

ЗАСТОСУВАННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДО ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ КРИПТОВАЛЮТ

В. Д. Дербенцев

Кандидат економічних наук, доцент,
професор кафедри інформатики та системології
Державний вищий навчальний заклад «Київський національний
економічний університет імені Вадима Гетьмана»
проспект Перемоги, 54/1, м. Київ, 03680, Україна
derbvd@kneu.edu.ua

Г. І. Великоіваненко

Кандидат фізико-математичних наук, професор,
завідувач кафедри економіко-математичного моделювання
Державний вищий навчальний заклад «Київський національний
економічний університет імені Вадима Гетьмана»
проспект Перемоги, 54/1, м. Київ, 03680, Україна
ivanenko@kneu.edu.ua

Н. В. Даценко

Старший викладач кафедри економіко-математичного моделювання
Державний вищий навчальний заклад «Київський національний
економічний університет імені Вадима Гетьмана»
проспект Перемоги, 54/1, м. Київ, 03680, Україна
d_tashakneu@ukr.net

Робота присвячена питанням прогнозування короткострокової динаміки часових рядів криптовалют за допомогою методів машинного навчання (ML). Проаналізовано методологічні засади, переваги та недоліки використання ML-алгоритмів при дослідженні фінансових часових рядів.

Проведено прогнозні розрахунки динаміки трьох найбільш капіталізованих криптовалют (Bitcoin, Ethereum, Ripple) на 90-денний часовий горизонт як за допомогою запропонованого методу бінарного авторегресійного дерева (BART), так і з використанням нейронних мереж (багаточарового перцептрон, MLP) та ансамблю моделей дерев регресії та класифікації (випадковий ліс, RF). Перевагою побудованих моделей є те, що їх застосування не накладає жорстких обмежень на статистичні властивості досліджуваних часових рядів, при цьому в якості предикторів використовуються лише минулі значення цільової змінної.

Проведено порівняльний аналіз прогностичних властивостей побудованих моделей, згідно з яким усі моделі в цілому достатньо адекватно описують динаміку досліджуваних криптовалют, при цьому похибки прогнозу цін за середньою абсолютною про-

центною похибкою (MAPE) для моделей BART та MLP склали в середньому 3,5 %, а для моделі RF — в межах 5 %.

Проведені комп'ютерні експерименти підтвердили доцільність застосування розглянутих ML моделей для задач короткострокового прогнозування фінансових часових рядів. Побудовані моделі та їх ансамблі можуть бути покладені в основу алгоритмів для автоматизованих торгових систем, призначених для інтернет-трейдингу.

Ключові слова: машинне навчання, модель бінарного авторегресійного дерева, випадковий ліс, нейронна мережа, короткострокове прогнозування, криптовалюта, фінансовий часовий ряд.

ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ К ПРОГНОЗИРОВАНИЮ ВРЕМЕННЫХ РЯДОВ КРИПТОВАЛЮТ

В. Д. Дербенцев

Кандидат экономических наук, доцент,
профессор кафедры информатики и системологии
Государственное высшее учебное заведение «Киевский национальный
экономический университет имени Вадима Гетьмана»
проспект Победы, 54/1, г. Киев, 03680, Украина
derbvd@kneu.edu.ua

Г. И. Великоиваненко

Кандидат физико-математических наук, профессор,
заведующий кафедры экономико-математического моделирования
Государственное высшее учебное заведение «Киевский национальный
экономический университет имени Вадима Гетьмана»
проспект Победы, 54/1, г. Киев, 03680, Украина
ivanenko@kneu.edu.ua

Н. В. Даценко

Старший преподаватель кафедры экономико-математического моделирования
Государственное высшее учебное заведение «Киевский национальный
экономический университет имени Вадима Гетьмана»
проспект Победы, 54/1, г. Киев, 03680, Украина
d_tashakneu@ukr.net

Работа посвящена вопросам прогнозирования краткосрочной динамики временных рядов криптовалют с помощью методов машинного обучения (ML). Проанализированы методологические основы, преимущества и недостатки использования ML-алгоритмов при исследовании финансовых временных рядов. Проведены прогнозные расчеты динамики трех наиболее капитализированных криптовалют (Bitcoin, Ethereum, Ripple) на 90-

дневный временной горизонт как с помощью предложенного метода бинарного авторегрессионного дерева (BART), так и с использованием нейронных сетей (многослойного перцептрона, MLP) и ансамбля моделей деревьев регрессии и классификации (случайный лес, RF). Преимуществом построенных моделей является то, что их применение не накладывает жестких ограничений на статистические свойства изучаемых временных рядов, причем в качестве предикторов используются только предыдущие значения целевой переменной.

Проведен сравнительный анализ прогностических свойств построенных моделей, согласно которому все модели в целом достаточно адекватно описывают динамику исследуемых криптовалют, при этом ошибки прогноза цен, рассчитанные по средней абсолютной процентной погрешности (MAPE), для моделей BART и MLP составили в среднем 3,5 %, а для модели RF — в пределах 5 %.

Проведенные компьютерные эксперименты подтвердили целесообразность применения рассмотренных ML моделей для задач краткосрочного прогнозирования финансовых временных рядов. Построенные модели и их ансамбли могут быть положены в основу алгоритмов для автоматизированных торговых систем, предназначенных для интернет-трейдинга.

Ключевые слова: машинное обучение, модель бинарного авторегрессионного дерева, случайный лес, нейронная сеть, краткосрочное прогнозирование, криптовалюта, финансовый временной ряд.

MACHINE LEARNING APPROACH FOR FORECASTING CRYPTOCURRENCIES TIME SERIES

Vasyl Derbentsev

PhD (Economic Sciences), Docent,
Professor of Informatics and Systemology Department
State Higher Educational Establishment
"Kyiv National Economic University named after Vadym Hetman"
54/1 Peremogy Avenue, Kyiv, 03680, Ukraine
derbvd@kneu.edu.ua

Halyna Velykoivanenko

PhD (Physics and Mathematical Sciences), Professor,
Head of Department of Economic and Mathematical Modeling
State Higher Educational Establishment
"Kyiv National Economic University named after Vadym Hetman"
54/1 Peremogy Avenue, Kyiv, 03680, Ukraine
ivanenko@kneu.edu.ua

Natalia Datsenko

Lecturer of Department of Economic and Mathematical Modeling
State Higher Educational Establishment
"Kyiv National Economic University named after Vadym Hetman"
54/1 Peremogy Avenue, Kyiv, 03680, Ukraine
d_tashakneu@ukr.net

This paper is devoted to the problems of the short-term forecasting cryptocurrencies time series using machine learning approach. The methodological principles, advantages and disadvantages of using ML-algorithms in the study of financial time series are analyzed.

The 90-day time horizon of the three most capitalized cryptocurrencies (Bitcoin, Ethereum, Ripple) dynamics has been estimated both using the proposed Binary Autoregressive Tree model (BART), and Neural Networks (Multilayer Perceptron, MLP) and an ensemble of Classification and Regression Trees models (Random Forest, RF). The built models have such advantages as their application does not impose strict limitation on the statistical properties of the studied time series, with only past values of the target variable being used as predictors.

Comparative performance of the predictive ability of the constructed models is carried out, according to which all the models adequately describe the dynamics of the cryptocurrencies. Price forecast errors calculated by the MAPE for the BART and MLP models were on average 3.5 %, and for RF model — within 5 %.

Computer experiments have confirmed the feasibility of using proposed ML-models for the short-term forecasting of financial time series. The constructed models and their ensembles can become the basis for the algorithms for automated trading systems for online trading.

Keywords: *machine learning, binary autoregressive tree model, regression and classification tree ensemble, neural network, short-term forecasting, cryptocurrency, financial time series.*

JEL Classification: C53, C63, G10

Вступ

Бурхливий розвиток і впровадження цифрових валют і технології блокчейн протягом останнього десятиріччя є однією із найсуперечливіших і найнеоднозначніших інноваційних технологій сучасної глобальної економіки. Значні коливання курсової вартості криптовалют, а також юридична неврегульованість операцій з ними у більшості країн світу зумовило значні ризики

інвестицій у криптоактиви. Це привело до гострих дискусій щодо їх місця та ролі у сучасній економіці.

Тому питання розробки адекватних методів і моделей прогнозування цін на криптовалюти є актуальними як для наукової спільноти, так і для фінансових аналітиків, інвесторів і трейдерів.

З методологічної точки зору можна виділити два принципово різні підходи до вирішення задачі прогнозування курсової динаміки. Перший полягає у побудові причинно-наслідкової казуальної моделі, яка описує взаємозв'язок між курсовою вартістю активу та іншими макроекономічними змінними. Інший підхід ґрунтується на аналізі спостережень досліджуваної змінної, а для здійснення прогнозу використовуються моделі часових рядів.

Аналіз останніх теоретичних і прикладних досліджень свідчить, що на цінову динаміку криптовалют впливають багато латентних чинників, при цьому ці ключові фактори (драйвери) ще недостатньо вивчені та ідентифіковані [1–4]. Переважна більшість дослідників схиляється до думки, що на курсову вартість криптовалют не мають суттєвого впливу фундаментальні фактори, а визначається вона співвідношенням між попитом і пропозицією.

Окрім цього, часові ряди криптовалют характеризуються високим ступенем волатильності, нестационарністю та негаусівськими законами розподілу [5]. Тому застосування традиційних методів прогнозування, що ґрунтуються на використанні казуальних моделей, побудованих в рамках відповідної теоретичної концепції, або класичних моделей часових рядів виявляється малоефективним.

Останні два десятиріччя до задач прогнозування фінансово-економічних часових рядів почали застосовуватись методи та алгоритми машинного навчання, які розробляються в рамках парадигми *Data Science* [6–8], а для трейдингу почали використовувати різноманітні автоматизовані торгові системи — боти, які побудовані на цих алгоритмах.

При цьому головна увага дослідників була зосереджена переважно на використанні таких методів, як нейронні мережі та регресії на опорних векторах (*Support Vector Regression*). Застосування цих методів виявилось ефективним для задач прогнозування як «традиційних» фінансових активів (фондових

індексів, курсової вартості акцій тощо) [9–13], так і для інноваційних фінансових інструментів, зокрема, криптографічних валют [14–17].

Іншими потужними методами машинного навчання для задач прогнозування виявились дерева класифікацій і регресій та їх ансамблі (*Classification and Regression Tree, C&RT*), запропоновані Лео Брейманом з колегами [18, 19]. Необхідно зауважити, що моделі *C&RT* є ефективнішими для задач класифікації та регресії для випадку наявності у моделі пояснюючих змінних, але гірше пристосовані для роботи з часовими рядами [12].

У роботах [20, 21] запропоновано модель бінарного авторегресійного дерева, що є узагальненням моделей *C&RT* для випадку скалярних часових рядів. Результати застосування цих моделей до задач прогнозування курсової вартості криптовалют показали їх більшу ефективність, ніж моделей часових рядів семейства *ARIMA-ARFIMA*.

Мета і завдання дослідження

Головною метою нашої роботи є побудова ефективної моделі короткострокового прогнозування динаміки курсів криптовалют з урахуванням результатів проведення порівняльного дослідження прогностичних властивостей алгоритму бінарного авторегресійного дерева, штучних нейронних мереж та ансамблів регресійних дерев.

Виклад основного матеріалу

Методологія машинного навчання

Протягом останнього часу серед дослідників і фахівців фінансового ринку все більшої популярності набуває новий напрям — машинне навчання (*Machine Learning, ML*), який знаходиться на стику математичної статистики, методів оптимізації та комп'ютерного моделювання, але має також і власну специфіку, обумовлену проблемами обчислювальної ефективності та перенавчання. Багато методів *ML* тісно пов'язані з видобутком інформації та інтелектуальним аналізом даних (*Data Mining, DM*) і розроблялися як альтернатива класичним статистичним підходам.

Головною відмінністю *ML* від класичного моделювання є те, що алгоритми самі інтерпретують дані, тобто немає методологічної необхідності проводити їх початкову декомпозицію. Залежно

від мети аналізу ці алгоритми самі «вибудовують» логіку моделювання на основі наявних даних.

Це дозволяє уникнути складного та тривалого предмодельного етапу статистичної перевірки різноманітних гіпотез щодо досліджуваного процесу. У зв'язку з цим основна задача дослідження зводиться до перевірки здатності методів *ML* ефективно аналізувати часові ряди криптовалют (як скалярні, так і векторні), виявляти в них приховані паттерни та часові кореляції, що є основою для здійснення якісних прогнозів.

Вся сучасна технологія *ML* побудована на концепції шаблону (паттерну), що відображає фрагменти багатоаспектних зв'язків аналізованих даних, зокрема закономірності, які характерні для певних їх наборів. Важливою особливістю *ML* є те, що методи, які використовуються для пошуку шаблонів, не передбачають апріорної структури даних, їх статистичних властивостей і типу взаємозв'язків.

У рамках парадигми *ML* розроблено низку потужних підходів, методів та алгоритмів: нейронні мережі (*Artificial Neural Network, ANN*), метод опорних векторів (*Support Vector Machine, SVM*), дерева класифікацій і регресій (*Classification and Regression Tree, C&RT*), ансамблі дерев регресій і класифікацій (*Random Forest, RF; Gradient Boosting, XGBoost*), глибокі нейронні мережі та глибоке навчання (*Deep Learning, DL*), ядерні методи (*Kernel Methods*) тощо [6].

Серед методів машинного навчання найбільшої популярності одержали нейронні мережі різноманітної архітектури, зокрема мережі глибинного навчання. Численні емпіричні дослідження свідчать про ефективність застосування *ANN* до задач розпізнавання образів, аналізу зображень і голосових повідомлень, машинного перекладу тощо. Їх усе частіше використовують для аналізу та прогнозування часових рядів, зокрема у фінансовій сфері. У роботах [9, 10, 14] показано, що *ANN* мають кращі прогностичні властивості, ніж моделі часових рядів та інші алгоритми *ML* для задач прогнозування фінансових показників.

Іншим потужним класом моделей *ML* є дерева регресій і класифікацій та їх ансамблі. Але питанням застосування для задач аналізу часових рядів алгоритмам даної групи було приділено значно менше уваги, зокрема і в фінансовій сфері [12, 13].

Кожен з цих методів і підходів має власні переваги та недоліки. Беззаперечною перевагою зазначених методів є те, що вони:

- не потребують жодних апріорних знань та/або припущень щодо вхідних даних;
- мають високий рівень автоматизації, тобто математичний апарат для їх реалізації присутній у більшості програмних продуктів з інтелектуального аналізу даних;
- здатні опрацьовувати масиви різномірних даних як кількісних (метричних), так і якісних (категоріальних).

Серед недоліків як *ANN*, так і *C&RT*, є їх схильність до перенавчання (*overfitting*) та велика кількість гіперпараметрів, що потребують налаштування. Перенавчання моделі призводить до того, що похибки прогнозу можуть суттєво зростати при застосуванні моделі до нових (або при збільшенні вибірки) даних.

Для *ANN* притаманна «непрозорість» функції, що описує залежність між вхідними та вихідними нейронами, та значні витрати часу на налаштування. Крім того, вибір архітектури мережі, кількості прихованих шарів, нейронів на кожному з них і функцій активації в загальному випадку є нетривіальною задачею.

До головних недоліків моделей дерев регресії та класифікації відносять їх меншу точність для задач регресії (порівняно, наприклад, з *ANN*) та неоднозначність вибору кращого фінального дерева (для задачі прогнозу). Але їх перевагами є те, що вони є інтуїтивно зрозумілими та простими при візуалізації та інтерпретації, тобто є прозорими для аналізу.

Однак надто складні розгалужені дерева також важко піддаються змістовній інтерпретації, тому при їх використанні доводиться шукати компроміс між складністю дерева та його точністю. Ця проблема притаманна для переважної більшості *ML* алгоритмів.

Що стосується алгоритму *RF*, який полягає у побудові ансамблю простих класифікаторів (дерев) та одержанні усередненої оцінки прогнозу кожного з дерев, які будуються на різних підмножинах ознак та випадково обраних навчальних підвибірках даних, то він набагато менш схильний до перенавчання, проте з точки зору змістовної інтерпретації є не надто «прозорим».

Ми будемо використовувати технологію машинного навчання з вчителем (*supervise learning*), тобто навчальна та тестова вибірки містять класифікаційний набір прикладів. У нашому випадку

це значення прогнозу цільової змінної на один крок вперед (one step ahead forecast) Y_t при відомих значеннях цільової змінної у p попередні моменти часу $Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}$ (рис. 1).

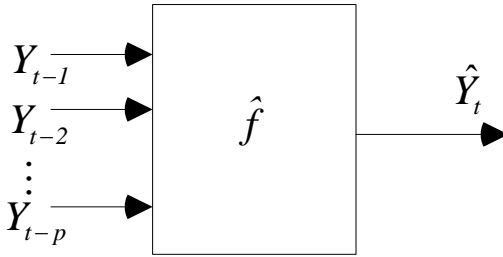


Рис. 1. Модель чорної скриньки для здійснення прогнозу

Таким чином, наша гіпотеза полягає у тому, що

$$P(Y_t | Y_1, Y_2, \dots, Y_{t-1}, \theta) = F(Y_t | Y_{t-p}, Y_{t-p+1}, \dots, Y_{t-1}, \theta), p < t \leq T, \quad (1)$$

де $F(\cdot | \cdot, \theta)$ — сімейство умовних розподілів імовірності, що представляють модель; θ — вектор невідомих параметрів; T — кількість спостережень у вихідному часовому ряді (навчальній вибірці).

При цьому функція гіпотези має вигляд:

$$\hat{Y}_t = \hat{f}(Y_{t-p}, Y_{t-p+1}, \dots, Y_{t-1}, \theta) + \varepsilon_t. \quad (2)$$

Тобто в якості факторів або ознак (в *ML* термінології — «фіч», feature) ми використовуємо лише минулі значення цільової змінної.

Для здійснення прогнозу динаміки криптовалют та порівняння прогностичних властивостей різних *ML* алгоритмів ми використали три моделі: алгоритм бінарного авторегресійного дерева (*Binary Auto Regressive Tree, BART*), модель багат шарового перцептрона (*Multilayer Perceptron, MLP*) та модель ансамблю дерев регресії «випадковий ліс» (*RF*).

Загальну схему проведення нашого дослідження представлено на рис. 2.

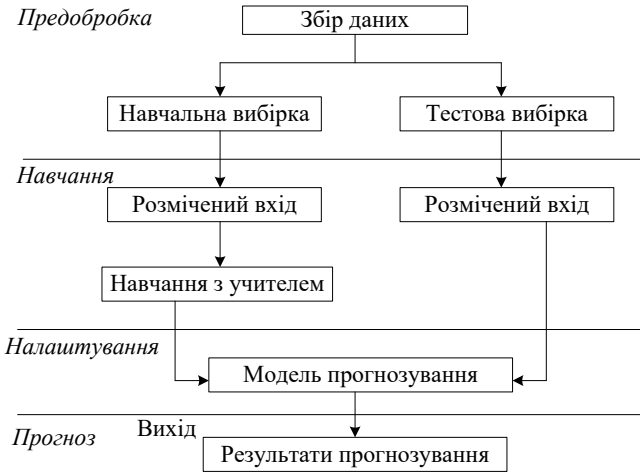


Рис. 2. Концептуальна схема проведення дослідження

Необхідно зауважити, що при застосуванні методів *ML* доводиться вирішувати проблему «*bias-variance trade-off*», яка полягає в тому, що при налаштуванні параметрів моделі доводиться шукати компроміс між похибкою прогнозу, зумовленою його зміщенням (*Bias*), та нестабільними значеннями параметрів (*Var*):

$$PE(Y_t) = M \left[(Y_t - \hat{f}(Y_t))^2 \right] = Bias^2(\hat{f}) + Var(\hat{f}) + \sigma^2, \quad (3)$$

де $PE(Y_t)$ — похибка прогнозу для довільного моменту часу t ; $M(\cdot)$ — оператор математичного сподівання; $Y_t, \hat{f}(Y_t)$ — точне значення часового ряду та його прогнозована оцінка, відповідно; $Bias(\cdot)$ — середнє зміщення прогнозу по усіх наборах даних; $Var(\cdot)$ — дисперсія похибки, яка в загальному випадку залежить від кількості параметрів моделі та їх точності (дисперсії); σ^2 — неусувна похибка.

Модель бінарного авторегресійного дерева

Бінарне авторегресійне дерево (*BART*) — модель дерева класифікації та регресії, на листях якого містяться авторегресійні моделі *ARIMA* [20, 21]. Ця модель є узагальненням стандартних моделей *C&RT* та адаптацією до задач прогнозування часових

рядів. *BART* поєднує класичний алгоритм *C&RT* [18] та стандартні авторегресійні моделі *ARIMA* Бокса-Дженкінса.

У *BART* для аналізу часового ряду застосовується метод «віконного» перетворення даних. Цільова змінна Y_t в даному алгоритмі залежить від p попередніх значень досліджуваного часового ряду $Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}$.

Таке розділення вхідного простору на сегменти (рис. 3) дозволяє побудувати для кожного з них власну (локальну) модель та представити кусково-задану функцію як дерево авторегресії в інтуїтивно наглядній формі.

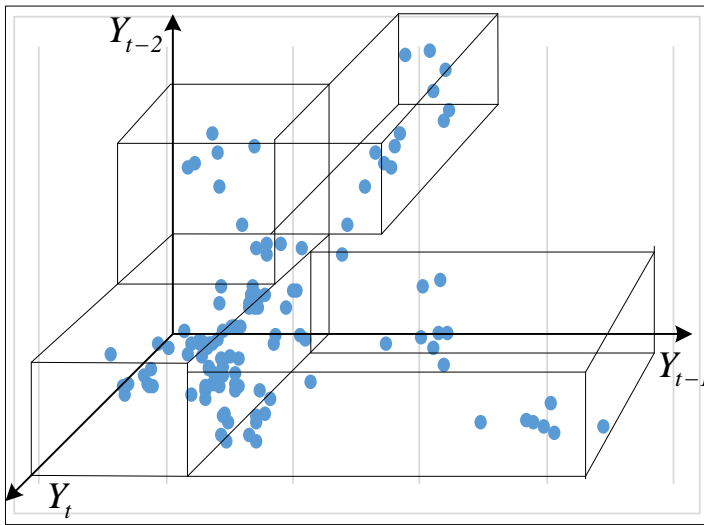


Рис. 3. Умовний приклад розбиття даних з двома лаговими змінними на підмножини

При побудові *BART* кількість гілок (розгалужень) на кожному рівні дорівнює 2, оскільки будується бінарне авторегресійне дерево (тобто таке, яке має двох нащадків з кожного вузла). Процес побудови авторегресійного дерева виконується послідовно (ітераційно) та описується наступним алгоритмом [20, 21].

Етап 1. На першому кроці визначається порогове значення для розщеплення початкового (кореневого) вузла, яке береться як

медіана Me (другий квантиль $Q_{50\%}$) навчального ряду (вибірки) Y до моменту часу t (Y^t) і розраховується за формулою:

$$Me(Y^t) = Q_{50\%} = 0,5 * (Y_t^{\min} + Y_t^{\max}), \quad (4)$$

де Y_t^{\min}, Y_t^{\max} — мінімальне та максимальне значення часового ряду Y до моменту часу t .

Медіана часового ряду визначається як медіана розподілу реалізації випадкової величини до моменту часу t . Для стаціонарного ряду та ряду з симетричним розподілом ця величина не залежить від моменту спостереження і тоді вибіркова медіана буде співпадати з середнім значенням усього часового ряду, тобто $Me(Y^t) = \bar{Y}$.

Авторегресійна оцінка дерева на першому рівні розбиття буде мати вигляд:

$$f(Y^t) = Me(Y^t) I_R(Y_{t-1}), \quad (5)$$

де R — простір даних піддерева (на першому кроці вся навчальна вибірка); $I_R(Y_{t-1})$ — індикаторна функція простору, що набуває бінарних значень (фактично це набір правил потрапляння змінної Y_{t-1} у лівий або правий підпростори даних). На першому кроці весь часовий ряд за критерієм $Y_{t-1} \leq Me(Y^t) \forall t: p < t \leq T$ розбивається на дві підмножини:

$$R_1 = \{Y_{t-1} \in R : Y_{t-1} \leq Me(Y^t)\}, R_2 = \{Y_{t-1} \in R : Y_{t-1} > Me(Y^t)\} \quad \forall t: p < t \leq T.$$

Етап 2. На другому кроці обирається одна з підмножин, одержаних на першому кроці, яка також поділяється на дві частини. Визначається лагова змінна для розщеплення, наприклад Y_{t-k} , $k \in (1, 2, \dots, p)$, та будуються спадкові лівий R_1 і правий R_2 підпростори даних (для спрощення запису індекси номеру ітерацій ми опустили):

$$R_1 = \{Y_{t-k} \in R : Y_{t-k} \leq \alpha\}, R_2 = \{Y_{t-k} \in R : Y_{t-k} > \alpha\} \quad \forall t: p < t \leq T. \quad (6)$$

Тоді регресійна оцінка на наступному рівні розбиття набуває вигляду

$$f(Y^t) = \left(\frac{1}{M} \sum_{I_1} Y_{t-k}^{(i)} \right) I_{R_1}(Y_{t-k}) + \left(\frac{1}{N} \sum_{I_2} Y_{t-k}^{(i)} \right) I_{R_2}(Y_{t-k}), \quad (7)$$

де $I_1 = \{i, Y_{t-k}^{(i)} \in R_1\}$, $I_2 = \{i, Y_{t-k}^{(i)} \in R_2\}$ — множина індексів спостережень $Y_{t-k}^{(i)}$, що потрапляють у підпростори R_1 та R_2 , відповідно; M, N — кількість елементів у цих підпросторах. Тут нижній індекс $t-k$ характеризує лаг спостереження (предиктора), за яким буде здійснюватись розщеплення, а верхній індекс у дужках (i) — це номери спостережень $Y_{t-k} \forall t: p < t \leq T$, що потрапляють у лівий R_1 і правий R_2 підпростори даних після здійснення розщеплення.

Фактично, регресійна оцінка (7) є середнім арифметичним спостережень, що потрапили у лівий і правий підпростори даних після здійснення розщеплення.

Необхідно зауважити, що як у класичному алгоритмі Бреймана *C&RT* [18], так і його модифікаціях (наприклад, *CHAID*, *C.4.5*), незважаючи на його назву (дерево регресії та класифікації), насправді прогноз не здійснюється за допомогою регресійної моделі. В якості прогнозу обирається середнє арифметичне спостережень, які потрапили у відповідний підпростір даних (атомарний вузол-лист дерева, який не підлягає подальшому розщепленню). Тому в якості прогнозованого значення береться константа, одержана за формулою виду (7) (після виконання певної кількості кроків алгоритму).

Отже, для використання класичного алгоритму *C&RT* та його модифікацій для задач прогнозування часових рядів необхідно будувати складні розгалужені дерева, намагаючись одержати на листках дерева невелику кількість однорідних спостережень. А це, як було зазначено вище, досить часто призводить до перенавчання моделі та, як наслідок, до значних похибок прогнозу на контрольній (або тестовій) вибірці, яку не було використано при оцінці параметрів моделі.

Для подолання цього недоліку класичного алгоритму нами було запропоновано будувати на листках локальні авторегресійні моделі *ARIMA*, які здатні краще апроксимувати дані, що потрапили у певну підмножину (лист дерева) після виконання деякої

кількості кроків розбиття простору вхідних даних. Оцінкою кращого розбиття за усією моделлю *BART* є найменша сума квадратів різниць:

$$R(\hat{f}) = \frac{1}{T} \sum_{t=p+1}^T (Y_t - \hat{f}(Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}))^2, \quad (8)$$

де $\hat{f}(\cdot)$ — функція гіпотези (2). В алгоритмі *BART* в якості цієї функції нами було запропоновано обрати модель типу *ARIMA*.

Етап 3. Для кожного вузла, одержаного на попередньому кроці, здійснюється пошук нового кращого розщеплення. Для цього задаються два аргументи: змінна Y_{t-k} , $k \in (1, 2, \dots, p)$, за якою буде здійснюватись розщеплення (сплітування), та порогове значення цієї змінної α .

У дослідженні ми використовували в якості порогових значень квантілі відповідного емпіричного розподілу випадкової величини Y (значення, яке задана випадкова величина не перевищує з певною ймовірністю):

$$\alpha \in (Q_{10\%}, Q_{25\%}, Q_{40\%}, Q_{50\%}, Q_{60\%}, Q_{75\%}, Q_{90\%}). \quad (9)$$

З можливих на цьому кроці варіантів розщеплення обирається «краще» за прийнятим правилом. Ці процедури проводяться аналогічно алгоритму *C&RT*. Відмінність полягає в прийнятих правилах, критеріях оцінки та зупинці розщеплення. У *BART* запропоновано альтернативний критерій обрання кращого розщеплення (або критерій інформативності) на основі показника ентропії, оскільки це приводить до зменшення складності дерева. У даному алгоритмі визначається ентропійний інформаційний виграш.

Вихідна ентропія всієї вибірки на першому кроці буде $\hat{H}(M, N)$. В якості функції ентропії ми використали ентропію Шенона (10). Вона характеризує кількість інформації (або, іншими словами, ступінь невизначеності) щодо початкового поділу вибірки на два класи (підмножини), що здійснюється на першому етапі згідно порогу (4). Частотна ймовірність потрапляння певного спостереження буде дорівнювати $\frac{M}{M+N}$ та $\frac{N}{M+N}$ для лівого R_1 та правого R_2 підпростору, відповідно.

На кожному кроці оцінюється ймовірність як частота віднесення деякого спостереження Y_{t-k} , $k \in (1, 2, \dots, p)$ для $\forall t \in p+1, \dots, T$, до певної підмножини (піддерева) і розраховується ентропія \hat{H} l -ї підвибірки до здійснення розщеплення ($y^l \in Y$):

$$\hat{H}(m, n) = H\left(\frac{m}{M+N} + \frac{n}{M+N}\right), \quad (10)$$

де m, n — кількість спостережень, які відповідають умовам належності підмножинам R_1 і R_2 для відповідного вузла до здійснення розщеплення за обраним предиктором ($m \in M, n \in N$).

Після здійснення l -го розщеплення в обраному вузлі за певним предиктором Y_{t-k} (внаслідок чого ми отримуємо додаткову інформацію ϕ щодо приналежності певного спостереження відповідному підпростору даних) ентропія підвибірки $y_1^l \in R_1^l$ буде $\hat{H}(m, n)$, а ймовірність появи об'єкта з цієї вибірки буде обчислюватись як $\frac{m+n}{M+N}$. Аналогічно для вибірки $y_2^l \in R_2^l$ ентропія буде $\hat{H}(M-m, N-n)$ з відповідною імовірністю $\frac{(M-m)+(N-n)}{M+N}$.

Тут R_1^l та R_2^l — лівий і правий підпростори даних після здійснення l -го розщеплення, а y_1^l та y_2^l — множини спостережень $Y_{t-k}^{(i)}$, що потрапили у ці підпростори, $y_1^l \cup y_2^l = y^l$.

Таким чином, ентропія всієї вибірки після отримання інформації ϕ обчислюється за формулою

$$\hat{H}_\phi(m, n) = \frac{m+n}{M+N} \hat{H}(m, n) + \frac{M+N-(m+n)}{M+N} \hat{H}(M-m, N-n). \quad (11)$$

Вираз (11) показує ентропію вибірки після того, як стало відомо, що при здійсненні розбиття певного вузла за умовою $Y_{t-k} \leq \alpha$ предиктор Y_{t-k} виділив m спостережень із M і n спостережень із N . У такому разі зменшення ентропії складатиме

$$IGain_C(\phi, y^l) = \hat{H}(m, n) - \hat{H}_\phi(m, n), \quad (12)$$

що називається *ентропійним інформаційним виграшем*.

Тут перший доданок характеризує ентропію у батьківському вузлі, а другий (обчислений за (11)) є зваженою сумою ентропії у вузлах нащадках після розщеплення.

Отже, вираз (12) характеризує зменшення вибіркової ентропії (або приріст інформації) до i після здійснення розщеплення в обраному вузлі за певним предиктором (в нашому випадку лаговою змінною Y_{t-k}).

Таким чином, для здійснення кожного наступного розщеплення обирається той вузол та лагова змінна (i , відповідно, порогове значення α), які забезпечать максимальне зменшення ентропії (12).

Етап 4. На наступному кроці здійснюється оцінка «вартості» дерева, яка характеризує співвідношення між точністю апроксимації та складністю (розгалуженістю) побудованого дерева.

У *BART* вартість дерева визначається на основі критерію ранньої зупинки. В якості такого критерію ми використовували розширений байєсівський інформаційний критерій (*Extended Bayesian Information Criterion, EBIC*), який мінімізує статистику

$$EBIC = T \cdot \ln R(\hat{f}) + J \cdot [\ln(T) + 2 \cdot \ln(b)], \quad (13)$$

де T — кількість прикладів навчальної вибірки; $R(\hat{f})$ — середньоквадратична похибка (8); J — кількість параметрів налаштування моделі; b — величина, яка характеризує складність моделі та дорівнює добутку розміру дерева (кількість розгалужень у дереві) на кількість лагових змінних (p).

Таким чином у виразі (13) перший доданок — це значення логарифмічної функції середньоквадратичної похибки, а другий є штрафом за складність моделі.

Етап 5. Розщеплення вузлів проводиться до тих пір, поки значення статистики *EBIC* зменшується. Якщо обране розщеплення ефективно за ентропійним інформаційним вирашем (12), то потрібно його виконати та перейти знову до етапу 3 (розглянути інші неопрацьовані вузли). В іншому випадку побудоване дерево вважається оптимальним, а процедура виконання алгоритму *BART* — завершеною.

Оскільки кінцевою метою наведеного алгоритму є прогнозування, то на вузлах-листях нами запропоновано будувати стан-

дартну авторегресійну модель класу *ARIMA*, яка є базовою галузеву моделлю аналізу та моделювання часових рядів.

Як у цілому для регресійних дерев *C&RT*, так і, зокрема, для *BART*, процедура спрощення (тобто ранньої зупинки розгалуження дерева) є важливішою, ніж для дерев класифікації. Це пояснюється тим, що дерева регресії, як правило, є складнішими та розгалуженішими, оскільки діапазон значень досліджуваного показника (наприклад, ціни криптовалюти) значно ширше, ніж для якісних (категоріальних) даних.

Модель «випадкового лісу»

Алгоритм випадкового лісу ґрунтується на побудові ансамблю класифікаційних (регресійних) дерев, кожне з яких утворюється з підвибірок вихідної навчальної вибірки з використанням *беггінгу* (скорочення від «**bootstrap aggregating**») [19]. Беггінг — це метод створення ансамблю моделей на основі різних випадкових вибірок з вихідного набору даних. Вибірki формуються рівномірно із заміною і називаються підсилюючими вибірками (*bootstrap samples*) [6].

Ефективність беггінгу досягається завдяки тому, що базові алгоритми, які пройшли навчання на різних підвибірках, будуть суттєво відрізнятися один від одного, при цьому їхні помилки взаємно компенсуються при «голосуванні». А також за рахунок того, що об'єкти-викиди (аномальні спостереження, стрибки часового ряду) можуть не потрапляти до деяких навчальних підвибірок.

Беггінг особливо корисний у поєднанні з деревоподібними моделями, які є чутливими до зміни навчальних даних. В алгоритмі *RF* беггінг поєднується з методом випадкових підпросторів, коли кожне дерево будується за різними випадково обраними підмножинами ознак (фіч), які називаються вибірками підпростору (*subspace sampling*).

Метод випадкових підпросторів дозволяє знизити корельованість між деревами та уникнути перенавчання, оскільки базові алгоритми навчаються на різних підмножинах ознак, які також обираються випадковим чином.

У результаті, різноманіття ансамблю буде ще більшим, при цьому зменшується час його налаштування, оскільки кожне з дерев можна навчати паралельно. Одержаний у такий спосіб ансамбль називається випадковим лісом (*random forest, RF*).

RF метод застосовується як для задач класифікації, так і для регресії, крім цього за допомогою *RF* також можна проводити відбір предикторів і пошук відхилень при аналізі даних.

Значення прогнозу з використанням алгоритму *RF* знаходиться шляхом усереднення прогнозів, одержаних кожним деревом ансамблю (або «голосуванням» дерев для задач класифікації). На відміну від окремих дерев, цей алгоритм набагато менше схильний до перенавчання та дає більш чутливу (гнучку) границю прийняття рішень.

Модель нейронної мережі

В якості моделі *ANN* ми використали найпростішу та найпоширенішу архітектуру багат шарового персептрона з одним вхідним шаром нейронів, одним прихованим шаром і вихідним шаром, що містить один вихід — оцінку прогнозу досліджуваного часового ряду на один крок (рис. 4).

Незважаючи на таку просту архітектуру, згідно з теоремою Колмогорова внаслідок використання суперпозиції нелінійних функцій активації на прихованому та вихідному шарах мережі можна виявити складні нелінійні паттерни в даних і змодельовати з достатньою точністю невідому нелінійну функцію досліджуваного часового ряду.

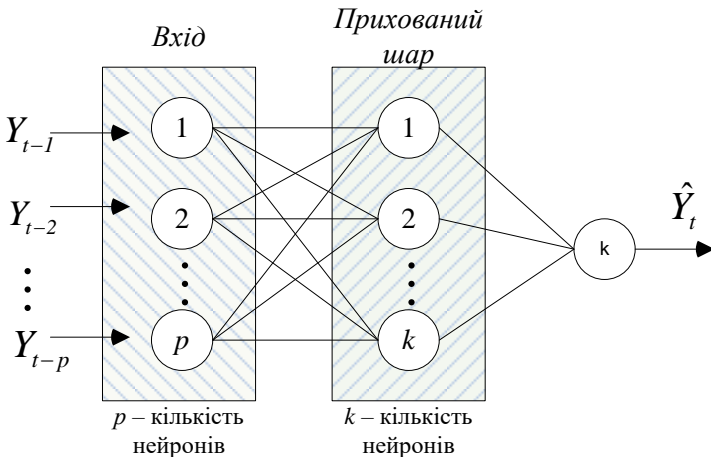


Рис. 4. Модель тришарового персептрона

Значення виходу мережі залежать від її конфігурації, вагових коефіцієнтів міжнейронних зв'язків та функцій активації нейронів прихованого та вихідного шарів:

$$\hat{Y}_t = g \left(\sum_{i=1}^k w_i f \left(\sum_{j=1}^p \omega_{ji} Y_{t-j} + b_i \right) + b_0 \right), \quad (14)$$

де $f(\cdot), g(\cdot)$ — функції активації нейронів прихованого та вихідного шару, відповідно; w_i — ваги зв'язків між i -им нейроном прихованого шару та виходом мережі; ω_{ji} — ваги зв'язків між j -им нейроном вхідного та i -им нейроном прихованого шарів; b_0, b_i — зміщення (bias) нейронів вихідного та прихованого шарів.

У процесі навчання *MLP*, яке полягає у знаходженні вагових коефіцієнтів (синаптичних ваг), використовується функція похибки (error or cost function), що визначає різницю між цільовою змінною та виходом мережі (аналогічно до (8)). Пошук мінімуму функції похибки здійснювався за допомогою методу градієнтного спуску (gradient descent), втіленого в алгоритмі зворотного поширення похибки (back-propagation algorithm).

Статистична база дослідження

Для чисельної реалізації моделей короткострокового прогнозування динаміки криптовалют з використанням алгоритмів *BART*, *RF* і *MLP* було обрано часові ряди щоденних котирувань трьох найбільш капіталізованих (станом на 21.08.2019) криптовалют: біткоіна (*BTC*), ефіріума (*ETH*) та ріпла (*XRP*). Вибірка даних складає 1482 спостереження за період з 01.08.2015 по 21.08.2019 за інформацією сервісу Yahoo Finance [22].

Для аналізу було обрано ціни закриття, які було взято як в абсолютному вираженні, так і в натуральних логарифмах, що дозволяє стабілізувати варіативність (дисперсію) досліджуваних рядів (рис. 5).

Перші 1392 спостереження для побудованих нами моделей було розділено у співвідношенні 80 % на 20 % між навчальною та тестовою вибіркою та використано для оптимізації структури і параметрів моделей, а останні 90 спостережень було зарезервовано для оцінки якості прогнозу на незалежних даних.

Емпіричні результати

Оскільки усі три типи моделей для здійснення прогнозу використовують лише минулі спостереження досліджуваного часового ряду, то одним із головних завдань їх ідентифікації є вибір глибини лагу p . Згідно з результатами багатьох емпіричних досліджень, для «традиційних» фінансових активів (фіатних валют, фондових та товарних індексів, курсів акцій тощо), які торгуються протягом п'яти днів на тиждень, існує циклічний лаг, який є кратним 5, якщо використовувати щоденні спостереження [10, 11].

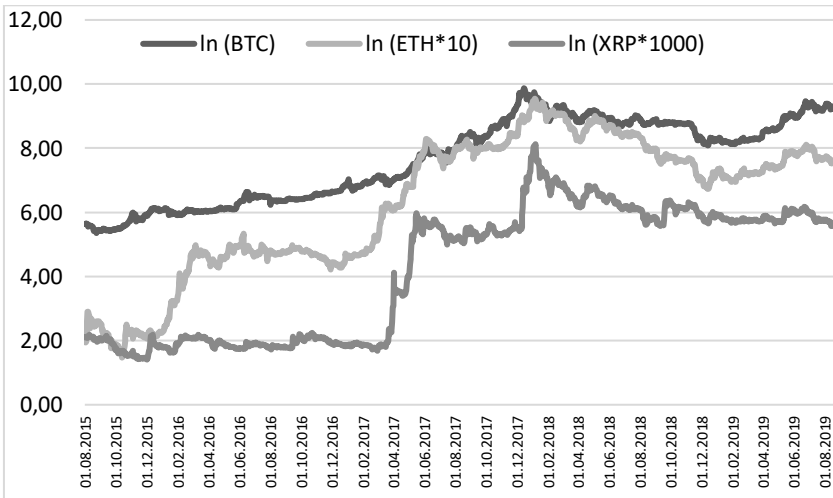


Рис. 5. Динаміка щоденних цін закриття *BTC*, *ETH* та *XRP* за період 1.08.2015 — 21.08.2019 (дол. США, логарифмічна шкала)

Оскільки криптовалюти торгуються у режимі 24/7, то цілком очікуваним буде наявність сезонного лагу, кратного 7 дням. Проведення кореляційного аналізу із використанням автокореляційної та часткової автокореляційної функції підтверджує нашу гіпотезу: для усіх трьох досліджуваних коїнів наявні статистично-значущі кореляції на лагах 7, 14, 21, 28, окрім цього існують кореляції і на деяких інших лагах. Схожі результати одержано, зокрема, у роботах [5, 17].

Для здійснення короткострокового прогнозу ми тестували 3 класи моделей (*BART*, *RF*, *MLP*) з різною глибиною лагу

$p = 7, 14, 21$ для кожної з криптовалют. Модельні розрахунки для *MLP* і *RF* проводились у пакеті *Satistica 8.0*, а для реалізації алгоритму *BART* було використано *MS Azure Machine Learning Studio* із застосуванням *R*-скрипту.

Згідно з прийнятою нами гіпотезою щодо глибини лагу в якості моделей *MLP* ми тестували такі архітектури:

- із 7 вхідними та від 4 до 12 нейронів прихованого шару;
- із 14 вхідними та від 5 до 15 нейронів прихованого шару;
- із 21 вхідними та від 6 до 21 нейронів прихованого шару.

В якості функцій активації було обрано такі найпоширеніші функції, як логістична, гіперболічний тангенс, експоненційна та тотожна (лінійна). Навчання мереж для кожної з криптовалют і різних значеннях лагу (кількості вхідних нейронів) проводилось протягом 100 епох, з яких для кожного випадку обирались п'ять кращих (з точки зору мінімальної похибки *PE* (3) на тестовій вибірці та відповідності залишків моделі нормальному розподілу) архітектур.

Остаточний прогноз для кожної з криптовалют було одержано як розрахунок ансамблю мереж, тобто середньоарифметичне п'яти відповідних кращих моделей *MLP*.

Для реалізації *RF* ми встановили такі метапараметри: загальна кількість дерев 200, максимальна глибина дерева 10, кількість предикторів у кожному дереві — 3, 5, 7 для моделей *RF-7*, *RF-14* та *RF-15*, відповідно.

Для реалізації *BART* було обрано максимальну глибину дерева 15, а мінімальна кількість прикладів (спостережень) у вузлі дорівнювала 20. Розрахунки було здійснено як для абсолютних значень цін, так і для їх логарифмів.

Для оцінки точності прогнозу ми скористались такими метриками, як середня абсолютна процентна похибка (*Mean Percentage Absolute Error*, *MAPE*) та середньоквадратична похибка (*Root Mean Square Error*, *RMSE*), що розраховуються за формулами:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|Y_t - \hat{f}(Y_t)|}{Y_t} \times 100\% ,$$

(15)

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (Y_t - \hat{f}(Y_t))^2} .$$

Слід зазначити, що показник $RMSE$ можна застосувати тільки для оцінки якості різних прогнозів для одного фінансового інструменту (часового ряду). Він надає інформацію про величину похибки, але не про те, наскільки ця похибка велика у порівнянні з реальним значенням котирування. На відміну від $RMSE$, $MAPE$ у процентах дозволяє оцінити ефективність прогнозів як окремих моделей, так і їх ансамблів для різних активів і порівнювати їх між собою.

Необхідно зауважити, що точність прогнозу за метриками (15) для логарифмів цін досліджуваних часових рядів виявилась у цілому не вищою, ніж для абсолютних значень цін. Цей факт підтверджує тезу про значно меншу чутливість алгоритмів ML , зокрема моделей нейронних мереж, регресійних дерев і їх ансамблів до статистичних властивостей досліджуваних рядів, ніж класичних статистичних та економетричних методів.

На рис. 6–8 наведено остаточні результати прогнозування динаміки абсолютних значень цін досліджуваних криптовалют на 90-денний часовий горизонт, який здійснювався за технологією однокрокового прогнозу без переналаштування параметрів моделей.

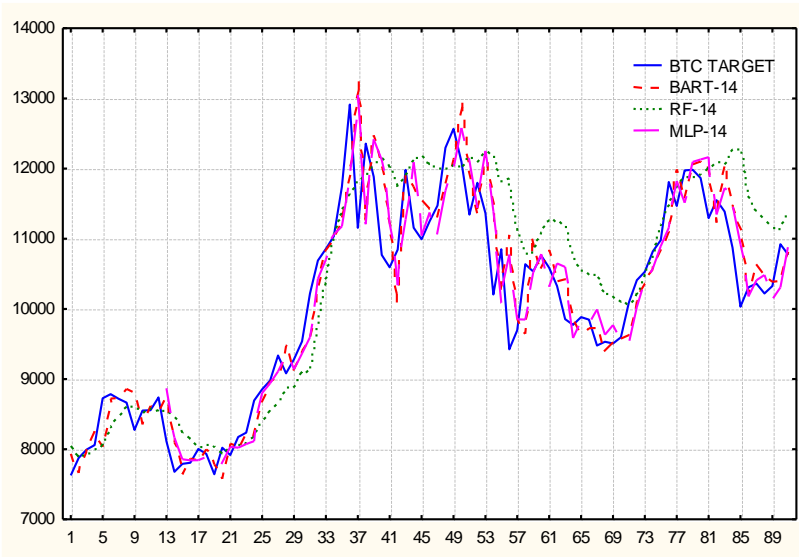


Рис. 6. Прогноз динаміки BTC (лаг $p=14$)

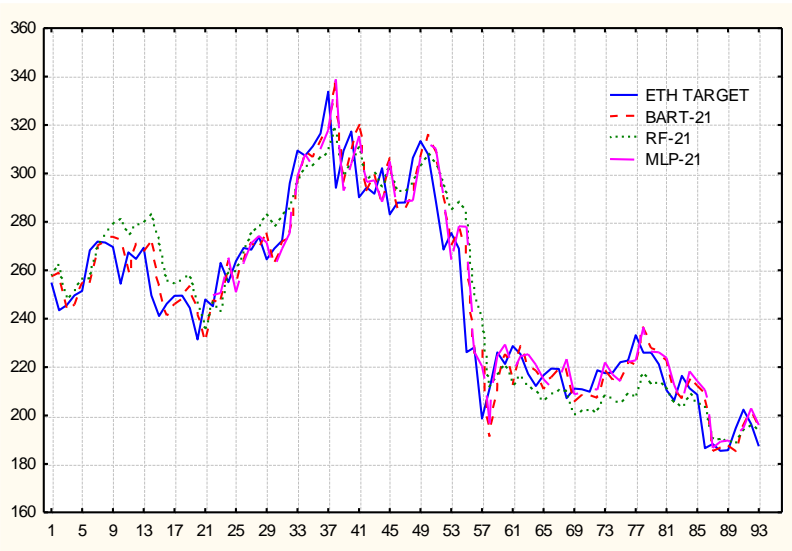


Рис. 7. Прогноз динаміки ETH (лаг $p=21$)

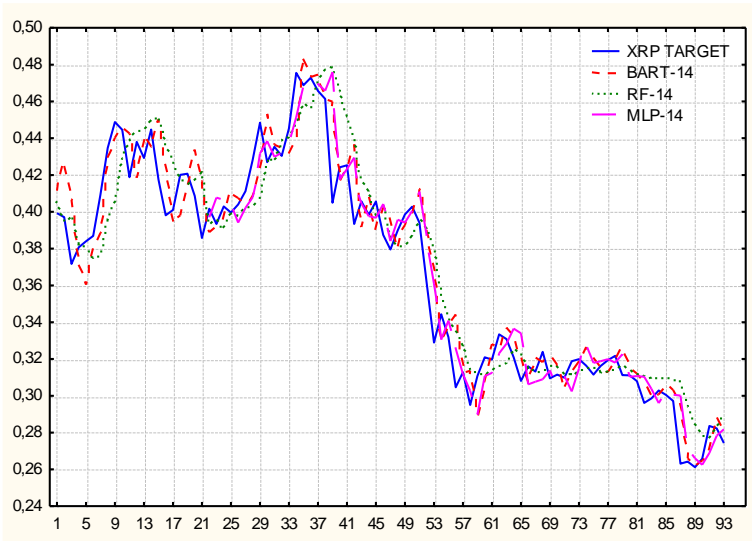


Рис. 8. Прогноз динаміки XRP (лаг $p=14$)

Безпосередній аналіз графіків дозволяє зробити висновок про в цілому достатньо адекватне слідування моделей реальним даним, враховуючи складну осцилюючу динамічну поведінку досліджуваних рядів, при цьому для *BTC* простежується наявність зростаючої тенденції, а для *ETH* та *XRP* — навпаки.

Також можна побачити, що всі моделі, незважаючи на в цілому адекватне врахування наявних тенденцій динаміки криптовалют, демонструють певне запізнення щодо реальних даних.

Підсумкові результати оцінки точності прогнозу на 90 днів за усіма моделями для досліджуваних рядів наведено у табл. 1.

Таблиця 1

РЕЗУЛЬТАТИ ОЦІНКИ ТОЧНОСТІ ПРОГНОЗУ

Модель	BTC		ETH		XRP	
	MAPE, %	RMSE	MAPE, %	RMSE	MAPE, %	RMSE
Глибина лагу $p=7$						
BART-7	3,71	535,2	3,39	11,74	3,07	0,0154
RF-7	7,11	971,9	7,44	21,8	3,94	0,0196
MLP-7	3,69	529,8	3,53	12,17	3,07	0,0153
Глибина лагу $p=14$						
BART-14	3,83	541,9	3,37	11,86	3,42	0,0167
RF-14	5,6	756,9	6,48	19,82	4,08	0,0203
MLP-14	3,95	559,1	3,51	12,16	3,41	0,0162
Глибина лагу $p=21$						
BART-21	3,94	558,5	3,69	12,55	3,83	0,0183
RF-21	5,54	739,3	4,52	14,55	3,92	0,0212
MLP-21	4,28	610,8	3,84	13,17	2,98	0,0151

Як свідчать результати моделювання, наведені в табл. 1, точність прогнозу як алгоритму *BART*, так і *MLP* виявилась значно вищою для усіх лагів і рядів, ніж для алгоритму *RF*. Порівняно невисока точність прогнозування *RF* може бути обумовлена тим, що для його ефективної реалізації необхідне використання значно більшої кількості факторів. Можна побачити, що точність *RF* зростає при збільшенні глибини лагу.

Що стосується порівняння ефективності *MLP* і *BART*, то як свідчать дані рис. 6–8 та табл. 1, ці моделі показали точність одного порядку: найменша похибка для *BTC* склала 3,69 % (модель *MLP*), для *ETH* — 3,37 % (*BART*), для *XRP* — 2,98 % (*MLP*).

Деяко неочікуваною виявилось незначне зменшення точності як *MLP*, так і *BART* (принаймні для *BTC* та *ETH*) при збільшенні глибини лагу. На нашу думку, це може бути пов'язано з перенавчанням зазначених моделей.

Висновки та перспективи подальших досліджень

Одержані нами результати моделювання короткострокової динаміки криптовалют дозволяють зробити висновок щодо ефективності використання для вирішення подібних задач методів та алгоритмів машинного навчання, зокрема, моделей нейронних мереж, регресійних (авторегресійних) дерев та їх ансамблів. Згідно з результатами проведеного дослідження, ці моделі дозволяють зробити короткостроковий прогноз із похибкою в межах 3–4 %.

При цьому необхідно зауважити, що ми використовували лише мінімальний набір даних — тільки лагові значення досліджуваних рядів (цін закриття). Точність прогнозу можна підвищити, якщо використовувати більш повний дата сет, наприклад, включити до моделей такі фактори, як ціни відкриття, максимальна, мінімальна та середня ціна, обсяг торгів тощо. Окрім цього можна використовувати також різноманітні індекси, осцилятори, зокрема плинні середні різних порядків, що враховують динаміку тренду.

Щодо використання для задач прогнозування фінансових часових рядів моделей нейронних мереж, то в цій роботі ми застосували просту модель мережі перцептронного типу з одним прихованим шаром. Використання мереж із більш складною архітектурою: рекурентних, самоорганізованих, глибинного навчання тощо має також підвищити точність прогнозних розрахунків.

Підсумовуючи, зауважимо, що, на наш погляд, перспективним напрямом розвитку технологій прогнозування фінансових часових рядів є побудова комбінованих деревоподібних моделей і нейронних мереж, зокрема ансамблів моделей типу випадкового лісу, в якому в якості атомарних дерев використовують моделі *BART*, при цьому в якості елементарних предикторів у деревах типу *BART* можна використати замість регресій (авторегресій) мережі перцептронного типу.

Список літератури

1. Top Economists Stiglitz, Roubini and Rogoff Renew Bitcoin Doom Scenarios. *CNBC*: website. URL: <https://www.cnb.com/2018/07/09/nobel-prize-winning-economist-joseph-stiglitz-criticizes-bitcoin.html>.
2. Selmi R., Tiwari A., Hammoudeh S. Efficiency or speculation? A dynamic analysis of the Bitcoin market. *Economic Bulletin*. 2018. Vol. 38, Iss. 4. P. 2037–2046.
3. Cheah E., Fry J. Speculative bubbles in Bitcoin markets? An empirical investigation into the fundamental value of bitcoin. *Economic Letters*. 2015. Vol. 130. P. 32–36.
4. Ciaian P., Rajcaniova M., Kancs D. The economics of BitCoin price formation. *Applied Economics*. 2016. Vol. 48, Iss. 19. P. 1799–1815.
5. Catania L., Grassi S. Modelling Crypto-Currencies Financial Time-Series. *CEIS Research Paper*. 2017. Vol. 15, Iss. 8, No. 417. P. 1–39. URL: <https://ideas.repec.org/p/rtv/ceisrp/417.html>.
6. Flach P. Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge: Cambridge University Press, 2012. 400 p.
7. Bontempi G., Taieb S., Borgne Y. Machine Learning Strategies for Time Series Forecasting. *Business Intelligence. eBISS 2012. Lecture Notes in Business Information Processing*. Berlin, Heidelberg: Springer-Verlag, 2013. Vol. 138. P. 62–77. DOI: 10.1007/978-3-642-36318-4_3.
8. Persio L., Honchar O. Multitask machine learning for financial forecasting. *International Journal of Circuits, Systems and Signal Processing*. 2018. Vol. 12. P. 444–451.
9. Boyacioglu M., Baykan O. K. Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the İstanbul Stock. *Exchange Expert Systems with Applications*. 2011. Vol. 38, Iss. 5. P. 5311–5319.
10. Матвійчук А. В. Моделювання економічних процесів із застосуванням методів нечіткої логіки: Монографія. К.: КНЕУ, 2007. 264 с.
11. Okasha M. K. Using Support Vector Machines in Financial Time Series Forecasting. *International Journal of Statistics and Applications*. 2014. Vol. 4, № 1. P. 28–39. DOI: 10.5923/j.statistics.20140401.03.
12. Varghade P., Patel R. Comparison of SVR and Decision Trees for Financial Series Prediction. *International Journal on Advanced Computer Theory and Engineering*. 2012. Vol. 1, Iss. 1. P. 101–105.
13. Kumar M. Forecasting Stock Index Movement: A Comparison of Support Vector Machines and Random Forest. *SSRN Working Paper. Indian Institute of Capital Markets 9th Capital Markets Conference Paper*. 2006. P. 1–16. DOI: 10.2139/ssrn.876544.
14. Hitam N. A., Ismail A. R. Comparative Performance of Machine Learning Algorithms for Cryptocurrency. *ResearchGate*. 2018. P. 1–11. URL: <https://www.researchgate.net/publication/327415267>.

15. Akyildirim E., Goncu A., Sensoy A. Prediction of Cryptocurrency Returns using Machine Learning. *ResearchGate*. 2018. P. 1–28. URL: <https://www.researchgate.net/publication/329322600>.
16. Yao Y., Yi J., Zhai S., Lin Y., Kim T., Zhang G., Lee L. Y. Predictive Analysis of Cryptocurrency Price Using Deep Learning. *International Journal of Engineering & Technology*. 2018. Vol. 7, No. 3.27. P. 258–264. DOI: 10.14419/ijet.v7i3.27.17889.
17. Alessandretti L., ElBahrawy A., Aiello L., Baronchelli A. Anticipating Cryptocurrency Prices Using Machine Learning. *Complexity*. 2018. Vol. 2018. P. 1–16. DOI: 10.1155/2018/8983590.
18. Breiman L., Friedman H., Olshen R. A., Stone C. J. Classification and Regression Trees. Belmont: Wadsworth International Group, 1984. 368 p.
19. Breiman L. Random Forests. *Machine Learning*. 2001. № 45. P. 5–32. DOI: 10.1023/A:1010933404324.
20. Даценко Н.В. Застосування дерев класифікації та регресії до прогнозування часових рядів фінансових інструментів. *Вчені записки. Сер. Економіко-математичні методи*. 2018. № 19. С. 169–181.
21. Derbentsev V., Datsenko N., Stepanenko O., Bezkorovainyi V. Forecasting Cryptocurrency Prices Time Series Using Machine Learning. *CEUR Workshop Proceedings*. 2019. Vol. 2422. P. 320–334. URL: <http://ceur-ws.org/Vol-2422/paper26.pdf>.
22. Stock Market. *Yahoo Finance* : website. URL: <https://finance.yahoo.com> (Last accessed: 22.08.2019).

References

1. CNBC. (2018, July 9). *Top Economists Stiglitz, Roubini and Rogoff Renew Bitcoin Doom Scenarios*. Retrieved from <https://www.cnbc.com/2018/07/09/nobel-prize-winning-economist-joseph-stiglitz-criticizes-bitcoin.html>.
2. Selmi, R., Tiwari, A., & Hammoudeh, S. (2018). Efficiency or speculation? A dynamic analysis of the Bitcoin market. *Economic Bulletin*, 38(4), 2037–2046.
3. Cheah, E., & Fry, J. (2015). Speculative bubbles in Bitcoin markets? An empirical investigation into the fundamental value of bitcoin. *Economic Letters*, 130, 32–36.
4. Ciaian, P., Rajcaniova, M., & Kancs, D. (2016). The economics of Bitcoin price formation. *Applied Economics*, 48(19), 1799–1815.
5. Catania, L., & Grassi, S. (2017). Modelling Crypto-Currencies Financial Time-Series. *CEIS Research Paper*, 15(8), 1–39. Retrieved from <https://ideas.repec.org/p/rtv/ceisrp/417.html>.
6. Flach, P. (2012). *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge, UK: Cambridge University Press.

7. Bontempi, G., Taieb, S., & Borgne, Y. (2013). Machine Learning Strategies for Time Series Forecasting. In: M. A. Auafeure, & E. Zimányi (Eds.), *Business Intelligence. eBISS 2012. Lecture Notes in Business Information Processing*, 138 (pp. 62–77). Berlin, Heidelberg, Germany: Springer-Verlag. DOI: 10.1007/978-3-642-36318-4_3.

8. Persio, L., & Honchar, O. (2018). Multitask machine learning for financial forecasting. *International Journal of Circuits, Systems and Signal Processing*, 12, 444–451.

9. Boyacioglu, M., & Baykan, O.K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock. *Exchange Expert Systems with Applications*, 38(5), 5311–5319.

10. Matviychuk, A. V. (2007). *Modeliuvannia ekonomichnykh protsesiv iz zastosuvanniam metodiv nechiitkoi lohiky*. Kyiv, Ukraine: KNEU [in Ukrainian].

11. Okasha, M. K. (2014). Using Support Vector Machines in Financial Time Series Forecasting. *International Journal of Statistics and Applications*, 4(1), 28–39. DOI: 10.5923/j.statistics.20140401.03.

12. Varghade, P., & Patel, R. (2012). Comparison of SVR and Decision Trees for Financial Series Prediction. *International Journal on Advanced Computer Theory and Engineering*, 1(1), 101–105.

13. Kumar, M. (2006). Forecasting Stock Index Movement: A Comparison of Support Vector Machines and Random Forest. *SSRN Working Paper. Indian Institute of Capital Markets 9th Capital Markets Conference Paper*, 1–16. DOI: 10.2139/ssrn.876544.

14. Hitam, N. A., & Ismail, A. R. (2018). Comparative Performance of Machine Learning Algorithms for Cryptocurrency Forecasting. *ResearchGate*, 1–11. Retrieved from <https://www.researchgate.net/publication/327415267>.

15. Akyildirim, E., Goncu, A., & Sensoy, A. (2018). Prediction of Cryptocurrency Returns using Machine Learning. *ResearchGate*, 1–28. Retrieved from <https://www.researchgate.net/publication/329322600>.

16. Yao, Y., Yi, J., Zhai, S., Lin, Y., Kim, T., Zhang, G., & Lee, L. Y. (2018). Predictive Analysis of Cryptocurrency Price Using Deep Learning. *International Journal of Engineering & Technology*, 7(3.27), 258–264. DOI: 10.14419/ijet.v7i3.27.17889.

17. Alessandretti, L., ElBahrawy, A., Aiello, L., & Baronchelli, A. (2018). Anticipating Cryptocurrency Prices Using Machine Learning. *Complexity*, 2018, 1–16. DOI: 10.1155/2018/8983590.

18. Breiman, L., Friedman, H., Olshen, R. A., & Stone, C. J. (1984). *Classification and Regression Trees*. Belmont, NJ: Wadsworth International Group.

19. Breiman, L. (2001). Random Forests. *Machine Learning*, 45, 5–32. DOI: 10.1023/A:1010933404324.

20. Datsenko, N.V. (2018). Zastosuvannia derev klasyfikatsii ta rehresii do prohnozuvannia chasovykh riadiv finansovykh instrumentiv. *Vcheni zapysky. Seriiia Ekonomiko-matematychni metody (Scientific notes. Series of Economic and Mathematical Methods)*, 19, 169-181 [in Ukrainian].

21. Derbentsev, V., Datsenko, N., Stepanenko, O., & Bezkorovainyi, V. (2019). Forecasting Cryptocurrency Prices Time Series Using Machine Learning. *CEUR Workshop Proceedings*, 2422, 320–334. Retrieved from <http://ceur-ws.org/Vol-2422/paper26.pdf>.

22. Yahoo Finance. (2019). *Stock Market*. Retrieved from <https://finance.yahoo.com>.

Стаття надійшла до редакції 28.08.2019