

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Чорноморський національний університет
імені Петра Могили**

Факультет комп'ютерних наук

Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.

_____ Ю. П. Кондратенко

«__» _____ 20__ р.

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

**Інтелектуальна система прогнозування на основі
індексів криптовалют**

Спеціальність «Комп'ютерні науки»

122 – МКР – 601.1830803

Студент _____ **М. Ф. Фінажин**

«__» _____ 20__ р.

Консультант _____ **І. О. Калініна**

к-т. техн. наук, доцент

«__» _____ 20__ р.

Миколаїв – 2022

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет ім. Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

Освітньо-кваліфікаційний рівень магістр

Галузь знань 12 «Інформаційні технології»
(шифр і назва)

Спеціальність 122 «Комп'ютерні науки»
(шифр і назва)

проф.

ЗАТВЕРДЖУЮ

Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук,

Ю. П. Кондратенко

« _____ » _____ 20__ р.

ЗАВДАННЯ
на магістерську кваліфікаційну роботу
Фінажину Микиті Федоровичу

1. Тема магістерської кваліфікаційної роботи «Інтелектуальна система прогнозування на основі індексів криптовалют»

Керівник роботи Калініна Ірина Олександрівна, к-т техн. наук, доцент

2. Строк представлення кваліфікаційної роботи « _____ » _____ 2022 р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні

Вхідні дані: необроблений набір даних, який містить інформацію о криптовалютних індексах. Очікуваним результатом інтелектуальна система прогнозування на основі індексів криптовалют, що здатна надати якісні результати прогнозу різноманітних видів криптовалют.

4. Перелік питань, що підлягають розробці

— аналіз сучасного стану розвитку технологій криптовалют та існуючих публікацій за темою аналізу та прогнозування криптовалют із застосуванням криптовалютних індексів і часових рядів у вчасності;

— огляд існуючих методів побудови моделей прогнозування;

— побудова інтелектуальної системи, згідно розроблених похідних ітераційних етапів розробки;

— використання новітніх методів при побудові моделей прогнозування, для отримання якісних результатів прогнозування;

— порівняння результатів отриманих прогнозів та обрання найкращого результату.

5. Перелік графічних матеріалів: презентація.

6. Завдання до спеціальної частини

Завдання до спеціальної частини: покращення умов освітленості робочого простору працівників, аналіз протипожежних мір на підприємстві та дій робітників при виникненні пожежі.

7. Консультанти:

Розділ	Прізвище, ініціали та посада консультанта	Підпис
Спеціальна частина з охорони праці		
Методична частина		

Керівник роботи к-т техн. наук, доцент Калініна І. О.
(посада, прізвище, ім'я, по батькові)

_____ (підпис)

Завдання прийнято до виконання Фінажин М. Ф.
(прізвище, ім'я, по батькові студента)

_____ (підпис)

Дата видачі завдання « ____ » _____ 20 ____ р.

АНОТАЦІЯ

до кваліфікаційної роботи магістра
«Інтелектуальна система прогнозування на основі індексів криптовалют»
Студент 601 гр.: Фінажин Микита Федорович
Керівник: к-т техн. наук, доц. Калініна І. О.

Робота присвячена розробці інтелектуальної системи, спроможної прогнозувати ціноутворення на ринку криптовалют, використовуючи індекси криптовалют як метрику навчання системи.

Об'єктом дослідження є процес прогнозування змін індексів криптовалют.

Предметом дослідження є методи прогнозування на основі часових рядів.

Метою дослідження є підвищення конкурентоздатності користувача на ринку криптовалют, за рахунок використання надійної інтелектуальної системи прогнозування на основі індексів криптовалют.

Методами дослідження є аналіз, проектування та алгоритмізація рішень, програмне кодування системи, тестування.

Дипломна робота включає в себе загальні відомості даної предметної галузі, проект програмного забезпечення, розділ з охорони праці. До розділів входять основні відомості щодо програмного продукту, його характеристики, проблеми та їх рішення.

Дипломна робота викладена на 63 аркушах, містить 2 додатки, 5 таблиць, 33 рисунки, перелік джерел посилання з 15 найменувань.

Ключові слова: *криптовалюта, часові ряди, прогнозування, рекурентні нейронні мережі, мова програмування R*

ABSTRACT

of the Master's Thesis

"Intelligent forecasting system based on cryptocurrency indices"

Student of group 601: Fynazhyn Mykyta Fedorovych

Head: Cand. Tech. Sciences, Assoc. Kalinin I. O.

The object of the study is the process of forecasting changes in cryptocurrency indices.

The subject of the research is forecasting methods based on time series.

The study's aim is to increase user competitiveness in the cryptocurrency market through the use of a reliable intelligent forecasting system based on cryptocurrency indices.

Research methods are analysis, design and algorithmization of solutions, software coding of the system, testing.

Thesis includes general information in this subject area, software project, section on labor protection. The sections include basic information about the software product, its characteristics, problems and solutions.

Thesis is presented on 63 sheets, contains 2 appendices, 5 tables, 33 figures, a list of reference sources of 15 titles.

Keywords: cryptocurrency, time series, forecasting, recurrent neural networks, R programming language

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ	3
ВСТУП	5
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ	7
1.1 Аналіз предметної області	7
1.2 Огляд та аналіз наявних аналогів та публікацій	12
1.3 Постановка задачі	16
Висновок до розділу 1	18
2 МАТЕМАТИЧНІ МОДЕЛІ, МЕТОДИ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ	19
2.1 Прогнозування з використанням часових рядів	19
2.2 Стаціонарні часові ряди	22
2.3 Методи і моделі для прогнозування з використанням часових рядів	25
2.3.1 Просте експоненційне згладжування.....	25
2.3.2 Подвійне експоненційне згладжування.....	26
2.3.3 Потрійне експоненційне згладжування.....	27
2.3.4 Моделі з урахування сезонності і тренду.....	28
Висновок до розділу 2	31
3 МОДЕЛЮВАННЯ ТА ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	32
3.1 Етапи збору та обробки даних	32
3.2 Етап агрегації даних	41
3.3 Етап навчання моделей та прогнозування	51
Висновок до розділу 3	67
4 РЕЗУЛЬТАТИ ВИКОНАННЯ РОБОТИ	68
Висновок до розділу 4	75
МЕТОДИЧНИЙ РОЗДІЛ	77
СПЕЦІАЛЬНИЙ РОЗДІЛ	88
ВИСНОВКИ	104
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	105

ПЕРЕЛІК СКОРОЧЕНЬ

- ARIMA – програмне забезпечення
- ETH – Ethereum
- IC – інтелектуальна система
- LSTM – мікроконтролер
- LOCF – Last Observation Carried Forward
- MSE – середньоквадратична помилка (mean squared error)
- MAE – середня абсолютна помилка (mean absolute error)
- MAPE – середня абсолютна відсоткова помилка (mean absolute percentage error)
- NaN – Not a Number
- RMSE – квадратний корінь з середньоквадратичної помилки (root mean squared error)
- SHF – імітовані історичні прогнози (simulated historical forecast)

Пояснювальна записка

до магістерської кваліфікаційної роботи

на тему:

«Інтелектуальна система прогнозування на основі індексів криптовалют»

Спеціальність 122 «Комп'ютерні науки»

122 – МКР – 601.1830803

Студент _____ **М. Ф. Фінажин**
«_____» _____ 2022 р.

Консультант _____ **І. О. Калініна**
к-т техн. наук, доцент
«_____» _____ 2022 р.

Миколаїв – 2022

ВСТУП

У сьогоденному технологічному ритмі життя інформація виступає одним з найголовніших ресурсів для людини, яка в деяких видах діяльності, інколи навіть грає більшу вартість ніж інші цінні матеріали, такі як робочі можливості людини, території і т. д. Найціннішою інформацією можна вважати лише найактуальніші дані. Інакше кажучи, за інших рівних умов, чим свіжіша інформація, тим вона цінніша. Іншим важливим елементом цінної інформації є її точність та достовірність. Можливо сказати, що вся цінність ринку полягає у отриманні актуальної інформації. Саме тому системи прогнозування набули такої популярності. Окрім того, з появою криптовалют, ринок зазнав значних змін, так як вони є важливим ресурсом, який надає багато нових можливостей. Можливість спрогнозувати зміни криптовалют, надає власнику цієї інформації значний контроль над ситуацією на ринку, що може захистити від ризикових наслідків.

Практичне застосування розробленої ІС забезпечить високий рівень надійності для користувача, при торгах на ринку криптовалют. Розроблена ІС забезпечує точні та актуальні дані, використовуючи які, користувач зменшує ризики невдалого використання фінансів.

Об'єктом дослідження є індекси криптовалют та часові ряди у окремоті, використовуючи які, ІС точно прогнозує вартість криптовалют на ринку.

Предметом дослідження є методи прогнозування криптовалют, за допомогою яких можливо навчити та розробити ефективну інтелектуальну систему прогнозування.

Метою дослідження є підвищення конкурентоздатності користувача на ринку криптовалют, за рахунок використання надійної інтелектуальної системи прогнозування на основі індексів криптовалют.

Для досягнення мети мають бути вирішені такі завдання:

- 1) проаналізувати існуючі на ринку рішення інтелектуальних систем прогнозування на основі індексів криптовалют;
- 2) провести огляд та аналіз існуючих математичних методів моделювання і прогнозування криптовалютних котирувань;
- 3) розробити модель системи для прогнозування курсу криптовалюти;
- 4) розробити інтелектуальну систему, в якій реалізовано роботу з індексами криптовалют, для подальшого прогнозування;
- 5) випробувати програмний продукт на реальних даних та провести порівняльний аналіз із обґрунтованим вибором кращої моделі.

Методами дослідження є аналіз, проектування та алгоритмізація рішень, програмне кодування системи, тестування.

Практичне значення результатів роботи полягає у розробці програмного продукту, здатного виконувати короткостроковий прогноз фінансової цінності криптовалюти, використовуючи індекси криптовалют та методи інтелектуального аналізу даних.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз предметної області

Приголомшлива швидкість розвитку криптовалютного ринку та його подальша інтеграція в системи господарських, операційних, фінансових та інших середовищ нагтовхують на необхідність комплексного вивчення даного явища. Окрім цього, підвищеного інтересу до теми додають новини про офіційну легалізацію криптовалютних операцій на державному рівні і можливостей, які надає подібна технологія у господарській діяльності економічних агентів. Навіть, якщо не звертати увагу на деякі песимістичні настрої та підходи, що на даний момент популярні серед експертів в даній середі, розвиток криптовалютних технологій йде дуже швидко не зважаючи на зовнішні фактори.

Зараз важко уявити роль, яку відіграватимуть криптовалюти в економіці наступних десятиліть. Але її можна передбачити. Саме точність прогнозів визначає стратегію успіху та приносить дохід.

В один момент цифрова торгівля почала набувати все більш примітної форми потоку грошей і самі гроші почали набути електронну форму. Вона виникла не одразу, але поєднання з все більшими сферами технологічних реалій, вона набула ту саму цифрову форму, в якій зараз відома багатьом людям – криптовалюта. Зважаючи на те, що зараз майже кожна людина використовує електронні платежі, певні групи людей почали блукати в недовірливих роздумах до криптовалюти, не розуміючи в чому саме різниця від звичайних грошей.

Дослідження криптовалют буде розглянуто на прикладі криптовалюти Bitcoin та ETH, оскільки вони є одними із найбільш популярних та розповсюджених в порівнянні із іншими, а принцип роботи суттєво не відрізняється від інших. Графічне представлення можливо побачити на рис. 1.1.

В 2008 році розробник програмного забезпечення Сатоши Накамото запропонував Bitcoin як систему електронних платежів, засновану на

криптографічній базі. Головною інновацією стало створення такого засобу обміну, який можливо використовувати окремо від централізованої влади і, який реалізовано в електронній формі та надає користувачам максимально безпечний, перевірений та неможливий для відстеження спосіб.

Отримати повне розуміння принципів роботи такої складної технології як Bitcoin можливо розділивши його на дві складові. Першою складовою виступає Bitcoin-токен – певний фрагмент коду, за допомогою якого представляють цифрове володіння (можливо порівняння з борговим зобов'язанням). Другою складовою є Bitcoin-протокол. Протокол – це розподілена на частини мережа, яка слугує для підтримки реєстру балансів Bitcoin-токенів. Об'єднання обох частин представляє собою відому усім технологію Bitcoin.

Як було зазначено, головна перевага криптовалютних технологій, таких як Bitcoin, в наданні можливості користувачам використовувати повністю децентралізовану та невідстежувану органами влади можливість виконання розрахункових операцій в мережі Інтернет. Це повністю електронний вид грошових одиниць, який неможливо роздрукувати, як наприклад долар, євро або юань. Подібні валюти без будь-яких обмежень друкуються по всьому світу.

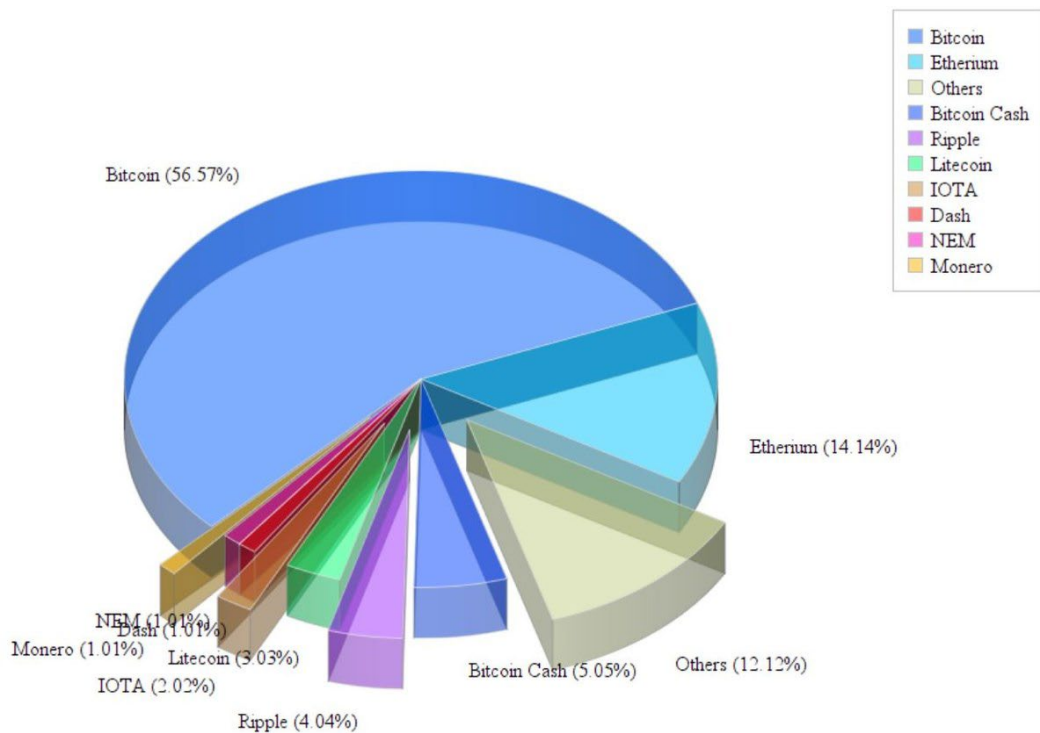


Рис. 1.1. Рейтинг найпопулярніших криптовалют

Для побудови актуальної та точної системи прогнозування зміни вартості крипто валют, можливо використовувати індекси необхідних криптовалют. Індекси являють собою певні метрики, які відображають важливі дані, що впливають на вартість крипто валют. Актуальну інформацію о крипто валютних індексах, зручно збирати з ресурсів агрегаторів подібних даних. Популярними прикладами є: CoinMarketCap, WorldCoinIndex та CryptoCompare. Подібні ресурси надають інформацію про актуальні ціни різних криптовалют, але також вони містять відомості для порівняння та моніторингу різних крипто валютних активів. Залежно від критеріїв індекс може містити інформацію про всі монети і токени, доступні на ринку, або лише деякі з них, наприклад, доступні на публічних біржах з достатнім рівнем обсягу торгів. Втім, здебільшого авторитетні крипто валютні індекси відстежують більшість існуючих монет, тож за їхньою допомогою зручно стежити за станом ринку — не потрібно вручну компіювати дані сотень окремих обмінників.

Окрім цього, при прогнозуванні використовуються часові ряди, які являють собою послідовність значень деякої змінної (або змінних), що реєструються через певні проміжки часу (регулярні або нерегулярні). Коли є тільки одна змінна, що спостерігається, тимчасовий ряд називають одномірним. У випадку ж з декількома змінними, що паралельно спостерігаються, говорять про багатовимірний часовий ряд. Варіант використання часових рядів представлено на рис. 1.2.

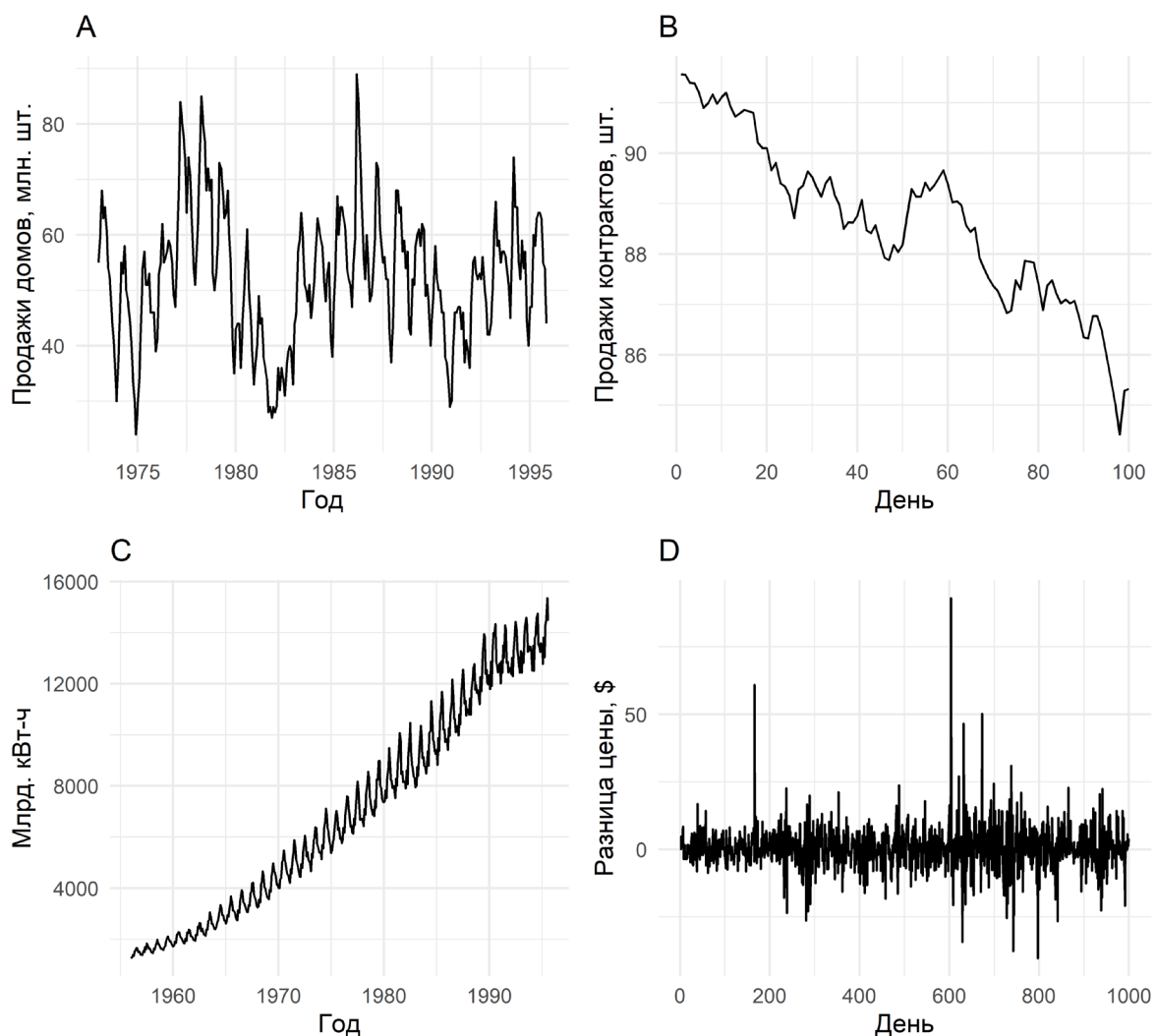


Рис. 1.2. Використання часових рядів

За останню одну-дві декади, теорія о використанні часових рядів переживає натуральну революцію. Створено неймовірну кількість різноманітних прикладних статичних моделей, вичерпний математичний аналіз котрих тільки почне розглядатись. Неможливо не визначити те, що звичайно головну причину вивчення та використання новітніх моделей зумовлено реальними практичними потребами.

Через суттєві покращення економічних ринків, виникненням нових сфер розвитку технологій та інформаційної революції, людство отримало важливе розуміння – для подальшого швидкого розвитку потрібне створення та використання нових математичних засобів для точного та зрозумілого опису і прогнозування даних. Всім відомі класичні методи, робота яких описує використання стаціонарності часових рядів, на даний час не може відобразити швидко прогресуючі, нелінійні процеси. З іншої сторони, після капітального розширення теорій стаціонарних, лінійних та гаусових моделей розумним наступним етапом стане відмова від подібних припущень обособлення отриманих результатів.

Першим кроком до оновлення стала еволюція розроблених математичних технік та засобів до моделей, що були схожі на розроблені раніше та мали деякі характерні подібності. На поточний час, розроблюють все більшу кількість суттєво нових моделей аналізу та прогнозування та методів, які дозволяють їх досліджувати. Вірним стане твердження, що розділи теорії часових рядів проходять етап становлення.

Описуючи класичний підхід побудови моделей часових рядів, потрібно зазначити структуру розкладу його на певні компоненти, суттєве значення котрих принципово відмінне від останніх. Потім кожна компонента починає аналізуватись специфічними для неї методами. Найпоширенішим на практиці методом розкладу часового ряду y_t є наступний:

$$y_t = tr_t + s_t + r_t$$

Компонента tr_t представляє складову під назвою тренд. Тренд змінює повільно, відносно інших компонент часового ряду. Він може мати вигляд простої сталої, многочлену або експоненти чи будь-якої іншої подібної функції. Тренд описує у часовому ряді довготривалі тенденції аналізованого явища. Компонента s_t описує періодичні зміни на часовому ряді. Ця компонента носить назву сезонної або циклічної компоненти. Найчастіше сезонною компонентою описують явища, які виникли на виробництві протягом довгого періоду, наприклад року, або різні економічні циклічності. Компонента r_t описує випадковість часового ряду. До випадковості можливо віднести події, що не було заздалегідь передбачено. Основний напрямок розвитку теорії часових рядів якраз є моделювання і аналіз процесу r_t .

1.2 Огляд та аналіз наявних аналогів та публікацій

При аналізуванні робіт на схожу тематику, було виділено статті, які гарно розкривають проблематику прогнозування вартості криптовалют.

Одна з досліджених статей, яка має назву «Improved Forecasting of Cryptocurrency Price using Social Signals» розглядає можливість прогнозування вартості криптовалют за допомогою аналізу соціальних мереж. Стислий опис статті представлено в абзаці нижче.

Сигнали соціальних мереж успішно використовуються для розробки великомасштабної предикативної та випереджальної аналітики. Наприклад, прогнозування цін на фондовому ринку та спалахів грипу.

Нещодавно соціальні дані використали для прогнозування коливань цін на криптовалюту. коливання цін на криптовалюту, які є новою руйнівною технологією, що має значні політичні та економічні наслідки наслідками.

У своїй роботі автори використовують та зіставляють прогностичну силу соціальних сигналів, зокрема, поведінки користувачів та ком-. комунікаційних моделей з кількох соціальних платформ GitHub та Reddit для прогнозування цін

на три криптовалюти з високим інтересом розробників та спільноти - Bitcoin, Ethereum, та Monero.

Вони оцінюють ефективність нейромережових моделей, заснованих на блоках довготривалої короткочасної пам'яті (LSTM) навчених на історичних даних про ціни та соціальні дані, в порівнянні з моделями, заснованими тільки на цінах тільки LSTM і базових авторегресивних інтегрованих похідних середніх (ARIMA) моделях, які зазвичай використовуються для прогнозування цін на акції. Їх результати не лише демонструють, що соціальні сигнали зменшують помилку під час прогнозування щоденної ціни криптовалют, але також показують, що мова, що використовується в коментарях в офіційних популярних спільнотах таких як Reddit (r/Bitcoin, r/Ethereum та r/Monero) є найкращими предикторами в цілому. Графіки представлено на рис. 1.3 та 1.4.

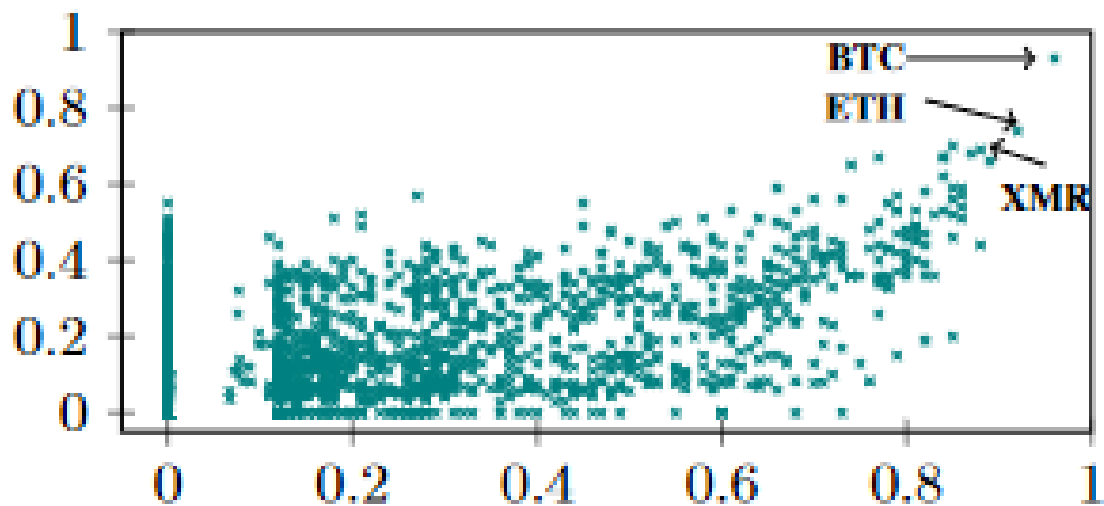


Рис. 1.3. Співвідношення інтересу групи людей до певних статей, зміні цін на криптовалюту

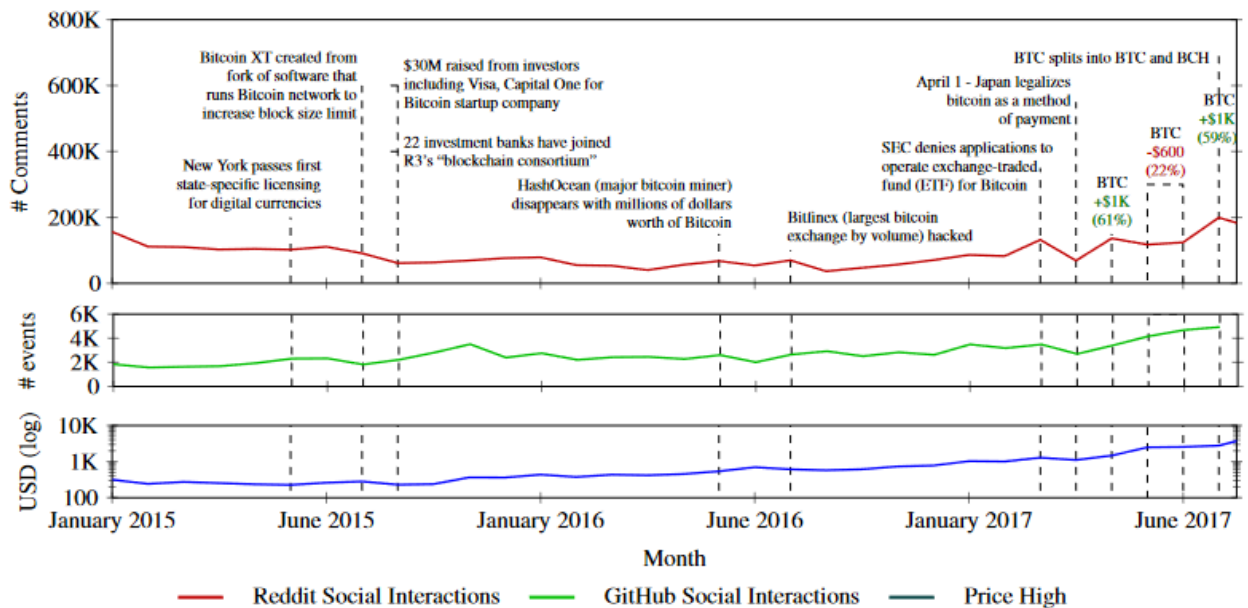


Рис. 1.4. Інтерес людей до статей на різних ресурсах та зміна ціни на криптовалюту

Друга досліджена стаття, яка має назву «Cryptocurrency volatility forecasting: What can we learn from the first wave of the COVID-19 outbreak?», досліджує мінливість цін на криптовалюту під час таких подій як епідемія коронавірусу та її наслідки.

Ринок криптовалют набув популярності серед засобів масової інформації, політиків, інвесторів, науковців та регуляторів, особливо після «бульбашки» 2017 року. Цей ринок також був джерелом критики та дискусій щодо його конкретних стилізованих фактів, особливо відновлених після суперечливості позиції популярних нобелівських лауреатів Джозефа Стігліца та Роберта Шиллера. Перший припустив, що біткойн має бути заборонений, тоді як другий підкреслив, що інтерес інвесторів до ринку криптовалюти пов'язаний з його антиурядовим, анти регуляційним збором. З академічної точки зору існує велика дискусія щодо того, чи можна вважати криптовалюту валютою чи активом

(Yuneline, 2019; White et al., 2020;), оскільки ціни демонструють значний ріст з 2017 року, що супроводжується високою мінливістю. Наприклад, ціна біткойна залишалася нижче 1000 доларів до лютого 2017 року; однак у грудні 2017 року він досяг 20 000 доларів США, а потім знизився приблизно до 8 000 доларів США в лютому 2018 року. У травні 2018 року він знову зріс і досяг 13 000 доларів США, а в грудні 2018 року швидко впав приблизно до 3 000 доларів США. З 2020 року та під час пандемії коронавірусної хвороби (COVID-19) на ринку біткойнів спостерігається ще одна ознака цінової бульбашки, оскільки ціни в грудні 2020 року становили близько 24 000 доларів США (найвищий показник з моменту його запровадження).

Цей сплеск цін на біткойн, який спостерігався під час пандемії COVID-19, супроводжувався значною кількістю літератури, яка досліджувала, чи можуть криптовалюти, зокрема біткойн, служити притулком у період негараздів, як-от триваюча криза охорони здоров'я (Huynh та ін., 2020; Paule-Vianez та ін.). У цьому контексті може бути важливим проаналізувати та запропонувати подальші ідеї щодо моделювання мінливості та прогнозування ринку криптовалют, особливо під час кризових періодів, таких як пандемія COVID-19, що дозволить інвесторам та хеджерам мінімізувати ризики за допомогою диверсифікації портфеля та розробити відповідне хеджування. позиції та допомагати політикам у формуванні регуляторної політики шляхом уточнення прогнозу волатильності цін на активи для оцінки ризику. Це дослідження має на меті внести свій внесок у літературу про моделювання та прогнозування мінливості ринку криптовалют, особливо в періоди кризи. Для прикладу наведено графік змін ціни на біткоїн під час епідемії коронавірусу на рисунку 1.5.

Цена биткоїна



Рис. 1.5. Зміна вартості під час пандемії

1.3 Постановка задачі

Розроблена інтелектуальна система прогнозування надає точні результати спрогнозованої вартості криптовалют. Подібна ефективність системи досягається завдяки виконанню поступових ітераційних етапів обробки даних. Основними етапами, які можливо виділити є наступні:

— Етап збірки та обробки даних. Даний етап слугує початковою ланкою роботи системи. Існує багато ресурсів-агрегаторів, які надають можливість отримати бази даних з колекціями крипто валютних індексів. Саме на цих даних, система буде навчатись та прогнозувати у подальшому. Окрім цього, на даному етапі, з набору криптовалютних індексів, вилучаються усі некоректні значення.

—Етап агрегації даних. Агрегування даних, потрібно для вилучення потрібних часових рядів з існуючого масиву даних. Завдяки розбиттю даних на

необхідні часові ряди, система має можливість використання найбільш точної моделі прогнозування.

—Етап навчання моделей. Виконується розбиття використаних у моделі даних на тренувальну та перевірочну вибірки. В подальшому, кожна з моделей використовує власний набір особливостей прогнозування, що дає різну точність прогнозування.

—Етап прогнозування даних. Кінцевий етап роботи системи. Система робить вибір найбільш правдивої моделі прогнозування за допомогою виконання перехресної перевірки серед вже навчених моделей.

Кожний з етапів хоч і є поступовим у виконанні, має можливість ітераційного застосування. На виконанні кожного з етапів, система має можливість повернутися до будь-якого з попередніх, для уточнення або зміни вже існуючого набору даних. Це дає можливість гнучкого навчання системи, що веде до високоточного прогнозування вартості криптовалют.



Рис. 1.6. Етапи розробки інтелектуальної системи

Висновок до розділу 1

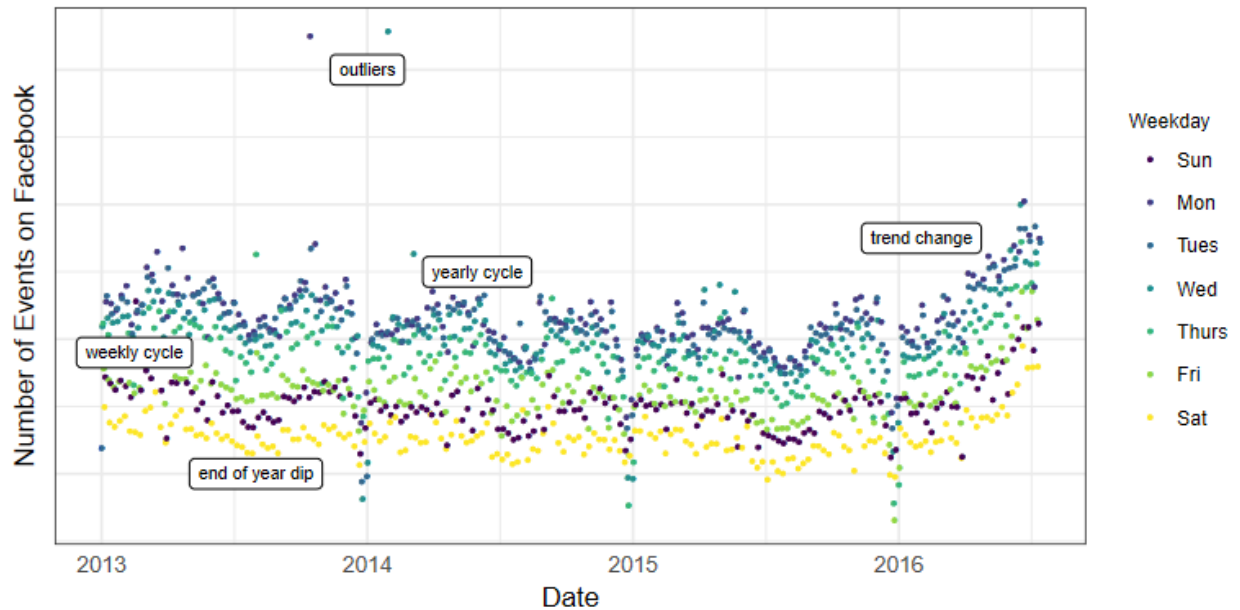
У результаті розробки першого розділу дипломної роботи відкрилась можливість дослідження та опису об'єкта та предмета дослідження, описано основні характеристики системи та її особливості. Розглянуто, що саме представляє собою технологія криптовалют та основи, завдяки яким можливо побудувати якісну інтелектуальну систему прогнозування. Розглянуто складові ціноутворення. Окрім того, розглянуто існуючі аналоги розроблюваної системи.

2 МАТЕМАТИЧНІ МОДЕЛІ, МЕТОДИ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

2.1 Прогнозування з використанням часових рядів

При розробці інтелектуальної системи прогнозування криптовалют для даної дипломної роботи, використовувався пакет для мови R під назвою “prophet”. Короткий опис даного пакету і часових рядів в цілому наведено нижче, і більш детально у наступній главі. Пакет було розроблено двома інженерами компанії Facebook – Sean J. Taylor & Benjamin Letham.

Існує велика різноманітність проблем бізнес-прогнозування, проте деякі особливості прогнозування є загальними для багатьох з них. Користувачі платформи Facebook можуть створювати різноманітні події за допомогою платформи Event, запрошувати на ці події інших користувачів та взаємодіяти з подіями різними способами з подіями інших користувачів.. На рис. 2.1 показані щоденні дані щодо кількості подій, створених на Facebook. У представленому часовому ряді чітко простежується кілька сезонних ефектів: тижневі та річні цикли, а також виражений спад у період Різдва та Нового року. Ці типи сезонних ефектів природно збільшуються або зменшуються і, очікувано, змінюються у тимчасових рядах, створених людськими діями. Тимчасовий ряд також показує явну зміну тенденції за останні шість місяців, що може виникати у тимчасових рядах, на які впливають нові продукти чи зміни на ринку. Зрештою, реальні набори даних часто мають деякі зміщення норми, і цей тимчасовий ряд не є винятком.



Цей тимчасовий ряд є корисною ілюстрацією труднощів у створенні обґрунтованих прогнозів за допомогою повністю автоматизованих методів. На рис. 2.2 показані прогнози з використанням кількох автоматизованих процедури з пакета прогнозування R, описаного в Hyndman et al. (2007). Прогнози були зроблені в трьох точках часу, кожна з яких використовувала лише частину тимчасового ряду до цього моменту, щоб імітувати складання прогнозу на дату. На малюнку представлені такі методи:

- `auto.arima` – підбирає ряд моделей ARIMA та автоматично вибирає найкращу з них;
- `ets` – підбирає набір моделей експоненційного згладжування та вибирає найкращу;
- `snaiive` – модель випадкового блукання, яка робить постійні прогнози з щотижневою сезонністю (`seasonal naive`);
- `tbats` – модель TBATS з тижневою та річною сезонністю.

Методи представлені на рис. 2.2, як правило, не можуть дати прогнози, які відповідають характеристикам цих тимчасових рядів. Автоматичні прогнози ARIMA схильні до великих помилок тренду, коли відбувається зміна тренду поблизу періоду відсікання, і вони не можуть вловити будь-яку сезонність. Прогнози експоненційного згладжування та сезонного наївного прогнозу вловлюють щотижневу сезонність, але дають не точні результати при більш довгостроковій сезонності. Окрім того, усі методи реагують понад мірою на період перед кінцем року.

При виникненні нечіткого прогнозу, виникає потреба мати можливість налаштувати параметри методу відповідно до поставленого завдання. Налаштування цих методів потребує глибокого розуміння того, як працюють базові моделі часових рядів. Першими вхідними параметрами для автоматизованої ARIMA наприклад, є максимальні порядки диференціювання, компоненти авторегресії, та компоненти ковзного середнього. Зазвичай типовий аналітик не знатиме, як саме відрегулювати ці методи щоб уникнути поведінки, показаної на рис. 2.2 – це той тип знань, який важко масштабувати.

Саме тому, використовується методологія описана в пакеті prophet.

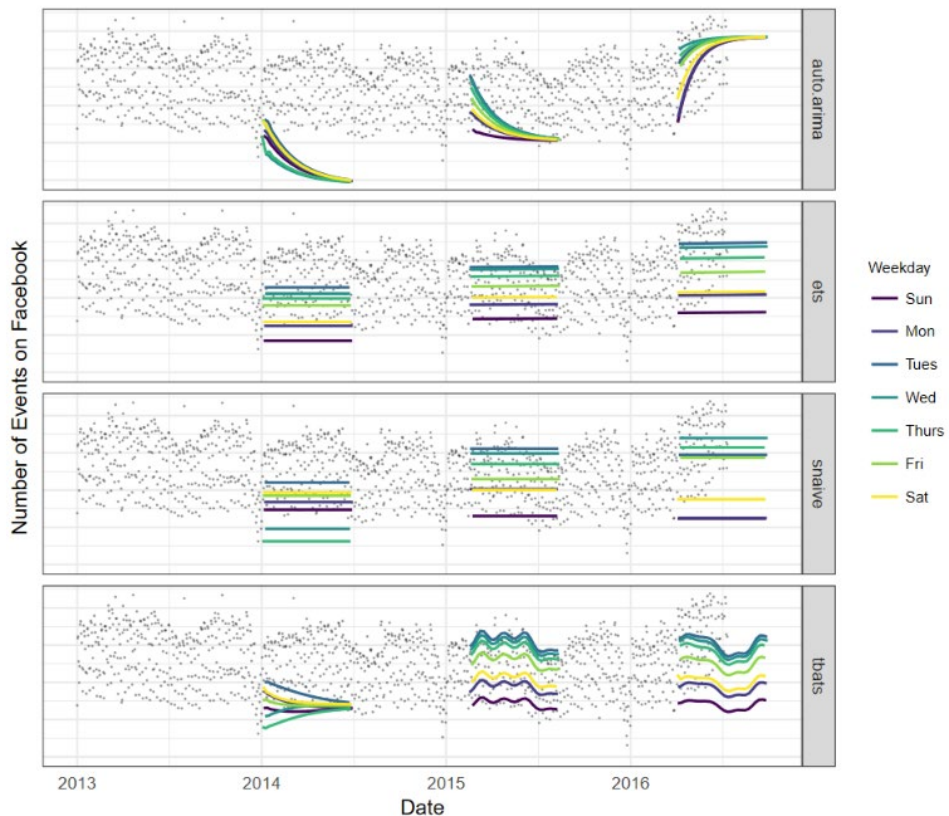


Рис. 2.2. Прогнозування часових рядів

2.2 Стаціонарні часові ряди

Стаціонарний процес - це стохастичний процес, який не змінює розподіл ймовірності при зміщенні в часі. Для розуміння стаціонарності, необхідно визначити, що це процес, який з плином часу не змінюватиме власних характеристик статистичного стану, тобто не буде змінюватись гомоскедатичність процесу (сталість дисперсії та мат. сподівання) і незалежність коваріаційної функції від часу (залежати може тільки від відстані між спостереженнями). В більшості статистичних процедур з використанням часових рядів, в основі лежить саме стаціонарність, тому нестаціонарні процеси можуть зводитись до стаціонарних без проблем.

Будь-який нестаціонарний процес, не маючий трендової поведінки, має назву цикло-статичного процесу, який інакше носить назву стохастичного

процесу, що може змінюватись з часом циклічно. Згідно робіть Шей Пелехі, з часових рядів можливо виділити наступне:

$$(\varepsilon_t, t \in T \subseteq \mathbb{R})$$

це – процес, що виник випадково та визначений на певному ймовірному просторі (Ω, \mathcal{A}, P) , називається 'стаціонарним в широкому сенсі', якщо

$$\forall_e \in T \subseteq \mathbb{R}$$

має наступні властивості:

— $\exists M \times \varepsilon_t$ та $\exists D\varepsilon_t = 0$

— Функція математичного сподівання постійна та не залежить від часу t ;

— Коваріаційна функція функціонально залежить лише від різниці аргументів

$$\text{cov}(\xi_t, \xi_s) = K(t, s) = \hat{K}(t - s), \forall s \in T.$$

Простим випадком стаціонарного процесу є білий шум. Наочно можливо подивитися на ці властивості на ілюстраціях представлених на рис. 2.3.

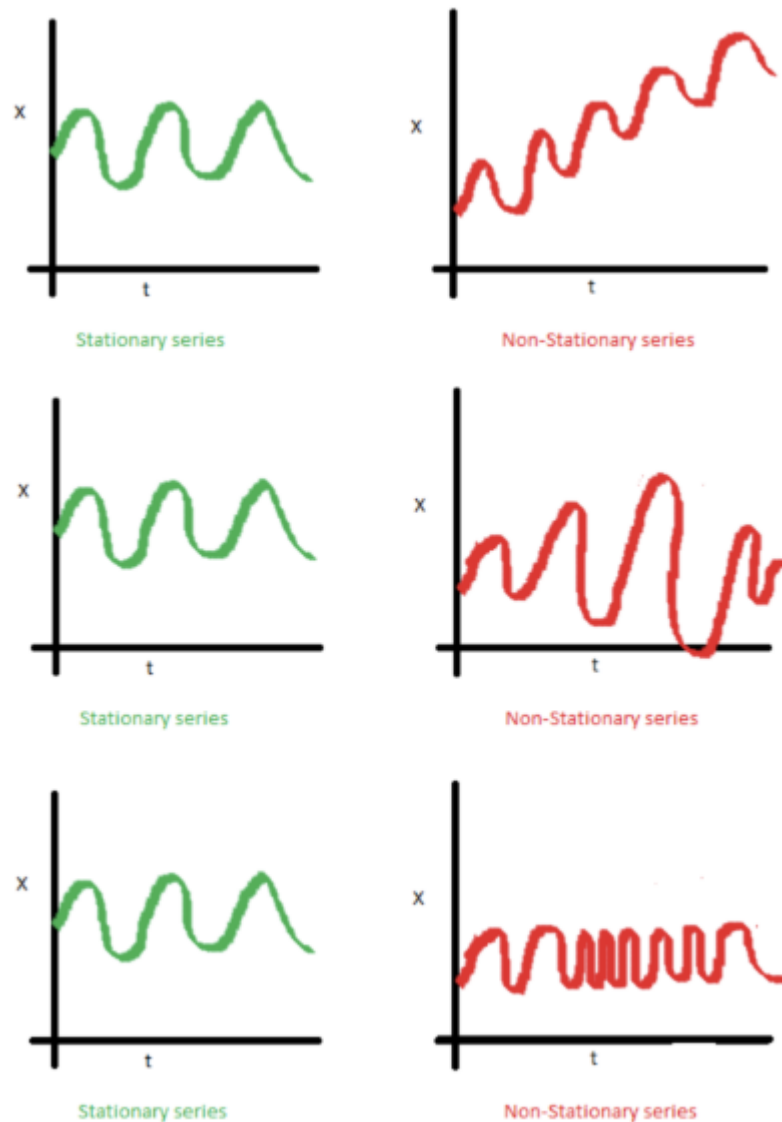


Рис. 2.3. Приклади стаціонарних(зліва) та нестаціонарних(зправа)

У першому випадку значення ряду розборсано в залежності від періоду. В другому ж рядкові значення незвично наближуються одне до одного, створюючи з себе певний кластер, в результаті чого отримується мінливість коваріацій. Також в останньому випадку часовий ряд з права не можливо назвати стаціонарним, адже його математичне сподівання тільки зростає.

Так як для побудови прогнозу використовується саме стаціонарність процесу, вона є важливою частиною системи, адже при подальшому аналізі даних його майбутні значення не будуть відрізнятися від поточних. Більша частина моделей часових рядів, в тій чи іншій мірі аналізують та прогнозують подальші результати (на кшталт математичного сподівання або дисперсії), тому якщо вихідний часовий ряд виявиться нестаціонарним результат може бути спрогнозовано невірно. Слід зазначити, що майже усі часові ряди, з якими доводиться працювати не є стаціонарними, але при необхідності це можливо обійти.

2.3 Методи і моделі для прогнозування з використанням часових рядів

2.3.1 Просте експоненційне згладжування

В основі методів експоненційного згладження виконується процес отримання спадних величин з попередніх спостережень часового ряду. Більша вага при цьому присвоюється тим спостереженням, які отримано пізніше. Розглядаючи роботу методу, можливо зрозуміти принцип присвоєння більшої ваги спостереженню п'ятиденної давнини, ніж спостереженню двох недільної давнини. Класифікація методів експоненційного згладження була об'єднана у загальну класифікацію у 1985 році. Вигляд моделі часового ряду має наступне описання:

$$X_t = b + \varepsilon_t$$

де b - константа, ε - випадкова помилка. Константу b можливо назвати стабільною на приведеному часовому ряду, проте з часом вона може повільно змінити значення.

Для описання компоненту b зручно використовувати метод згладження рухомого середнього. Методологія схожа на описану раніше. Останнім отриманим спостереженням надають більшої ваги, ніж передостаннім і так далі. Саме це є основою роботи принципу. Кожне давнє спостереження отримує також

давні значення ваги, проте на відміну від методу змінного середнього, при присвоєні враховують усі попередні спостереження часового ряду, не обмежуючись окремим часовим вікном. Формула простого експоненціального згладжування представлена у наступному вигляді:

$$S_t = \alpha X_t + (1 - \alpha)S_{t-1}$$

При використанні формули методу у рекурсивному вигляді, кожний новий отриманий прогноз отримує значення зваженого середнього поточного значення і згладженого ряду. Саме від параметра α у формулі отримується згладження. При значенні $\alpha = 1$, попередні отримані спостереження не будуть враховані. Якщо значення α дорівнює 0, будуть пропущені поточні отримані значення. Значення між 0, 1 дають проміжні результати.

Емпіричні дослідження надали досить точні результати прогнозу при використанні експоненційного згладжування.

Параметр α для отримання кращого значення оцінки використовує певні дані. При використанні на практиці параметр зазвичай визначають за допомогою створюваної сітки. На сітці можливі до використання параметри розбиваються з певним кроком віддаленості. Теоретичний вигляд сітки може мати наприклад значення від 0.01 до 0.99 та мати крок віддаленості, який дорівнює 0.07. Створенню сітку чередує визначення суми середніх квадратів залишків (спостережувані значення мінус прогнози на крок вперед), для яких значення є мінімальними. Іншими словами, обирається краща модель за результатами метрики.

2.3.2 Подвійне експоненційне згладжування

Порівнюючи просте експоненційне згладжування та подвійне виділяється недолік першого. Прогнозування лише на одиницю вперед часто не надає необхідних можливостей. Подвійне експоненційне згладжування в цьому випадку переважає та надає можливість прогнозування на дві одиниці вперед.

Для отримання необхідного зсуву часовий ряд розбивається на дві складові – рівень (level – l) і тренд (trend – b). В попередньому методі використовувався прогноз рівня, в поточному методі додається експоненційне згладження тренду. Для згладження тренду, до уваги беруться до уваги зважені значення попередніх змінних. В результаті отримується набір функцій:

$$l_x = \alpha y_x + (1 - \alpha)(l_{x-1} - b_{x-1})$$

$$b_x = \beta(l_x - l_{x-1}) + (1 - \beta)b_{x-1}$$

$$\hat{y}_{x+1} = l_x + b + x$$

Перша функція відображає рівень. Значення рівня залежить від поточного значення ряду, а другий доданок отримує значення попереднє значення рівня та тренду. Значення другої функції відображає тренд. Тренд в свою чергу, залежить від зміни значень на поточному кроці виконання та від попереднього значення. Коефіцієнт β в даному випадку відіграє роль ваги в експоненційному згладжуванні. Третя формула – це сума модельних значень тренду і рівня.

Надалі виступає налаштування двох параметрів – α і β . Перший параметр призначений за згладження часового ряду біля тренду, другий параметри згладжує часовий ряд тренду. Більші значення параметрів вказують на більшу перевагу останнім спостереженням ряду, з чого слідує зменшення плавності модельного ряду.

2.3.3 Потрійне експоненційне згладжування

Головна відмінність цього методу від попередніх в доданні ще однієї, третьої компоненти. Виходячи з цього, метод можливо використовувати, коли часовий ряд сезонності має достатньо якісних даних. Сезонна компонента моделі відповідає за опис коливань, що відносяться до рівня і тренду, а головною характеристикою виступатиме довжина сезону, або період, завершення котрого знаменується повторними коливаннями. Компоненти сезону отримують власне

значення відповідно до довжини сезону, тобто сезон складається з 7 днів тижня, формується 7 сезонних компонентів.

Отримується нова система:

$$l_x = \alpha(y_x - s_{x-L}) + (1 - \alpha)(l_{x-1} + b_{x-1})$$

$$b_x = \beta(l_{x-1} + b_{x-1}) + (1 - \beta)b_{x-1}$$

$$s_x = \gamma(y_x - l_x) + (1 - \gamma)s_{x-L}$$

$$\hat{y}_{x+m} = l_x + mb_x + s_{x-L+1+(m-1)\text{mod}L}$$

Визначення рівня тепер розраховується з використанням значення ряду від якого віднімають значення сезонної компоненти, тренд зміни не підлягає, сезонна компонента змінюється залежачи від поточного значення ряду без додання рівня та від попередніх значень компоненти. Компоненти часового ряду проходять згладження використовуючи всі доступні сезони, наприклад усі середи будуть згладжуватись за усіма іншими середами. На даний момент, використовуючи сезону компоненту отримується можливість прогнозувати вже більше ніж на один чи два кроки, прогнозування ведеться на довільну кількість кроків.

2.3.4 Моделі з урахування сезонності і тренду

Нові та більш комплексні моделі використовують тренд та сезонну компоненту на відміну від звичайного простого експоненційного згладжування. Головна перевага подібних моделей, в можливості обчислювати прогнози залежачи не тільки від попередніх спостережень, а й заздалегідь визначеними затримками, що дозволяють якісно проводити оцінку і сезонну складову. Аналіз та класифікація різних моделей відбулась у 1985 році.

В сезонності можливо визначити наступні елементи:

- відсутність сезонності;
- адитивна сезонність;

- мультиплікативна.

В категорії тренду можна розділити на наступні групи:

- відсутність тренду;
- лінійний тренд;
- рухливий тренд;
- демпфований

Більшість існуючих часових рядів в тому чи іншому вигляді мають серед компонент сезонність. Наприклад більшість ялинок продається саме у зимовий сезон. Тільки зимою виникає пік на часовому ряді, інша частина року не викликає таких коливань. Подібна сезонність виникає щороку, проте характер та розмір продажів може мати відмінності від року в рік. Виходячи з цього, при роботі з сезонною компонентою є сенс експоненційно згладити її, використовуючи додатковий параметр δ .

Сезонні компоненти поділяють на дві групи – адитивні та мультиплікативні. Цінові піки є зручним прикладом представлення різниці між групами. Продажі ялинок протягом січня збільшились на один мільйон гривень відносно попереднього року і продовжують постійне збільшення. Сезонне коливання можливо виразити, додавши до відносного прогнозу за місяць січень один мільйон гривень. Це адитивне представлення сезонності. У разі розділення піку вартості за певний місяць відносно всього року, буде представлено мультиплікативну групу.

Загалом, прогноз на один крок вперед обчислюється наступними формулами:

Адитивна модель:

$$P_{rt} = S_t + I_t - p.$$

Мультиплікативна модель:

$$P_{rt} = S_t * I_t - p,$$

де S_t – просте згладження значення часового ряду в момент t , а I_{t-p} – відповідає згладженому сезонному значенню в момент часу $t-p$. p у свій час являє довжину сезону. Отже, даний метод в порівнянні з простим експоненційним згладжуванням може поліпшити якість прогнозу завдяки використанню сезонної компоненти значень.

Компонента своїм чином оцінюється окремо з використанням простого експоненційного згладжування:

Аддитивна модель:

$$I_t = I_{t-p} + \delta * (1 - \alpha) * e_t.$$

Мультиплікативна модель:

$$I_t = I_{t-p} + \delta * (1 - \alpha) * e_t / S_t,$$

де сезонна компонента дорівнює компоненті останнього сезонного циклу в момент t плюс помилка (e_t – значення, що відображає зміни у між прогнозованим та спостереженим значенням в момент t). Параметр δ приймає значення між 0 і 1. При значенні рівному 0 сезонна складова залишається незмінною на наступному циклі. Якщо ж вона дорівнює 1 – сезонна складова через вплив помилки змінює власне значення на стільки, на скільки це можливо на кожному кроці. При прогнозуванні значень часового ряду з використанням сезонності, оптимальне значення δ буде знаходитись у границях між 0 та 1. Простий приклад сезонного згладження представлено на рис. 2.4.

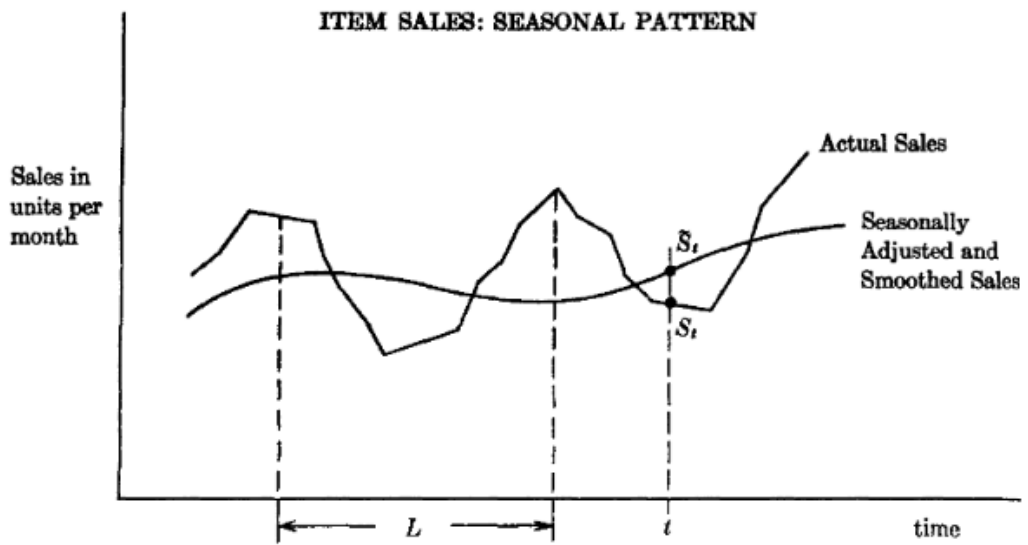


Рис. 2.4. Приклад сезонного згладження

Висновок до розділу 2

Результатом роботи над другим розділом дипломної роботи є безпосередній опис моделей та методів, завдяки яким можливо побудувати інтелектуальну систему прогнозування криптовалют.

3 МОДЕЛЮВАННЯ ТА ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Етапи збору та обробки даних

Перед початком побудови інтелектуальної системи прогнозування, було обрано потрібні для цього програмні інструменти.

Мовою програмування обрано R. Це одна з найпопулярніших мов, які використовують для статистичного аналізу. R — мова програмування, що інтерпретується, основним способом роботи з яким є командний інтерпретатор. Мова є реєстр залежною, у плані синтаксису вона схожа, з одного боку, на функціональні мови на кшталт Scheme, з іншого — на типові сучасні сценарні мови, з простим синтаксисом і невеликим набором основних конструкцій. Мова об'єктна: будь-який програмний об'єкт має набір атрибутів — іменований список значень, що визначають його.

Мова підтримує мінімальний набір примітивних типів даних: символічний (character), числовий (numeric), логічний (logical) та комплексний (complex). Числові змінні, крім звичайних чисел, можуть набувати спеціальних значень NaN (Not a Number — «не число») і Inf (Infinity — «нескінченність»). Нескінченність (позитивна або негативна) виходить при виході результату обчислень за межі діапазону, що надається реалізацією, NaN - при операціях з невизначеним результатом. Крім цих, є ще одне дуже важливе спеціальне значення, NA (Not Available - "не доступно"). Воно може бути використане для фіксації того факту, що відповідне значення, що бере участь у обчисленнях, з якоїсь причини не було отримано (досить звичайна у статистичних розрахунках ситуація, коли через збої у зборі даних деякі спостереження залишаються без результатів).

Значення примітивних типів можуть поєднуватися у вектори (vector), списки (list), матриці або масиви (matrix), у тому числі багатовимірні; ці

комбіновані типи зберігають набори даних одного й того самого примітивного типу. Крім цього, мова містить поняття факторів (factor) - наборів категоріальних або шкальних даних, що приймають строго певний набір значень. Нарешті, можуть створюватися таблиці (data frame) — структури даних, які кожної рядки (індивіда) зберігають набір різних (і мають різні типи) параметрів (ознак). Особливістю R є те, що операції з векторами та матрицями підтримуються на рівні самої мови, як, наприклад, APL.

Мова R була розроблена співробітниками статистичного факультету Оклендського університету Россом Айхекою (англ. Ross Ihaka) та Робертом Джентлменом (англ. Robert Gentleman) (перша літера їх імен - R), логотип якої представлено на рис. 3.1.; на поточний час мова та середовище підтримуються та розвиваються організацією R Foundation.



Рис. 3.1. Логотип мови програмування R

Середою розробки обрано популярну для мови R IDE RStudio.

RStudio — вільне середовище розробки програмного забезпечення з відкритим вихідним кодом мови програмування R, який призначений для статистичної обробки даних і роботи з графікою.

RStudio доступна у двох версіях: RStudio Desktop, в якій програма виконується на локальній машині як звичайна програма; та RStudio Server, в якій надається доступ через браузер до RStudio, встановленої на віддаленому Linux-сервері. Інтерфейс Desktop версії представлено на рис. 3.2.

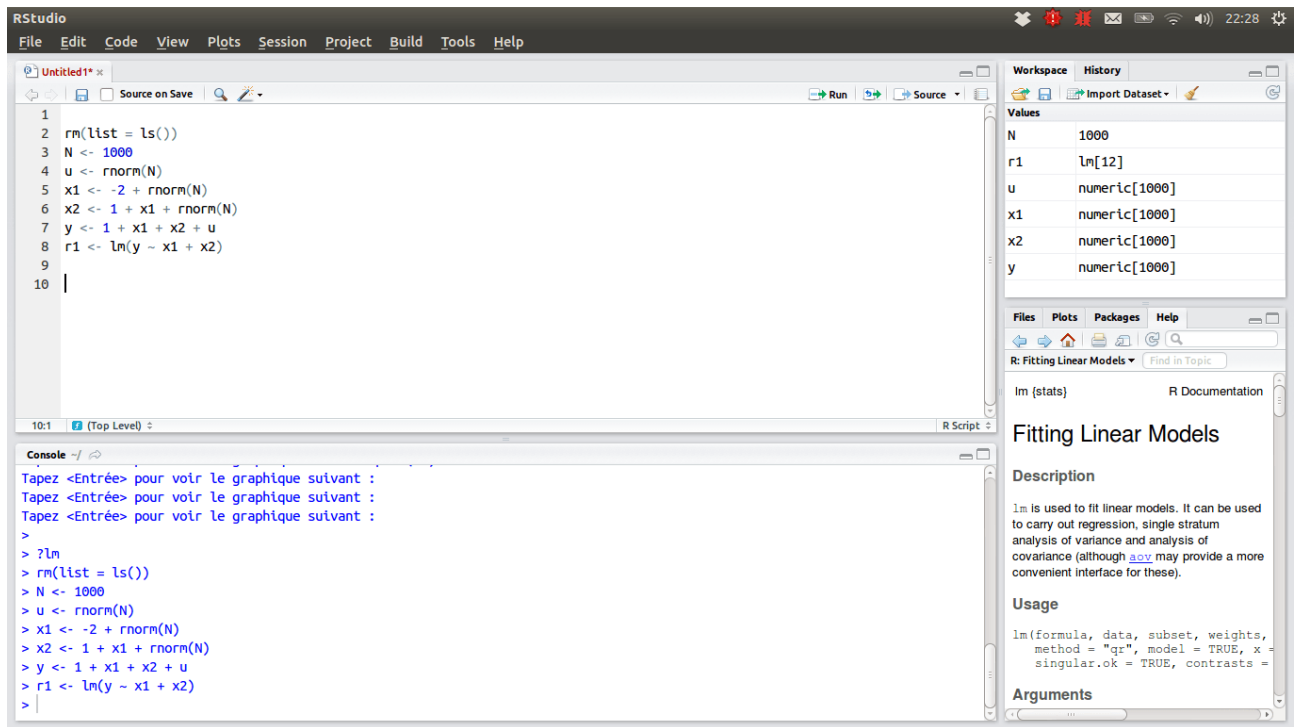


Рис. 3.2. Інтерфейс RStudio

Актуальну інформацію о крипто валютних індексах, зручно збирати з ресурсів агрегаторів подібних даних. Популярними прикладами є: CoinMarketCap, WorldCoinIndex та CryptoCompare. Подібні ресурси надають інформацію про актуальні ціни різних крипто валют, але також вони містять відомості для порівняння та моніторингу різних крипто валютних активів. Залежно від критеріїв індекс може містити інформацію про всі монети і токени, доступні на ринку, або лише деякі з них, наприклад, доступні на публічних біржах з достатнім рівнем обсягу торгів. Втім, здебільшого авторитетні крипто валютні індекси відстежують більшість існуючих монет, тож за їхньою

допомогою зручно стежити за станом ринку — не потрібно вручну компілювати дані сотень окремих обмінників.

В якості ресурса-агрегатора для даного проекту, обрано ресурс CoinMarketCap. Це один з найвідоміших у світі ресурсів, який надає різноманітну інформацію пов'язану з криптовалютами, яку представлено на рис. 3.3.

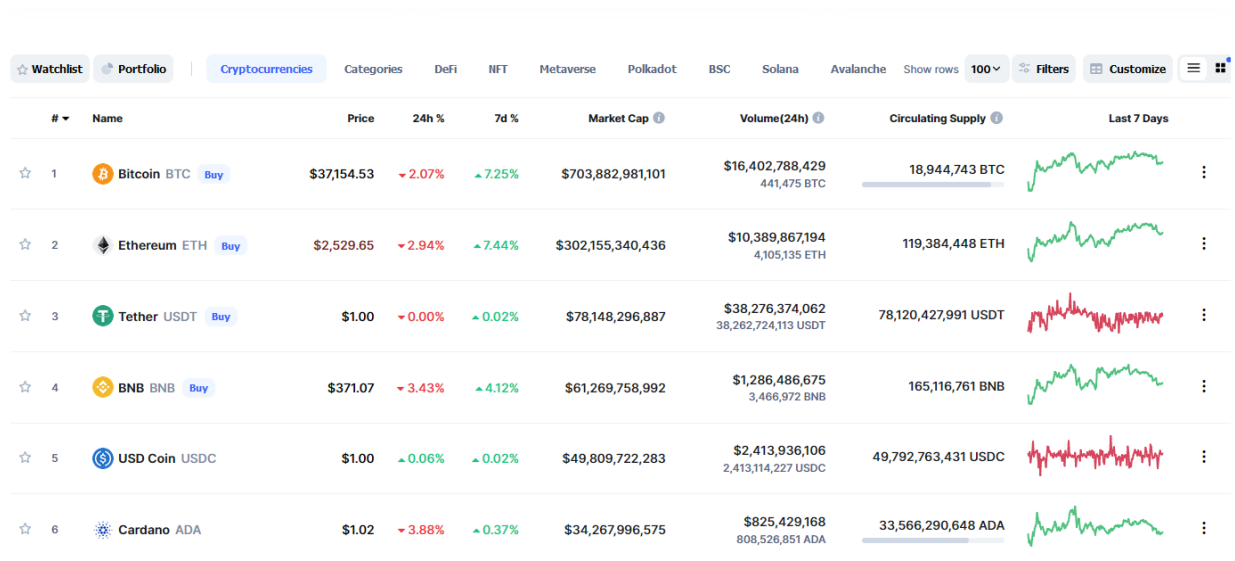


Рис. 3.3. Ресурс CoinMarketCap

Першим кроком, при початку роботи з криптовалютними індексами, є їх приведення до зручного для праці вигляду. Так як етапи розробки та подальшої роботи з системою є ітераційними, ми на постійній основі будемо повертатись до обраного набору даних. Тому приведення до зручного для праці вигляду, є одним з найважливіших кроків.

Представленні дані на мові R по стандарту оброблюються такими класами, як `ts`, `zoо` або `xts`. Але робота з цими класами, дуже часто приводить до зайвих проблем, які можуть виникнути під час проектування системи. Основними проблемами класів є нерегулярна реєстрацією спостережень у часі, наявністю кількох змінних різних типів, кількох групуючих змінних і т.д.

Для вирішення даної проблеми, були розроблені принципи “tidy data” (охайні дані). Принципи охайних даних забезпечують стандартний спосіб організації значень даних у наборі даних. Стандарт полегшує очищення початкових даних. Зникає необхідність щоразу починати з нуля і винаходити велосипед. Стандарт Tidy Data був розроблений для полегшення початкового дослідження та аналізу даних, а також для спрощення розробки інструментів аналізу даних, які добре працюють разом.

Охайні набори даних забезпечують стандартизований спосіб зв’язування структури набору даних (його фізичного розташування) з його семантикою (його значенням).

Для мови R існує пакет “tsibble”, який саме реалізує принципи tidy data. Основні переваги пакету можливо охарактеризувати наступним чином:

- дані зберігаються у табличному вигляді;
- у таблиці повинні бути присутніми як мінімум два стовпці — зі значеннями кількісної змінної, що спостерігається в часі, і з упорядкованими за зростанням (тобто від минулого до майбутнього) тимчасовими відмітками (стовпець з тимчасовими відмітками називається індексуєчим — index);
- в таблицю можуть входити одна або кілька групуючих змінних (key) - значення цих змінних вказують на належність кожного спостереження до відповідного тимчасового ряду;
- будь-яке спостереження в таблиці можливо унікально ідентифікувати за поєднанням значень індексуєчої та групуючих змінних.

Частковий набір криптовалютних індексів, які використано та приведено да зручного вигляду, представлено на рис. 3.4.

	A	B	C
1	y	ds	coin
2	7547	06.12.2019	bitcoin
3	7448,31	05.12.2019	bitcoin
4	7252,03	04.12.2019	bitcoin
5	7320,15	03.12.2019	bitcoin
6	7321,99	02.12.2019	bitcoin
7	7424,29	01.12.2019	bitcoin
8	7569,63	30.11.2019	bitcoin
9	7761,24	29.11.2019	bitcoin
10	7463,11	28.11.2019	bitcoin
11	7531,66	27.11.2019	bitcoin
12	7218,37	26.11.2019	bitcoin
13	7146,13	25.11.2019	bitcoin
14	7047,92	24.11.2019	bitcoin
15	7397,8	23.11.2019	bitcoin

Рис. 3.4. Набір криптовалютних індексів

Вже на даному етапі, отримується можливість виконати прості команди для перегляду даних з отриманого набору індексів. Так, наприклад з обраного набору даних, який містить 15510 елементів, ми можемо переглянути кількість існуючих часових рядів та динаміку зміни вартості, які відносяться до окремої криптовалюти (Bitcoin в даному прикладі). Результати представлені на рис. 3.5 та 3.6.

```
> glimpse(cryptos, width = 60)
Rows: 15,510
Columns: 3
$ y      <dbl> 7547.00, 7448.31, 7252.03, 7320.15, 7321.99, ~
$ ds     <date> 2019-12-06, 2019-12-05, 2019-12-04, 2019-12--~
$ coin   <chr> "bitcoin", "bitcoin", "bitcoin", "bitcoin", "~
>
```

Рис. 3.5. Кількість елементів у наборі даних

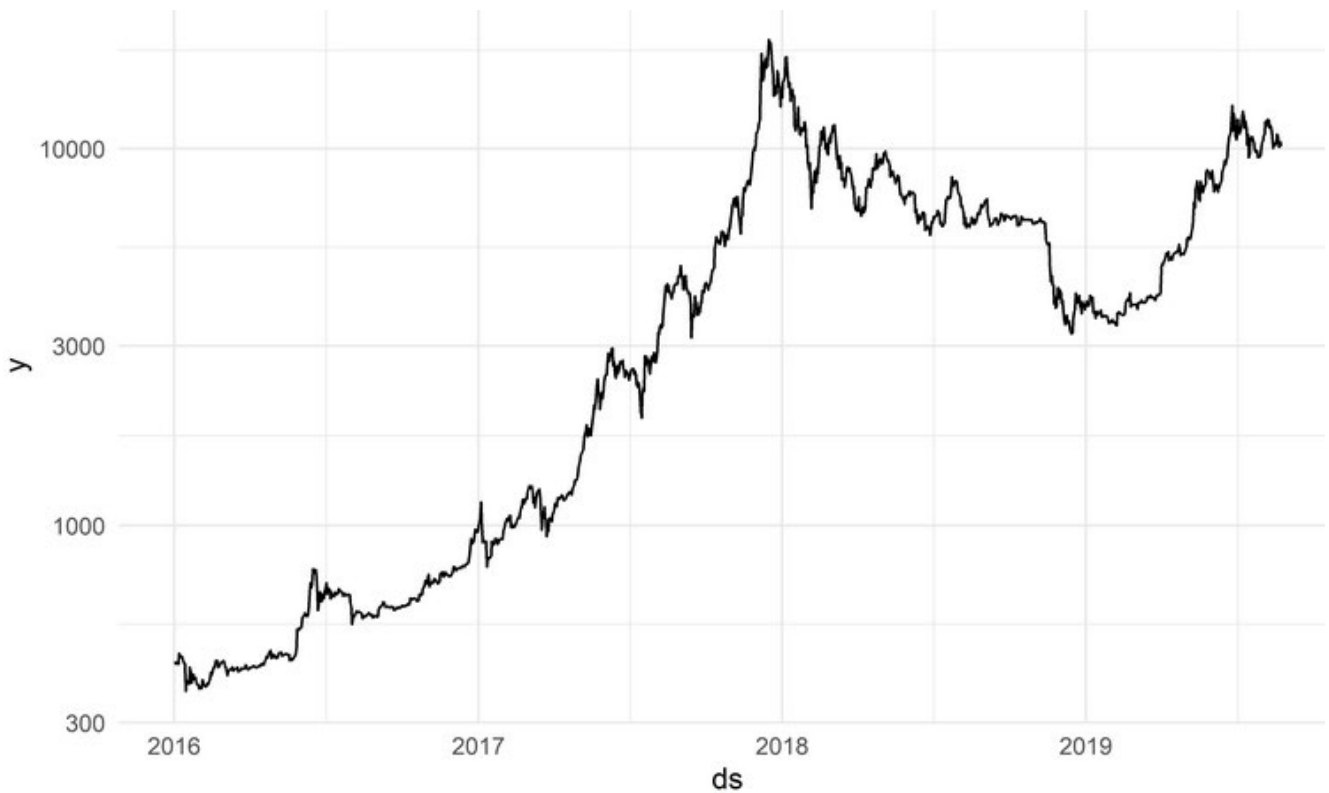


Рис. 3.6. Динаміка зміни вартості криптовалюти Bitcoin

Використовуючи дані отримані на момент закриття торгів криптовалют, також отримується можливість переглянути та проаналізувати, які саме криптовалюти займають головні позиції на ринку. Окрім того, графік відображає відсутність торгів у США під час святкових днів, що можливо побачити за допомогою пропущених ланок на графіку відображеному на рис. 3.7.

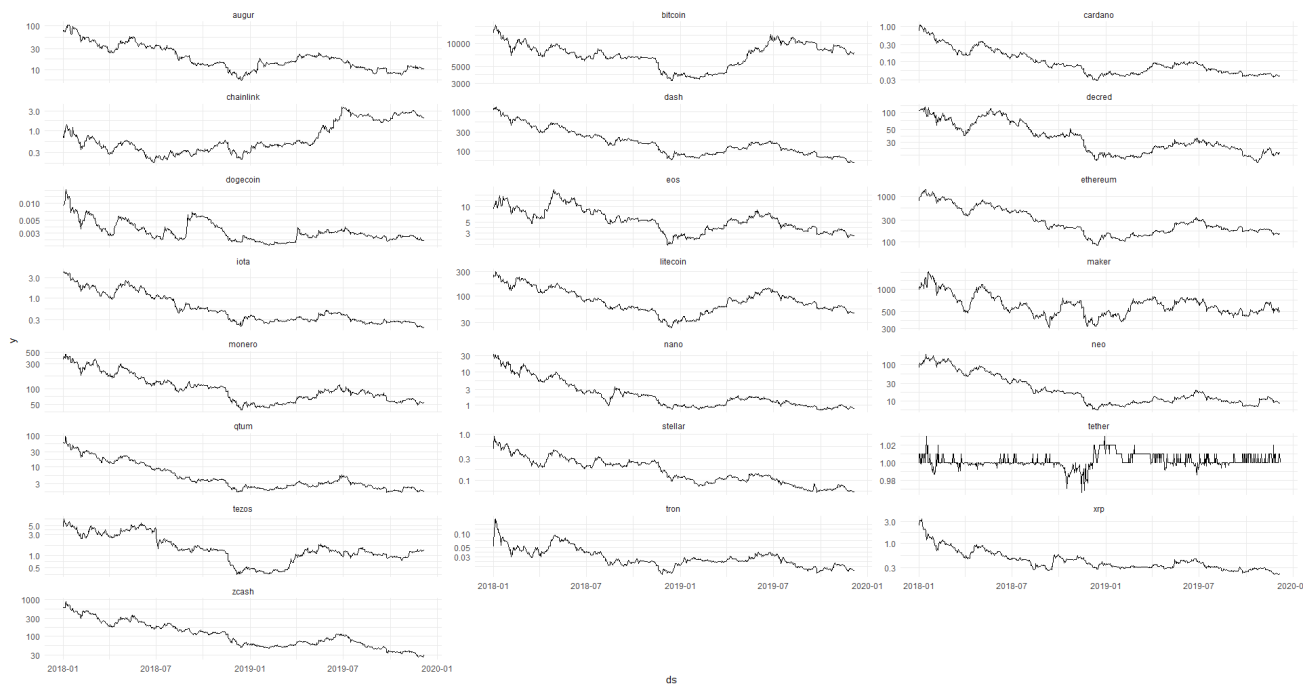


Рис. 3.7. Динаміка зміни криптовалют

Наступним кроком, йде видалення пропущених значень з набору даних. Дуже часто практично доводиться стикатися з пропущеними значеннями в часових рядах. Це може стати проблемою при використанні багатьох методів аналізу та моделювання, які передбачають наявність повних наборів упорядкованих у часі спостережень. Пакет `tsibble` надає ряд функцій, що дозволяють виконати діагностику даних на наявність пропущених спостережень.

Для виявлення і видалення пропущених та пустих значень необхідно виконати декілька етапів перевірки даних для виявлення потрібних діапазонів та значень. Даний проект використовує дані торгів з перевіреного ресурсу, який надає криптовалютні індекси у перевіреному та готовому до роботі вигляді. Описані кроки використовуються, лише для переконання та застереження від надзвичайного фактору.

Першим кроком при обробці даних потрібно виявити та видалити NA значення (NaN), тобто пропущені значення, які передусмотренні мовою

програмування R. На рис. 3.8 відображено, що обраний набір даних подібних значень не має.

```
> has_gaps(cryptos)
# A tibble: 22 x 2
  coin      .gaps
  <chr>    <lgl>
1 augur    FALSE
2 bitcoin  FALSE
3 cardano  FALSE
4 chainlink FALSE
5 dash     FALSE
6 decred   FALSE
7 dogecoin FALSE
8 eos      FALSE
9 ethereum FALSE
10 iota    FALSE
# ... with 12 more rows
```

Рис. 3.8. Перевірка на NA-значення

В випадку виявлення подібних значень, отримується можливість виявлення місць, де саме присутні пропущені значення. Наступним логічним кроком після виявлення подібних місць, йде їх заповнення для надання можливості подальшого аналізу даних та моделювання системи, так як більшість статичних методів для мови R не вміють працювати з NA значеннями. Для вирішення цієї проблеми виконується відновлення пропущених значень тим чи іншим доступним для даної ситуації способом. Популярним рішенням для подібної задачі, є використання стратегії LOCF (“last observation carried forward”), яка полягає в тому, що кожне пропущене значення замінюється на останнє попереднє йому непропущене значення (або на найближче наступне за ним непропущене значення, якщо ряд починається з NA). Принцип роботи стратегії представлено на схемі, зображеній на рис. 3.9.

Subject	Group	Baseline	Follow-up Week		
			1st month	3th month	6th month
1	1	296	175	187	192
2	1	376	329	236	76
3	1	150	150	150	150
4	2	282	186	225	134
5	2	317	31	85	120
6	2	362	104	104	104

Рис. 3.9. Схематичне представлення стратегії LOCF

Після виконаної обробки, набір даних вважається підготовленим для подальшого використання системою прогнозування криптовалют.

3.2 Етап агрегації даних

При побудові системи прогнозування, першим кроком, який необхідно зробити після отримання робочого набору даних – це побудова та вивчення графіків часових рядів. Це надає можливість обрати, які саме методи аналізу будуть використанні при подальшому моделюванні. Це перша ланка, яка надає можливість системі в наступних ітеративних етапах отримувати високоточні результати.

Часовий ряд можливо розкласти на наступні основні компоненти, завдяки яким проводиться аналіз:

— тренд (Tt): характеризує довгострокову тенденцію в даних (зниження або зростання). Тренд може бути лінійним або нелінійним. У деяких тимчасових рядах може також спостерігатися зміна напрямку тренду (наприклад, коли зростання змінюється спадом).

— циклічна компонента (Ct): довгострокові циклічні коливання, зазвичай займають не менше 2 років. Як правило, частота таких змін непостійна.

— сезонна компонента (St): короткочасні періодичні зміни, що володіють фіксованою частотою (наприклад, добові зміни кількості сонячного світла, що падає на одиницю поверхні Землі).

— нерегулярне компонента (Et): ефекти випадкових факторів ("шум").

Існує кілька методів декомпозиції часових рядів. Одним з найбільш широко використовуваних є розкладання на тренд і сезонну складову за допомогою локальної полиномиальної регресії. Графічний вигляд розкладання криптовалют Ethereum та XRP представлено на рис. 3.10 та 3.11.

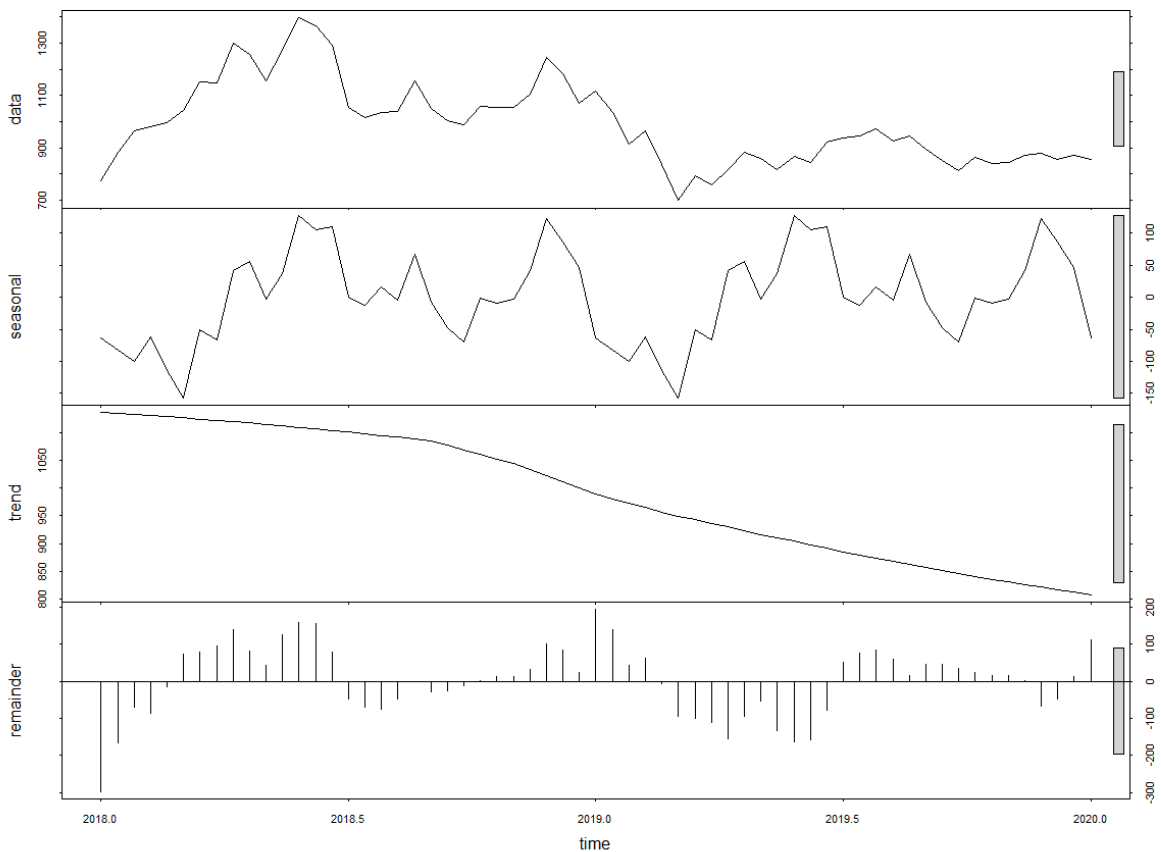


Рис. 3.10. Декомпозиція часового ряду Ethereum

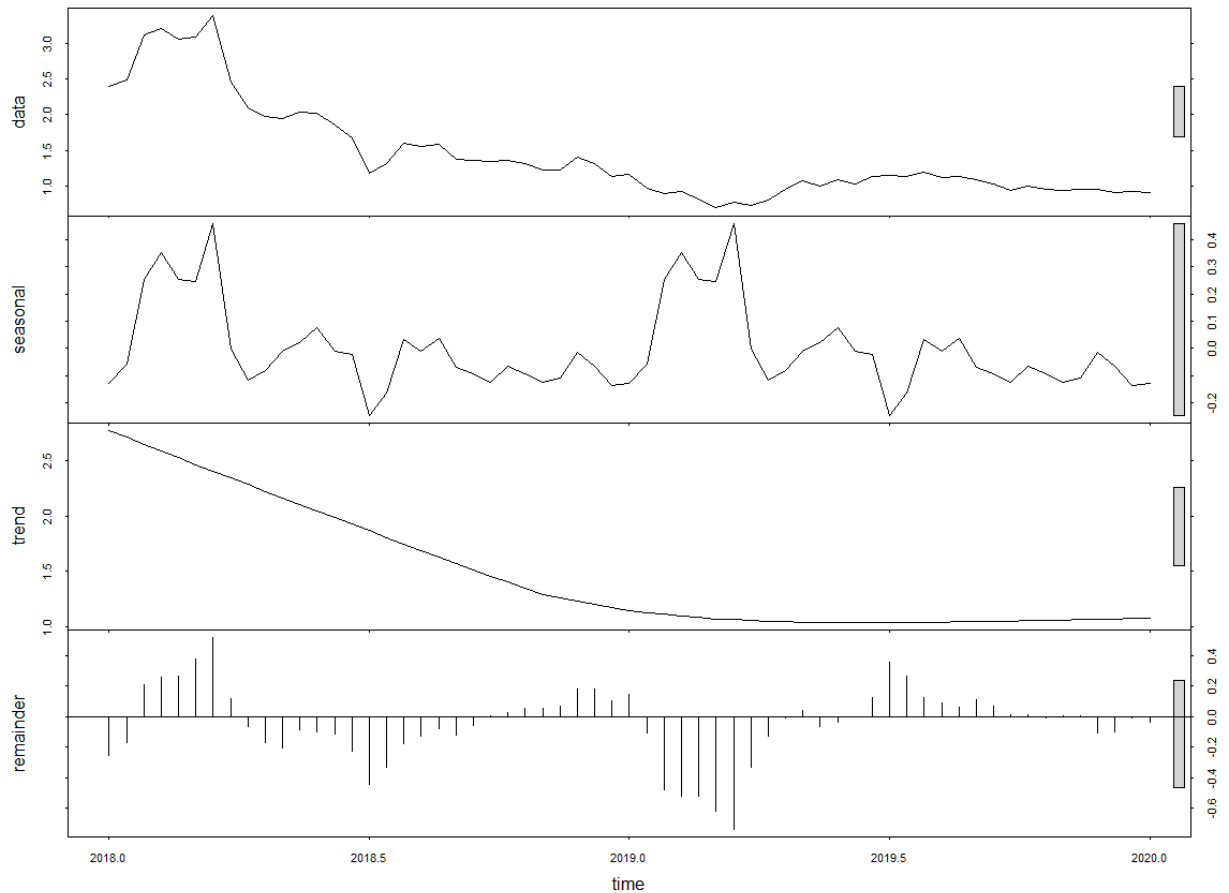


Рис. 3.11. Декомпозиція часового ряду XRP

Графіки представлені вище, відображають три компоненти часового ряду, а саме сезонність, тренд та залишок. З приведеної у другій главі формули принципів роботи часових рядів, отримується можливість перевірити точність даних. Якщо ми просумуємо усі компоненти, то отримує вихідні значення представлені у сročці data.

Виходячи з цього, визначаємо що часові ряди є нестационарними.

Для додаткового представлення інформації о трендах часових рядів, можливо представити дані використовуючи гістограми. Дані гістограми відображають тренд, на якому зображено, що ціна криптовалют збільшується і

зменшується майже на одну й ту саму величину, проте простежується саме спад вартості обох криптовалют.

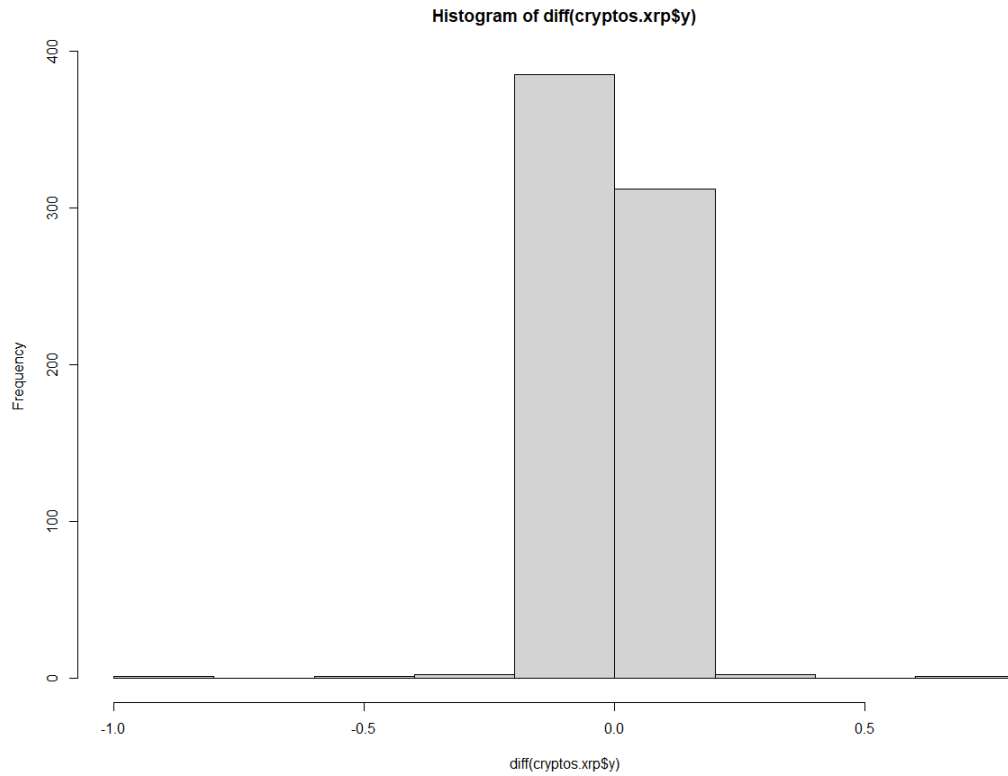


Рис. 3.12. Гістограма різниць XRP

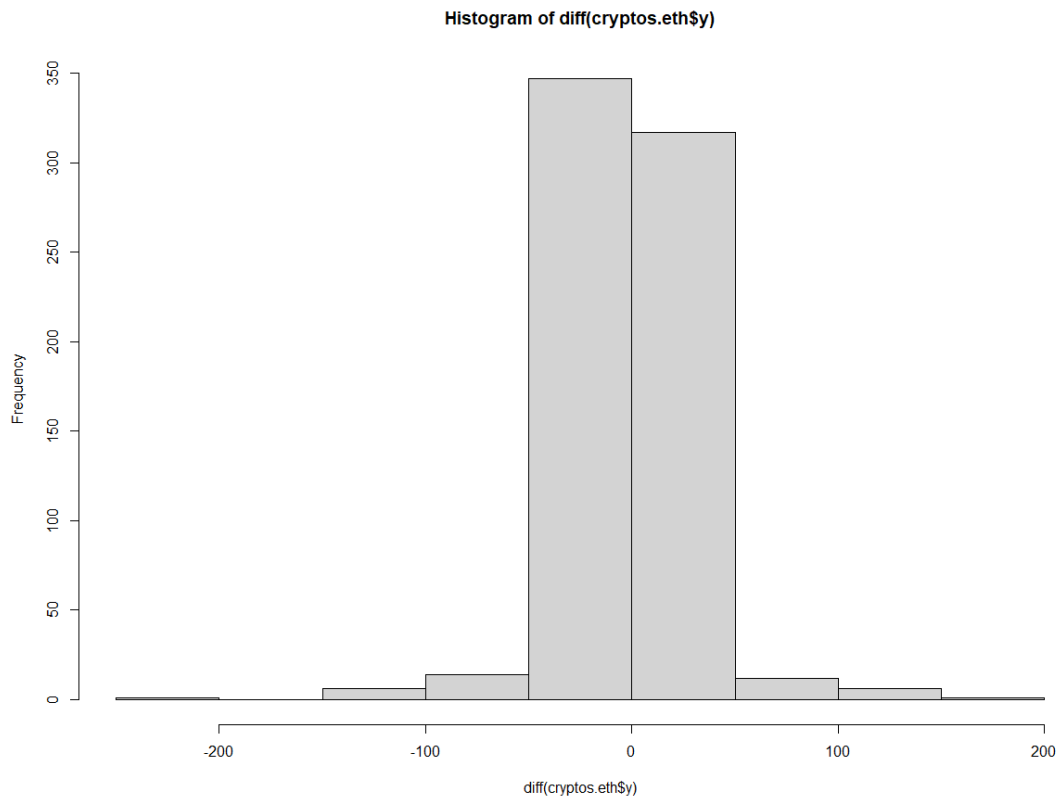


Рис. 3.13 Гістограма різниць Ethereum

При роботі з часовими рядами, основною задачею агрегації предстає сортування даних за певний календарний період, для подальшого використання в моделях прогнозування. Використовуючи описані раніше пакети `tsibble` та `dplyr`, часові ряди можливо привести до зручної форми відсортованої за потрібним періодом часу (день, місяць, квартал і т.д.). Приклад сортування криптовалютних індексів представлено на рис. 3.14, в даному прикладі, ведеться пошук середньомісячної вартості криптовалюти та кількості щомісячних оновлень даних криптовалюти. Приклад з використанням случайно обраної валюти представлено на рис. 3.15.

```
# A tibble: 528 x 4 [1M]
# Key:   coin [22]
  coin year_month avg_y   n
  <chr>   <mt> <dbl> <int>
1 augur 2018 янв  84.1   31
2 augur 2018 фев  51.1   28
3 augur 2018 мар  35.9   31
4 augur 2018 апр  32.5   30
5 augur 2018 май  45.8   31
6 augur 2018 июн  34.5   30
7 augur 2018 июл  31.9   31
8 augur 2018 авг  21.7   31
9 augur 2018 сен  14.7   30
10 augur 2018 окт  13.1   31
# ... with 518 more rows
```

Рис. 3.14. Приклад агрегації криптовалютних індексів



Рис. 3.15. Середньомісячна вартість криптовалюти Монеро

Часові ряди, окрім наведених вище властивостей, мають важливу характеристику – автокореляція значень. Автокореляція визначає ступінь залежності між сусідніми значеннями, які знаходяться на певній відстані, одна від одної, на «лаговій» відстані. Завдяки використанню автокореляції отримується можливість простежити зміну взаємовпливу окремих значень одна на одну. Саме ця характеристика, може бути використана для отримання точних та надійних прогнозів з використанням індексів криптовалют. На рис. 3.16 можливо простежити зв'язок між тимчасовим рядом оригіналом, та рядами копіями, які змінились через зсув. Даний графік відображає коефіцієнт зсуву значень криптовалюти Ethereum. Числове представлення коефіцієнту зображено на рис. 3.17.

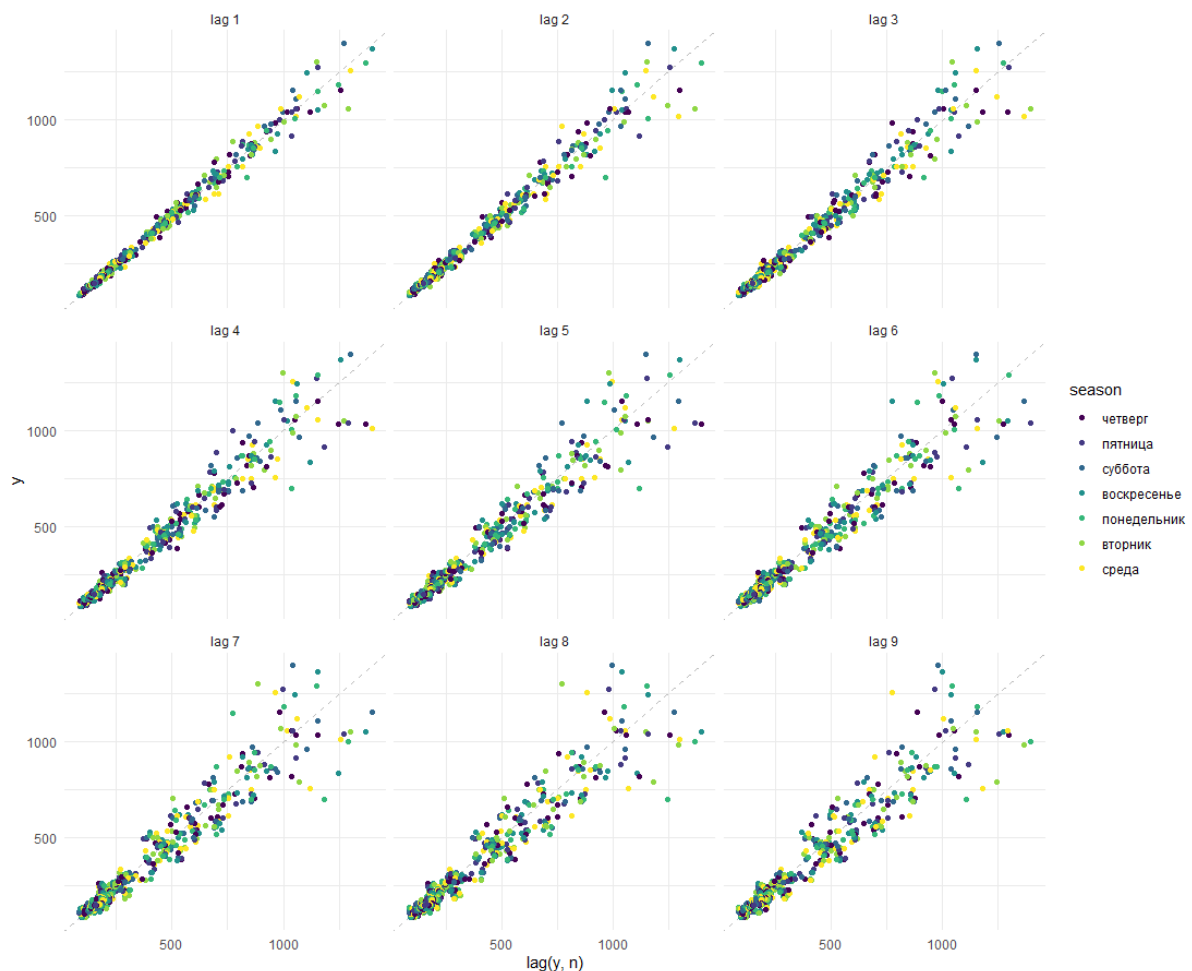


Рис. 3.16. Графічний коефіцієнт зсуву

```
# A tsibble: 6 x 3 [1D]
# Key:      coin [1]
  coin      lag  acf
  <chr>    <lag> <dbl>
1 ethereum 1D 0.993
2 ethereum 2D 0.984
3 ethereum 3D 0.975
4 ethereum 4D 0.965
5 ethereum 5D 0.956
6 ethereum 6D 0.945
> |
```

Рис. 3.17. Числове представлення коефіцієнту зсуву

Отримані дані можливо представити у вигляді корелограм часового ряду. Дані діаграми відображають в зручному вигляді коефіцієнт автокореляції. На діаграмі зображеній на рис. 3.18 можливо простежити, що коефіцієнт є позитивним та вказує на зростання тренду. Відображені пунктирною стрічкою значення є критичними. Так як отримані дані більше критичних, то кореляції вважаються значно більшими від нуля.

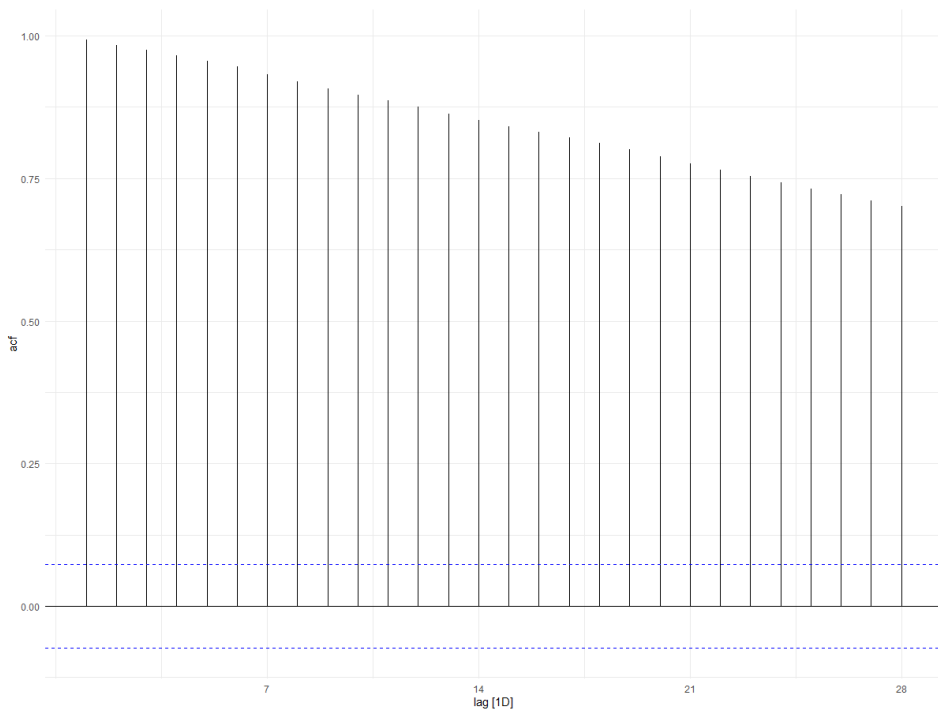


Рис. 3.18. Корелограма часового ряду криптовалюти Ethereum

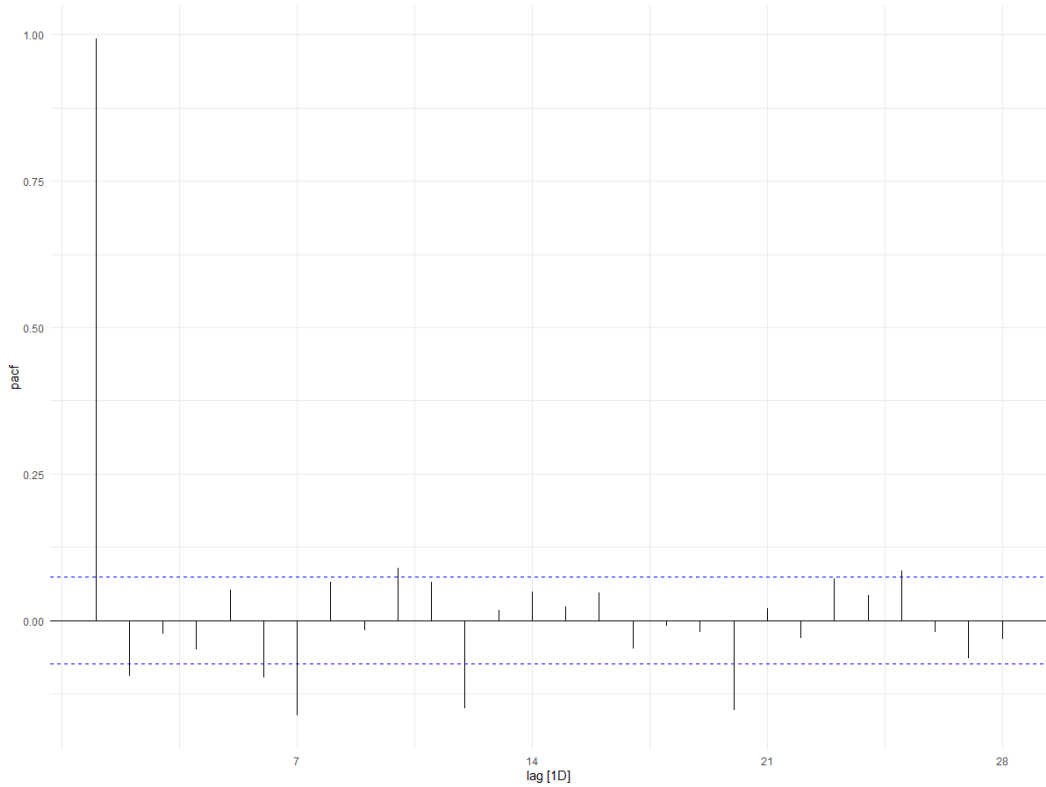


Рис. 3.19. Корелограма часткової автокореляційної функції

Фінальними кроками даного етапу моделювання системи можливо назвати остаточне тестування додатковими тестами. Перший тест – тест Дікі-Фуллера, результат якого представлено на рис. 3.20. Головна ідея тесту залежить у перевірці стаціонарності та тенденції стаціонарності часового ряду.

Якщо значення $p > 0,05$ під час тесту ADF (доповнений Дікі-Фуллер) часового ряду, тоді серія називається нестаціонарною і приймає гіпотезу NULL. Якщо значення $p \leq 0,05$, воно відхиляє гіпотезу NULL, яка символізується як H_0 , і вона називається стаціонарною, коли дані не мають одиничного кореня. Альтернативна гіпотеза символізується як H_1 . З результату виконання тесту можна сказати, що часовий ряд нестаціонарний, так як $p=0.38 > 0.05$.

```
> tseries::adf.test(tsData[,1])  
  
      Augmented Dickey-Fuller Test  
  
data: tsData[, 1]  
Dickey-Fuller = -2.4623, Lag order = 3, p-value = 0.3879  
alternative hypothesis: stationary  
  
> |
```

Рис. 3.20. Тест Дікі-Фуллера

Другий тест – тест KPSS. В економетриці тести Квятковського–Філіпса–Шмідта–Шина (KPSS) використовуються для перевірки нульової гіпотези про те, що спостережуваний часовий ряд є стаціонарним навколо детермінованої тенденції (тобто стаціонарної тенденції) проти альтернативи одиничного кореня. Результат на рис. 3.21.

На відміну від більшості тестів одиничного кореня, наявність одиничного кореня є не нульовою гіпотезою, а альтернативою. Крім того, у тесті KPSS відсутність одиничного кореня є доказом не стаціонарності, а, за задумом, стаціонарності тренду. Це важлива відмінність, оскільки часовий ряд може бути нестаціонарним, не мати одиничного кореня, але бути стаціонарним за трендом.

Як в одиничних кореневих процесах, так і в стаціонарних процесах із трендом середнє може зростати або зменшуватися з часом; однак за наявності шоку стаціонарні тенденційні процеси повертаються до середнього (тобто тимчасові, часові ряди знову зближаться до зростаючого середнього, на яке не вплинув шок), тоді як процеси з одиничним коренем мають постійний вплив на середнє (тобто відсутність збіжності в часі).

```
> kpss.test(tsData[,1])  
  
KPSS Test for Level Stationarity  
  
data: tsData[, 1]  
KPSS Level = 1.2073, Truncation lag parameter = 3, p-value = 0.01
```

Рис. 3.21. Тести Квятковського – Філіпса – Шмідта – Шіна

У статистиці тест Філіпса – Перрона (названий на честь Пітера С. Б. Філіпса та П'єра Перрона) є одиничним тестом. Тобто він використовується для аналізу часових рядів для перевірки нульової гіпотези про те, що часовий ряд інтегрований із порядку. Результат на рис. 3.22.

```
> tseries::pp.test(tsData[,1])  
  
Phillips-Perron Unit Root Test  
  
data: tsData[, 1]  
Dickey-Fuller Z(alpha) = -7.3083, Truncation lag parameter = 3, p-value = 0.6827  
alternative hypothesis: stationary  
  
> |
```

Рис. 3.22. Тест Філіпса – Перрона

3.3 Етап навчання моделей та прогнозування

автоматичному режимі, що значно зменшує можливість виникнення непередбачуваних результатів.

Для ілюстрації роботи розробленої інтелектуальної системи прогнозування на основі індексів криптовалют обрано криптовалюту Ethereum. Система може працювати з будь-якою криптовалютою, Ethereum обрано у випадковому порядку для зручності демонстрування результатів роботи.

Вартість криптовалют – не найпростіша змінна для моделювання (що справедливо для переважної більшості фінансових тимчасових рядів). Подібний ряд має складний тренд, дисперсія значень зростає з часом, мають місце різкі зміни рівнів, ймовірно пов'язані з якимись (у більшості випадків невідомими) особливими подіями.

Для прикладу буде змодельовано випадок, коли необхідно отримати прогноз вартості криптовалют на наступні 90 днів. Наведений у лістингу 1 код виконує підготовку даних для побудови подібної моделі: спочатку відбувається логарифмування значень вартості криптовалюти (для зниження дисперсії), а потім розбиття вихідної вибірки на навчальну (всі спостереження за винятком останніх 90 днів) та перевірочну (останні 90 днів).

Лістинг 1

```
ethereum_train <- cryptos.eth %>%  
  mutate(y = log(y)) %>%  
  slice(1:(n() - 90)) %>%  
  as.data.frame()  
  
ethereum_test <- cryptos.eth %>%  
  mutate(y = log(y), ds = as.Date(ds)) %>%  
  tail(90) %>%  
  as.data.frame()
```

Відокремлені навчальна та перевірочна вибірки даних по криптовалюти Ethereum представлена на рис. 3.23 та 3.24.



Рис. 3.23. Навчальна вибірка даних Ethereum

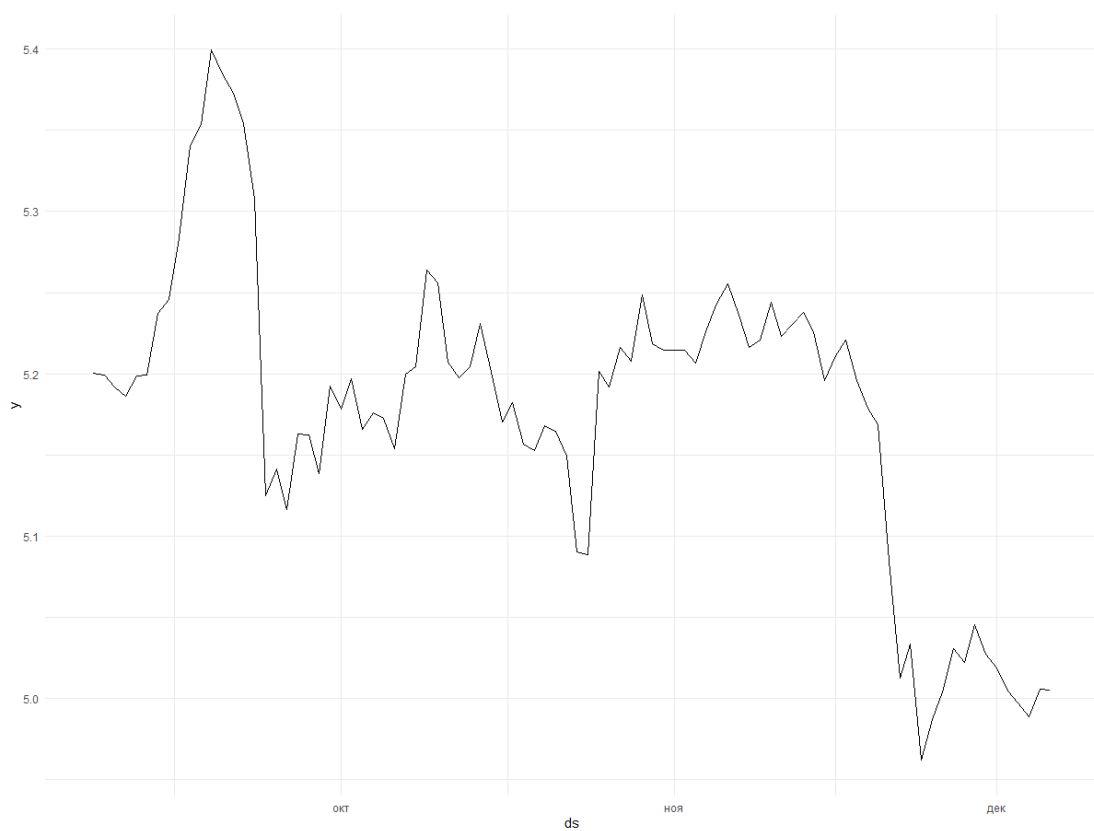


Рис. 3.24. Перевірочна вибірка даних Ethereum

Підготувавши навчальну та перевірочну вибірки починається процес побудови моделей прогнозування. Модель prophet представляє собою великий список змінних, які відповідають за прогнозування, таких як сезонність, свята і т.д. Частковий вигляд моделі представлено на рис. 3.25.

```
List of 32
$ growth : chr "linear"
$ changepoints : POSIXct[1:25], format: "2018-01-21" "2018-02-09" "2018-03-01" "2018-03-21" ...
$ n.changepoints : num 25
$ changepoint.range : num 0.8
$ yearly.seasonality : chr "auto"
$ weekly.seasonality : chr "auto"
$ daily.seasonality : chr "auto"
$ holidays : NULL
$ seasonality.mode : chr "additive"
$ seasonality.prior.scale: num 10
$ changepoint.prior.scale: num 0.05
$ holidays.prior.scale : num 10
$ mcmc.samples : num 0
$ interval.width : num 0.8
$ uncertainty.samples : num 1000
$ specified.changepoints : logi FALSE
$ start : POSIXct[1:1], format: "2018-01-01"
$ y.scale : num 7.24
$ logistic.floor : logi FALSE
$ t.scale : num 5.3e+07
$ changepoints.t : num [1:25] 0.0326 0.0635 0.0961 0.1287 0.1596 ...
```

Рис. 3.25. Модель M0

Після створення моделі, починається процес навчання. Модель представляє собою звичайну таблицю, в якій зберігаються значення кількох розрахованих на основі моделі M0 величин, зпрогнозовані значення відгуків, а також верхні та нижні межі крайніх інтервалів відповідних величин.

Зпрогнозовані результати моделі M0 відображені у графічному вигляді на рис. 3.26.

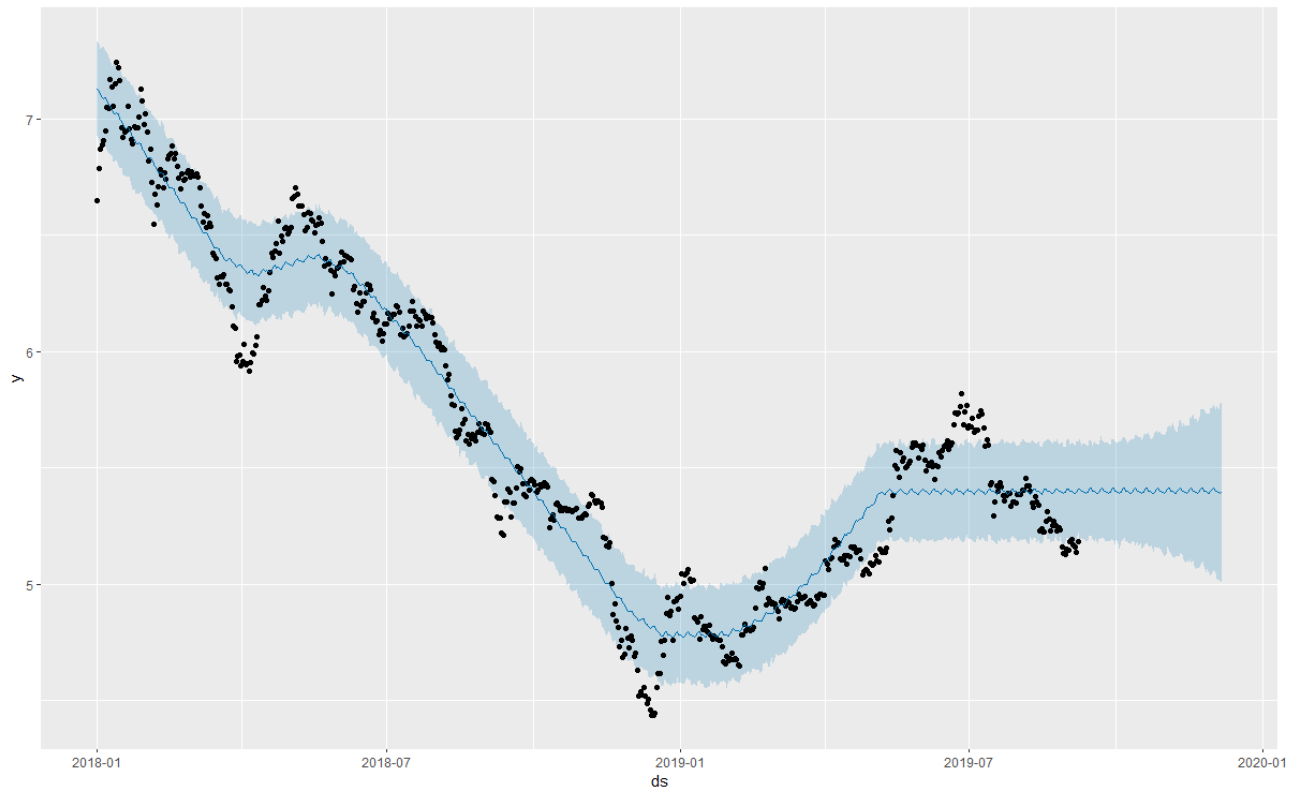


Рис. 3.26. Спрогнозована модель M0

Точки на представленому графіку відповідають (логорифмічним) значенням вартості Ethereum з навчальної вибірки. Суцільна синя лінія - це передбачені моделлю значення вартості, а світло-блакитна "стрічка", що огинає лінію позначає 80% довірчі границі передбачених значень. Прогнозні значення у на наступні 90 днів видно в правій частині графіка (після 2020-01).

З графіку видно, що прогноз незадовільний, так як він недостатньо точно передає структуру вибірки. Для покращення результатів можливо змінити навчальний термін з 90 до 240 днів. Результат представлено на рис. 3.27.

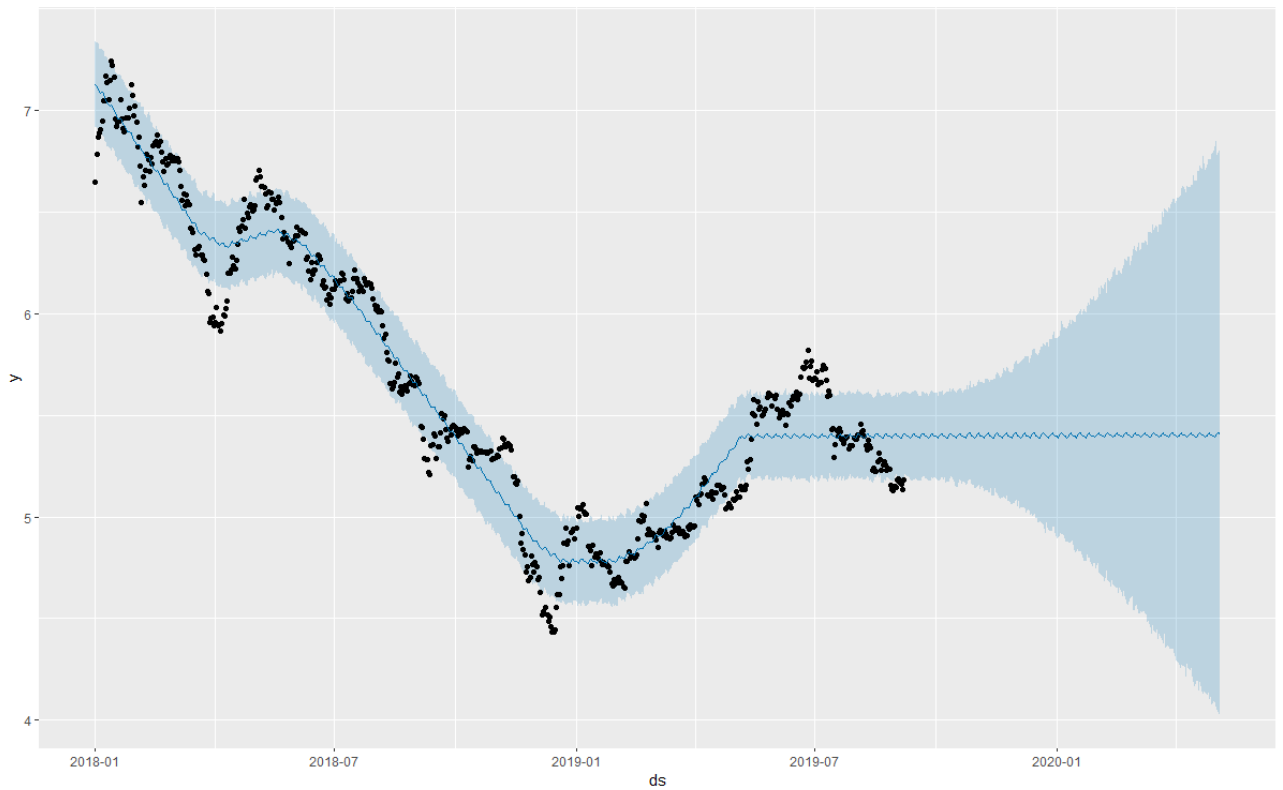


Рис. 3.27. Новий прогноз моделі M0

Для більш детального аналізу можливо відобразити на рис. 3.28 окремі компоненти моделі, на яких базується дана методологія.

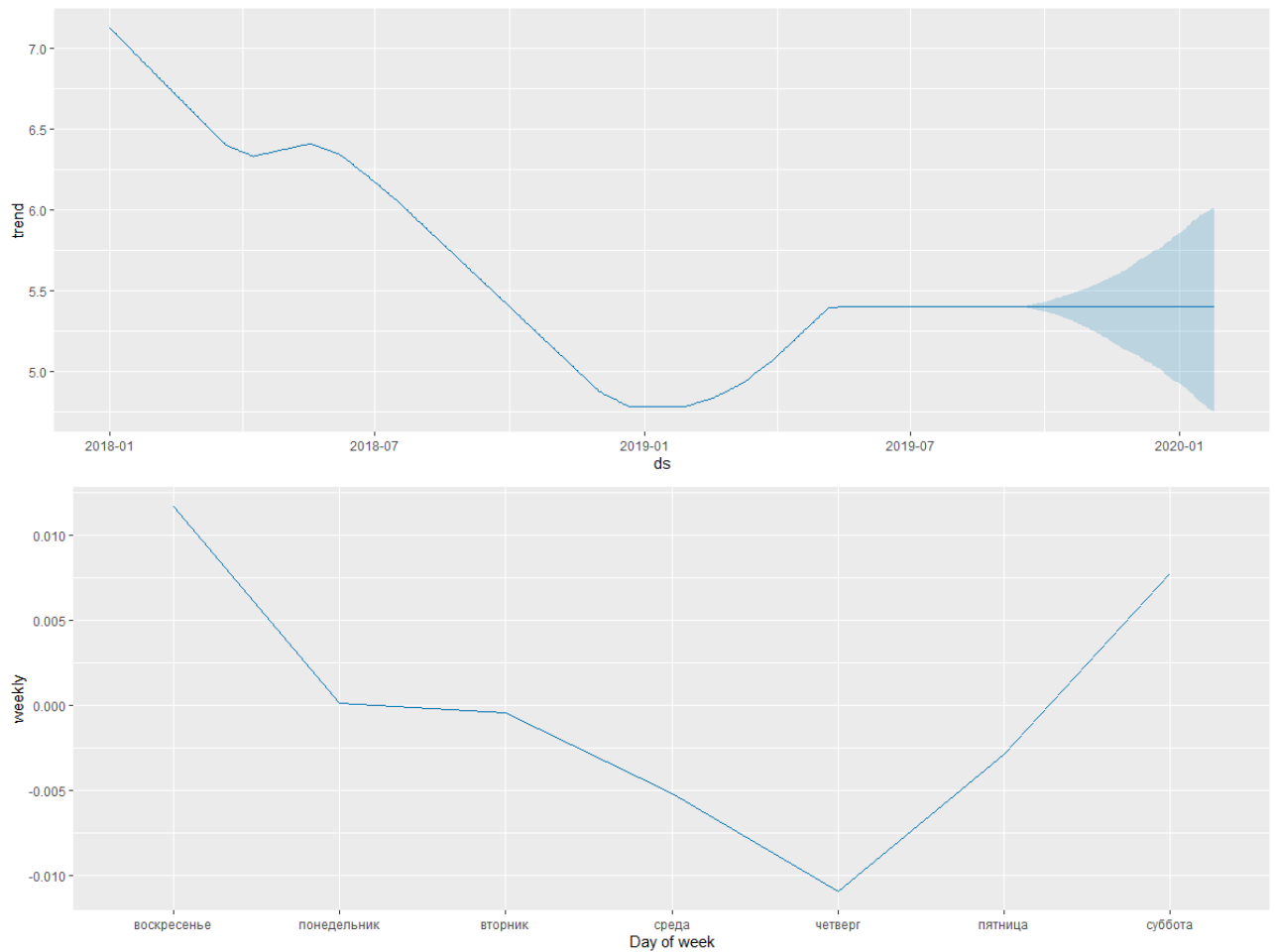


Рис.3.28. Тренд та недільна сезонність моделі M0

Можливо побачити добре відображений складний тренд присутній в моделі M0, а також побачити на графіку недільної сезонності вираженні коливання. Ці дані допоможуть надалі оцінити внесок окремих компонентів для точного прогнозу вартості криптовалюти.

Пакет prophet надає можливість оцінки тренду моделі за допомогою точок перегину. При побудові моделі у автоматичному режимі, модель бере за основу 80% даних навчального набору даних та пропоставляє 13 точок перегину. Це відбувається через вбудований механізм регуляризації даних – обирається мінімальне значення точок для оцінки та прогнозування. Автоматично згенерований графік зображено на рис. 3.29.

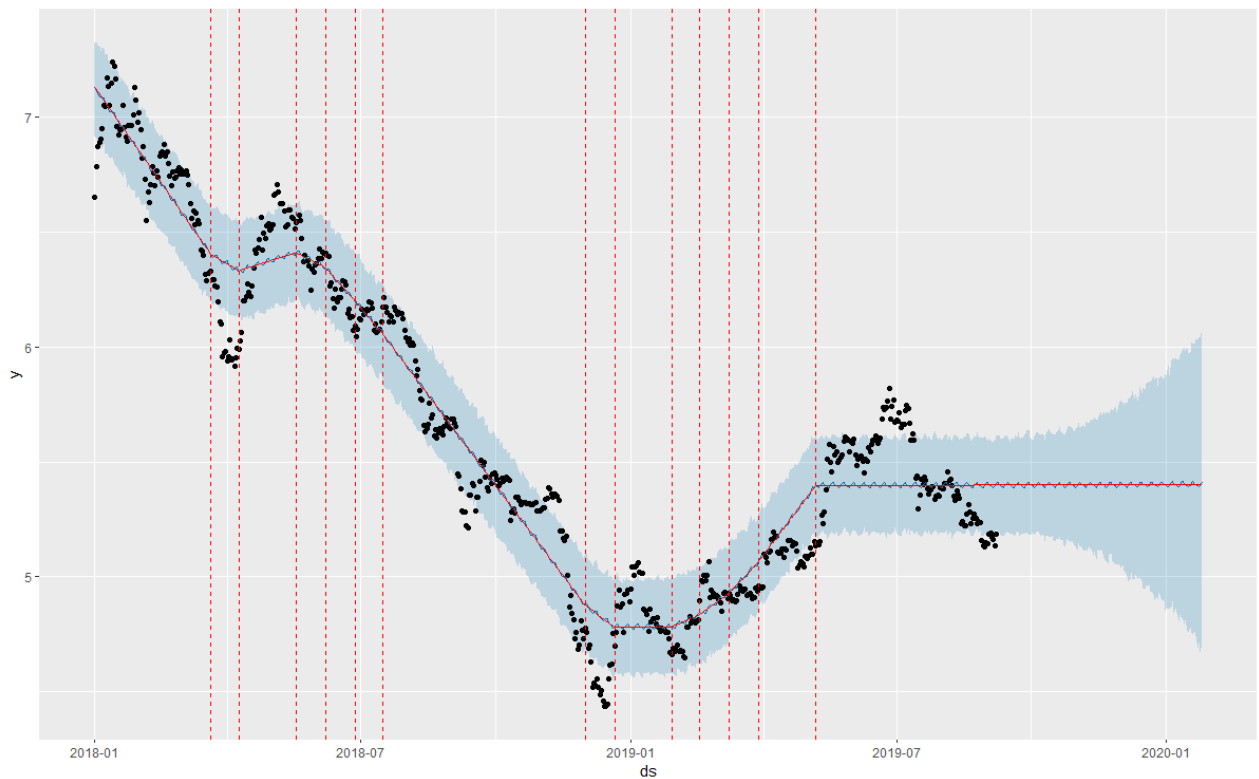


Рис. 3.29. Автоматична оцінка моделі

Судячи з графіку, модель обрала досить точно і змінювати кількість точок перегину не потрібно, для покращення результату прогнозу. Проте автоматична навчальна вибірка охопила не всі важливі точки перегину. На графіку видно підвищення вартості Ethereum у вересні 2019 року, але даний часовий інтервал не бере участі у прогнозуванні. Завдяки корегуванню часового інтервалу покращується якість аналізування даної моделі.

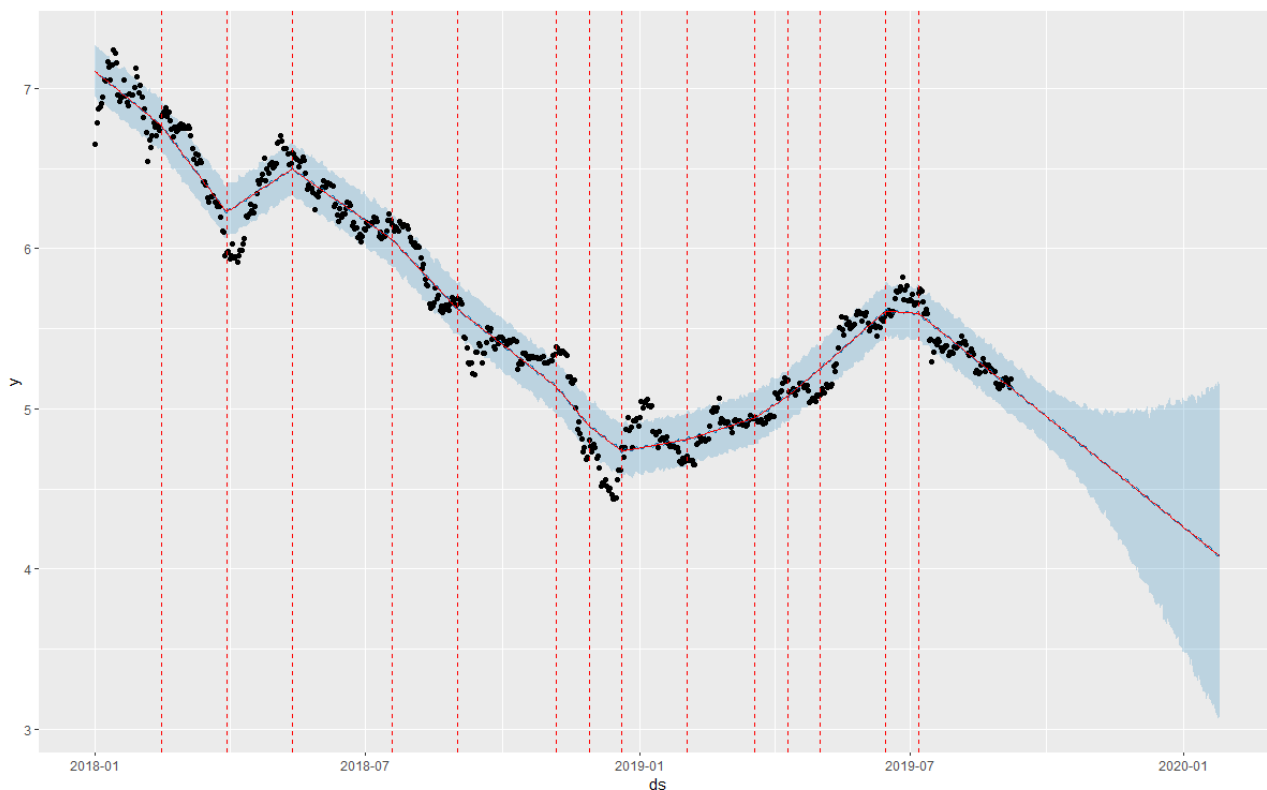


Рис. 3.30. Модель з підвищеним часовим інтервалом оцінки

Однією з переваг, які надає пакет prophet є оцінка часових рядів з урахуванням свят. Третя зпрогнозована модель буде враховувати важливі дати для покращення результатів власних прогнозів. Зпрогнозовані компоненти моделі зображено на рис. 3.31.

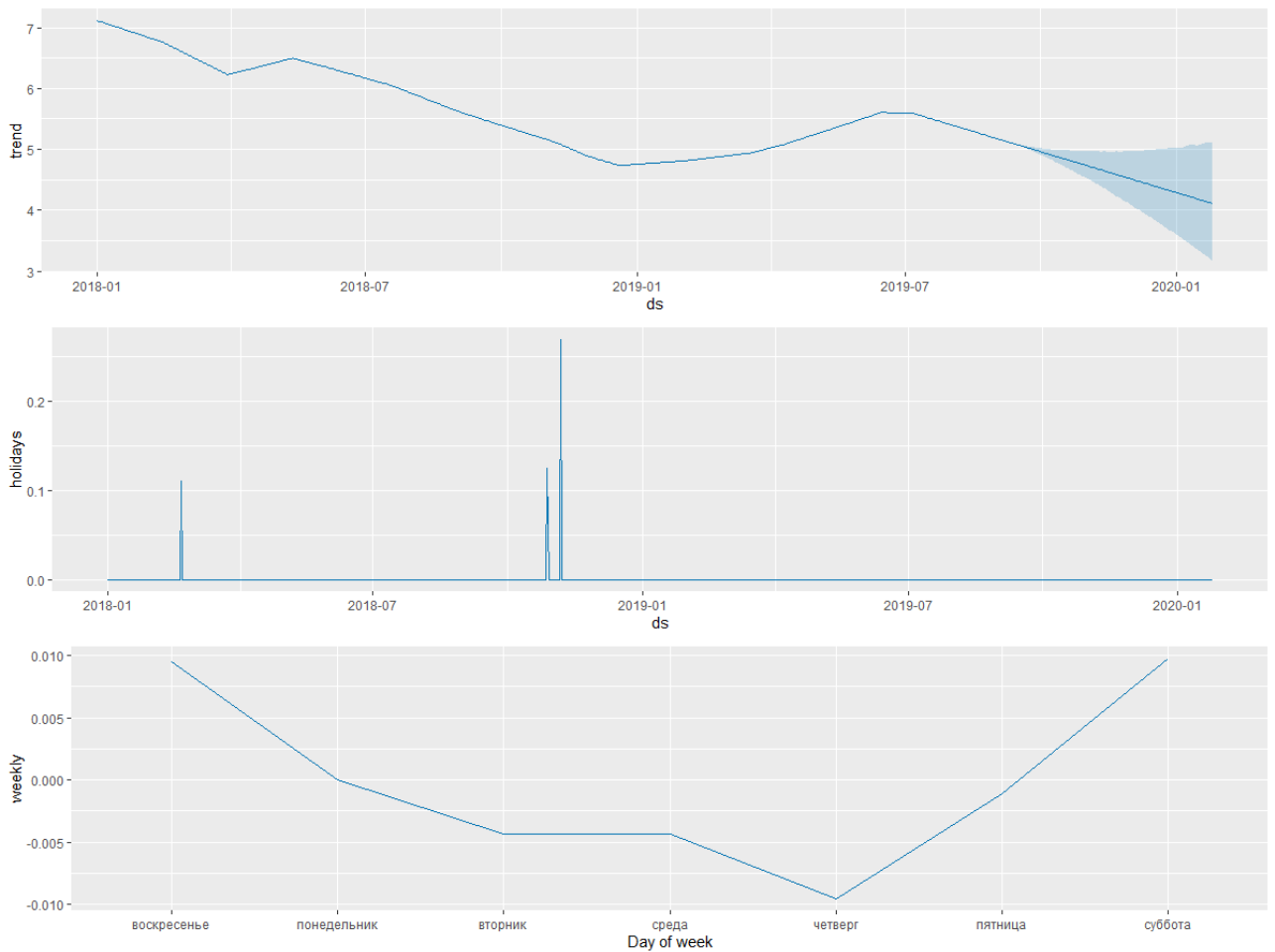


Рис. 3.31. Компоненти третьої моделі

Наступна модель розроблена з урахуванням офіційних свят країни. Пакет містить у собі святкові дати для більше ніж 60 країн світу. Україна наявна у списку можливих для використання країн, але жодних святкових днів не наявно для оцінки. Виходячи с цього, для цілості якості прогнозу обрано США з її святковими днями. Список використаних свят зображено на рис. 3.32, а графік впливу на рис. 3.33.

```
> M3$train.holiday.names
[1] "event_1"           "event_2"           "event_3"           "event_4"
[5] "event_5"           "New Year's Day"   "Martin Luther king Jr. Day" "washington's birthday"
[9] "Memorial Day"     "Independence Day" "Labor Day"         "Columbus Day"
[13] "veterans Day"     "veterans Day (Observed)" "Thanksgiving"     "Christmas Day"
> |
```

Рис. 3.32. Святкові дні

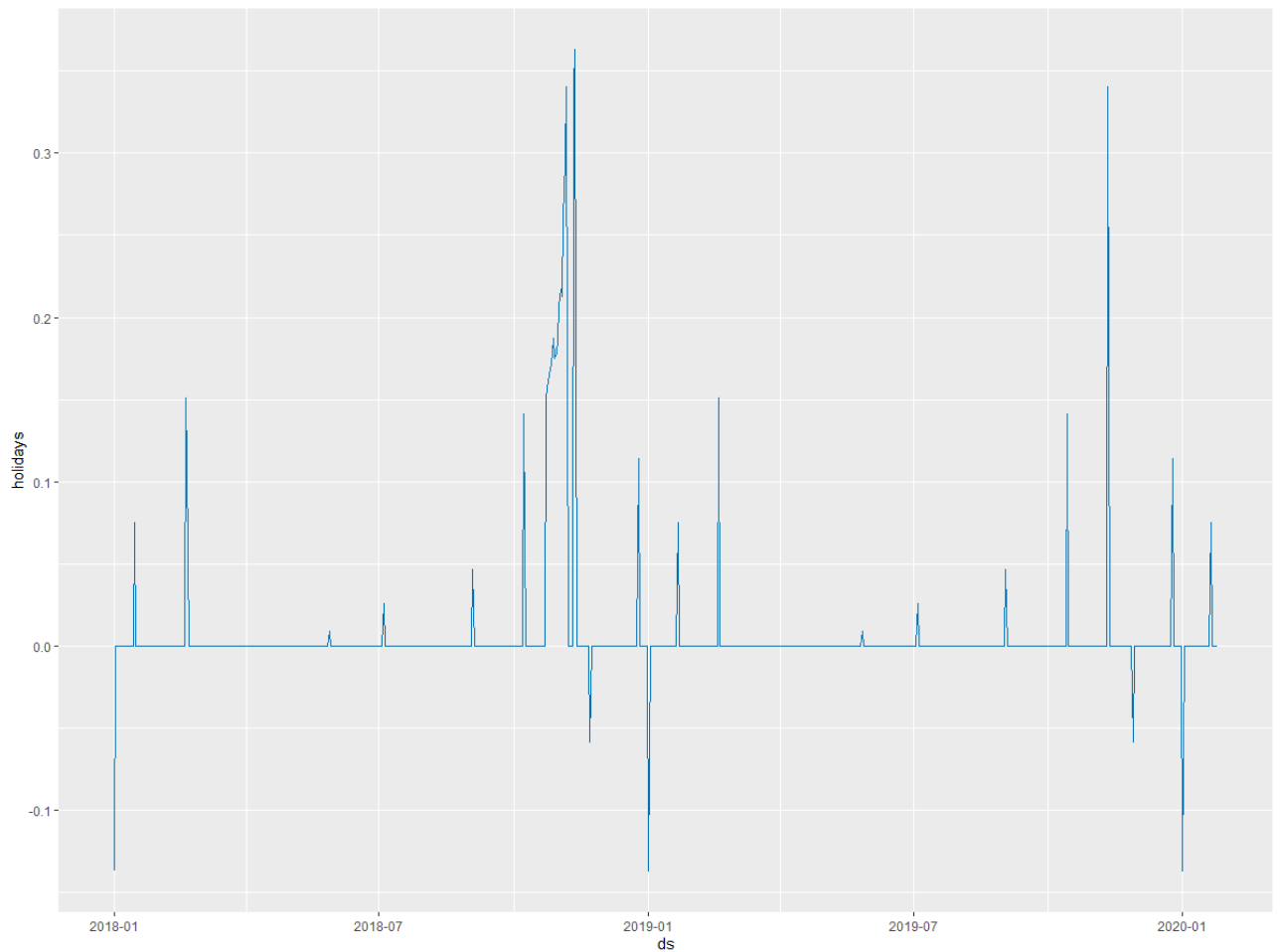


Рис. 3.33. Ефект святкових днів на модель

Точність прогнозу можливо регулювати за допомогою контролю сезонності моделі. Автоматично модель будує недільну сезонність, але можливо змінити її для корегування результатів прогнозування. Четверта модель має відрегульовану сезонність, яка має місячний характер, протяжність якого 29 днів. Вплив компонентів відображено на рис. 3.34.

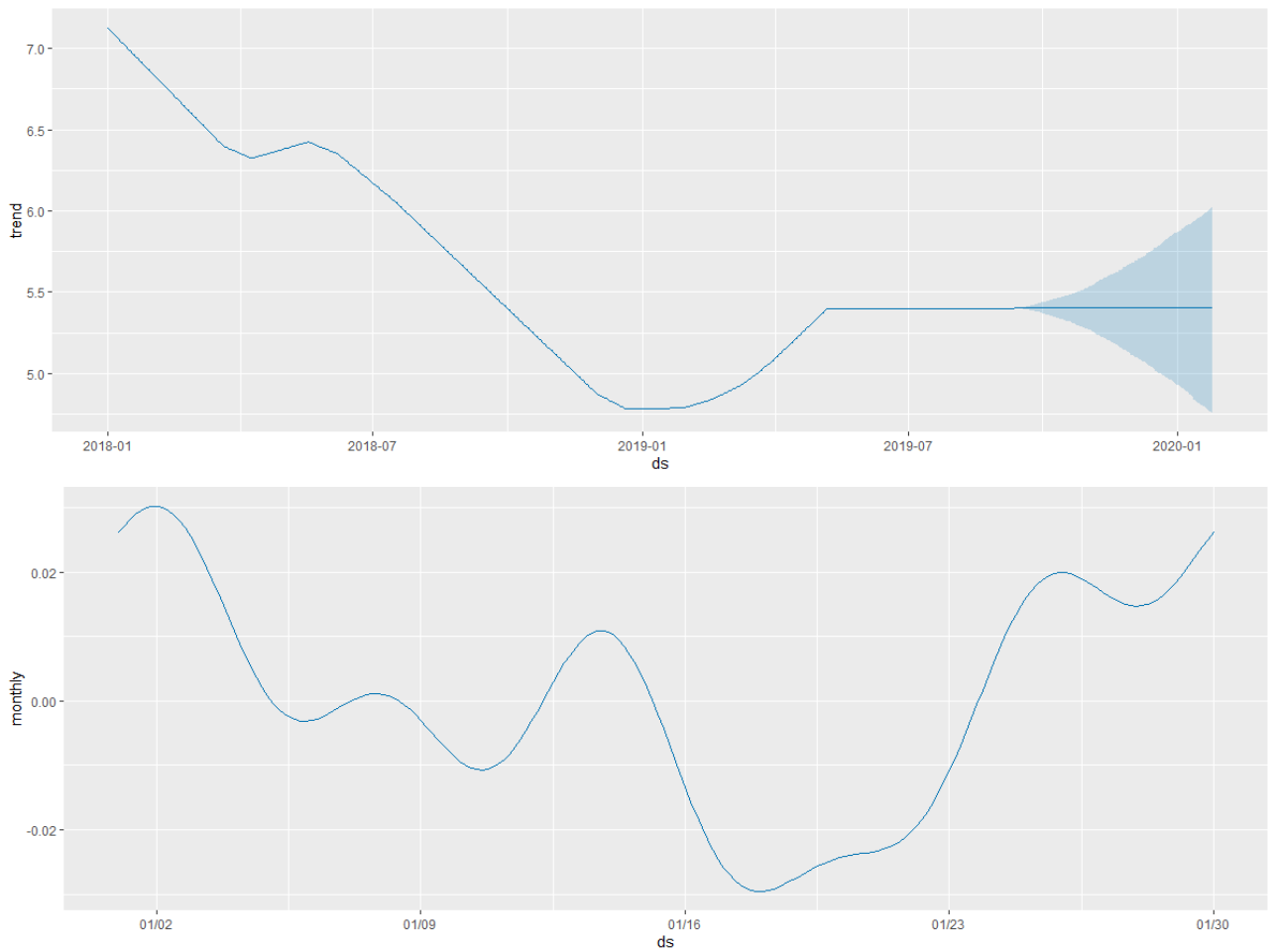


Рис. 3.34. Місячна сезонність моделі

П'ята модель наслідує поведінку попередньої моделі, але регулює квартальну сезонність замість місячної та має довжину періоду $365.25/4$ днів. Вплив компонентів зображено на рис. 3.35.

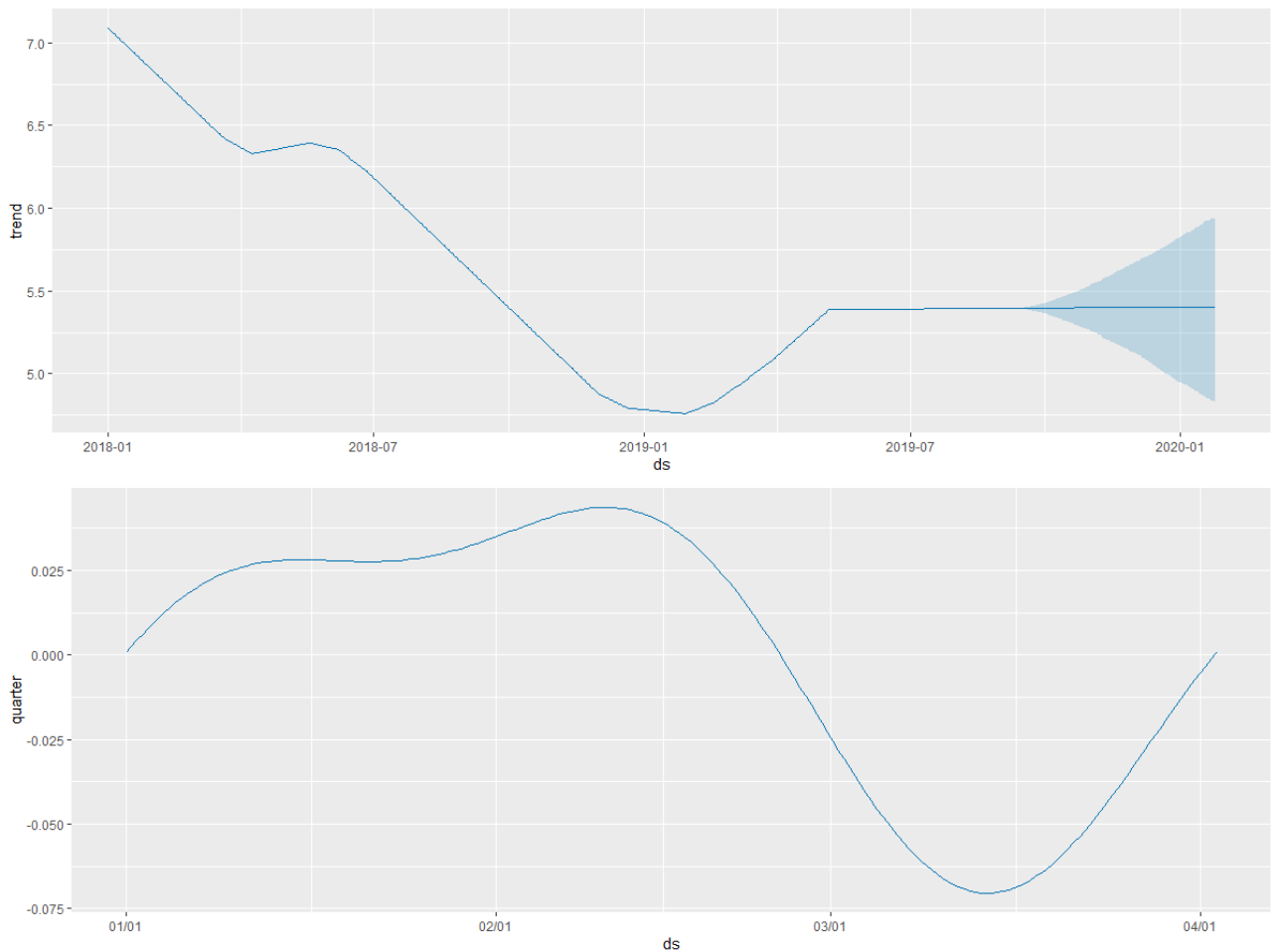


Рис. 3.35. Квартальна сезонність моделі

Нарешті, після створення достатньої кількості моделей, отримується можливість за допомогою тестів визначити, яка саме модель надає найточніші прогнози з використанням криптовалютних індексів.

Для проведення оцінки якості моделі використовують різні методи оцінки якості. Зручним методом, який можливо використати з розробленими моделями – є метод перехресної перевірки.

В основі роботи цього методи застосовується розбиття тренувальних наборів даних випадковим чином на K рівних частин, після чого модель n -ну кількість разів проходить по $K-1$ частинам, а остання частина використовується для перевірки якості на обраній метриці.

Але подібний метод неможливо використовувати у стандартному вигляді у системах з часовими рядами. Саме для цього у пакеті prophet надається метод імітованих історичних прогнозів, який вирішує дану проблему.

Метод SHF використовує стандартний шлях розбиття часового ряду на частини, але всі історичні спостереження, що передують точці k відліку, утворюють тренувальні дані для роботи обраної моделі, а N історичних спостережень, що йдуть за точкою відліку, утворюють прогнозний горизонт. Відстань між точками (період) дорівнює $N/2$. Тренувальні набори даних в першому з K блоків утворюють так званий "initial period" ("початковий відрізок"). У prophet довжина цього відрізка за замовчуванням становить $3 \times N$.

Щоразу після навчання моделі на тренувальних даних з k блоку розраховуються передбачення для прогнозного горизонту того ж блоку, що дозволяє оцінити якість прогнозу за допомогою обраної метрики. Схема роботи даного методу представлена на рис. 3.36.

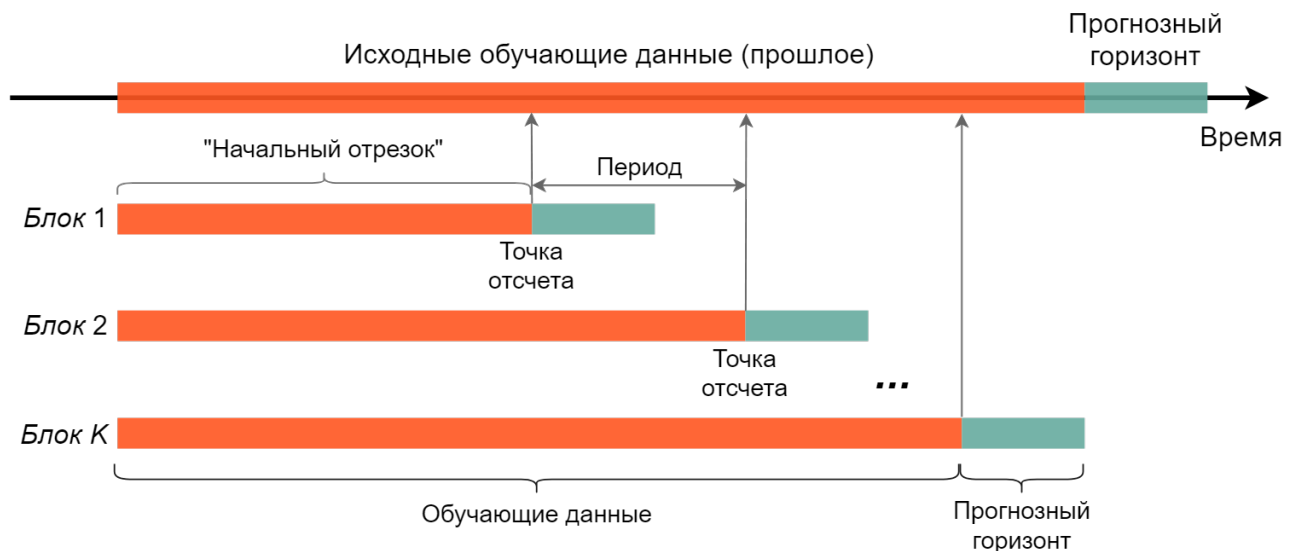


Рис. 3.36. Работа методу SHF

Для отримання інформації, щодо якості моделей у роботі також використано метрики для оцінки якості моделі.

Пакет prophet надає можливість провести оцінку моделі за наступними метриками:

— Середньоквадратична помикав (MSE):

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

— Квадратний корінь із середньоквадратичної помилки (RMSE):

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

— Середня абсолютна помилка (MAE):

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

— Середня абсолютна відсоткова помилка (MAPE):

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

У наведених формулах y_i і \hat{y}_i — це початкове та спрогнозоване значення змінної, що моделюється, а n — кількість спостережень.

— Використання методу SHF або cross-validation. На ній можливо побачити, що y – початкові дані для оцінки, y_{hat} – спрогнозований результат, $y_{\text{hat_lower}}$ та $y_{\text{hat_upper}}$ – нижня та верхня точки довіри. В якості прикладу на рис. 3.37 наведено перевірку однієї з моделей:

```
> head(m3_cv)
      y      ds      yhat yhat_lower yhat_upper cutoff
1 6.196444 2018-06-15 6.254176 6.167851 6.351725 2018-06-14
2 6.213888 2018-06-16 6.262167 6.177045 6.353589 2018-06-14
3 6.215508 2018-06-17 6.254383 6.162418 6.346648 2018-06-14
4 6.251692 2018-06-18 6.225600 6.133032 6.316030 2018-06-14
5 6.287784 2018-06-19 6.206660 6.121986 6.301983 2018-06-14
6 6.284638 2018-06-20 6.186211 6.090054 6.284176 2018-06-14
```

Рис. 3.37. Перехресна перевірка моделі

— Використання метрик оцінки якості. Приклад оцінено за допомогою метрики середньої абсолютної помилки (MSE) та представлено на рис. 3.38:

```
> performance_metrics(m3_cv, metrics = "mse",
+                     rolling_window = 0.1) %>% head()
  horizon      mse
1 9 days 0.01836917
2 10 days 0.02501688
3 11 days 0.03357384
4 12 days 0.04516734
5 13 days 0.06006785
6 14 days 0.07353472
```

Рис. 3.38. Метрики оцінки якості

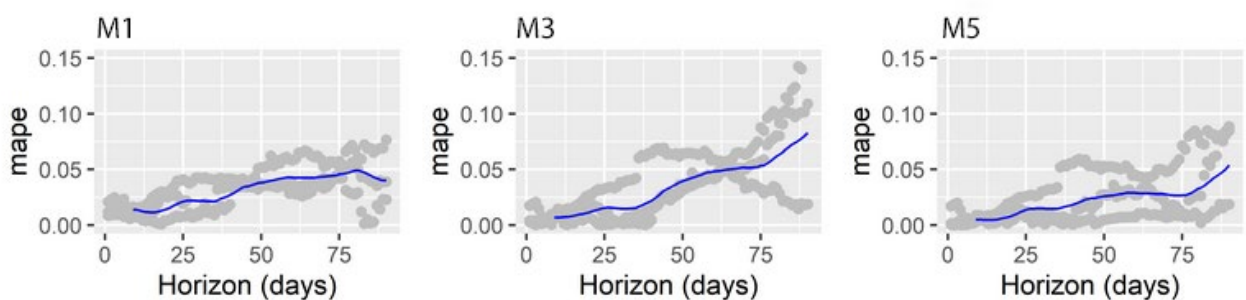


Рис. 3.39. Зпрогнозовані моделі

При початку роботи над інтелектуальною системою прогнозування на основі індексів криптовалют, з початкового набору даних було виділено перевірочну частину. Використовуючи даний набір, перевірено точність прогнозування даних системою та представлено результат на рис. 3.40.

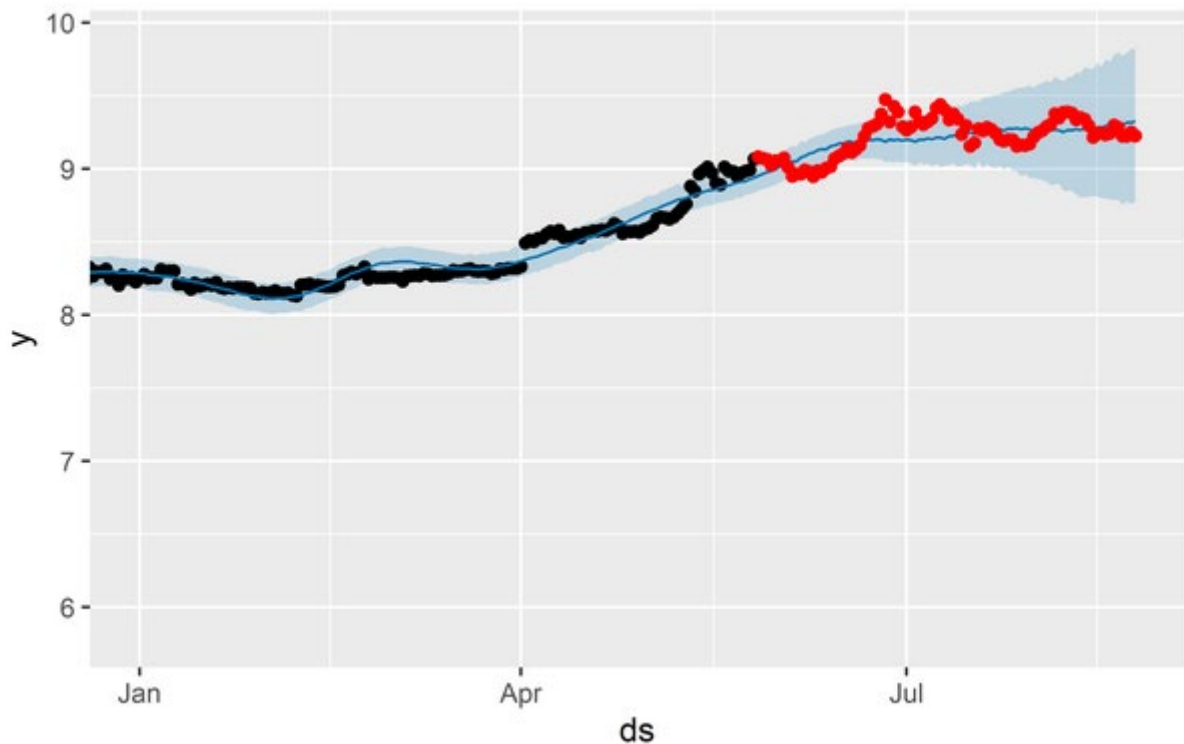


Рис. 3.40 Перевірка початкових даних з спрогнозованими

Висновок до розділу 3

У результаті розробки третього розділу, описано роботу з моделювання та реалізації описаних методологій. Представлено результати тренувань та прогнозів моделей з використанням сучасних можливостей роботи з часовими рядами. Представлено програмні засоби та пакети, використанні при розробці ІС. Наведено результат прогнозування криптовалюти з використанням розробленої ІС прогнозування на основі індексів криптовалют.

4 РЕЗУЛЬТАТИ ВИКОНАННЯ РОБОТИ

Результатом виконання дипломної роботи стала інтелектуальна система прогнозування на основі індексів криптовалют.

Виконуючи процес кодування проекту, використана мова програмування R. Дана мова надає можливість автоматизація та спрощення процесу кодування за допомогою використання пакетів, які надають можливості використовувати необхідні методи аналізу даних.

Нижче наведено лістинг, який відповідає етапам розробки інтелектуальної системи:

Лістинг 4.1. Початкова підготовка даних для подальшої роботи

```
require(dplyr)
require(tsibble)
has_gaps(cryptos)

require(dplyr)
require(ggrepel)
require(ggplot2)
require(readr)
require(tibble)
cryptos <- read_csv("data/cryptos_price.csv") %>%
  as_tsibble(., key = coin, index = ds)
glimpse(cryptos, width = 60)
cryptos.final <- cryptos[cryptos$coin == "monero", ]
cryptos.final %>%
  ggplot(., aes(ds, y)) +
  geom_line() + scale_y_log10() +
  facet_wrap(~coin, ncol = 3, scales = "free_y") +
  theme_minimal()

require(dplyr)
require(tsibble)
require(ggrepel)
require(ggplot2)
```



```
require(readr)
require(tibble)
library(xlsx)
cryptos = read.xlsx("data/cryptos.xlsx", sheetName = "sheet1") %>%
  as_tsibble(key = coin, index = ds)
glimpse(cryptos)
cryptos_na_dropped <- cryptos %>%
  na.omit() %>%
  as_tsibble(key = coin, index = ds)
scan_gaps(cryptos_na_dropped)
#has_gaps(cryptos_na_dropped)
(gaps <- count_gaps(cryptos_na_dropped))
gaps %>%
  filter(.to <= as.Date("2016-06-30")) %>%
  ggplot(., aes(x = coin)) +
  geom_linerange(aes(ymin = .from, ymax = .to)) +
  geom_point(aes(y = .from)) + geom_point(aes(y = .to)) +
  coord_flip() + theme_minimal()
```

Лістинг 4.2. Агрегації даних

```
require(feasts)
require(tseries)
cryptos <- read_csv("data/cryptos_price.csv") %>%
  as_tsibble(., key = coin, index = ds)
glimpse(cryptos, width = 60)
cryptos.eth <- cryptos[cryptos$coin == "ethereum", ]
tsData <- ts(data = cryptos.eth, start = c(2018,1), end = c(2020,1), frequency = 30)
ethData <- stl(tsData[,1], s.window = "periodic")
plot(ethData)
cryptos.xrp <- cryptos[cryptos$coin == "xrp", ]
tsData <- ts(data = cryptos.xrp, start = c(2018,1), end = c(2020,1), frequency = 30)
xrpData <- stl(tsData[,1], s.window = "periodic")
plot(xrpData)
tseries::adf.test(tsData[,1])
tseries::kpss.test(tsData[,1])
tseries::pp.test(tsData[,1])
hist(cryptos.xrp$y, main = "Histogram of XRP Price", xlab = "cryptos.xrp")
```

```
hist(diff(cryptos.xrp$y))
# hist это гистограммы
hist(cryptos.eth$y,main = "Histogram of Ethereum Price",xlab = "cryptos.eth")
hist(diff(cryptos.eth$y))
cryptos.eth %>%
  feasts::gg_lag(geom = "point") + theme_minimal()
cryptos.eth %>%
  feasts::ACF(lag_max = 6)
cryptos.eth %>%
  feasts::PACF() %>%
  autoplot() + theme_minimal()
require(dplyr)
require(ggrepel)
cryptos <- read_csv("data/cryptos_price.csv") %>%
  as_tsibble(., key = coin, index = ds)
monthly.bitcoin<- cryptos[cryptos$coin == "bitcoin",] %>%
  group_by_key() %>%
  index_by(year_month = ~ yearmonth(.)) %>%
  summarise(
    avg_y = mean(y),
    n = n()
  )
monthly
monthly.monero<- cryptos[cryptos$coin == "monero",] %>%
  group_by_key() %>%
  index_by(year_month = ~ yearmonth(.)) %>%
  summarise(
    avg_y = mean(y),
    n = n()
  )
monthly.monero %>%
  mutate(label = ifelse(year_month == max(year_month), coin, NA)) %>%
  ggplot(., aes(year_month, avg_y, group = coin)) +
  geom_line() + scale_y_log10() +
  geom_text_repel(aes(label = label),
                 size = 3, nudge_x = 50,
                 segment.size = 0.4,
                 segment.color = "gray60",
```

```
      point.padding = 0.2,  
      force = 4, na.rm = TRUE) +  
  #xlim(c(yearmonth("2018-01"), yearmonth("2020-06"))) +  
  theme_minimal()
```

Лістинг 4.3 Створення та тренування моделей

```
require(prophet)  
M0 <- prophet(ethereum_train)  
str(M0)  
future_df <- make_future_dataframe(M0, periods = 240)  
forecast_M0 <- predict(M0, future_df)  
forecast_M0 %>%  
  dplyr::select(yhat, yhat_lower, yhat_upper) %>% head()  
plot(M0, forecast_M0)  
prophet_plot_components(M0, forecast_M0)  
plot(M0, forecast_M0) + add_changepoints_to_plot(M0)  
M1 <- prophet(ethereum_train,  
              changepoint.range = 0.9)  
forecast_M1 <- predict(M1, future_df)  
plot(M1, forecast_M1) + add_changepoints_to_plot(M1)  
key_dates <- dplyr::tibble(  
  holiday = paste0("event_", 1:5),  
  ds = as.Date(c("2017-10-21",  
                 "2017-05-09",  
                 "2018-10-28",  
                 "2018-02-20",  
                 "2018-11-06"))  
)  
M2 <- prophet(ethereum_train,  
              holidays = key_dates,  
              changepoint.range = 0.9)  
forecast_M2 <- predict(M2, future_df)  
prophet_plot_components(M2, forecast_M2)  
key_dates2 <- dplyr::bind_cols(key_dates,  
                               lower_window = c(0, 0, 0, 0, -14),  
                               upper_window = c(0, 0, 0, 0, 0))  
M3 <- prophet(holidays = key_dates2, changepoint.range = 0.9)  
M3 <- add_country_holidays(m = M3, country_name = 'US')  
2022 р.                               Фінажин М. Ф.
```

```
M3 <- fit.prophet(M3, ethereum_train)
forecast_M3 <- predict(M3, future_df)
plot_forecast_component(M3, forecast_M3, name = "holidays")
M3$train.holiday.names
M4 <- prophet(weekly.seasonality = FALSE)
M4 <- add_seasonality(m = M4,
                     name = "monthly",
                     period = 29,
                     fourier.order = 5)
M4 <- fit.prophet(M4, ethereum_train)
forecast_M4 <- predict(M4, future_df)
prophet_plot_components(M4, forecast_M4)
M5 <- prophet(weekly.seasonality = FALSE)
M5 <- add_seasonality(m = M5,
                     name = "quarter",
                     period = 365.25/4,
                     fourier.order = 2)
M5 <- fit.prophet(M5, ethereum_train)
forecast_M5 <- predict(M5, future_df)
prophet_plot_components(M5, forecast_M5)
ethereum_train <- cryptos.eth %>%
  mutate(y = log(y)) %>%
  slice(1:(n() - 90)) %>%
  as.data.frame()
ethereum_test <- cryptos.eth %>%
  mutate(y = log(y), ds = as.Date(ds)) %>%
  tail(90) %>%
  as.data.frame()
ethereum_train %>%
  ggplot(., aes(ds, y)) +
  geom_line() +
  theme_minimal()
```

Лістинг 4.4 Прогнозування криптовалют

```
require(tidyr)
require(feasts)

require(tseries)
```

```
cryptos <- read_csv("data/cryptos_price.csv") %>%
  as_tsibble(., key = coin, index = ds)
glimpse(cryptos, width = 60)
cryptos.eth <- cryptos[cryptos$coin == "ethereum", ]
ethereum_train <- cryptos.eth %>%
  mutate(y = log(y)) %>%
  slice(1:(n() - 90)) %>%
  as.data.frame()
ethereum_test <- cryptos.eth %>%
  mutate(y = log(y), ds = as.Date(ds)) %>%
  tail(90) %>%
  as.data.frame()
# Размах дат в таблицях:
ethereum_train %>% pull(ds) %>% range()
shares_na_filled %>% pull(ds) %>% range()
M3_cv <- cross_validation(M3, initial = 90,
  period = 90,
  horizon = 90,
  units = "days")
head(M3_cv)
performance_metrics(M3_cv, metrics = "mse",
  rolling_window = 0.1) %>% head()
plot_cross_validation_metric(M3_cv, metric = "mse",
  rolling_window = 0.1)
M1_cv <- cross_validation(M3, initial = 90,
  period = 90,
  horizon = 90,
  units = "days")
M2_cv <- cross_validation(M3, initial = 90,
  period = 90,
  horizon = 90,
```

```
        units = "days")
M3_cv <- cross_validation(M3, initial = 90,
        period = 90,
        horizon = 90,
        units = "days")
M4_cv <- cross_validation(M3, initial = 90,
        period = 90,
        horizon = 90,
        units = "days")
M5_cv <- cross_validation(M3, initial = 90,
        period = 90,
        horizon = 90,
        units = "days")
M1_perf <- performance_metrics(M1_cv,
        metrics = c("mape", "coverage"),
        rolling_window = 1)
M2_perf <- performance_metrics(M2_cv,
        metrics = c("mape", "coverage"),
        rolling_window = 1)
M3_perf <- performance_metrics(M3_cv,
        metrics = c("mape", "coverage"),
        rolling_window = 1)
M4_perf <- performance_metrics(M4_cv,
        metrics = c("mape", "coverage"),
        rolling_window = 1)
M5_perf <- performance_metrics(M5_cv,
        metrics = c("mape", "coverage"),
        rolling_window = 1)
M1_cv_plot <- plot_cross_validation_metric(M4_cv,
        metric = "mape",
        rolling_window = 0.1) +
```

```
ylim(c(0, 0.15)) + ggtitle("M1")
M2_cv_plot <- plot_cross_validation_metric(M2_cv,
                                          metric = "mape",
                                          rolling_window = 0.1) +
ylim(c(0, 0.15)) + ggtitle("M2")
M3_cv_plot <- plot_cross_validation_metric(M3_cv,
                                          metric = "mape",
                                          rolling_window = 0.1) +
ylim(c(0, 0.15)) + ggtitle("M3")
M4_cv_plot <- plot_cross_validation_metric(M4_cv,
                                          metric = "mape",
                                          rolling_window = 0.1) +
ylim(c(0, 0.15)) + ggtitle("M4")
M5_cv_plot <- plot_cross_validation_metric(M5_cv,
                                          metric = "mape", rolling_window = 0.1) +
ylim(c(0, 0.15)) + ggtitle("M5")
gridExtra::grid.arrange(M1_cv_plot, M2_cv_plot, M3_cv_plot, M4_cv_plot, M5_cv_plot,
ncol = 3)
plot(M5, forecast_M5) +
  coord_cartesian(xlim = c(as.POSIXct("2019-01-01"), as.POSIXct("2019-08-24"))) +
  geom_point(data = bitcoin_test,
            aes(as.POSIXct(ds), y), col = "red")
```

Висновок до розділу 4

Результатом розробки четвертого розділу дипломної роботи стало представлення виконаної роботи з кодування системи та представлення розроблених модулів системи, поділених на описані етапи створення інтелектуальної системи прогнозування.

ВИСНОВКИ

Результатом роботи над магістерською кваліфікаційною роботою став розроблений проект інтелектуальної системи прогнозування на основі індексів криптовалют. Система використовує новітні методології прогнозування часових рядів для отримання якісних результатів прогнозу цін різноманітних криптовалют.

Під час виконання роботи виконані такі завдання:

- 1) проведений аналіз існуючих прикладів створення інтелектуальних систем прогнозування, тематика яких пов'язана з криптовалютами;
- 2) проведена оцінка моделей та методів прогнозування часових рядів та виявлено найкращі існуючі рішення для отримання найточніших результатів прогнозу моделей. Розроблено етапи поступової ітераційної розробки системи, для гнучкого подальшого використання;
- 3) створено інтелектуальну систему прогнозування на основі індексів криптовалют, яка надає високоточні прогнози будь-яких видів криптовалют, використовуючи провідні пакети рішень.

У проектному рішенні реалізовано усі необхідні для подальшого користувача можливості, а саме: можливість аналізу наборів даних будь-яких криптовалют та подальшої швидкої побудови моделей для отримання точних прогнозів у графічному представленні.

Результати проведених перевірок надають наглядної можливості впевнитись в результатах точності роботи системи, використовуючи візуальне представлення у вигляді графіків, гістограм і корелограм.

Застосунок має комерційну спроможність та у подальшому має можливість вийти на ринок як готове рішення для користувачів торгових бірж криптовалют і ентузіастів, які досліджують постійну мінливість вартості криптовалютних індексів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бідюк П. І., Савенков О. І., Баклан І. В. Часові ряди: моделювання та прогнозування. Київ: ЕКМО, 2003. 82 с.
2. Бідюк П. І., Романенко В. Д., Тимощук О. Л. Аналіз часових рядів. Київ: НТУУ «КПІ», 2010. 223 с.
3. Афанасьев В. Н., Юзбашев М. М. Аналіз часових рядів и прогнозування. Москва: Фінанси и статистика, 2001. 231 с.
4. Hyndman R. J., Athanasopoulos G. Forecasting: Principles and Practice (3rd ed). Australia: OTexts, 2021. 442 p.
5. Wickham H., Grolemund G., R for Data Science: Import, Tidy, Transform, Visualize, and Model Data (1st Edition). Canada: O'Reilly, 2021. 521 p.
6. Improved Forecasting of Cryptocurrency Price using Social Signals. вебсайт. URL: <https://arxiv.org/pdf/1907.00558.pdf>
7. Machine learning applications in time series hierarchical forecastin. вебсайт. URL: <https://arxiv.org/pdf/1912.00370.pdf>
8. Bitcoin Energy Requirements Climbing. вебсайт. URL: <https://knoema.ru/suxqfz/bitcoin-energy-requirements-climbing>
9. Bitcoin Price from 2009 to 2021. вебсайт. URL: <https://public.knoema.com/uedfgt/bitcoin-price-from-2009-to-2021>
10. Legal Status of Cryptocurrency. вебсайт. URL: <https://knoema.ru/zstozl/legal-status-of-cryptocurrencies>
11. Quantitative cryptocurrency trading: exploring the use of machine learning techniques. вебсайт. URL: <https://dl.acm.org/doi/10.1145/3336499.3338003>
12. Forecasting at Scale. вебсайт. URL: <https://www.tandfonline.com/doi/full/10.1080/00031305.2017.1380080>
13. Armstrong J.S. Principles of Forecasting: A Handbook for Researchers and Practitioners. German: Springer, 2001. 862 p.

14. Ord K., Fildes R. Kourentzes N. Principles of Business Forecasting (2nd ed). England: wessex, inc., 2017. 588 p.
15. Anscombe F. J., Graphs in Statistical Analysis. New Haven: Yale University, 2012. 21p.
16. Cleveland W. S. Visualizing Data. USA: Hobart Pr, 1993. 360 p.
17. Unwin A. Graphical Data Analysis with R. Augsburg: Taylor and Francis inc., 2015. 310 p.
18. Bickel P. J., & Doksum K. A. An analysis of transformations revisited. Journal of the American Statistical Association, 1981. Vol. 76, P. 296–311.
19. Box G. E. P., Cox D. R. An analysis of transformations. Journal of the Royal Statistical Society. Series B, Statistical Methodology, 1964. Vol. 26. P. 211–252.
20. Guerrero V. M. Time-series analysis supported by power transformations. Journal of Forecasting, 1993 Vol.12. P. 37–48.
21. Dagum E. B., & Bianconcini S. Seasonal adjustment methods and real time trend-cycle estimation. German: Springer, 2016. 299 p.
22. Cleveland R. B., Cleveland W. S., McRae J. E. STL: A seasonal-trend decomposition procedure based on loess. Journal of Official Statistics, 1990. Vol.6. P. 3–33.
23. Izenman, A. J. Modern multivariate statistical techniques: Regression, classification and manifold learning. German: Springer, 2008. 758 p.
24. Wang X., Smith K. A., Hyndman R. J. Large-Scale Unusual Time Series Detection. Proceedings of the IEEE International Conference on Data Mining, 2015. Vol.1 P. 1616–1619.
25. Kang Y., Hyndman R. J., & Smith-Miles K. Visualising forecasting algorithm performance using time series instance spaces. International Journal of Forecasting, 2017. Vol.33, P. 345–358.
26. Wang X., Smith K. A., Hyndman R. J. Characteristic-based clustering for time series data. Data Mining and Knowledge Discovery, 2006. Vol.13. P. 335–364.

27. Gneiting T., & Katzfuss M. Probabilistic forecasting. *Annual Review of Statistics and Its Application*, 2014. Vol.1. P. 125–151.
28. Winkler R. L. A decision-theoretic approach to interval estimation. *Journal of the American Statistical Association*, 1972. Vol.67. P. 187–191.
29. Hyndman R. J., Koehler A. B. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 2006. Vol.22. P. 679–688.
30. Ord J. K., Fildes R., Kourentzes N. *Principles of business forecasting* (2nd ed.). England: Wessex Press Publishing Co. 2017. 588 p.
31. Theodosiou M. Forecasting monthly and quarterly time series using STL decomposition. *International Journal of Forecasting*, 2011. Vol.27. P. 1178–1195.
32. Lawrence M., Goodwin P., O'Conno, M. Judgmental forecasting: A review of progress over the last 25 years. *International Journal of Forecasting*, 2006. Vol.22. P.493–518.
33. Fildes R., Goodwin P. Good and bad judgment in forecasting: Lessons from four companies. *Foresight: The International Journal of Applied Forecasting*, 2007. Vol.8. P. 5–10.
34. Buehler R., Messervey D., Griffin, D. Collaborative planning and prediction: Does group discussion affect optimistic biases in time estimation? *Organizational Behavior and Human Decision Processes*, 2005. Vol.97. P.47–63.
35. Rowe G. A guide to Delphi. *Foresight: The International Journal of Applied Forecasting*, 2007. Vol.8. P.11–16.
36. Rowe G., Wright G. The Delphi technique as a forecasting tool: Issues and analysis. *International Journal of Forecasting*, 1999. Vol.15. P. 353–375.
37. Green K. C., Armstrong J. S. Structured analogies for forecasting. *International Journal of Forecasting*, 2007. Vol.23. P.365–376.
38. Kahneman D., Lovallo D. Timid choices and bold forecasts: A cognitive perspective on risk taking. *Management Science*, 1993. Vol.39. P.17–31.

39. Groves R. M., Fowler F. J., Couper M. P. Survey methodology (2nd ed). USA: John Wiley & Sons, 2009. 486 p.
40. Randall D. M., Wolff J. A. The time interval in the intention-behaviour relationship: Meta-analysis. *British Journal of Social Psychology*, 1994. Vol.33. P.405–418.
41. Harrell F. E. Regression modeling strategies: With applications to linear models, logistic and ordinal regression, and survival analysis (2nd ed). Germany: Springer, 2015. 607p.
42. Sheather S. J. A modern approach to regression with R. Germany: Springer, 2009. 588 p.
43. Gardner E. S. Exponential smoothing: The state of the art. *Journal of Forecasting*, 1985. Vol.4. P.1–28.
44. Gardner, E. S. Exponential smoothing: The state of the art — Part II. *International Journal of Forecasting*, 2006. Vol.22. P.637–666.
45. Hyndman R. J., Koehler A. B., Ord J. K. Forecasting with exponential smoothing: The state space approach. Switzerland: Springer-Verlag, 2008. 324 p.

ВІДГУК

на магістерську кваліфікаційну роботу студента групи 601м
Чорноморського національного університету імені Петра Могили
Фінажина Микити Федоровича
«Інтелектуальна система прогнозування на основі індексів криптовалют»

Магістерська кваліфікаційна робота (МКР) стосується розробки інтелектуальної системи прогнозування з використанням криптовалютних індексів. Метою даної роботи є створення системи, яка здатна проводити якісні прогнози криптовалют для пониження фінансових ризиків користувачів. В роботі проведено аналіз існуючих досліджень прогнозування криптовалют та моделей і методів прогнозування з використанням часових рядів. Досліджено різні методології роботи з часовими рядами та визначено новітні варіанти якісних прогнозів. Визначено методи та засоби розробки функціоналу системи.

Автор провів аналіз існуючих інтелектуальних систем, методологій навчання моделей та варіантів оцінки якості прогнозів. Показав здатність самостійно аналізувати матеріал, викладений у наукових працях і публікаціях міжнародного рівня.

Матеріал у магістерській роботі викладено грамотно та структуровано, з використанням достатньої кількості графічних та схематичних матеріалів. Пояснювальна записка оформлена відповідно до існуючих вимог до оформлення технічної документації.

Магістерська кваліфікаційна робота Фінажина Микити Федоровича є самостійною, цілісною та завершеною працею, яка повністю відповідає встановленим методичним вимогам. Робота виконана на високому рівні і може бути допущена до захисту.

Фінажин Микита Федорович виявив себе працелюбним, дисциплінованим і творчим фахівцем, вміло застосував теоретичні знання та різноманітні сучасні засоби, методи та інформаційні технології для розв'язання практичних задач. Оформлення роботи відповідає вимогам чинних стандартів.

Враховуючи усе вищесказане вважаю за можливе допустити роботу Фінажина Микити Федоровича до захисту та присвоїти освітню кваліфікацію «Магістр комп'ютерних наук» в галузі знань 12 «Інформаційні технології» по спеціальності 122 «Комп'ютерні науки».

Керівник МКР

к-т техн. наук, доцент кафедри ІС
ЧНУ ім. Петра Могили

І. О. Калініна

РЕЦЕНЗІЯ

на магістерську кваліфікаційну роботу студента 601- групи Фінажина Микити Федоровича «Інтелектуальна система прогнозування на основі індексів криптовалют»

Магістерська кваліфікаційна робота Фінажина М. Ф. «Інтелектуальна система прогнозування на основі індексів криптовалют» присвячена розробці програмної системи, яка зможе допомогти отримувати якісні прогнози курсу криптовалют, що забезпечить мінімізацію фінансових ризиків та підвищення конкурентоспроможність користувача.

Точність отримуваних результатів прогнозів є важливим фактором для ефективних торгів на ринках цінних валют. Оскільки криптовалюти в наш час мають велику популярність, створення системи з використанням новітніх методів оцінки, є досить **актуальним** дослідженням.

Фахова частина роботи складається з вступу, 6 розділів, висновків та переліку посилань на використані джерела.

У **вступі** автор дає загальну характеристику досліджуваної теми, відзначає актуальність магістерського дослідження, формулює мету та завдання роботи, відзначає наукову новизну та практичну цінність дослідження, подає інформацію про апробацію, структуру та обсяг роботи.

У **першому розділі** магістрант розкриває загальний зміст понять та визначень, що характеризують систему в цілому, а також дає порівняльну характеристику програмних систем-аналогів, в якій розкриваються їх основні переваги і недоліки.

У **другому розділі** подано загальну характеристику методів і моделей побудови інтелектуальних систем з використанням часових рядів та криптовалютних індексів. Проведений аналіз на підставі якого, формується висновок і подальший хід дослідження. Вводяться основні поняття для поняття для подальшої побудови системи.

У **третьому розділі** автор описує основні етапи створення системи. Даний розділ описує повну розробку системи, починаючи з збору даних і завершуючи результатами наведених системою прогнозів.

У **четвертому розділі** показана програмна реалізація, наведено повний програмний код інтелектуальної системи та розбито його по системним розділам.

У **п'ятому розділі** для закріплення знань і навичок студентам автор пропонує виконати ряд практичних робіт з використанням методів, підходів та

алгоритмів оптимізації. За допомогою яких, студент має можливість оволодіти фаховими знаннями в даній предметній області.

Шостий розділ присвячений охороні праці місця, де проводилась магістерська практика студента. Представлений аналіз факторів виробничого середовища офісного приміщенні і висвітлено питання забезпечення вимог електро- та пожежної безпеки. Проаналізовано питання дій робітників при виникненні надзвичайних ситуацій, а саме випадку пожежної небезпеки.

Таким чином, програмне забезпечення, яке розроблено автором роботи, призначене для прогнозування курсу криптовалют використовуючи криптовалютні індекси допоможе користувачам отримати продукт, здатний сперечатись з іншими гравцями торгів.

Подальше розширення можливе, за рахунок введення більш користувацько зручного інтерфейсу та аналізу можливостей покращення системи використанням новітніх моделей прогнозування.

Матеріал у магістерській роботі викладено грамотно та структурно, з використанням достатньої кількості графічного та схематичного матеріалу. Робота написана сучасною діловою українською мовою. Зміст роботи повністю відповідає завданню виданому на магістерську роботу.

В цілому робота Фінажина М. Ф. відповідає існуючим вимогам до наукових магістерських робіт, має достатньо високий науково-технічний рівень і заслуговує оцінки «відмінно», а її автор заслуговує присвоєння академічної кваліфікації «магістр» за спеціальністю «Комп'ютерні науки».

Рецензент

К. Т. Н., доцент
кафедри комп'ютерної інженерії
ЧНУ ім. Петра Могили

Л. В. Солобуто

«___» _____ 2022 р.

ЗВІТ

про унікальність пояснювальної записки
магістерської кваліфікаційної роботи на тему:
«Інтелектуальна система прогнозування на основі індексів криптовалют»

студента спеціальності 122 «Комп'ютерні науки», 601 групи

Фінажина Микити Федоровича

прізвище, ім'я, по-батькові

Перевірку тексту здійснено сервісом: онлайн-сервіс Unicheck

Результат перевірки тексту магістерської кваліфікаційної роботи: схожість
складає 7,11%.

The screenshot shows the Unicheck web interface. The main content area displays a table of contents for a thesis titled 'Інтелектуальна система прогнозування основ індексів криптовалют' (Intelligent system for forecasting the main indices of cryptocurrencies) from the 'Кафедра інтелектуальних інформаційних систем' (Department of Intelligent Information Systems). The table of contents includes sections like 'ПЕРЕЛІК СКОРОЧЕНЬ', 'ВСТУП', '1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ', and '2 МАТЕМАТИЧНІ МОДЕЛІ, МЕТОДИ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ'. On the right side, a sidebar shows the similarity score of 7.11% and a list of sources with their respective similarity percentages: 'analytics.github.io/tsa-with-r/ch-intro-to-pro...' (4.60%), 'analytics.blogspot.com/2019/12/tsibble.html' (0.68%), 'Фінажин Диплом +в.4 (на Антиплагиате)' (0.45%), 'uk.wikijaa.ru/wiki/KTCC_тест' (0.27%), and 'Юр'єва, 601, Unicheck, 2022' (0.21%).

Студент:

_____ М. Ф. Фінажин
підпис ініціали, прізвище

Керівник:

К-Т. Т. Н., доцент

_____ І. О. Калініна
підпис ініціали, прізвище

Дата: «__» лютого 2022 р.