаємо на виході початковий графік

2.1.4 Автокореляція

Ще один тип інформації, який може бути присутній в даних часових рядів – це автокореляція.

Автокореляція [37] – це кореляція, або залежність, між поточним та минулими значеннями часового ряду. У такому випадку це означає, що використовуючи автокореляцію ми можемо також використовувати поточні значення для прогнозування майбутніх значень.

Автокореляція може бути позитивною або негативною:

- Позитивна автокореляція означає, що високе значення зараз, швидше за все, дасть високе значення в майбутньому, і навпаки. Для прикладу можна взяти фондовий ринок: якщо всі купують акції, то ціна зростає. Коли ціна зростає, люди думають, що ця акція вигідна для покупки, і також її купують, тим самим підвищуючи ціну. Але якщо ціна падає, то всі бояться краху, що провокує продаж акцій, і як результат – ціна продовжує, або навіть пришвидшує своє падіння.
- Негативна автокореляція має протилежний характер: високе значення сьогодні дає низьке значення завтра, або низьке значення сьогодні дасть високе значення завтра. Як приклад можна взяти популяцію кроликів у природному середовищі. Якщо влітку певного року диких кроликів буде багато, то за літо вони з'їдять усі наявні природні ресурси. Взимку не буде нічого їсти, тому багато з них загине і популяція кроликів що виживе – буде невеликою. З невеликою популяцією кроликів протягом року, природні ресурси відновляться, що дозволить збільшити популяцію кролів у наступному році.

Існують два відомі графіки, які можуть допомогти у виявленні автокореляції у певному наборі даних: графік ACF та графік PACF.

2.1.4.1 АСФ – автокореляційна функція

Автокореляційна функція визначає те, як дані часового ряду, у певний момент часу, в середньому, залежать від даних у минулому [38]. Іншими словами, вона визначає закономірність у залежності теперішніх даних від даних у минулому через різні інтервали часу.

Щоб краще зрозуміти, що з себе представляє автокореляційна функція, застосуємо її до набору даних про вміст вуглекислого газу в повітрі за 1958-2001 роки (рис. 2.7)

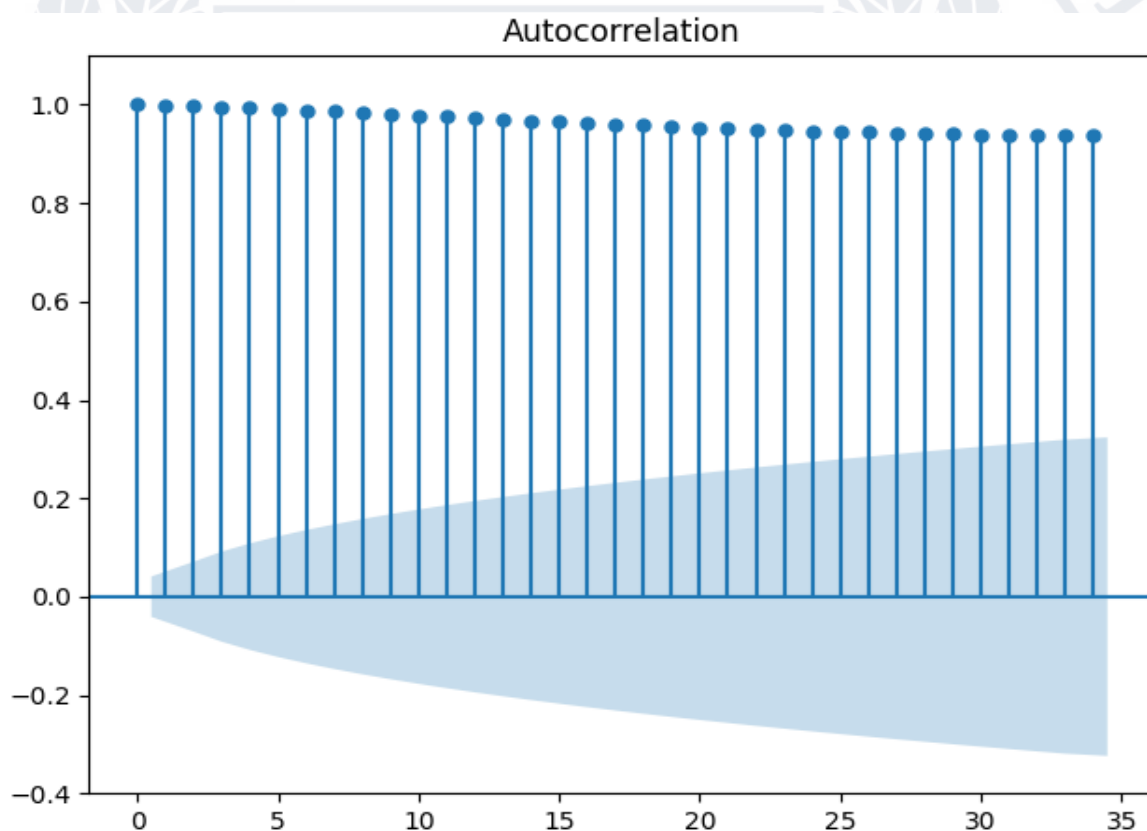


Рисунок 2.7 – Результат застосування автокореляційної функції до набору даних про вміст вуглекислого газу в повітрі

На рисунку 2.7 по осі абсцис зображено кількість кроків назад у часі. Ці кроки також називають кількістю лагів або затримок. На осі ординат можна побачити ступінь кореляції кожного кроку у певний момент часу з поточним кроком. Отож,

як можна бачити з графіку, очевидно, що у даному часовому ряді присутня значна автокореляція між показниками. Найвища кореляція показника теперішнього часу (V_t) є від показника, який знаходиться на одну позначку у часі в минулому (V_{t-1}). Тобто якщо вміст вуглекислого газу вчора був великий, то сьогодні він також буде великий і навпаки - якщо вчора вміст вуглекислого газу був низький, то сьогодні він також буде низький відповідно.

2.1.4.2 ПАСФ – часткова автокореляційна функція

Часткова автокореляційна функція [39] є альтернативою звичайній автокореляційній функції. Ця автокореляція називається частковою через те, що з кожним порівняльним кроком назад у минуле в списку буде відображатись лише пряма кореляція між теперішніми даними та минулими. Цей підхід відрізняється від АСФ, оскільки при визначенні кореляції між показниками у часі АСФ враховує також проміжні значення. Наприклад, якщо ми хочемо знайти кореляцію між показником теперішнього часу (V_t) від показника часу два дні тому (V_{t-2}), то з використанням автокореляційної функції, результат буде також включати проміжне значення показника часу один день тому (V_{t-1}). Натомість, при застосуванні часткової автокореляційної функції, результат буде показувати лише пряму залежність V_t від V_{t-2} .

Наприклад, якщо сьогоднішня вартість така ж сама, як і вчорашня і така ж сама, як і позавчорашня, то автокореляційна функція покаже два сильно корельовані кроки. Натомість часткова автокореляційна функція покаже сильну кореляцію лише від вчорашнього дня, а показник позавчорашній дня буде приближений до нуля.

Застосуємо часткову автокореляційну функцію до набору даних про вміст вуглекислого газу в повітрі за 1958-2001 роки (рис. 2.8).

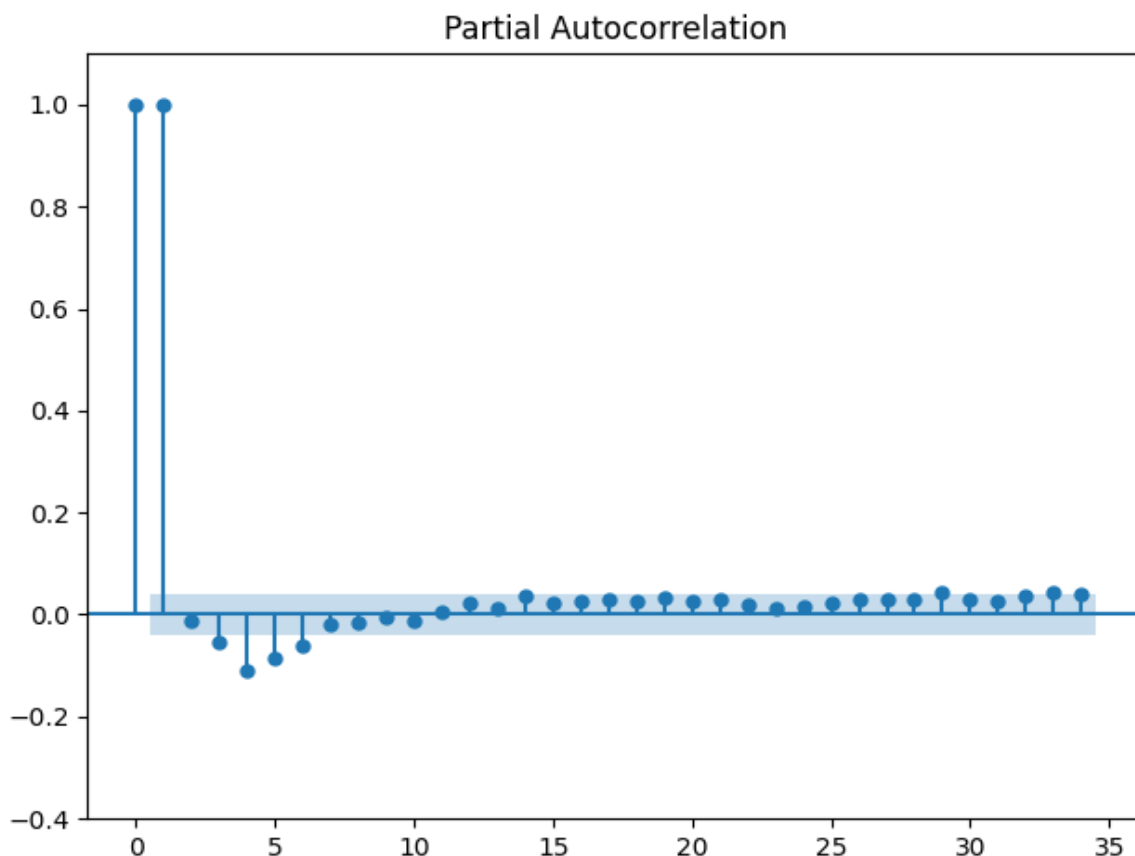


Рисунок 2.8 – Результат застосування часткової автокореляційної функції до набору даних про вміст вуглекислого газу в повітрі

Аналізуючи рисунок 2.8 можна побачити, що графік PACF дає набагато краще представлення кореляції даних CO₂. Існує сильна позитивна автокореляція з лагом один: високе значення вмісту вуглекислого газу вчора означатиме, що швидше за все сьогодні графік також буде мати високе значення і навпаки – низьке значення вчора скоріш за все дасть низьке значення сьогодні. Оскільки автокореляція є частковою, то можна бачити, що відсутні ефекти із попередніми затримками, що робить графік PACF більш правдивим і чіткішим.

2.1.5 Стаціонарність

Ще одною важливою концепцією часових рядів є стаціонарність [40]. Стаціонарний часовий ряд – це часовий ряд, який відповідає таким трьом критеріям:

- Відсутній тренд
- Відсутня сезонність
- Волатильність є постійною величиною

Знаючи те, що ми маємо справу з стаціонарним часовим рядом, перед нами відкривається багато варіантів застосування моделей для прогнозу, наприклад такі як AR або MA моделі, тому що ці моделі були створені з припущенням, що їх будуть намагатись використовувати саме на стаціонарних часових рядах.

Звичайно, визначити те, що часовий ряд є стаціонарним або нестаціонарним можна візуально, але так як твердження що до властивостей часового ряду побудовані на візуальному сприйнятті можуть виявитись хибними та не точними – було придумано тест на одиничний корінь [41] до якого відноситься найбільш популярний розширений тест Дікі-Фулера [42], який може надати математичне підтвердження припущення що до стаціонарності часового ряду.

2.1.5.1 Тест Дікі-Фулера

Перш ніж перейти до розширеного тесту Дікі-Фулера, потрібно зрозуміти, яку проблему вирішує звичайний тест Дікі-Фулера.

Тест Дікі-Фулера – це тест на одиничний корінь [42]. Нульова гіпотеза тесту полягає в тому, що в часовому ряді існує одиничний корінь, що означає, що ряд даних не є стаціонарним. Альтернативна гіпотеза, як правило, говорить про стаціонарність або стаціонарність тренду, але твердження може відрізнитися залежно від варіації тесту, який використовується. Розглянемо наступну формулу, щоб краще зрозуміти тест:

$$y_t = \rho y_{t-1} + \mu_t, \quad (2.1)$$

Де y_t – це поточне змодельоване значення часового ряду;

ρ – вільний коефіцієнт;

y_{t-1} – значення часового ряду на минулому кроці;

μ_t – похибка за час t ;

Рівняння представляє процес першого порядку AR(1) і означає, що показник часового ряду y_t залежить лише від його минулого значення, тобто змінна пояснюється показником часового ряду з затримкою в одиницю. ρ – це коефіцієнт інтересу для визначення одиничного кореня або стаціонарності. Якщо значення ρ менше одиниці, це означає, що великі зміщення у часовому ряді є тимчасовими і згаснуть з плином часу. У цьому випадку ряд буде мати стаціонарний характер. У іншому випадку, якщо значення ρ дорівнює одиниці, то великі зміщення показників ряду мають постійний характер і не зникнуть з плином часу. У такому випадку це буде означати, що у часовому ряді присутній одиничний корінь. І на останок, якщо значення ρ буде більше за одиницю, то часовий ряд з плином часу буде збільшувати свою волатильність.

Проблема з цим рівнянням полягає в тому, що початковим нашим припущенням є те, що часовий ряд стаціонарний. Коли ми хочемо щось довести, то теорія рекомендує брати за відправну точку сумніви, які ми хочемо спростувати. Сумніви мають полягати у тому, що ряд має одиничний корінь, отже ми можемо переписати рівняння враховуючи це і в результаті рівняння матиме такий вигляд:

$$\Delta y_t = (\rho - 1)y_{t-1} + \mu_t = \delta y_{t-1} + \mu_t, \quad (2.2)$$

Де Δy_t – різниця між значеннями поточного кроку t та минулого $t-1$;

ρ – вільний коефіцієнт;

y_{t-1} – значення часового ряду на минулому кроці;

μ_t – похибка за час t ;

δ – вільний коефіцієнт $(p - 1)$.

Тепер початкове припущення полягає в тому, що ряд є нестационарним і тому модель оцінюється по іншому. Тепер тут δ – це коефіцієнт інтересу для визначення одиничного кореня або стаціонарності. Якщо значення δ менше нуля, то зміщення в часовому ряді є тимчасовими що приведе до їх згасання з плином часу і у цьому випадку ряд матиме стаціонарний характер. З іншого боку, якщо значення δ дорівнює нулю, то коливання в y є постійними за своєю природою і не зникають з плином часу. Відповідно, у цьому випадку це буде означати, що у часовому ряді присутній одиничний корінь. Це називається гіпотезою одиничного кореня.

У будь-якому випадку, тест DF розглядається як один із найслабших тестів на наявність одиничного кореня, оскільки він припускає лише те, що наша модель – це модель AR(1). Це є дуже різким і строгим припущенням, оскільки у більшості випадків зміни часових рядів пов'язані з історичними подіями. Отож альтернативним і кращим тестом є розширений тест Дікі-Фуллера (ADF) [43].

2.1.5.2 Розширений тест Дікі-Фуллера

Як згадувалося раніше, нульова гіпотеза тесту Дікі-Фуллера полягає в тому, що одиничний корінь присутній у моделі AR першого порядку. Процес AR першого порядку лише враховує автокореляцію першого порядку. Якщо ми хочемо враховувати автокореляції вищого порядку в моделі, тоді тест Дікі-Фуллера не підходить і буде порушене припущення того, що помилка представлена у вигляді білого шуму. Тому Дікі та Фуллер розширили рівняння затримками вищого порядку, щоб охопити автокореляцію вищого порядку. Якщо бути більш точними, то розширений тест Дікі-Фуллера використовується для параметричного

виправлення кореляції вищого порядку, додаючи у рівняння більшу кількість затримок у праву частину.

$$\Delta y_t = \delta y_{t-1} + \sum_{i=1}^p \beta_i \Delta y_{t-i} + \mu_t, \quad (2.3)$$

Де Δy_t – різниця між значеннями поточного кроку t та минулого $t-1$;

δ – вільний коефіцієнт $(p - 1)$;

y_{t-1} – значення часового ряду на минулому кроці;

p – глибина процесу або кількість затримок;

β_1, \dots, β_p – коефіцієнт ваги значень кроку Δy_{t-i} ;

$\Delta y_{t-1}, \dots, \Delta y_{t-p}$ – сума значень різниць часового ряду;

μ_t – похибка за час t .

Включення кількості затримок у наведене вище рівняння дозволяє врахувати автокореляцію вищого порядку. Загалом прикладні економетристи радять, що слід включити достатню кількість затримок, щоб автокореляція в залишках не мала суттєвого ефекту на результат. Однак іноді це не особливо корисно через втрату ступеня свободи.

Застосуємо розширений тест Дікі-Фулера до набору даних про вміст вуглекислого газу в повітрі за 1958-2001 роки (рис. 2.9).

Statistic: 0.03378459745825786
P-value: 0.9612384528286101

Рисунок 2.9 – Результат застосування розширеного тесту Дікі-Фулера до набору даних про вміст вуглекислого газу в повітрі

Як можна бачити на рисунку 2.9, другим значенням є p -значення. Якщо це значення p було би менше 0.05, то можна б було відхилити нульову гіпотезу про те,

що часовий ряд має одиничний корінь і прийняти альтернативну гіпотезу про те, що часовий ряд є стаціонарним. У даному випадку ми не можемо відхилити нульову гіпотезу і припускаємо, що часовий ряд має одиничний корінь. Оскільки ми візуально вже бачили графічне представлення даних (див. рис. 2.5), то ми знаємо, що у графіку присутня сезонність і тренд, що порушує перше і друге правило стаціонарності, отож тест підтверджує результат візуального дослідження, що часовий ряд не є стаціонарним.

2.1.5.3 Метод різниць

Метод різниць дозволяє видалити тренд з часового ряду, що дозволяє зробити ряд стаціонарним, якщо в ньому присутня лише трендова компонента [44].

Метод різниць виконується шляхом віднімання попереднього показника часового ряду від поточного, і так для кожного показника ряду.

$$D_t = y_t - y_{t-1}, \quad (2.4)$$

Де D_t – показник різниці між поточним і минулим значеннями часового ряду;

y_t – поточне значення часового ряду;

y_{t-1} – попереднє значення часового ряду.

Зазвичай після виконання прогнозу, необхідне інвертування процесу, коли прогноз потрібно перетворити назад у вихідний масштаб. Повернути дані у вихідний стан можна через додавання до показника часового ряду на попередньому кроці – значення різниці.

$$y_t = y_{t-1} + D_t, \quad (2.5)$$

Де y_t – поточне значення часового ряду;

y_{t-1} – попереднє значення часового ряду;

D_t – показник різниці між поточним і минулим значеннями часового ряду.

Для прикладу застосуємо метод різниць до набору даних про вміст вуглекислого газу в повітрі за 1958-2001 роки (рис. 2.10).

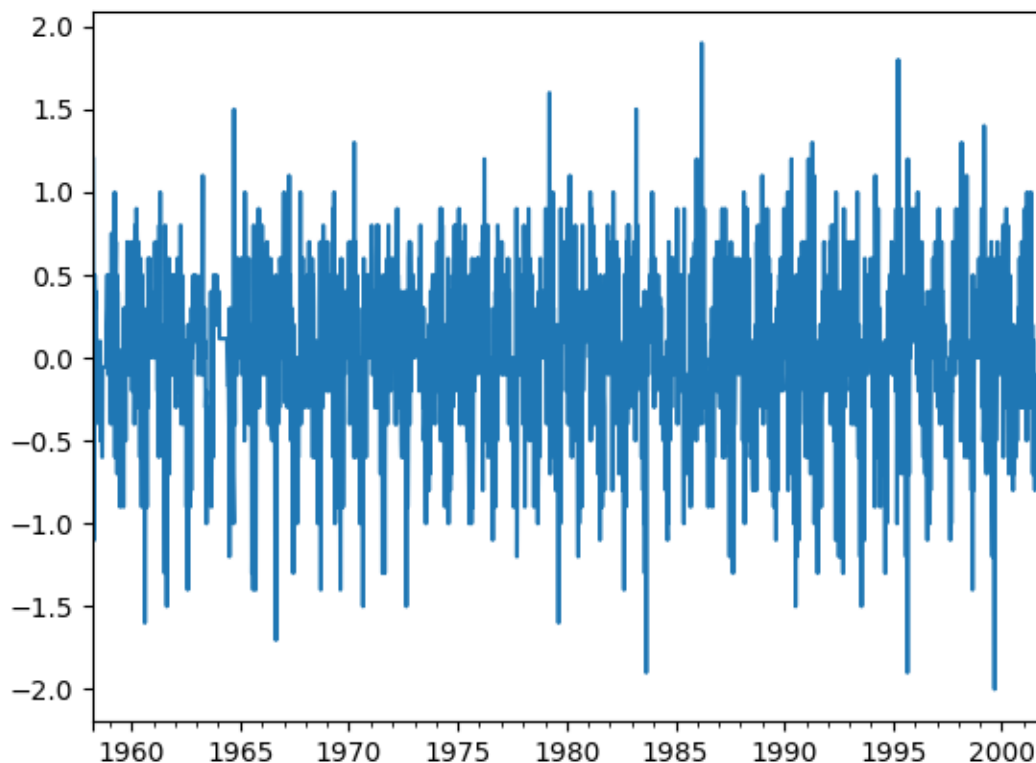


Рисунок 2.10 – Результат застосування методу диференціювання до набору даних про вміст вуглекислого газу в повітрі

Аналізуючи рисунок 2.10 можна візуально бачити, що ряд виглядає стаціонарним, оскільки в ньому немає сезонності, тренду та волатильність є постійною величиною.

Спробуємо знову застосувати розширений тест Дікі-Фуллера, але вже до часового ряду отриманого в результаті застосування до нього методу різниць (рис. 2.11)

Statistic: -15.727522408375838
P-value: 1.3013480157810615e-28

Рисунок 2.11 – Результат застосування розширеного тесту Дікі-Фулера до набору даних отриманого в результаті застосування методу різниць

Як можна бачити з результатів (див. рис 2.11), значення p є дуже малим і попадає під альтернативну гіпотезу Дікі-Фулера, коли це значення менше за 0.05, що дозволяє нам відхилити нульову гіпотезу про те, що у часовому ряді присутній одиничний корінь. Отже тест підтвердив результат візуального дослідження, що часовий ряд є стаціонарним.

2.1.6 Однокрокові та багатокрокові моделі прогнозування часових рядів

Остання концепція, яку важливо зрозуміти, перш ніж переходити до огляду методів прогнозування часових рядів – це концепція однокрокових моделей та багатокрокових моделей.

Деякі моделі чудово працюють лише для прогнозування наступного кроку часового ряду, але не мають можливості прогнозувати на декілька кроків вперед. Ці моделі називають однокроковими. Звісно, можна створювати багатокрокові передбачення з однокроковими моделями, додаючи на кожному наступному кроці результат прогнозу минулого кроку, але існує ризик при використанні такого підходу. Якщо використовувати значення, які були прогнозовані до цього – для наступних прогнозів, то помилка при такому підході може зростати дуже швидко і буде збільшуватись з кожним наступним кроком. Тому для довготривалих прогнозів рекомендується використовувати моделі, які передбачають їх використання для таких випадків.

Багатокрокові моделі – це моделі, які мають здатність робити прогнози на декілька кроків уперед. Зазвичай вони є кращим вибором для довгострокових

прогнозів, але і також підтримують їхнє застосування для прогнозування на один крок. Отож дуже важливо визначилися з кількістю кроків, на які розраховується передбачити рух часового ряду, перш ніж почати будувати та використовувати моделі. Кількість кроків повністю залежить від випадку використання.

2.2 Детальний огляд базових методів прогнозування часових рядів

2.2.3 Сімейство методів прогнозування ARIMA

Сімейство моделей ARIMA – це набір менших моделей, які можна комбінувати. Кожну частину моделі ARIMA можна використовувати як окремий компонент або поєднувати їх у різні комбінації з блоків. Коли всі окремі компоненти зібрані разом, в результаті формується модель SARIMAX. Розглянемо кожен з блоків окремо.

2.2.3.1 Авторегресія (AR)

Авторегресія [45] є першим блоком сімейства ARIMA. Можна розглядати модель AR як регресійну модель, яка пояснює майбутнє значення змінної, використовуючи лінійну комбінацію її минулих значень (запізнь). Термін авторегресія вказує на те, що це регресія змінної відносно самої себе. Таким чином, авторегресійну модель порядку p можна записати як:

$$y_t = c + \sum_{i=1}^p \phi_i y_{t-i} + \varepsilon_t, \quad (2.6)$$

Де c – постійна величина (зміщення значень відносно осі ординат);

p – порядок моделі (кількість враховуючих зміщень);

ϕ_1, \dots, ϕ_p – коефіцієнти моделі для значень часового ряду;

y_{t-1}, \dots, y_{t-p} – значення часового ряду на попередніх кроках;

ε_t – білий шум.

Порядок моделі AR позначається як p , і він представляє кількість значень із запізненням, які потрібно включити в модель. Найпростішою моделлю є модель AR(1): вона використовує лише значення попереднього кроку часового ряду для прогнозування поточного значення. Максимальна кількість значень, які можна використовувати для навчання – дорівнює загальній довжині часового ряду, але у більшості випадків такий крок призводить до занадто складної моделі і перенавчання.

2.2.3.2 Ковзне середнє (MA)

Модель ковзного середнього [46] є другим блоком моделі ARIMA. Логіка її роботи дещо відрізняється від моделі AR: замість того, щоб використовувати в регресії минулі значення змінної прогнозу, модель ковзного середнього в моделі використовує попередні помилки прогнозу. Модель ковзного середнього порядку q можна записати як:

$$y_t = c + \sum_{i=1}^q \theta_i \varepsilon_{t-i} + \varepsilon_t, \quad (2.7)$$

Де c – постійна величина (зміщення значень відносно осі ординат);

q – порядок моделі (кількість враховуючих зміщень);

$\theta_1, \dots, \theta_q$ – коефіцієнти моделі для значень часового ряду;

$\varepsilon_{t-1}, \dots, \varepsilon_{t-q}$ – помилки передбачень;

ε_t – білий шум.

Може здатись, що між помилками і передбаченням майбутнього немає ніякого зв'язку, але за цим є логіка. Коли модель має якісь невідомі, але регулярні зовнішні чинники впливу на передбачення: наприклад, коли модель має сезонність або іншу закономірність у похибці, то модель середнього ковзного дозволяє визначити цей шаблон, навіть не визначаючи його походження. Модель MA може

використовуватись як з одним, так і з кількома кроками назад у часі. Це представлено в параметрі порядку під назвою q . Наприклад, модель MA(1) має перший порядок і враховує лише один крок назад.

2.2.3.3 Авторегресивне ковзне середнє (ARMA)

Модель авторегресивного ковзного середнього, або ARMA [47], поєднує два попередні блоки AR та MA – в одну модель. Тому ARMA може використовувати як минулі значення, так і минулі помилки прогнозу. Повну ARMA(p, q) модель можна записати як:

$$y_t = c + \sum_{i=1}^p \phi_i y_{t-i} + \sum_{i=1}^q \theta_i \varepsilon_{t-i} + \varepsilon_t, \quad (2.8)$$

Де c – постійна величина (зміщення значень відносно осі ординат);

p – порядок моделі AR;

ϕ_1, \dots, ϕ_p – коефіцієнти моделі AR для значень часового ряду;

y_{t-1}, \dots, y_{t-p} – значення часового ряду на попередніх кроках;

q – порядок моделі MA;

$\theta_1, \dots, \theta_q$ – коефіцієнти моделі MA для значень часового ряду;

$\varepsilon_{t-1}, \dots, \varepsilon_{t-q}$ – помилки передбачень;

ε_t – білий шум.

ARMA може мати різні значення для врахування кількості запізнень p для AR та q для MA. Наприклад, модель ARMA(1, 0) має перший порядок для AR ($p = 1$) і нульовий порядок для MA ($q = 0$). Тобто насправді, ми можемо сприймати цю модель так само, як модель AR(1). Таке саме правило працює для MA(1), якщо порядок p для AR буде нульовим. Можливі також й інші комбінації, наприклад: ARMA(3, 1), де AR має третій порядок і MA має перший порядок.

2.2.3.4 Авторегресивне інтегроване ковзне середнє (ARIMA)

Усі вище згадані моделі мають один недолік: для їх застосування рекомендується використовувати стаціонарний часовий ряд. Стаціонарність означає, що часовий ряд відповідає трьом вимогам:

- Відсутній тренд
- Відсутня сезонність
- Волатильність є постійною величиною

Для того, щоб переконатись, чи є часовий ряд стаціонарним, можна використати розширений тест Дікі-Фуллера, і якщо це не так – застосувати метод різниць.

Отож модель ARIMA додає автоматичне застосування методу різниць до моделі ARMA. Вона має додатковий параметр, який визначає те, скільки разів метод різниць буде застосований до часового ряду. Модель ARIMA(p, d, q) можна записати як:

$$y'_t = c + \sum_{i=1}^p \phi_i y'_{t-i} + \sum_{i=1}^q \theta_i \varepsilon_{t-i} + \varepsilon_t, \quad (2.9)$$

Де c – постійна величина (зміщення значень відносно осі ординат);

p – порядок моделі AR;

ϕ_1, \dots, ϕ_p – коефіцієнти моделі AR для значень часового ряду;

$y'_{t-i}, \dots, y'_{t-p}$ – значення часового ряду на попередніх кроках з застосованим методом різниць d разів;

q – порядок моделі MA;

$\theta_1, \dots, \theta_q$ – коефіцієнти моделі MA для значень часового ряду;

$\varepsilon_{t-i}, \dots, \varepsilon_{t-q}$ – помилки передбачень;

ε_t – білий шум.

Наприклад, модель ARMA(1, 1), до якої потрібно застосувати метод різниць один раз, матиме наступне позначення – ARIMA(1, 1, 1), де:

- перша одиниця – порядок p для AR
- друга одиниця – кількість застосування методу різниць d
- третя одиниця – порядок q для MA.

Результуюча модель при використанні ARIMA(1, 0, 1) буде такою самою, як ARMA(1, 1).

2.2.3.5 Сезонне авторегресивне інтегроване ковзне середнє (SARIMA)

SARIMA [48] додає сезонні ефекти в модель ARIMA. Якщо у часовому ряді присутня сезонність, дуже важливо врахувати її у своєму прогнозі.

Нотація SARIMA є трохи складнішою, ніж ARIMA, оскільки кожен із компонентів тепер отримує сезонний параметр на додаток до звичайного параметра.

Наприклад, якщо до цього ми мали три параметра в моделі ARIMA: p , d та q , то нотація SARIMA буде виглядати наступним чином: SARIMA(p , d , q)(P , D , Q) m .

Модель SARIMA(p , d , q)(P , D , Q) m можна записати як:

$$y'_t = c + \sum_{i=1}^p \phi_i y'_{t-i} + \sum_{i=1}^q \theta_i \varepsilon_{t-i} + \sum_{i=1}^P \Phi_i y''_{t-mi} + \sum_{i=1}^Q \vartheta_i \varepsilon_{t-mi} + \varepsilon_t, \quad (2.10)$$

Де c – постійна величина (зміщення значень відносно осі ординат);

p – порядок моделі AR;

ϕ_1, \dots, ϕ_p – коефіцієнти моделі AR для значень часового ряду;

$y'_{t-i}, \dots, y'_{t-p}$ – значення часового ряду на попередніх кроках з застосованим методом різниць d разів;

q – порядок моделі MA;

$\theta_1, \dots, \theta_q$ – коефіцієнти моделі MA для значень часового ряду;

$\varepsilon_{t-1}, \dots, \varepsilon_{t-q}$ – помилки передбачень;

P – порядок сезонного компоненту моделі AR;

Φ_1, \dots, Φ_P – сезонні коефіцієнти моделі AR для значень часового ряду;

m – кількість спостережень за рік

$y''_{t-m}, \dots, y''_{t-mP}$ – значення часового ряду на попередніх кроках з застосованим методом різниць D разів;

Q – порядок сезонного компоненту моделі MA;

$\vartheta_1, \dots, \vartheta_Q$ – сезонні коефіцієнти моделі MA для значень часового ряду;

$\varepsilon_{t-1}, \dots, \varepsilon_{t-mQ}$ – сезонні помилки передбачень;

ε_t – білий шум.

Показник m задається відповідно кількості спостережень за рік. Наприклад якщо дані мають місячний сезонний характер, то показник m буде рівний 12; якщо дані мають сезонний квартальний характер, то показник m буде дорівнювати 4 і т.д. Як можна бачити з формули, маленькі літери (p, d, q) відносяться до несезонних параметрів, а великі літери (P, D, Q) відповідно – до сезонних.

2.2.3.6 Сезонне авторегресивне інтегроване ковзне середнє з екзогенними регресорами (SARIMAX)

Найскладнішою варіацією сімейства методів прогнозування ARIMA – є модель SARIMAX. Вона включає в себе усі вище згадані моделі і компоненти, такі як

- AR – авторегресія
- MA – ковзне середнє
- Метод різниць
- Сезонність

Але окрім того, вона дозволяє ще врахувати екзогенні змінні, або іншими словами, врахувати зовнішні чинники прогнозу. Деякі реальні приклади екзогенних змінних включають ціну на золото, ціну на нафту, зовнішню температуру, обмінний курс і т. д.

З огляду на це, якщо ми включимо зовнішні дані у прогноз, то модель реагуватиме на їх вплив набагато швидше, ніж якщо ми будемо покладатись лише на вплив запізнілих термінів.

2.2.4 Сімейство методів прогнозування GARCH

Розглянемо сімейство методів прогнозування ARCH та GARCH.

GARCH є акронімом для чотирьох термінів:

- G – узагальнена
- AR – авторегресивна
- C – умовна
- H – гетероскадестичність

Гетероскадестичність часто називають ще волатильністю, саме тому ці моделі також називаються моделями волатильності. У більшості випадків дані моделі використовуються у фінансовій галузі, оскільки ціни на багато активів є умовними гетероскедастиками.

2.2.4.1 Авторегресивна умовна гетероскедастичність (ARCH)

Модель ARCH [49] дуже схожа на ARIMA, але її компонент CH моделює квадрати квадратичних залишків часового ряду до кожного попереднього моменту часу. Для моделі ARCH властиво використовувати термін «волатильність» для позначення дисперсії або квадратичних залишків. Модель CH надає можливість

передбачити майбутню волатильність і виступає в якості окремого блоку в моделі ARCH. Можна записати процес ARCH(q), як:

$$\sigma_t^2 = \omega + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2 \quad (2.11)$$

Де ω – білий шум;

q – кількість враховуючих зміщень;

$\alpha_1, \dots, \alpha_q$ – коефіцієнти помилок;

$\varepsilon_{t-1}^2, \dots, \varepsilon_{t-q}^2$ – квадратичні значення помилок.

При застосуванні ARCH дуже важливо враховувати те, щоб часові ряди не мали трендів чи сезонних ефектів, тобто такі, які не мають очевидної вторинної кореляції. Як правило, ARCH можна застосовувати тоді, коли до часового ряду не підходять моделі ARIMA та SARIMA.

2.2.4.1 Узагальнена авторегресивна умовна гетероскедастичність (GARCH)

Узагальнена авторегресивна умовна гетероскедастичність – це статистична модель, яка використовується для аналізу даних часових рядів, де помилка дисперсії вважається послідовною автокореляцією (тобто коли показник волатильності у часовому ряді може відхилятися на доволі тривалий час) [49]. Модель GARCH допускає, що дисперсія члена помилки відповідає процесу авторегресії ковзного середнього.

Модель GARCH (p, q) записується як:

$$\sigma_t^2 = \omega + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 \quad (2.12)$$

Де ω – білий шум;

q – кількість враховуючих зміщень;

$\alpha_i, \dots, \alpha_q$ – коефіцієнти помилок;

$\varepsilon_{t-i}^2, \dots, \varepsilon_{t-q}^2$ – квадратичні значення помилок;

p – кількість враховуючих минулих значень волатильності;

β_i, \dots, β_p – коефіцієнти значень волатильності;

$\sigma_{t-i}^2, \dots, \sigma_{t-p}^2$ – квадратичні значення волатильності на минулих кроках.

Параметри конфігурацій для моделей ARCH та GARCH найкраще обирати базуючись на результатах графіків ACF і PACF волатильності часового ряду.

Цього можна досягти шляхом віднімання середнього арифметичного від кожного показника в ряді та зведення результату в квадрат (або просто піднести спостереження у квадрат, якщо ви вже працюєте із показниками волатильності з іншої моделі)

Потім графіки ACF і PACF можна використовувати для оцінки значень для p і q , подібно до того, як це робиться для моделі ARMA.

2.2.4 Модель TBATS

Модель була створена в 2011 році як рішення для прогнозування часових рядів з кількома сезонними періодами. Оскільки ця модель відносно нова і ускладнена – вона є менш поширеною і не так часто використовується, як моделі сімейства ARIMA.

Акронім TBATS [50] розкладається на компоненти наступним чином:

1. T – тригонометрична сезонність;
2. B – модель простору станів експоненційного згладжування з перетворенням Бокса-Кокса;
3. A – помилки ARMA;
4. T – трендова складова;
5. S – сезонна складова.

Розберемо значення кожного компоненту у моделі:

- Тригонометрична сезонність дозволяє представляти кожен сезонний період як тригонометричне представлення на основі рядів Фур'є, що дозволяє використовувати модель з великими сезонними періодами, а також задавати сезонним періодам не тільки цілочисельні значення – а й з плаваючою точкою;
- Моделювання простору станів – це структура, у якій часовий ряд розглядається як набір спостережуваних даних, на які впливає набір факторів поза межами спостереження. Отож модель простору станів виражає зв'язок між двома наборами. Знову ж таки, хоч в назві і присутнє слово «модель», її слід розглядати як структуру, оскільки модель ARMA теж можна виразити як модель простору станів;
- Експоненціальне згладжування – це сімейство методів прогнозування. Загальна ідея, яка лежить в основі цих методів прогнозування, полягає в тому, що майбутні значення є середньозваженими показниками значень з минулого, де ваги зменшуються експоненційно, що далі ми повертаємося назад у часі. Методи прогнозування включають просте експоненціальне згладжування, подвійне експоненціальне згладжування, метод Холта (для часових рядів з трендом) і потрійне експоненціальне згладжування або метод Холта-Вінтера (для часових рядів з трендом і сезонністю);

- Перетворення Бокса-Кокса є степеневими перетвореннями, які допомагають зробити ряд стаціонарним шляхом стабілізації дисперсії та середнього значення з часом;
- Помилки ARMA – це процес, у якому ми застосовуємо модель ARMA до залишків часового ряду, щоб знайти будь-які зв'язки, які не можна пояснити. Зазвичай залишки моделі є абсолютно випадковими, за винятком випадків, коли якась інформація не була зафіксована моделлю;
- Тренд – це компонент часового ряду, який пояснює довгострокову зміну середнього значення ряду. Коли ми маємо позитивну динаміку, то наша серія з часом збільшується. При негативному тренді ряд з часом зменшується;
- Сезонна складова пояснює періодичну зміну ряду.

Підсумовуючи, TBATS є розширенням методів експоненціального згладжування, який поєднує перетворення Бокса-Кокса для обробки нелінійних даних, використовує тригонометричну сезонність, що дозволяє задавати сезонним періодам значення з плаваючою точкою і використовує модель ARMA для фіксації автокореляції в залишках.

2.2.5 Довга короткочасна пам'ять (LSTM)

LSTM – це рекурентні нейронні мережі спеціального типу, які здатні проводити навчання на послідовних залежностях і вирішувати проблеми прогнозування цих послідовностей [51]. Вони можуть бути корисними у різних випадках, починаючи з прогнозування часових рядів (які є послідовностями значень у часі), а також тексту (речення – це послідовності слів) і мовлення.

Нейронні мережі – це дуже складні моделі машинного навчання, які приймають на вхід якийсь набір даних, пропускають їх через мережу зв'язків і

повертають на виході результат [52]. Кожен вузол у мережі проводить дуже просту операцію, але оскільки нейронна мережа складається з багатьох таких вузлів – загальна мережа має дуже складну структуру. Саме тому нейронні мережі можуть використовуватись на дуже складних та нелінійних наборах даних.

Дану нейронну мережу називають як довга короткочасна пам'ять, оскільки програмна реалізація мережі використовує структуру, засновану на процесах короткочасної пам'яті, для створення довгострокової пам'яті. Загалом, LSTM є загальноприйнятою концепцією в новаторських рекурентних нейронних мережах.

LSTM використовується для даних часових рядів таким чином, що послідовність спостережень ділиться на декілька тренувальних наборів даних, з яких потім LSTM проводить навчання. Зазвичай, послідовність поділяється на чотири кроки, де з них: три використовуються як вхідні дані, а один часовий крок – використовується як вихідний для однокрокового передбачення.

Найважливішими механізмами в LSTM моделі є:

- Механізм забування: дозволяє забувати всю інформацію про сцену, яка не є ключовою та не впливає на передбачення.
- Механізм збереження: дозволяє зберігати інформацію, яка є важливою та може допомогти в прогнозах у майбутньому.

Оскільки LSTM працює як з довгостроковою пам'яттю (LTM), так і з короткочасною пам'яттю (STM). Для того, щоб спростити обчислення зробити їх та ефективними, модель використовує концепцію так званих вентилів:

1. Забувальний вентиль: при надходженні даних у модель, вони разом з довгостроковою пам'яттю пропускаються через сигмоїдну функцію і на виході вентиля повертається результат, який говорить про те яка інформація з вхідних даних є корисною, а яка – ні.

2. Вхідний клапан: вектор даних і короткотривала пам'ять поєднуються разом, для того, щоб інформація, яка нещодавно була збережена в короткотривалій пам'яті, була застосована до поточних даних.
3. Запам'ятовувальний клапан: інформація у довготривалій пам'яті, яку ми не забули, короткочасна пам'ять і дані об'єднуються разом у запам'ятовувальному клапані, який в результаті повертає нове значення для довготривалої пам'яті.
4. Вихідний клапан: цей клапан пропускає через себе довготривалу пам'ять, короткочасну пам'ять і дані для прогнозування результату поточної події для того, щоб оновити короткотривалу пам'ять.

Наведений нижче малюнок показує математичну архітектуру LSTM (рис. 2.12), де жовтим кольором позначений забувальний клапан; блакитним – вхідний клапан; червоним – запам'ятовувальний клапан; фіолетовим – вихідний клапан:

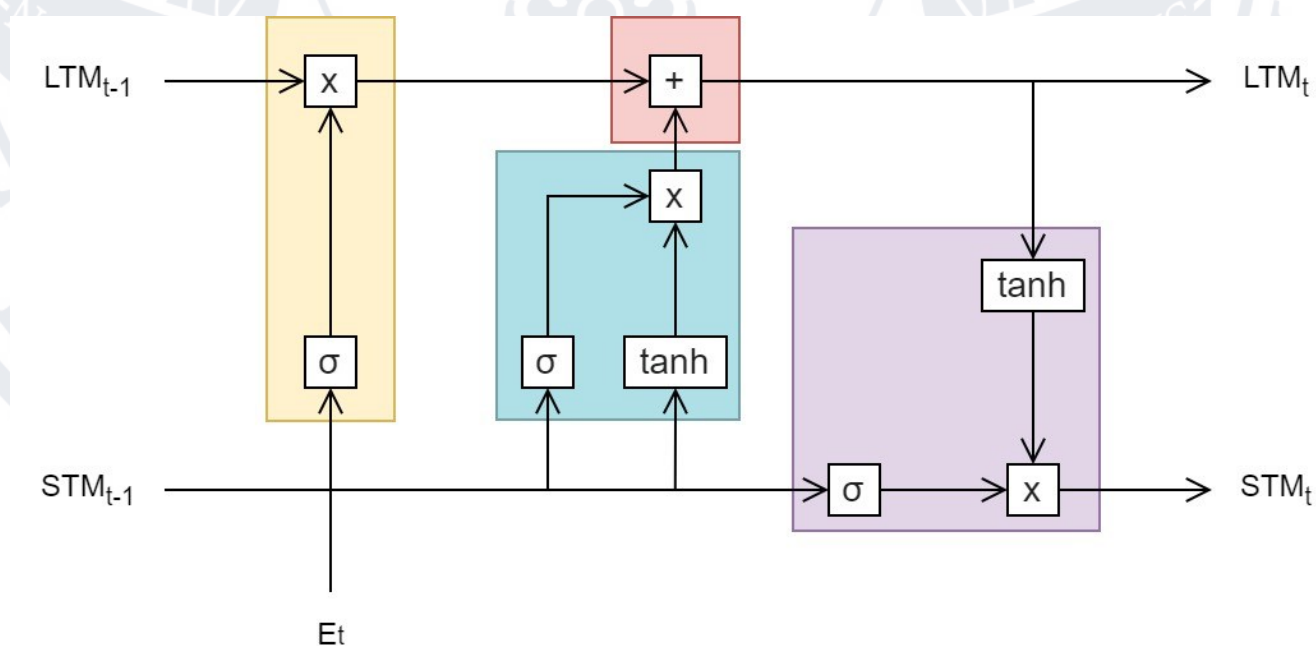


Рисунок 2.12 – Архітектурна модель нейронної мережі LSTM

Дану архітектуру (див. рис. 2.12) краще всього розглядати з точки зору кожного вентиля, щоб краще зрозуміти математичні процеси, які в них відбуваються:

- Забувальний вентиль: використовує довгострокову пам'ять з минулого кроку навчання LTM_{t-1} та вектор даних E_t у якості вхідних даних, та вирішує, яку інформацію слід зберегти, а яку забути (рис. 2.13).

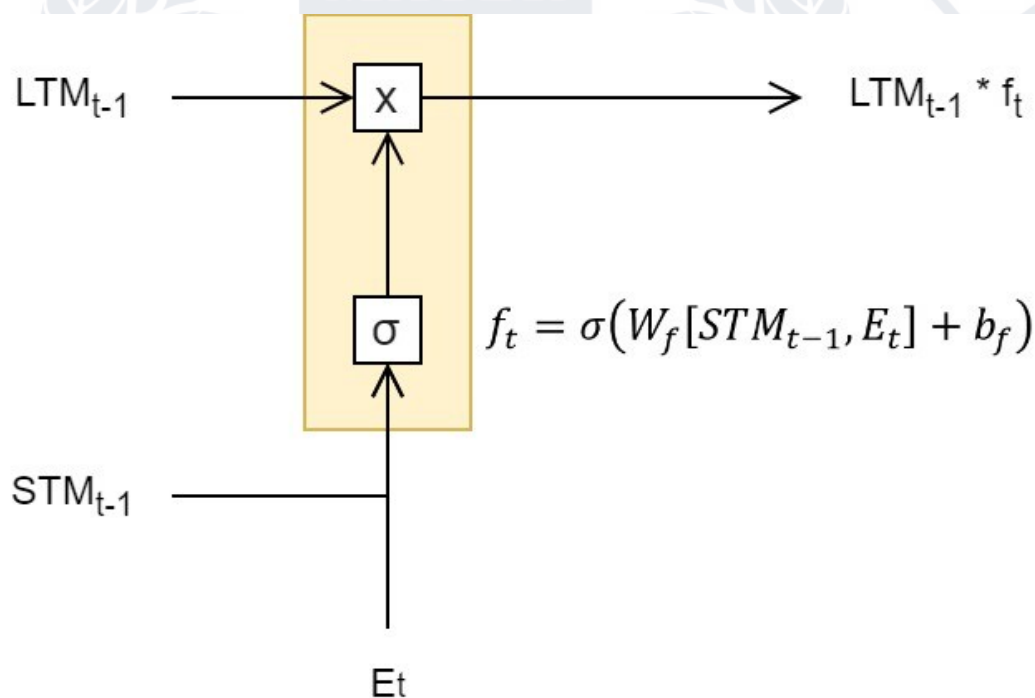


Рисунок 2.13 – Забувальний вентиль в архітектурній моделі неймережі LSTM

На рисунку 2.13 попередня короточасна пам'ять STM_{t-1} і вектор поточних даних E_t об'єднуються разом $[STM_{t-1}, E_t]$, множаться на вагову матрицю W_f і пропускаються через сигмоїдну функцію активації σ з деяким зміщенням b_f для формування коефіцієнта забування f_t .

Коефіцієнт забування f_t потім множиться на попередню довгострокову пам'ять LTM_{t-1} для формування результату на виході забувального вентиля.

- Вхідний вентиль: приймає вектор даних E_t та попередню короткочасну пам'ять STM_{t-1} у якості вхідної інформації, та повертає інформацію, яка є актуальною для прогнозу (рис. 2.14).

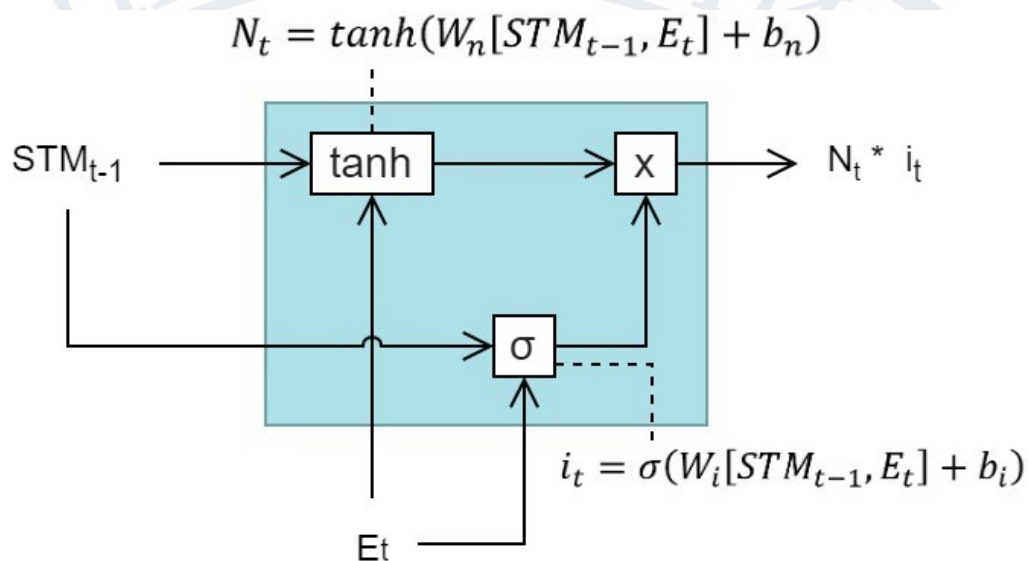


Рисунок 2.14 – Вхідний вентиль в архітектурній моделі нейромережі LSTM

На рисунку 2.14 попередня короткочасна пам'ять STM_{t-1} і вектор поточних даних E_t об'єднуються разом $[STM_{t-1}, E_t]$ і множаться на вагову матрицю W_n , яка має деяке зміщення b_n . Потім результат передається до функції \tanh (гіперболічний тангенс) для введення нелінійності і створюється матриця N_t .

Для ігнорування несуттєвої інформації створюється фактор ігнорування i_t , для якого об'єднується короткочасна пам'ять STM_{t-1} з вектором поточних даних E_t – $[STM_{t-1}, E_t]$ і множиться на вагову матрицю W_i з подальшим пропусканням через сигмоїдну функцію активації σ з деяким зміщенням b_i .

На виході матриця навчання N_t і коефіцієнт ігнорування i_t множаться разом для отримання результату вхідного вентиляю.

- Запам'ятовуючий вентиль: комбінує виходи з забувального і входового вентилів для того, щоб в результаті повернути нове значення для довготривалої пам'яті (рис. 2.15).

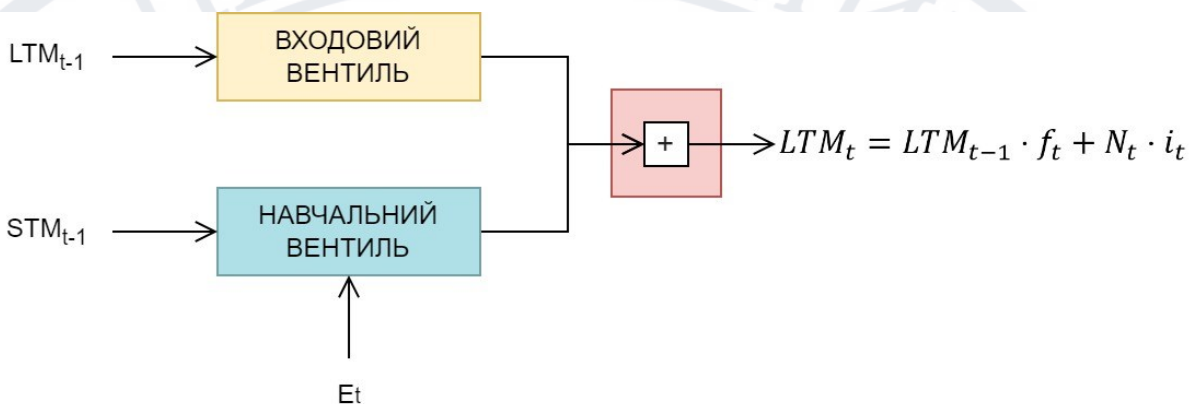


Рисунок 2.15 – Запам'ятовуючий вентиль в архітектурній моделі нейромережі LSTM

- Виходовий вентиль: об'єднує важливу інформацію з минулої довгострокової пам'яті та минулої короткострокової пам'яті, щоб створити нову короткострокову пам'ять для наступного кроку навчання та отримати вихід для поточного вектору даних (рис. 2.16).

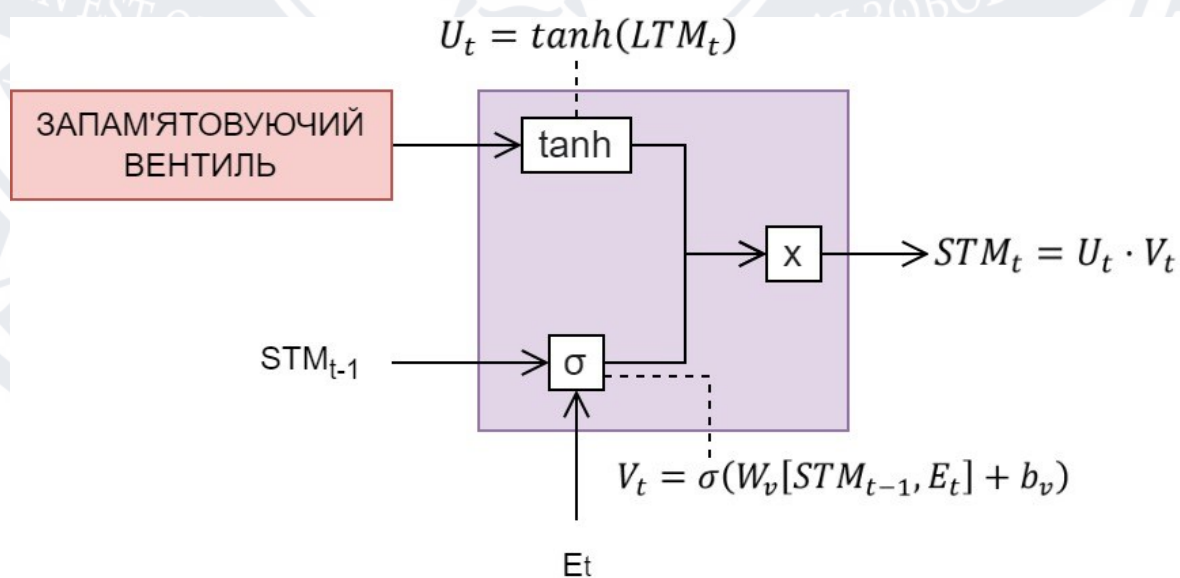


Рисунок 2.16 – Виходовий вентиль в архітектурній моделі нейромережі LSTM

Як можна бачити на рисунку 2.16, спочатку, значення поточного вектору даних E_t та попереднього значення короткочасної пам'яті стану передаються в сигмоїдну функцію σ . Потім значення нової довгострокової пам'яті, створеної із запам'ятовуючого вентиля, передається через функцію гіперболічного тангенса \tanh . Обидва ці виходи перемножуються і на основі результуючого значення мережа поновлює короткострокову пам'ять і повертає вихідний сигнал.

2.3 Висновки до розділу

В рамках цього розділу було проведено аналіз особливостей у прогнозуванні часових рядів та виділено їх основні компоненти, такі як:

- Тренд
- Сезонність
- Випадкова похибка

Також було проведено детальний огляд основних підходів і моделей у методах аналізу і прогнозування часових рядів. Було оглянуто такі методи та поняття, як: автокореляція, часткова автокореляція, декомпозиція, стаціонарність, тест Дікі-Фулера, метод різниць, сімейство методів прогнозування ARIMA, сімейство методів прогнозування GARCH, модель TBATS та нейронна мережа LSTM.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ПОРІВНЯННЯ ЕФЕКТИВНОСТІ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

3.1 Опис інструментів для програмної реалізації

В якості основних інструментів для реалізації програм було обрано мову програмування Python 3 та такий набір бібліотек, як:

- Matplotlib – бібліотека для створення статичних, анімованих та інтерактивних графіків для Python [53].
- Pandas – швидкий та ефективний інструмент, який працює з об'єктами типу DataFrame і має вбудовану функцію індексування для швидкої обробки даних [54].
- Numpy – бібліотека для Python, яка дозволяє використовувати об'єкти багатовимірного масиву, а також різні похідні об'єкти, такі як замасковані масиви та матриці, а також, надає набір процедур для швидких операцій над масивами, включаючи математичні, логічні, маніпуляції формою, сортування, базової лінійної алгебри, основні статистичні операції, випадкове моделювання та багато іншого [55].
- Statsmodels – це бібліотека для Python, яка надає класи та функції для оцінки багатьох різних статистичних моделей, а також для проведення статистичних тестів і дослідження статистичних даних [56].
- Sktime – надає API для машинного навчання з часовими рядами з метою визначення, підгонки, застосування та перевірки моделей машинного навчання [57].
- Sklearn – це модуль Python для машинного навчання, створений на основі SciPy [58].

- Keras – це API глибокого навчання, написаний на Python, який працює на базі платформи для машинного навчання TensorFlow. Він був розроблений для того, щоб надавати можливості для швидкого експериментування над даними [59].

3.2 Практичне застосування методів на часовому ряді криптовалюти

Почати практичне застосування методів прогнозування логічніше усього буде з графіку GARCH, який покаже волатильність часового ряду на момент прогнозування.

Прогнозування буде виконуватись на графіку ціни криптовалюти біткойн (BTC). Для оптимізації прогнозування було вирішено використовувати дані лише за період у два місяці, оскільки актив є дуже волатильним і показники ціни можуть змінюватись дуже кардинально за короткі проміжки часу (рис. 3.1).

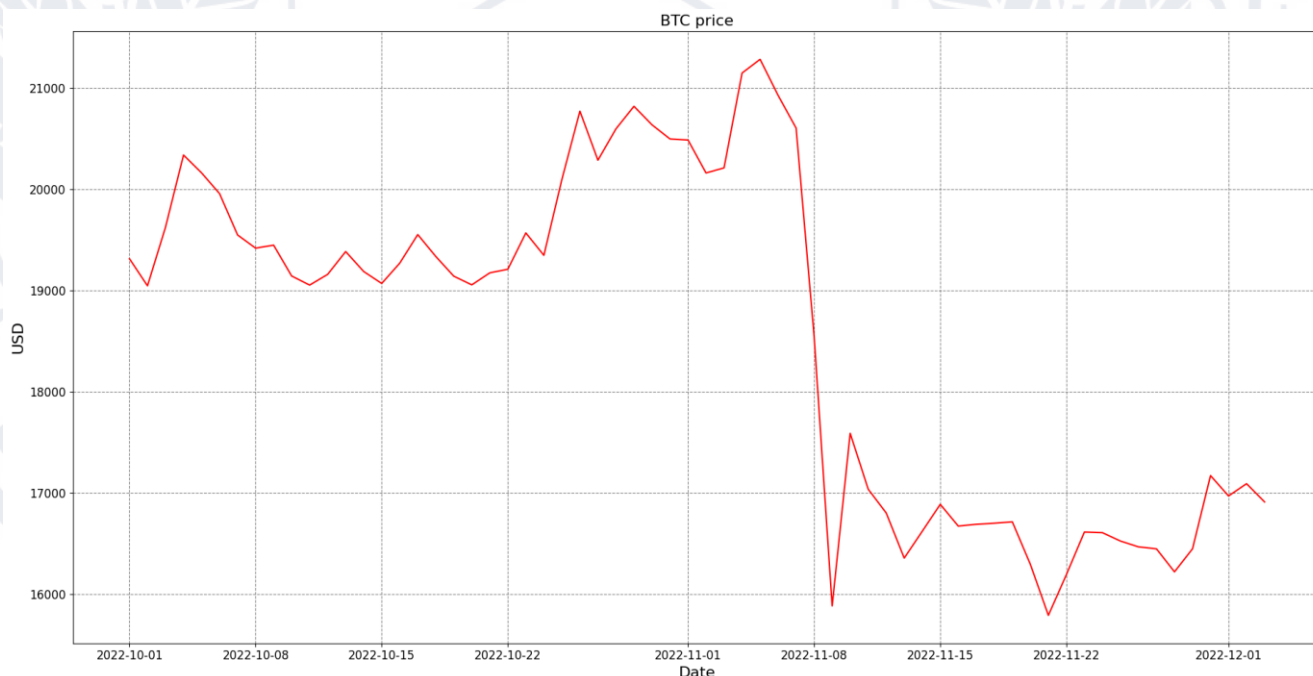


Рисунок 3.1 – Показник ціни BTC у валюті USD за період від 2022.10.01 до 2022.12.03

Аналізуючи рисунок 3.1 можна бачити, що у актива відсутня сезонна складова і тренд, але середнє значення ціни дуже сильно змінюється за короткий час. Застосуємо тест Дікі-Фуллера, щоб переконатись в тому, що графік не є стаціонарним (рис 3.2).

```
Statistic: -1.2218464431528193  
P-value: 0.6641280359413175  
Is stationary: No
```

Рисунок 3.2 – Результат застосування тесту Дікі-Фуллера до ціни BTC

Застосуємо метод різниць часового ряду та побудуємо графік з отриманих значень (рис 3.3).

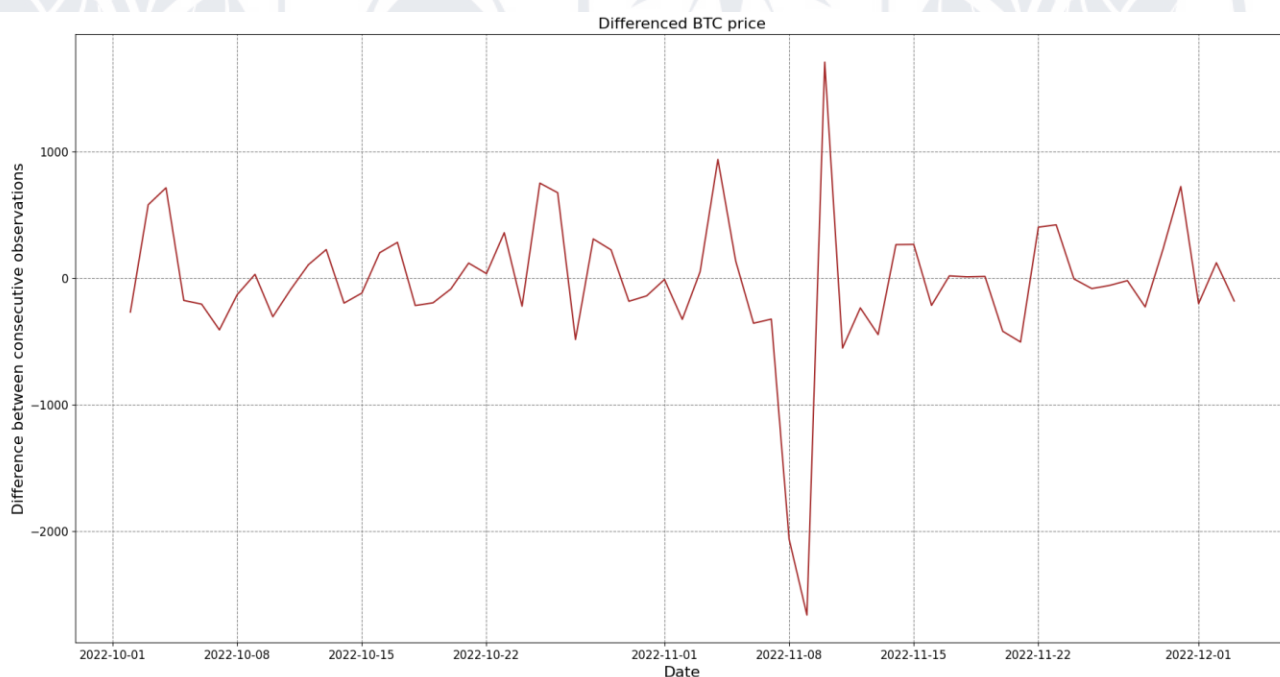


Рисунок 3.3 – Графік після застосування методу різниць до ціни BTC

Як можна бачити з графіку на рисунку 3.3, за допомогою методу різниць вдалось приблизити до нуля середнє значення показника ціни.

Застосуємо тест Дікі-Фуллера, щоб переконатись в тому, що графік змінив своє значення стаціонарності на протилежне (рис 3.4).

```
Statistic: -7.3041347147669216  
P-value: 1.3150030347330842e-10  
Is stationary: Yes
```

Рисунок 3.4 – Результат застосування тесту Дікі-Фуллера до ціни BTC з методом різниць

Графік є стаціонарним (див. рис. 3.4), а це означає, що тепер ми можемо застосувати функції автокореляції (рис. 3.5) та часткової автокореляції (рис. 3.6) для того, щоб знайти показники ρ та q для моделі GARCH.

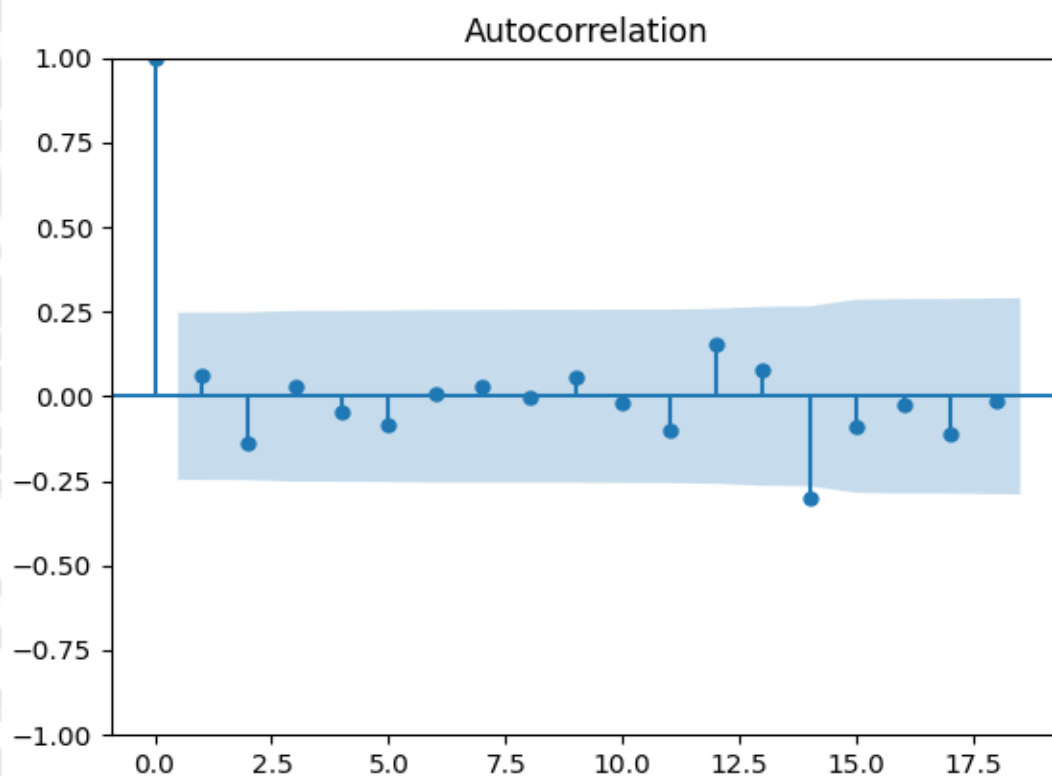


Рисунок 3.5 – Результат застосування автокореляційної функції до показників методу різниць ціни BTC

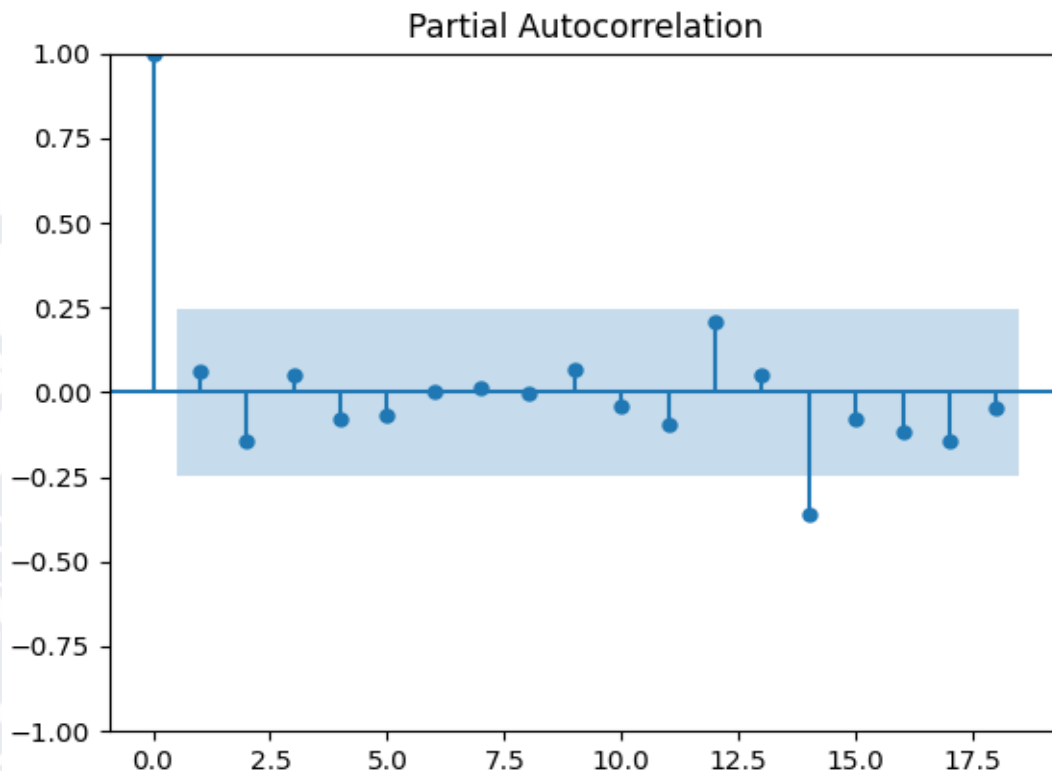


Рисунок 3.6 – Результат застосування часткової автокореляційної функції до показників методу різниць ціни BTC

Як можна бачити на рисунку 3.5 та на рисунку 3.6 – найбільша залежність спостерігається на показнику 14, що означає те, що існує залежність значення теперішнього показника часового ряду від показника часового ряду чотирнадцять кроків тому. Відповідно параметри p та q для GARCH будуть рівними 14. Побудуємо модель GARCH, застосувавши її до показників початкового часового ряду (див. рис. 3.1) та проаналізуємо результат (рис. 3.7).

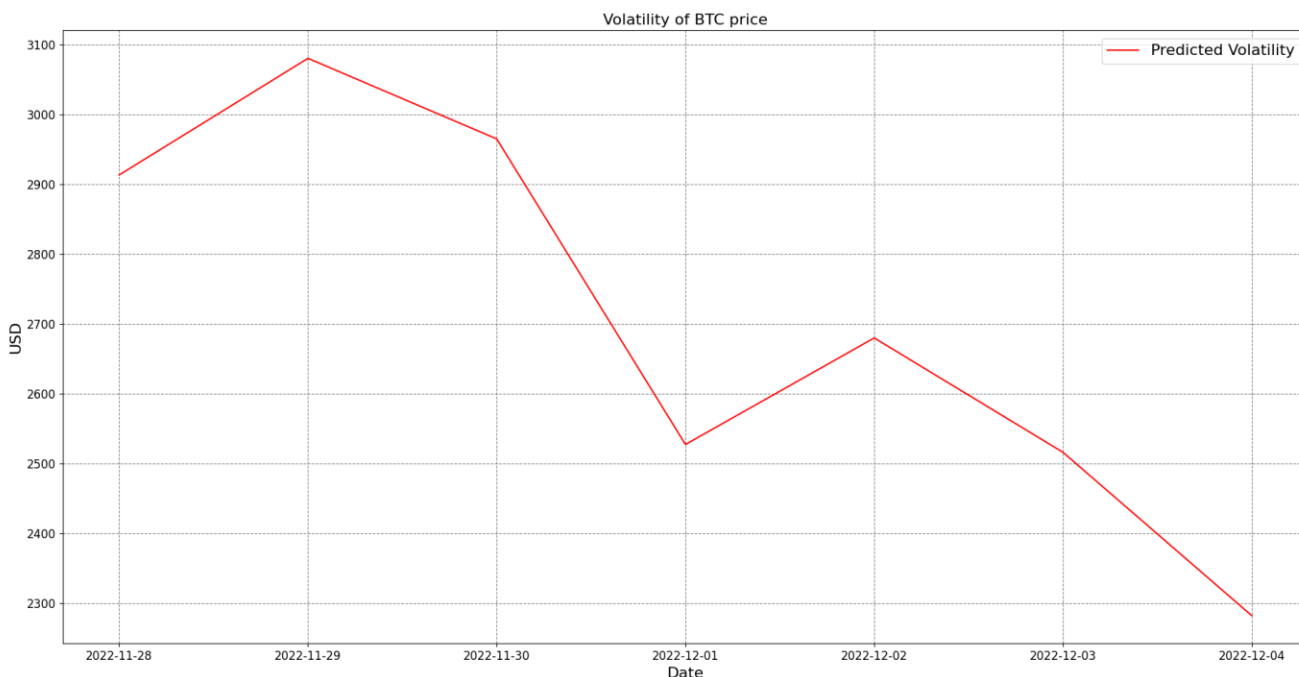


Рисунок 3.7 – Результат застосування моделі GARCH до показників часового ряду ціни BTC за період від 2022.11.28 до 2022.12.04

Як можна бачити на рисунку 3.7, було проведено навчання моделі до 28 листопада 2022-го року і виконано прогноз волатильності до 4 грудня 2022-го року включно. Отож роблячи висновки можна сказати, що зпрогнозована волатильність активу на 4 грудня є низькою порівнюючи її з значеннями за тиждень, що дозволяє бути більш впевненими в прогнозах на цей день.

Отож знаючи те, що показники значень ціни на біткойн стають стаціонарними після застосування методу різниць один раз, а також те, що значення теперішнього показника часового ряду t залежить від показника часового ряду чотирнадцять кроків тому $t-14$, ми можемо побудувати модель SARIMAX з параметрами: $p = 14$, $d = 1$, $q = 14$ (рис. 3.8).



Рисунок 3.8 – Результат застосування моделі SARIMAX до показників часового ряду ціни BTC за період від 2022.11.28 до 2022.12.04

Як можна бачити на рисунку 3.8, модель SARIMAX показала досить непоганий результат за останні три дні прогнозу коли графік більш менш стабілізувався після високого викиду вгору у 800 одиниць.

За тим-самим періодом, побудуємо модель TBATS для прогнозу ціни. При побудові моделі будемо враховувати тренд та перетворення Бокса-Кокса, але не будемо враховувати сезонність, оскільки ціна не має ніякої візуальної періодичності (рис. 3.9).

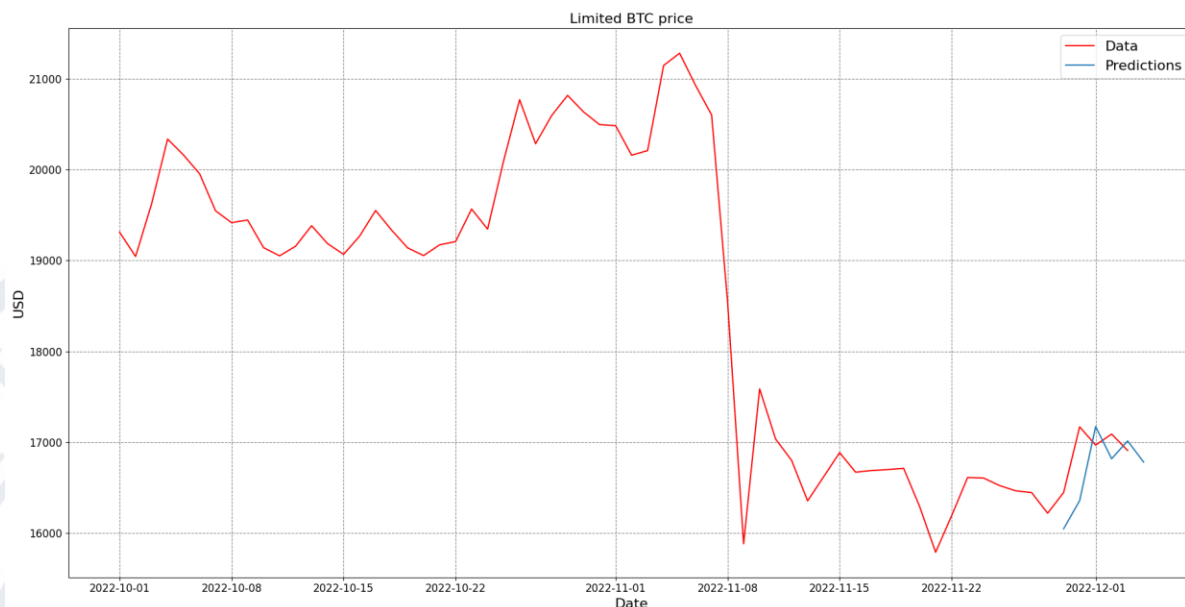


Рисунок 3.9 – Результат застосування моделі TBATS до показників часового ряду ціни BTC за період від 2022.11.28 до 2022.12.04

Дивлячись на результат застосування моделі TBATS до показників часового ряду ціни BTC (див. рис. 3.9), можна бачити, що прогноз є дуже не точним, оскільки модель просто копіює значення з минулого кроку з невеликим відхиленням. Отож роблячи висновок по застосуванню моделі TBATS можна сказати, що при відсутній сезонній компоненті модель майже не дає ніякого чіткого прогнозу і дуже близька за своїм характером до моделі AR(1), яка говорить про те, що поточний показник часового ряду t лінійно залежить від показника часового ряду $t-1$. У більшості випадків таке припущення є дуже різким і майже ніколи не відповідає дійсності, так як часові ряди дуже часто мають набагато складніші залежності.

На останок, застосуємо модель рекурентної нейронної мережі LSTM до часового ряду і проаналізуємо результат її роботи. Для створення моделі було використано бібліотеку Keras з чотирма шарами: два LSTM і два Dense для глибокого в'язання з шарами LSTM і надання одиничного вихідного результату (рис. 3.10).

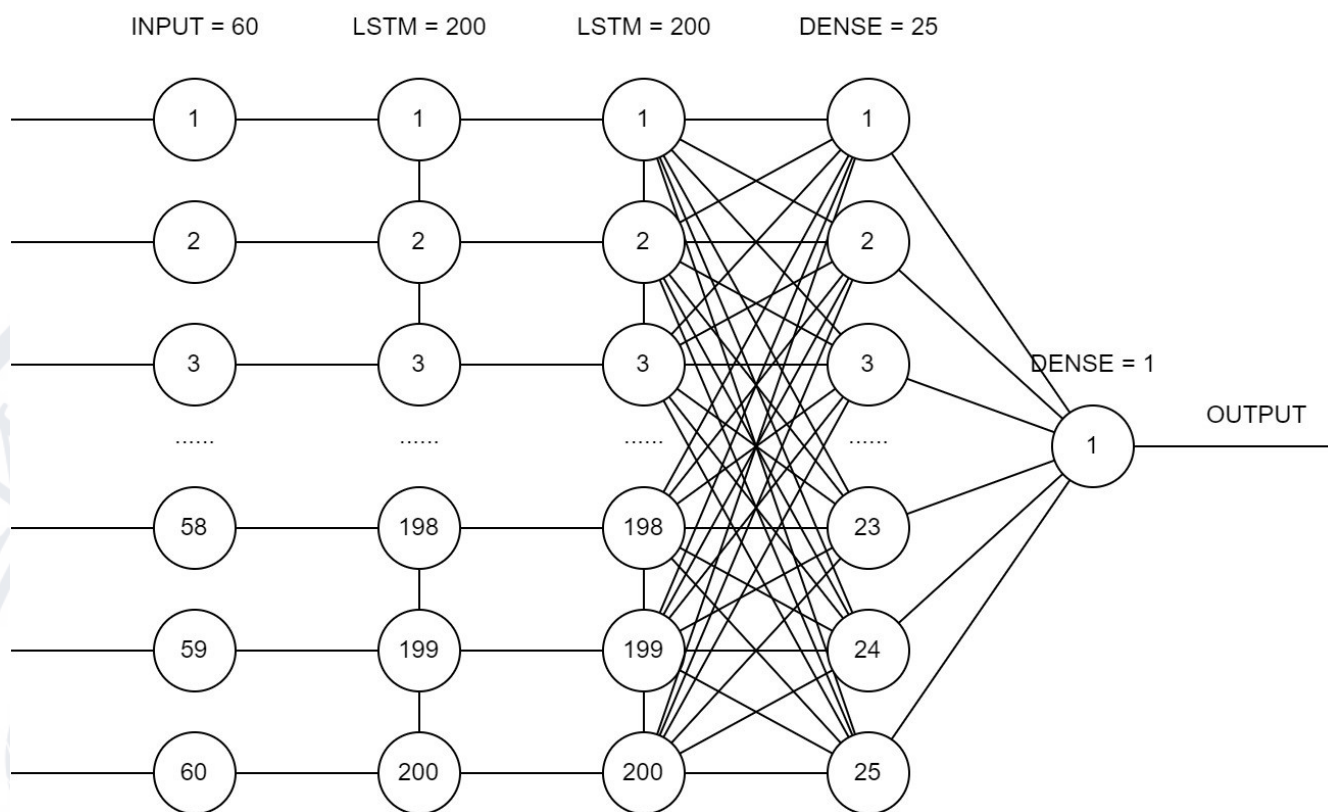


Рисунок 3.10 – Модель нейронної мережі на базі шарів LSTM

Як можна бачити на рисунку 3.10, на вході модель приймає значення показників часового ряду за два місяці, тобто 60 днів, а на виході присутній лише одиничний вихід який і надає значення прогнозу. Для навчання моделі було використано дещо більший набір даних на відміну від інших моделей, а саме показники починаючи з 1 січня 2020-го року. Показник кількості використаних епох був виставлений – дві. Похибку за епохами можна бачити на рисунку 3.11, а результат прогнозу на рисунку 3.12.

```
Epoch 1/2
909/909 [=====] - 307s 327ms/step - loss: 0.0039
Epoch 2/2
909/909 [=====] - 306s 336ms/step - loss: 0.0015
Model: "sequential"
```

Рисунок 3.11 – Похибка прогнозів моделі при навчанні за епохами

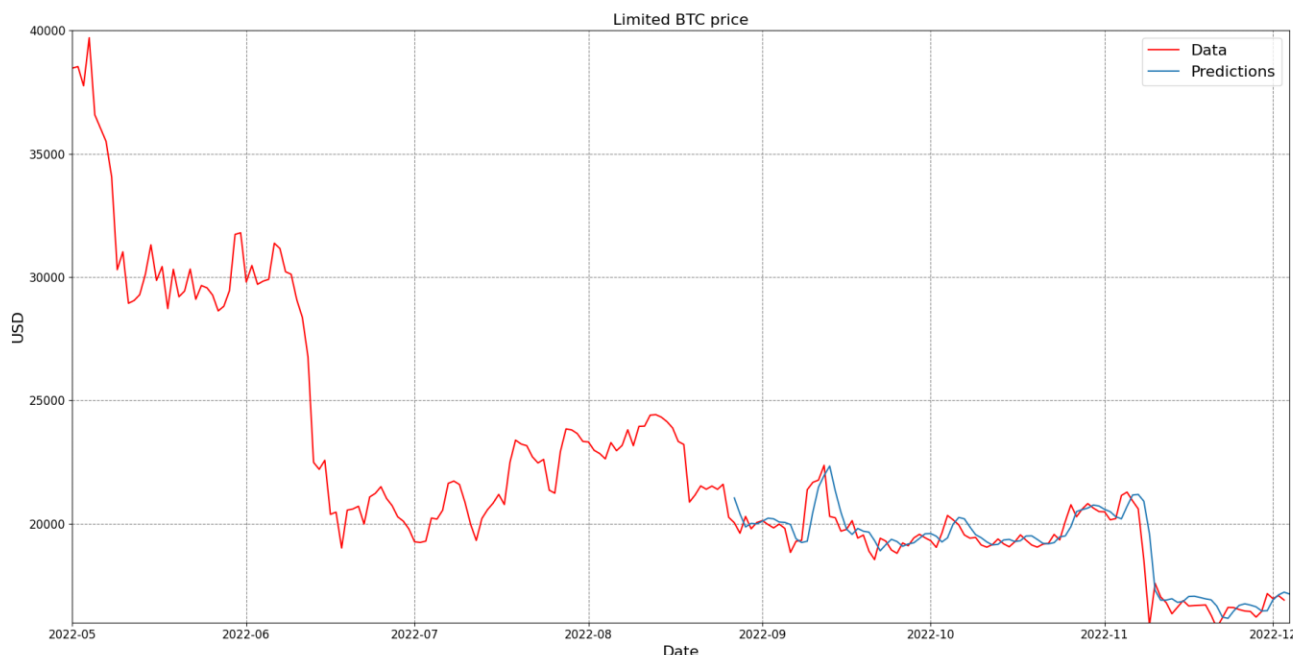


Рисунок 3.12 – Результат застосування нейромережі з шарами LSTM до показників часового ряду ціни BTC за період від 2022.08.26 до 2022.12.04

Як можна бачити на графіку, нейромережа реагує з деяким запізненням на різкі зміни у графіку, але непогано себе показує на більш-менш стаціонарних ділянках часового ряду.

3.3 Висновки до розділу

Отже у рамках даного розділу було застосовано різні методи до прогнозування часового ряду криптовалюти біткойн (BTC), такі як:

- GARCH – узагальнена авторегресивна умовна гетероскедастичність.
- SARIMAX – сезонне авторегресивне інтегроване ковзне середнє з екзогенними регресорами
- TBATS
- LSTM – Довга короткочасна пам'ять

Найкращі значення прогнозу показали моделі SARIMAX та LSTM, що також з наданими значеннями волатильності GARCH дозволяє більш якісно робити прогноз.

ВИСНОВКИ

У ході виконання дипломної роботи було визначено основні поняття ринку, розглянуті різні типи структур ринку, а також проведене детальне порівняння різних типів ринку: фондового та криптовалютного. Описано поняття біржі і детально оглянуті її основні елементи, поняття і властивості. Проведений огляд існуючих методів аналізу і прогнозування часових рядів з метою ознайомлення, визначення їх особливостей і основних компонентів, таких як:

- Тренд
- Сезонність
- Випадкова похибка

Також, було детально розглянуто такі методи і поняття, як: автокореляція, часткова автокореляція, декомпозиція, стаціонарність, тест Дікі-Фулера, метод різниць, сімейство методів прогнозування ARIMA, сімейство методів прогнозування GARCH, модель TBATS та нейронна мережа LSTM.

Було виконано аналіз та порівняння ефективності застосування методів до прогнозування часового ряду криптовалюти біткойн (BTC), таких як:

- GARCH – узагальнена авторегресивна умовна гетероскедастичність.
- SARIMAX – сезонне авторегресивне інтегроване ковзне середнє з екзогенними регресорами
- TBATS
- LSTM – Довга короткочасна пам'ять

Найкращі значення прогнозу показали моделі SARIMAX та LSTM, що також з наданими значеннями волатильності моделі GARCH дозволяє більш точніше та якісніше виконувати прогноз майбутнього руху ціни на криптовалюту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ

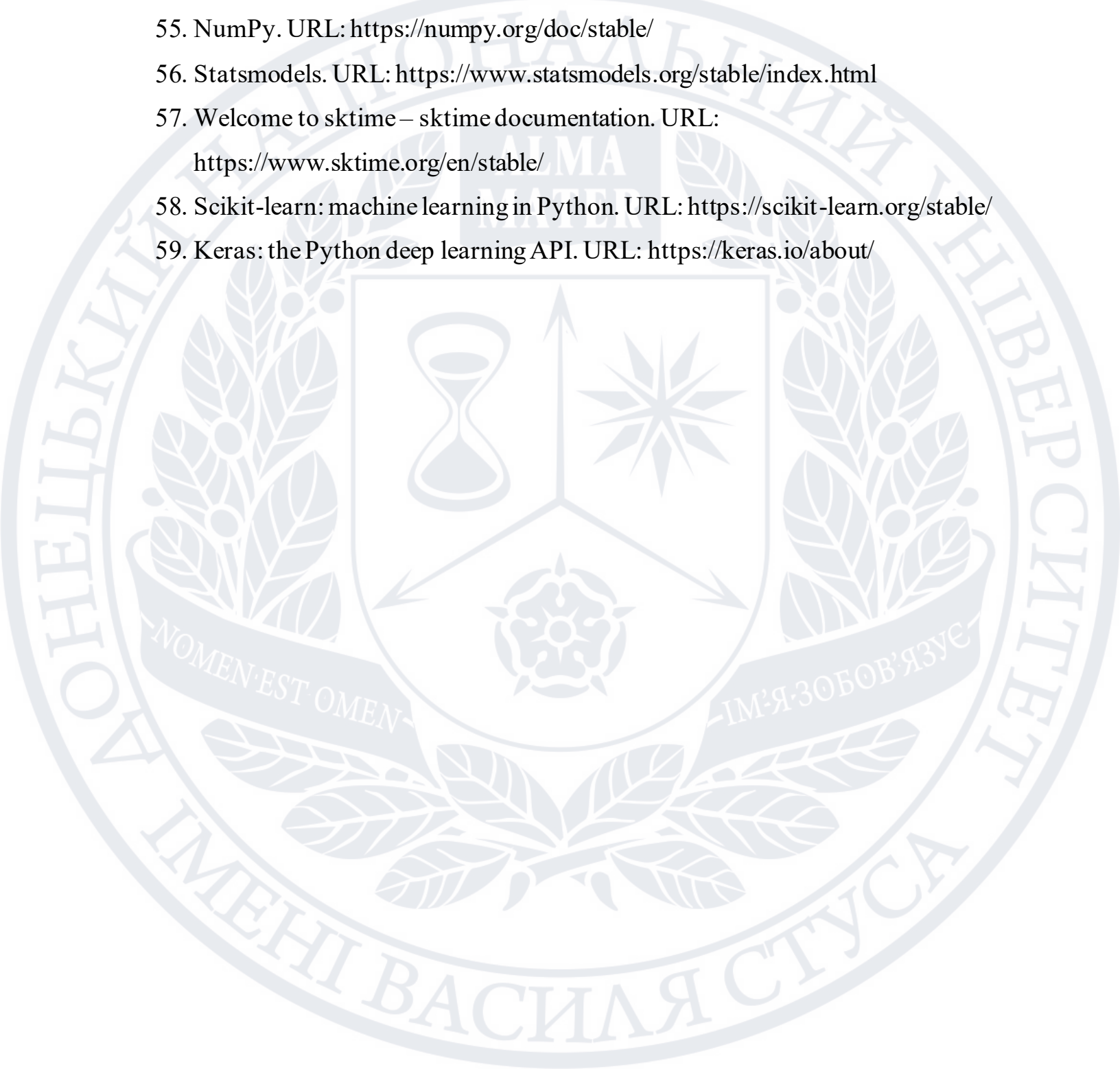
1. Will Kenton & Charles Potters (2021). *Market: What It Means in Economics, Types and Common Features*. Retrieved from <https://www.investopedia.com/terms/m/market.asp>
2. Key Players in the Capital Markets. URL: <https://corporatefinanceinstitute.com/resources/career/key-players-in-capital-markets/>
3. Структура ринку: типи і види. URL: <https://buklib.net/books/33784/>
4. Adam Hayes & Robert C. Kell (2020). *Price-Taker: Definition, Perfect Competition, and Examples*. Retrieved from <https://www.investopedia.com/terms/p/pricetaker.asp>
5. Загальна характеристика ринкових структур. URL: https://pidru4niki.com/13000611/politekonomiya/zagalna_harakteristika_rinkovih_struktur
6. Monopolistic Competition. URL: https://saylordotorg.github.io/text_developing-new-products-and-services/s04-01-monopolistic-competition.html
7. Характеристика олігопольного ринку. URL: <https://buklib.net/books/33877/>
8. R. S. Khemani & D. M. Shapiro (1993). *Glossary of Industrial Organisation Economics and Competition Law*. Retrieved from <https://www.oecd.org/regreform/sectors/2376087.pdf>
9. What is 'Stock Market'. URL: <https://economictimes.indiatimes.com/definition/stock-market>
10. Andrew Beattie & Chop Stapleton (2022). *The Birth of Stock Exchanges*. Retrieved from <https://www.investopedia.com/articles/07/stock-exchange-history.asp>
11. Nasdaq. URL: <https://www.nasdaq.com/>

12. What is online trading? URL: <https://upstox.com/learning-center/online-trading/what-is-online-trading/>
13. U.S. Securities and Exchange Commission. URL: https://en.wikipedia.org/wiki/U.S._Securities_and_Exchange_Commission
14. Key Players in the Capital Markets. URL: <https://corporatefinanceinstitute.com/resources/career/key-players-in-capital-markets/>
15. Caplinger D. (2020). 'What Is a Stock Market Index?'. Retrieved from <https://www.fool.com/investing/stock-market/indexes/>
16. What are bulls and bears? URL: <https://www.bitstamp.net/learn/crypto-trading/what-are-bulls-and-bears/>
17. Anna-Louise Jackson & Benjamin Curry (2022). *Short Selling Guide*. Retrieved from <https://www.forbes.com/advisor/investing/short-selling/>
18. Understanding market capitalization. URL: <https://www.fidelity.com/learning-center/trading-investing/fundamental-analysis/understanding-market-capitalization>
19. What is cryptocurrency? URL: <https://www.coinbase.com/learn/crypto-basics/what-is-cryptocurrency>
20. How Does Blockchain Work? URL: <https://www.simplilearn.com/tutorials/blockchain-tutorial/blockchain-technology>
21. History of bitcoin. URL: https://en.wikipedia.org/wiki/History_of_bitcoin
22. Kat Tretina & Benjamin Curry (2021). *What Is A Stock Exchange?* Retrieved from <https://www.forbes.com/advisor/investing/stock-exchange/>
23. Що таке волатильність на біржі? URL: <https://ffin.ua/blog/faq/investytsii/post/shcho-take-volatylnist-na-birzhi>
24. Abhishek R. (2019). *ARIMA simplified. - Towards Data Science*. Retrieved from <https://towardsdatascience.com/arima-simplified-b63315f27cbc>

25. Francq, C. & Zakoian, J. (2011). *GARCH Models: Structure, Statistical Inference and Financial Applications*. Retrieved from:
<https://faculty.washington.edu/ezivot/econ589/ch18-garch.pdf>
26. Time Series Forecasting using TBATS Model. URL: <https://medium.com/analytics-vidhya/time-series-forecasting-using-tbats-model-ce8c429442a9>
27. Understanding LSTM Networks. URL: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>
28. Niklas Donges (2022). *Understanding RNN and LSTM Networks*. Retrieved from
<https://builtin.com/data-science/recurrent-neural-networks-and-lstm>
29. Time Series – Definition, Analysis, Forecasting, Components. URL:
<https://www.wallstreetmojo.com/time-series/>
30. Екстраполяція – Вікіпедія. URL: <https://uk.wikipedia.org/wiki/Екстраполяція>
31. Інтерполяція – Вікіпедія. URL: <https://uk.wikipedia.org/wiki/Інтерполяція>
32. Univariate Time Series Models. URL:
<https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc44.htm>
33. Multivariate Time Series. URL: <https://www.sciencedirect.com/topics/computer-science/multivariate-time-series>
34. Time series decomposition. URL: <https://otexts.com/fpp2/decomposition.html>
35. Will Kenton & Robert C. Kelly (2020). *Seasonality: What It Means in Business and Economics*. Retrieved from <https://www.investopedia.com/terms/s/seasonality.asp>
36. What is a trend in time series? URL: <https://www.geeksforgeeks.org/what-is-a-trend-in-time-series/>
37. Tim Smith & Clierra Murry (2022). *Autocorrelation*. Retrieved from
<https://www.investopedia.com/terms/a/autocorrelation.asp>
38. Autocorrelation Function and its Properties. URL:
<https://www.tutorialspoint.com/autocorrelation-function-and-its-properties>

39. Partial autocorrelation function – Wikipedia. URL:
https://en.wikipedia.org/wiki/Partial_autocorrelation_function
40. Detecting stationarity in time series data. URL:
<https://www.kdnuggets.com/2019/08/stationarity-time-series-data.html>
41. Unit root – Wikipedia. URL: https://en.wikipedia.org/wiki/Unit_root
42. Dickey–Fuller test – Wikipedia. URL: https://en.wikipedia.org/wiki/Dickey–Fuller_test
43. Augmented Dickey–Fuller test – Wikipedia. URL:
https://en.wikipedia.org/wiki/Augmented_Dickey–Fuller_test
44. Stationarity and differencing. URL: <https://otexts.com/fpp2/stationarity.html>
45. Autoregressive Model: Definition & The AR Process. URL:
<https://www.statisticshowto.com/autoregressive-model/>
46. Moving average models. URL: <https://otexts.com/fpp2/MA.html>
47. ARMA model. URL: <https://www.statisticshowto.com/arma-model/>
48. Aayush Bajaj (2022). *ARIMA & SARIMA: Real-World Time Series Forecasting*. Retrieved from <https://neptune.ai/blog/arima-sarima-real-world-time-series-forecasting-guide>
49. ARCH/GARCH Models. URL: <https://online.stat.psu.edu/stat510/lesson/11/11.1>
50. Time-Series Forecasting using TBATS model. URL: <https://blog.tenthplanet.in/time-series-forecasting-tbats/>
51. Introduction to Long Short Term Memory (LSTM). URL:
<https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/>
52. Нейронні мережі - шлях до глибинного навчання. URL:
<https://codeguida.com/post/739>
53. Matplotlib – Visualization with Python. URL: <https://matplotlib.org/>

54. Pandas - Python Data Analysis Library. URL:
<https://pandas.pydata.org/about/index.html>
55. NumPy. URL: <https://numpy.org/doc/stable/>
56. Statsmodels. URL: <https://www.statsmodels.org/stable/index.html>
57. Welcome to sktime – sktime documentation. URL:
<https://www.sktime.org/en/stable/>
58. Scikit-learn: machine learning in Python. URL: <https://scikit-learn.org/stable/>
59. Keras: the Python deep learning API. URL: <https://keras.io/about/>



ДОДАТОК А

Лістинги

```

from statsmodels.tsa.stattools import adfuller
import pandas as pd
import yfinance as yf

def get_differenced_series(dataframe, column_name, depth = 1):
    dataframe = dataframe.fillna(dataframe.interpolate())
    prev_dataframe_values = dataframe.loc[:, column_name].shift()
    differenced_series = dataframe.loc[:, column_name] - prev_dataframe_values
    if depth > 1:
        df_from_series = pd.DataFrame(differenced_series, columns=[column_name])
        return get_differenced_series(df_from_series, column_name, depth - 1)
    else:
        return differenced_series.dropna()

def get_adf(dataframe):
    dataframe = dataframe.dropna()
    adf_result = adfuller(dataframe)
    adf_info = dict()
    adf_info['Statistic'] = adf_result[0]
    adf_info['P-value'] = adf_result[1]
    adf_info['Is stationary'] = 'Yes' if adf_result[1] < 0.05 else 'No'
    return adf_info

def print_dictionary(dictionary):
    for key, value in dictionary.items():
        print(f'{key}: {value}')

def get_ticker_data(ticker_symbol, ticker_period, ticker_date_from, ticker_date_to,
                    ticker_col_name):
    ticker_data = yf.Ticker(ticker_symbol)
    ticker_df = ticker_data.history(period=ticker_period, start=ticker_date_from,
    end=ticker_date_to)
    ticker_df = pd.DataFrame(ticker_df, columns=[ticker_col_name])
    ticker_df = __fillna_interpolation(ticker_df)

```

```
    return ticker_df

def __fillna_interpolation(dataframe):
    return dataframe.fillna(dataframe.interpolate())
```

Лістинг А.1 prediction helpers

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.seasonal import seasonal_decompose
from datetime import timedelta
from statsmodels.tsa.statespace.sarimax import SARIMAX
from prediction_helpers import get_differenced_series, get_adf, print_dictionary,
get_ticker_data

TICKER_SYMBOL = 'BTC-USD'
TICKER_PERIOD = '1d'
TICKER_COL_NAME = 'Close'

TICKER_DATE_FROM = '2010-01-01'
TICKER_DATE_TO = '2022-12-04'

LIM_DATE_FROM = '2022-10-01'

TRAIN_DATE_TO = '2022-11-28'

TEST_DATE_FROM = '2022-11-29'

PLOT_INFO_SIZE = 16
PLOT_TICKS_SIZE = 12

def main():
    # Get data to predict
    ticker_data = get_ticker_data(TICKER_SYMBOL, TICKER_PERIOD,
    TICKER_DATE_FROM, TICKER_DATE_TO, TICKER_COL_NAME)
```

```
# Plot price to date
fig, ticker_plot = plt.subplots()
ticker_plot.set_title('BTC price', fontsize=PLOT_INFO_SIZE)
ticker_plot.set_xlabel('Date', fontsize=PLOT_INFO_SIZE)
ticker_plot.set_ylabel('USD', fontsize=PLOT_INFO_SIZE)
ticker_plot.xaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
ticker_plot.yaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
ticker_plot.grid(color='gray', linestyle='dashed')
ticker_plot.plot(ticker_data, color='red')

# Limit train data
lim_ticker_df = ticker_data[LIM_DATE_FROM:TICKER_DATE_TO]

# Plot limited price to date
fig, lim_ticker_plot = plt.subplots()
lim_ticker_plot.set_title('Limited BTC price', fontsize=PLOT_INFO_SIZE)
lim_ticker_plot.set_xlabel('Date', fontsize=PLOT_INFO_SIZE)
lim_ticker_plot.set_ylabel('USD', fontsize=PLOT_INFO_SIZE)
lim_ticker_plot.xaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
lim_ticker_plot.yaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
lim_ticker_plot.grid(color='gray', linestyle='dashed')
lim_ticker_plot.plot(lim_ticker_df, color='red')

# Decomposing and plotting data
ticker_df_close_decompose = seasonal_decompose(lim_ticker_df)
ticker_df_close_decompose.plot()

# Use ADF to check if differencing is needed
adf_info = get_adf(lim_ticker_df)
print_dictionary(adf_info) # Is stationary: No

# Apply differencing to remove unit root
ticker_differenced = get_differenced_series(lim_ticker_df, TICKER_COL_NAME, 1)

# Use ADF to check if differencing is needed
adf_info = get_adf(ticker_differenced)
print_dictionary(adf_info) # Is stationary: Yes
```

```

# Plot result of differencing
fig, differenced_plot = plt.subplots()
differenced_plot.set_title('Differenced BTC price', fontsize=PLOT_INFO_SIZE)
differenced_plot.set_xlabel('Date', fontsize=PLOT_INFO_SIZE)
differenced_plot.set_ylabel('Difference between consecutive observations',
fontsize=PLOT_INFO_SIZE)
differenced_plot.xaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
differenced_plot.yaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
differenced_plot.grid(color='gray', linestyle='dashed')
differenced_plot.plot(ticker_differenced, color='brown')

# Use ACF to find value for MA
plot_acf(ticker_differenced) # 2, 12 lags are the best
plt.ylim(-1, 1)

# Use PACF to find value for AR
plot_pacf(ticker_differenced) # 2, 12 lags are the best
plt.ylim(-1, 1)

# Prepare test data
test_data = lim_ticker_df[TEST_DATE_FROM:TICKER_DATE_TO]

# Define and train model and make a prediction on t+1
rolling_predictions = pd.DataFrame()
prediction_dates = test_data.index.copy()
prediction_dates = pd.DatetimeIndex(np.append(prediction_dates, prediction_dates[-1] +
timedelta(days=1)))
for prediction_date in prediction_dates:
    train_data = lim_ticker_df[:prediction_date - timedelta(days=1)]
    model = SARIMAX(train_data, order=(16, 1, 1))
    model_fit = model.fit(dispatch=False)
    predicted_value = model_fit.forecast()
    prediction_row = pd.DataFrame({TICKER_COL_NAME: predicted_value},
index=[prediction_date])
    rolling_predictions = pd.concat([prediction_row, rolling_predictions.loc[:]])

# Plotting results
lim_ticker_plot.plot(rolling_predictions)

```

```
lim_ticker_plot.legend(('Data', 'Predictions'), fontsize=PLOT_INFO_SIZE)

plt.show()

if __name__ == "__main__":
    main()
```

Лістинг A.2 sarimax implementation

```
import numpy as np
import pandas as pd
from datetime import timedelta
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt
from tensorflow import keras
from tensorflow.python.keras import layers
from prediction_helpers import get_ticker_data
from datetime import datetime

TICKER_SYMBOL = 'BTC-USD'
TICKER_PERIOD = '1d'
TICKER_COL_NAME = 'Close'

TICKER_DATE_FROM = '2020-01-01'
TICKER_DATE_TO = '2022-12-04'

LIM_DATE_FROM = '2022-05-01'

TRAIN_DATE_TO = '2022-08-26'

PLOT_INFO_SIZE = 16
PLOT_TICKS_SIZE = 12

TRAINING_PERIOD_SIZE = 60

def main():
    # Get data to predict
```



```

ticker_data = get_ticker_data(TICKER_SYMBOL, TICKER_PERIOD,
TICKER_DATE_FROM, TICKER_DATE_TO, TICKER_COL_NAME)

# Plot limited date price
fig, lim_ticker_plot = plt.subplots()
lim_ticker_plot.set_title('Limited BTC price', fontsize=PLOT_INFO_SIZE)
lim_ticker_plot.set_xlabel('Date', fontsize=PLOT_INFO_SIZE)
lim_ticker_plot.set_ylabel('USD', fontsize=PLOT_INFO_SIZE)
lim_ticker_plot.xaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
lim_ticker_plot.yaxis.set_tick_params(labelsize=PLOT_TICKS_SIZE)
lim_ticker_plot.grid(color='gray', linestyle='dashed')
lim_ticker_plot.plot(ticker_data, color='red')
lim_ticker_plot.set_xlim(datetime.strptime(LIM_DATE_FROM, '%Y-%m-%d'),
datetime.strptime(TICKER_DATE_TO, '%Y-%m-%d'))
lim_ticker_plot.set_ylim(16000, 40000)

values = ticker_data.values
normalizer = MinMaxScaler(feature_range=(0, 1)) # to set range values of ticker data
from 0 to 1
normalized_data = normalizer.fit_transform(values)

training_data = ticker_data[:TRAIN_DATE_TO]
training_data_length = len(training_data)
normalized_training_data = normalized_data[0: training_data_length, :]

x_train = []
y_train = []

for i in range(60, training_data_length):
    x_train.append(normalized_training_data[i-TRAINING_PERIOD_SIZE: i, 0]) # train on
TRAINING_PERIOD_SIZE values
    y_train.append(normalized_training_data[i, 0]) # expected results of training

# place each record in the array and group them in another arrays of length
x_train.shape[1]
x_train = np.array(x_train)
x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))

```

```

y_train = np.array(y_train)

test_data_with_date = ticker_data[training_data_length:]
test_data = normalized_data[training_data_length-TRAINING_PERIOD_SIZE:, :]
test_data_length = len(test_data)

x_test = []

for i in range(TRAINING_PERIOD_SIZE, test_data_length + 1):
    x_test.append(test_data[i-TRAINING_PERIOD_SIZE: i, 0])

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

model = keras.Sequential()
model.add(layers.LSTM(200, return_sequences=True, input_shape=(x_train.shape[1], 1)))
model.add(layers.LSTM(200))
model.add(layers.Dense(25))
model.add(layers.Dense(1))
model.compile(optimizer='adam', loss='mean_squared_error')
model.fit(x_train, y_train, batch_size=1, epochs=2)
model.summary()

predictions = model.predict(x_test)
predictions = normalizer.inverse_transform(predictions) # remove normalization

rolling_predictions = pd.DataFrame()
prediction_dates = test_data_with_date.index.copy()
prediction_dates = pd.DatetimeIndex(np.append(prediction_dates, prediction_dates[-1] +
timedelta(days=1)))
prediction_index = 0
for prediction_date in prediction_dates:
    prediction_row = pd.DataFrame({TICKER_COL_NAME: predictions[prediction_index]},
index=[prediction_date])
    rolling_predictions = pd.concat([prediction_row, rolling_predictions.loc[:]])
    prediction_index += 1

lim_ticker_plot.plot(rolling_predictions)

```

```
lim_ticker_plot.legend(('Data', 'Predictions'), fontsize=PLOT_INFO_SIZE)

plt.show()

if __name__ == "__main__":
    main()
```

Лістинг А.3 lstm implementation

