

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО

Факультет інформатики та обчислювальної техніки
(назва факультету, інституту)

Кафедра автоматизованих систем обробки інформації і управління
(назва кафедри)

"На правах рукопису"
УДК 004.02:519.246.8

«До захисту допущено»
Завідувач кафедри

_____ О.А.Павлов
(підпис) (ініціали, прізвище)

“ _____ ” _____ 20 18 р.

МАГІСТЕРСЬКА ДИСЕРТАЦІЯ

на здобуття ступеня магістра

за спеціальністю 122 Комп'ютерні науки та інформаційні технології
(код та назва спеціальності)

спеціалізацією Інформаційні управляючі системи та технології
(код та назва спеціалізації)

на тему: Фінансове прогнозування часових рядів

Виконав: студент VI курсу групи ІС-61м
(шифр групи)

Перерва Антон Сергійович
(прізвище, ім'я, по батькові)

_____ (підпис)

Науковий керівник

доц., к.т.н., доц. Ковалюк Т. В.
(посада, науковий ступінь, вчене звання, прізвище та ініціали)

_____ (підпис)

Консультант

к.т.н., доц. Жданова О.Г.
(науковий ступінь, вчене звання, прізвище, ініціали)

_____ (підпис)

Рецензент

_____ (посада, науковий ступінь, вчене звання, прізвище та ініціали)

_____ (підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент

_____ (підпис)

Київ – 2018

РЕФЕРАТ

Магістерська дисертація: 93 с., 11 рис., 3 табл., 7 додатків, 50 джерел.

Актуальність теми. Завдяки стрімкому розвитку інформаційних технологій, з'явилася можливість за лічені секунди проводити аналіз великого обсягу інформації, будувати складні математичні моделі. Дослідники, які займаються питаннями циклічного розвитку економіки, стали розробляти теорії, вважаючи, що відстеження тенденцій ряду економічних змінних дозволить прояснити і передбачити періоди підйому і спаду. Одним з об'єктів для вивчення був обраний фондовий ринок. При створенні систем аналізу фінансових ринків і розробці прогнозних моделей наряду з іншими виникають дві принципові проблеми: визначення необхідних і достатніх параметрів для оцінки стану ринку, а також цільових функцій, тобто вибір критеріїв ефективності дій; проблема розмірності – бажання врахувати в моделі якомога більше показників і критеріїв оцінки може призвести до практичної неможливості обчислень пов'язаної з обсягами обчислювальної складності.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського» в рамках теми «Інформаційна підтримка взаємодії ІТ освіти та ІТ індустрії в Україні» (Державний реєстраційний номер 0117U000917).

Метою дослідження є виявлення закономірностей у показниках ринкових котирувань, визначення напрямку у якому буде рухатись ринок.

Завдання дослідження:

- охарактеризувати наявні методи аналізу ринкових котирувань;

- проаналізувати наявні методи аналізу ринкових котирувань;
- визначити ефективність наявних методів аналізу ринкових котирувань;
- визначити ознаки для створення нового методу аналізу ринкових котирувань;
- розробити метод аналізу ринкових котирувань;
- визначити ефективність створеного рішення.

Об’єкт дослідження – котирування на фінансовому ринку, які представлені у вигляді часового ряду.

Предмет дослідження – методи аналізу ринкових котирувань.

Методи дослідження застосовані в роботі, базуються на методах теорії штучних нейронних мереж, еволюційних обчислень, дискретної математики, основних положеннях теорії ймовірності та математичної статистики, генетичному алгоритмі.

Наукова новизна отриманих результатів. Розроблено алгоритм, що дозволяє визначити параметри конфігурації ефективної нейромережі для прогнозування фінансових часових рядів.

Публікації. За напрямком дослідження було опубліковано такі матеріали: “Огляд методів аналізу фінансових часових рядів”; “Порівняльний аналіз методів прогнозування стаціонарних та нестаціонарних часових рядів”; “Застосування генетичного алгоритму для визначення ефективної конфігурації штучних нейронних мереж”.

ЧАСОВІ РЯДИ, ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ, МЕТОДИ ВІДБОРУ ІНФОРМАТИВНИХ ДАНИХ, НЕЙРОННІ МЕРЕЖІ, ГЕНЕТИЧНИЙ АЛГОРИТМ

ABSTRACT

Master's Thesis: 93 pages, 11 images, 3 tables, 7 attachments, 50 references.

Research rationale. Due to the rapid development of information technologies, it becomes possible to analyze a large amount of information in a few seconds, build complex mathematical models and solve problems of multi-criteria optimization. Researchers, who are involved in cyclical economic development, began to theorize, assuming that tracking of the number's trends in economic would clarify and predict periods of recovery and decline. One of the objects of study is stock market. Creating systems for financial market analysis and predictive models development with other problems involves solving of two fundamental problems: identification and determination of required and sufficient parameters for assessing the state of the market, as well as target functions, which includes, selection of criteria for the actions' efficacy. Formalization is the simulation of the behavior of a system consisted of heterogeneous components. And it requires the use of a single metric for their description; dimensional problem – the desire to consider in the model as many indicators and evaluation criteria as possible can lead to a practical impossibility due to computational complexity.

Connection of research with scientific programs, plans, topics. The work was performed at the the department of Computer-Aided Management And Data Processing Systems (ASOIU) of the National Technical University of Ukraine "Kyiv Polytechnic Institute. Igor Sikorsky" within the frame of the topic "Information support of the interaction of IT education and IT industry in Ukraine" (State registration number 0117U000917).

The goal of the research is to identify the patterns in the indicators of market stock quotes, to determinate the direction in which the market will move.

Research tasks:

- to characterize existing methods of market quotation analysis;
- to evaluate available methods of market quotation analysis;
- to determine the efficiency of existing market quotation analysis methods;
- to identify features to create a new method for analyzing market quotes;
- to develop a new method for market quotation analysis;
- to determine the effectiveness of the method.

Object of research – quotations on the stock market, which are presented in the form of a time series.

Subject of research – methods of analysis of market quotations.

Research involves **methods** based on the methods of the theory of artificial neural networks, evolutionary calculations, discrete mathematics, the fundamentals provisions of the theory of probability and mathematical statistics, genetic algorithm.

The scientific novelty of the study. The algorithm is developed, which allows to determine the parameters of the configuration of effective neural network for prediction of financial time series.

Publications. The following publications were produced from described research: “An overview of methods for financial time series analysis”; “Comparative analysis of forecasting methods of stationary and non-stationary time series”; “Application of the genetic algorithm to determine the effective configuration of artificial neural networks”.

TIME SERIES, INTELLECTUAL ANALYSIS OF DATA, SELECTION METHODS FOR INFORMATIVE DATA, NEURAL NETWORKS, GENETIC ALGORITHM.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ	9
ВСТУП	10
1 ВИБІР НАПРЯМКІВ ДОСЛІДЖЕННЯ (СУЧАСНИЙ СТАН ПРОБЛЕМИ)	12
1.1 Модель оцінки капітальних активів.....	14
1.2 Лінія ринку цінних паперів.....	16
1.3 Технічний аналіз.....	17
1.4 Фундаментальний аналіз.....	21
1.5 Основні форми ефективності ринку.....	26
1.6 Висновок.....	28
2 МЕТОДИ ТА ЗАСОБИ АНАЛІЗУ ФІНАНСОВИХ ЧАСОВИХ РЯДІВ	30
2.1 Модель ціноутворення капітальних активів.....	30
2.2 Авторегресійні моделі.....	31
2.2.1 Авторегресійна модель (AR).....	31
2.2.2 Модель ковзаючого середнього.....	32
2.2.3 Модель авторегресії - ковзаючого середнього (ARMA).....	32
2.2.4 Модель Бокса-Дженкінса (ARIMA).....	33
2.3 Аналіз сингулярного спектру (SSA).....	34
2.4 Пошук прихованих шаблонів (hidden T-patterns).....	37
2.4.3 Формалізований опис.....	40
2.5 Нейромережеві методи.....	41
2.5.1 Штучна нейронна мережа.....	42
2.5.2 Штучний нейрон.....	44
2.5.3 Перцептрон.....	46
2.5.4 Нейромережа зворотного поширення похибки.....	47
2.5.5 Мережа Кохонена.....	50
2.5.6 Нейронна мережа Гопфілда.....	50
2.6 Не математичні методи.....	52
2.7 Аналіз.....	53
2.8 Висновок.....	59

3 ВИЗНАЧЕННЯ ЕФЕКТИВНОЇ КОНФІГУРАЦІЇ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ	60
3.1 Вхідні дані.....	60
3.2 Застосування генетичного алгоритму для вирішення задачі конфігурації нейронної мережі.....	62
3.3 Математична постановка задачі.....	63
3.4 Практичні аспекти застосування алгоритму.....	64
3.5 Висновок.....	65
4 ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ПРОДУКТУ	66
4.1 Загальні вимоги до архітектури програмного забезпечення.....	66
4.2 Вхідні та вихідні дані.....	70
4.3 Структура класів програмного забезпечення.....	71
4.4 Керівництво користувача.....	72
4.5 Висновок.....	75
5 АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ	76
5.1 Вхідні дані.....	76
5.2 Інтерпретація отриманих результатів.....	76
5.3 Висновки.....	78
ВИСНОВКИ	79
ПЕРЕЛІК ПОСИЛАНЬ	81
ДОДАТОК А Графічний матеріал	86
Порівняння підходів	87
Математична модель	88
Схема алгоритму	89
Архітектура програмного продукту	90
Діаграма класів	91
Результати експерименту	92
Екранні форми	93

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

ШНМ	Штучна нейронна мережа
ЧШ	Часовий шаблон
РНМ	Рекурентні нейронні мережі
ГА	Генетичний алгоритм

ВСТУП

Завдяки стрімкому розвитку інформаційних технологій, з'явилася можливість за лічені секунди проводити аналіз великого обсягу інформації, будувати складні математичні моделі. Дослідники, які займаються питаннями циклічного розвитку економіки, стали розробляти теорії, вважаючи, що відстеження тенденцій ряду економічних змінних дозволить прояснити і передбачити періоди підйому і спаду. Одним з об'єктів для вивчення був обраний фондовий ринок.

Серед факторів, що характеризують динаміку ринку є неабияка кількість даних нечислової природи, значення яких відомі тільки з певною часткою впевненості / вірогідності. Можна виділити різні типи невизначеностей, з яких для фінансового аналізу важливі наступні:

- пов'язані з незнанням або неточним знанням деяких факторів або процесів, що впливають на розвиток ситуації;
- пов'язані з математичною неспівмірністю чисельних оцінок величин, що характеризують динаміку системи;
- пов'язані з нелінійністю і наявністю у системи декількох станів рівноваги;
- пов'язані з нестачею або неадекватністю понятійного апарату і неможливістю ототожнення фактів.

При створенні систем аналізу фінансових ринків і розробці прогнозних моделей виникають три принципові проблеми:

- визначення необхідних і достатніх параметрів для оцінки стану ринку, а також цільових функцій, тобто вибір критеріїв ефективності дій. Формалізація, тобто моделювання поведінки системи, що складається з різнорідних компонентів, вимагає використання єдиної метрики для їх опису.

- проблема розмірності. Бажання врахувати в моделі якомога більше показників і критеріїв оцінки може призвести до збільшення обчислювальної складності та неможливості виконання розрахунків за розумний час. Іншими словами, суть цієї проблеми зводиться до обмеження на швидкодію і розміри обчислювального комплексу в залежності від кількості інформації, що обробляється в одиницю часу.
- прояв ознак надсистемності. Відомо, що взаємодіючі системи утворюють надсистему - систему більш високого рівня, яка має власні (надсистемні) властивості, яких не має жодна зі складових систем. Проблема полягає у принциповій неможливості виявити зазначені прояви надсистемного відображення засобами, що входять до складу взаємодіючих систем.

Нові підходи до прогнозування з'явилися саме з метою подолання деяких з перелічених проблем. Ці підходи базуються на застосуванні таких розділів сучасної математики, як еволюційний, теорія стохастичного моделювання (теорія хаосу), теорія катастроф, синергетика і теорія систем, що самоорганізуються, включаючи генетичні алгоритми, теорію фракталів (клітинні автомати) і нечіткі логіки. Вважається, що ці методи дозволять збільшити глибину прогнозу на фінансових ринках за рахунок виявлення прихованих закономірностей, властивих цим ринкам.

1 ВИБІР НАПРЯМКІВ ДОСЛІДЖЕННЯ (СУЧАСНИЙ СТАН ПРОБЛЕМИ)

Фінансовий ринок відображає гіпотезу ефективного ринку, однак ефективний ринок - інституціональна структура, в рамках якої сторони, що обмінюються, отримують вигоду від торгівлі в світі з нульовими транзакційними витратами. Але оскільки укладання угод тягне за собою витрати на всіх ринках, то ефективні ринки (як вони визначаються в неокласичних моделях) недосяжні. Тому ефективність ринку, як і всіх економічних інститутів, визначається мінімізацією транзакційних витрат, які несуть суб'єкти в умовах взаємодії в рамках цього інституту.

В аналізі ринку однією з базових концепцій є гіпотеза ефективного ринку - гіпотеза, згідно з якою вся суттєва інформація негайно і в повній мірі відбивається на ринковій курсовій вартості цінних паперів. У 60-х роках ХХ століття Юджин Фама і Пол Самуельсон незалежно один від одного сформулювали гіпотезу ефективного ринку, основою якої є твердження, що ринкові ціни відображають всю доступну інформацію [1].

Важливо виділити основні положення на яких базується дана теорія [2]:

- всі учасники ринку мають рівний доступ до історичних даних про ринкові ціни, публічно доступної інформації і приватної інформації; таким чином, жоден з учасників ринку не має переваги перед іншими в прийнятті інвестиційних рішень;
- спочатку передбачається, що всі учасники ринку діють раціонально, тобто не приймають ризик, якщо він не компенсується адекватним рівнем прибутковості;
- жоден з учасників ринку не зможе отримати в довгостроковій перспективі прибутковість, яка буде істотно перевищувати ринкову для заданого рівня ризику;

- ринкові ціни змінюються випадковим чином, тому минулі тенденції або патерни не дозволяють судити про їх рух в майбутньому. Іншими словами, в умовах ефективного ринку застосування інструментів технічного і фундаментального аналізу є абсолютно марним.

Хоча запропонована теорія не була спростована, її основні положення зазнали серйозної критики.

На реальних ринках доступ до всієї інформації мають не всі учасники. Частина інформації завжди залишається приватною, тобто доступною, як правило, вузькому колу осіб, яких називають інсайдерами. До них, наприклад, можуть відноситися топ-менеджери корпорацій, високопоставлені співробітники уряду і центрального банку, які в силу свого службового становища мають доступ до інформації, яка недоступна широкому загалу. І хоча законодавство зазвичай містить норми, покликані перешкоджати інсайдерській торгівлі, такі особи мають перевагу перед іншими учасниками ринку, що підриває основні положення гіпотези ефективного ринку.

Практична психологія також обґрунтовано ставить під сумнів твердження про те, що всі люди діють раціонально. Більш того, в учасників ринку може також бути різна схильність до ризику. Це призводить до того, що в одній і тій же ситуації інвестори можуть обрати різні, а часто і взаємовиключаючі рішення, що також суперечить основним положенням гіпотези ефективного ринку.

На реальних ринках періодично виникають патерни (шаблони) [3, 4], звані також фігурами аналізу, які добре відомі інвесторам і дозволяють з високою ймовірністю зробити припущення про подальший рух цін. Наприклад, до таких патернів належать «голова і плечі», «подвійна вершина», «плоске дно», «трикутник» і т.д.

З огляду на зазначену вище критику, стає очевидним, що в чистому вигляді ефективний ринок не може існувати в реальному світі, тому в результаті подальшого доопрацювання теорії були виділені три основних форми ефективності ринку, що розрізняються за ступенем доступу до інформації.

Однак перш ніж розглянути форми ефективності ринку, необхідно ознайомитися з поняттями “модель оцінки капітальних активів”, “лінія ринку цінних паперів”, “технічний аналіз” та “фундаментальний аналіз”.

1.1 Модель оцінки капітальних активів

У фінансах модель оцінки капітальних активів (модель CAPM [6]) застосовується для визначення необхідної норми прибутковості активу або портфеля активів. Модель CAPM враховує систематичний ризик, мірою якого виступає бета-коефіцієнт, очікувану ринкову прибутковість і без ризикову процентну ставку.

Більшість вихідних положень моделі будуються, виходячи з гіпотези ефективності ринку [2]:

- всі інвестори діють раціонально і не схильні до ризику, маючи на меті максимізацію свого добробуту на кінець періоду;
- ринки є досконалими: модель оцінки капітальних активів не враховує транзакційні витрати, податки, інфляцію і існуючі обмеження на короткий продаж;
- всі інвестори можуть отримувати або надавати необмежене фінансування за без ризиковою процентною ставкою;
- всі інвестори мають однаковий доступ до інформації, а середньоквадратичне відхилення прибутковості активу є єдиним показником міри ризику. Отже, у всіх інвесторів будуть однакові очікування щодо прибутковості певного активу;

- всі активи нескінченно дробляться і є абсолютно ліквідними;
- загальна кількість активів на ринку є фіксованим в межах певного періоду часу;
- розподіл прибутковості активів є нормальним або близьким до нормального;
- всі ринки знаходяться в рівновазі, і жоден учасник самостійно не може вплинути на ціну активу.

Хоча модель САМР є досить простою в застосуванні, багато її вихідних положень повністю або частково не виконуються на реальних ринках:

- відсутність транзакційних витрат. Всі угоди на реальних ринках припускають наявність транзакційних витрат, причому їх рівень може істотно відрізнятись для різних учасників ринку. Наприклад, для великих інституційних інвесторів вони будуть значно нижче, ніж для дрібних приватних інвесторів, за рахунок ефекту масштабу діяльності;
- нульові ставки оподаткування. Сучасні системи оподаткування можуть бути досить складними, особливо щодо фінансових інвестицій. Податок на приріст капіталу, податок на дивіденди, відстрочений податок можуть мати різні ставки, що буде стимулювати інвесторів формувати свої портфелі таким чином, щоб мінімізувати витрати, пов'язані з виплатою податків. Все це знижує ефективність інвестицій і має суттєвий вплив на ціноутворення активів;
- однорідні очікування інвесторів. Така ситуація можлива тільки при наявності абсолютно ефективного ринку, що не зустрічається на практиці. Однак слід зазначити, що деякі ринки характеризуються високим ступенем ефективності;

- можливість інвестування в безризикові активи. Наявність безризикових цінних паперів є одним з базових припущень моделі оцінки капітальних активів.

Отримуючи додаткове фінансування під безризикову процентну ставку, інвестори збільшують частку ризикових активів в своїх портфелях. Однак в реальній практиці вартість залучення фінансування для дрібних інвесторів, як правило вище, ніж для великих інституційних.

Бета-коефіцієнт є повною мірою ризику. Модель CAPM передбачає, що єдиною мірою ризику є бета-коефіцієнт, який характеризує волатильність (тенденція до змін) дохідності цінного паперу щодо волатильності прибутковості ринкового портфеля. Однак на практиці існує множина інших видів ризику, які мають істотний вплив на ціноутворення активів і вибір інвесторів: ризик інфляції, ризик ліквідності, ризик реінвестування.

Розподіл прибутковості активів є нормальним або близьким до нормального. На практиці розподіл прибутковості активів є близьким до нормального в дуже рідкісних випадках, що також впливає на вибір інвесторів при формуванні портфелів.

1.2 Лінія ринку цінних паперів

Лінія ринку цінних паперів є графічною інтерпретацією залежності ризику окремого цінного паперу [17], мірою якого виступає бета-коефіцієнт, і нормою прибутковості, яку вимагатимуть інвестори за його прийняття. При цьому, чим вище буде рівень прийнятого ризику, тим більша компенсація повинна бути запропонована інвестору.

Основною проблемою практичного застосування лінії ринку цінних паперів є те, що вона базується на тих же вихідних положеннях, що і модель оцінки капітальних активів CAPM. В силу тих обставин, що реальні ринки не характеризуються абсолютним ступенем ефективності, різні інвестори мають

різні можливості по залученню додаткового фінансування (як за обсягом, так і за процентними ставками), а податки і транзакційні витрати значно впливають на формування індивідуального портфеля, множина доступних на ринку цін являють собою не пряму лінію, а нечітку сукупність. Також однією з основних причин такої ситуації є те, що бета-коефіцієнт використовується в якості повної міри ризику, пов'язаного з інвестуванням в певний цінний папір. На реальних ринках існують і інші ризики, які впливають на необхідну норму прибутковості, і призводять до зсуву окремого цінного паперу від лінії ринку цінних паперів.

1.3 Технічний аналіз

Технічний аналіз - сукупність методик дослідження динаміки ринку на основі закономірностей зміни цін в минулому у аналогічних обставинах, зазвичай засобом графіків, з метою прогнозування майбутнього напрямку руху цін. Теоретично, технічний аналіз може бути застосований на будь-якому ринку. Але найбільшого поширення технічний аналіз отримав на високоліквідних вільних ринках, наприклад на біржах.

Технічні аналітики не намагаються виміряти внутрішню вартість цінного паперу, а замість цього використовують діаграми і інші інструменти, щоб ідентифікувати патерни (фігури аналізу), які можуть допомогти спрогнозувати майбутній розвиток ринку.

На початку 20 століття теорія Доу заклала основу для того, що надалі стало сучасним технічним аналізом [4]. Теорія Доу не була якимось одним документом, а була зібрана з численних записів Чарльза Доу. З теорем, висунутих Доу, варто відзначити три:

- ціна враховує все;
- рух цін - не хаотичний;
- «що» - більш важливо, ніж «чому».

Ця теорема схожа з сильною і відносною формами ефективності ринку [5]. Технічні аналітики вважають, що поточна ціна повністю відображає всю доступну інформацію. Оскільки вся інформація вже відображена в ціні, ціна є справедливою і повинна бути основою для аналізу. Зрештою, ринкова ціна відображає знання всіх учасників торгів, включаючи трейдерів, інвесторів, менеджерів портфелів, аналітиків з боку покупців і продавців, стратегів, технічних аналітиків, фундаментальних аналітиків і багатьох інших. Було б нерозумно і недоречно не погодитися з ціною, сформованою такою великою кількістю фахівців і професіоналів. Технічний аналіз використовує інформацію, зафіксовану ціною, для інтерпретування того, про що «говорить» ринок, з метою прогнозування.

Більшість технічних аналітиків погоджуються з тим, що ціни рухаються направлено. Разом з цим, багато фахівців також погоджуються з тим, що є періоди, коли ціни рухаються не направлено. Якби ціни були завжди хаотичні, то було б вкрай важко заробити гроші за допомогою технічного аналізу. У книзі “Технічний аналіз” Джек Швагер [4] пише, що “ринки складаються з довгих періодів хаотичних коливань, і коротких періодів спрямованого не хаотичного руху. Завдання фахівця з технічного аналізу - виявити і відрізнити ці періоди”.

Ідея технічного аналізу полягає в тому, що можна розпізнати тренд, інвестувати або торгувати в залежності від тренда, і заробити грошей при виконанні (реалізації) тренда. Оскільки технічний аналіз застосовується до різних періодів, з його допомогою можна розпізнати як довгострокові, так і короткострокові тренди.

У розумінні технічного аналізу, ціна – це кінцевий результат бою між силами попиту і пропозиції за акції компанії. Основна мета технічного

аналізу – прогнозування напряму руху ціни, спираючись виключно на історичні дані – попередні значення раду.

Значна кількість технічних аналітиків використовують метод зверху-вниз, який починається з великого макроаналізу – дослідження на великих часових проміжках (роки). Великі обсяги інформації розкладаються на дрібні, і так триває поки аналітик не проаналізує мікродані – проміжки довжиною у декілька годин. Такий підхід складається з трьох фаз:

- великий аналіз ринку, з використанням індексів (такі як S & P 500, Dow Industrials, NYSE Composite, NASDAQ і ін);
- аналіз конкретної галузі для виявлення сильніших і слабших груп інструментів в межах галузі;
- аналіз конкретного інструменту по черзі для виявлення сильніших і слабших інструментів в даній групі.

Одна з позитивних рис технічного аналізу – це його багатосторонність. Оскільки принципи технічного аналізу застосовні практично до багатьох предметних областей, у яких дані можна подати за допомогою графіків, адже кожен з перерахованих кроків можна виконати за допомогою однієї і тієї ж методики і теорії.

У технічному аналізі вводять поняття технічного індикатора - це функція, побудована на значеннях статистичних показників торгів (ціни, обсяг торгів і т. д.), аналіз поведінки якої покликаний відповісти на питання зміниться або збережеться поточна тенденція на ринку. Можна виділити три основні класи індикаторів:

- індикатори тенденцій - підтвердження тенденції;
- осцилятори - передбачення зміни тенденції;
- каналні індикатори;

Існує велика множина різних індикаторів, деякими з яких є [8]:

- канал Дончиана;
- KST;
- Williams %R (процентний діапазон);
- Trix;
- адаптивна змінна середня Кауфмана (AMA / КАМА / АМкА);
- балансовий обсяг (OBV);
- індекс Армса (Arms' Short-Term Trading Index);
- індекс грошового потоку (MFI);
- індекс маси (MI);
- індекс накопичення / розподілу;
- індекс відносної сили (RSI);
- індекс підсумовування Макклеллана (MCSI);
- індекс негативного обсягу (NVI);
- індекс позитивного обсягу (PVI);
- індекс товарного каналу (CCI);
- індикатор MACD;
- індикатори настрою ринку;
- канал Кельтнера;
- крива Коппока;
- легкість руху (EMV);
- лінія зростання / падіння (англ. ADL, англ. Advance-decline line);
- метод хвильової симетрії;
- лінії тренда;
- тренд ціни і обсягу (PVT / VPT);
- японські свічки.

Аналіз проводять за допомогою пошуку ознак появи фігур технічного аналізу на графіку, наприклад:

- висхідний трикутник;
- низхідний трикутник;
- симетричний трикутник;
- вимпел;
- голова та плечі;
- бриліант;
- подвійна вершина;
- потрійна вершина;
- лінія Ганна;
- сітка Ганна;
- дуги Фібоначі;
- канал лінійної регресії.

1.4 Фундаментальний аналіз

Фундаментальний аналіз - це метод оцінки цінних паперів, що має на увазі під собою спробу виміряти її внутрішню вартість шляхом дослідження пов'язаних з нею економічних, фінансових та інших якісних і кількісних факторів. Фундаментальні аналітики намагаються вивчити все, що може впливати на вартість цінного паперу, включаючи макроекономічні чинники (як стан всієї економіки і умови окремої галузі) і специфічні фактори, що впливають на діяльність конкретної компанії (фінансовий стан, менеджмент і т.д.).

Мета виконання фундаментального аналізу [9] полягає в тому, щоб отримати вартість, яку інвестор може порівняти з поточною ціною цінного паперу. Ця вартість допоможе визначити, яку позицію слід зайняти по даному цінному папері. Якщо вона є недооціненою (вартість, отримана в результаті проведення фундаментального аналізу, нижче поточної ринкової), то інвестору слід зайняти довгу позицію (купити цей цінний папір). І навпаки,

якщо цінний папір є переоціненою, то краще зайняти по ній коротку позицію (тобто продати її).

Цей метод аналізу цінних паперів, є повною протилежністю технічного аналізу і намагається відповісти на питання “чому”, а не “як” зміниться ціна.

Фундаментальний аналіз використовує реальні дані, щоб оцінити вартість цінного паперу. Хоча більшість аналітиків використовує фундаментальний аналіз для оцінки акцій, цей метод може використовуватися практично для будь-якого типу цінних паперів.

Наприклад, інвестор може виконати фундаментальний аналіз вартості облігації, розглядаючи економічні чинники, такі як процентні ставки і стан економіки, інформація про емітента облігації і потенційні зміни його кредитного рейтингу. Для оцінки акцій цей метод використовує такі показники, як дохід, виручка, майбутнє зростання, рентабельність акціонерного капіталу, маржа прибутку та інші дані, щоб визначити базову вартість компанії і потенціал для її майбутнього зростання. З точки зору акцій фундаментальний аналіз зосереджується на фінансових звітах оцінюваної компанії.

Один з найвідоміших і успішних фундаментальних аналітиків є Уоррен Баффетт [8], який відомий успішним використанням фундаментального аналізу при виборі цінних паперів. Його здібності перетворили його в мільярдера.

У фундаментальному аналізі виділяють безліч показників, які в свою чергу можна розбити на групи:

- а) показники руху торгового та інвестиційного капіталу:
 - 1) торговий баланс;
 - 2) баланс поточних рахунків;
 - 3) баланс капітальних рахунків;

- 4) платіжний баланс;
- б) показники, що характеризують стан фінансового ринку країни:
 - 1) динаміка фондових індексів;
 - 2) сума розміщених облігацій казначейства США і ставка і динаміка обсягу та ставки розміщення;
 - 3) динаміка прибутковості по середньострокових і довгострокових державних цінних паперів;
 - 4) динаміка процентних ставок на міжбанківському кредитному ринку;
 - 5) динаміка ставок за іпотечними кредитами;
- в) агреговані макроекономічні показники:
 - 1) валовий внутрішній продукт;
 - 2) обсяг і динаміка державних витрат;
 - 3) обсяг і динаміка надходжень до бюджету країни;
 - 4) дефіцит/профіцит бюджету країни;
 - 5) сукупне споживання;
 - 6) сукупні приватні інвестиції;
 - 7) рівень особистих заощаджень;
 - 8) обсяг експорту;
 - 9) обсяг імпорту;
 - 10) безробіття;
- г) показники динаміки виробництва і торгівлі:
 - 1) індекс промислового виробництва;
 - 2) індекс використання виробничих потужностей;
 - 3) обсяг промислових замовлень;
 - 4) промислові запаси;
 - 5) замовлення на товари тривалого користування;

- б) роздрібні продажі;
 - 7) обсяг комерційного кредиту;
 - 8) обсяг кредитів під нерухомість;
 - 9) обсяг виданих споживчих кредитів;
- д) показники трудової статистики:
- 1) продуктивності праці;
 - 2) різні показники зміни кількості зайнятих і безробітних;
- е) показники інфляції:
- 1) індекс споживчих цін;
 - 2) індекс промислових цін;
 - 3) індекс споживчих енергетичних цін;
 - 4) індекс цін експорту;
 - 5) індекс цін на імпорт;
 - 6) середня погодинна заробітна плата;
 - 7) індекс вартості робочої сили (витрат компаній на оплату праці співробітників);
 - 8) динаміка цін на нафту;
- ж) індикатори кредитно-грошового регулювання економіки:
- 1) облікова ставка центрального банку країни;
 - 2) плановані обсяги розміщення та погашення державних цінних паперів;
 - 3) плановані обсяги залучення або погашення урядом зовнішніх кредитів (динаміка державного боргу);
 - 4) показники пропозиції грошей (грошова маса): M1, M2, M3.
- з) індикатори ділового циклу, а також індекси економічної (ділової і споживчої) активності і оптимізму:
- 1) індекс випереджальних економічних індикаторів;

- 2) індекс співпадаючих економічних індикаторів;
 - 3) індекс запізнених економічних індикаторів;
 - 4) індекс ділового оптимізму менеджерів із закупівель;
 - 5) індекс ділового оптимізму менеджерів підприємств сфери послуг;
 - 6) індекс споживчої довіри (оптимізму);
 - 7) індекс ділової довіри (оптимізму);
- к) показники будівельної галузі:
- 1) кількість споруджуваних будинків;
 - 2) кількість виданих дозволів на будівництво;
 - 3) продажі нових будинків;
 - 4) продажу ненових будинків;
 - 5) витрати на будівництво;

У процесі прийняття рішень необхідно аналізувати, з огляду на наступні фактори:

- що і яким чином даний технічний індикатор вимірює, на основі яких даних він розраховується, яка волатильність при його виході, в яких одиницях він розраховується (номінальні одиниці, динаміка в порівнянні з попереднім періодом або до якогось періоду минулого);
- до якого класу належить показник (випереджає по відношенню до економічного циклу, запізнюється або співпадає);
- яка періодичність і час виходу показника;
- публікується він один раз або існують також його попередні, і остаточні, переглянуті значення;
- які діапазони коливання показника в різні періоди економічного циклу країни, а також максимальні і мінімальні значення за весь історичний період;
- який його основний вплив на інші фундаментальні показники;

– які фундаментальні показники надають критичний вплив.

1.5 Основні форми ефективності ринку

Розглянемо три основні форми ефективності ринку. Основним постулатом технічного аналізу можна вважати наступне твердження: “Ціни враховують всю інформацію” [4]. Іншими словами, аналітик, який володіє історичними даними про ринкові ціни може шляхом проведення аналізу виявити тенденції і спрогнозувати їхню поведінку в майбутньому. Однак при слабкій формі ефективності ринку це твердження не є вірним, оскільки в цьому випадку рух ринкових цін носить випадковий характер, і інвестор не зможе стабільно отримувати прибуток в довгостроковій перспективі, засновуючи свої дії виключно на результатах технічного аналізу.

Отже, слабка форма ефективності ринку передбачає, що всі учасники ринку мають рівний доступ до всієї минулої публічно доступної інформації, а вона повністю врахована в поточних ринкових цінах. Таким чином, руху цін в минулому не дають інвестору уявлення про їх рух в майбутньому, і в значній мірі залежать від факторів, які не були враховані в минулих рухах цін.

Середня форма ефективності ринку передбачає, що поточні ринкові ціни відображають всю публічно доступну інформацію, при цьому частина інформації залишається приватною, тобто доступна обмеженому, як правило, вузькому колу осіб. Оскільки всі учасники ринку одночасно отримують публічно доступну інформацію, то жоден з них не зможе отримати на цьому додатковий дохід. Іншими словами, учасники ринку, які мають доступ виключно до публічно доступної інформації, не можуть розраховувати на прибутковість, що перевищує ту, яку показує лінія ринку цінних паперів.

Однак на такому ринку існують інсайдери - особи, які мають доступ до приватної інформації, до яких, наприклад, відносяться головний виконавчий директор і члени ради директорів. Завдяки своїй перевазі в доступі до

інформації перед іншими учасниками ринку вони можуть отримати прибутковість, що істотно перевищує ту, яку показує лінія ринку цінних паперів.

Таким чином, використання технічного і фундаментального аналізу стосовно публічно доступної інформації не дозволяє отримувати на постійній основі прибутковість, яка буде істотно вищою за ринкову.

При сильній формі ефективності ринку всім його учасники мають рівний доступ не тільки до минулого і публічно доступної інформації, а й до приватної (інсайдерської) інформації. В таких умовах жоден з учасників ринку не може в довгостроковій перспективі розраховувати на отримання прибутковості, яка буде істотно перевищувати ринкову.

На сьогоднішній день жоден з існуючих фондових ринків у світі не може бути названий повністю інформаційно ефективним. Більш того, беручи до уваги сучасні емпіричні дослідження, можна зробити висновок, що теорія ефективного ринку є скоріше утопією, тому що не здатна повною мірою раціонально пояснити реальні процеси, що протікають на фінансових ринках.

Зокрема, професором Єльського університету Робертом Шиллером [8] був виявлений феномен, який він в подальшому назвав надмірною мінливістю цін фондових активів. Суть феномена полягає в частій зміні котирувань, які не піддаються раціональному поясненню, а саме, відсутня можливість інтерпретувати дане явище відповідними змінами в фундаментальних факторах.

В кінці 1980-х років були зроблені перші кроки до створення моделі, яка на відміну від концепції ефективного ринку, дозволила б точніше пояснити реальну поведінку фондових ринків. У 1986 році Фішер Блек у своїй публікації вводить новий термін - «шумова торгівля» [11, 12].

«Шумова торгівля - це торгівля на шумі, сприймається так, як би шум був би інформацією. Люди, які торгують на шумі, будуть торгувати навіть тоді, коли об'єктивно вони повинні були б утримуватися від цього. Можливо, вони вважають, що шум, на основі якого вони торгують, є інформацією. Або, можливо, їм просто подобається торгувати». Хоча Ф. Блек не вказує, яких операторів слід відносити до категорії «шумових трейдерів», в роботі Де Лонга, Шляйфер, Саммерса і Вальдмана [12] можна знайти опис таких учасників ринку. Шумові трейдери помилково вважають, що у них є унікальна інформація про майбутні ціни на активи. Джерелами такої інформації можуть бути помилкові сигнали про неіснуючі тренди, що подаються індикаторами технічного аналізу, чутки, рекомендації фінансових експертів. Шумові трейдери сильно переоцінюють значення наявної інформації і готові приймати на себе необгрунтовано великий ризик. Проведені емпіричні дослідження також вказують на те, що до шумових трейдерів в першу чергу слід віднести індивідуальних інвесторів, тобто фізичних осіб. Більш того, саме ця група трейдерів несе систематичні збитки від торгівлі через ірраціональність своїх дій. Для західних фондових ринків емпіричне підтвердження цього явища можна знайти в дослідженнях Барбера і Одіна [14]. Теорія шумової торгівлі дозволяє пояснити і феномен Р. Шиллера. Саме ірраціональні дії трейдерів викликають надмірну мінливість цін.

1.6 Висновок

Резюмуючи вищесказане, можна зробити висновок, що незважаючи на велику кількість існуючих метрик, показчиків, індикаторів дієвого методу аналізу ринкових котирувань немає, проте існують прецеденти на прикладі багатьох людей які доводять, що це можливо. В той же час є наукові дослідження, що ставлять під сумнів можливість будь-якого аналізу ринку.

Метою дослідження є виявлення закономірностей числових рядах - показниках ринкових котирувань, визначення напрямку у якому буде рухатись ринок.

За завдання дослідження ставляться наступні положення:

- охарактеризувати наявні методи аналізу ринкових котирувань;
- проаналізувати наявні методи аналізу ринкових котирувань;
- визначити ознаки для створення нового методу аналізу ринкових котирувань;
- розробити метод аналізу ринкових котирувань;
- визначити ефективність створеного рішення.

2 МЕТОДИ ТА ЗАСОБИ АНАЛІЗУ ФІНАНСОВИХ ЧАСОВИХ РЯДІВ

Вхідні дані представлені у вигляді числового ряду, як таблиця значень від певної функції від часу $f(t) = x$, де t - час, x -показник. У якості $f(t)$ може виступати як ціна певного суб'єкту ринку на певний момент часу, так і певний фінансовий індикатор. Для використання у подальшому аналізі таких числових рядів з метою усунення шумів можуть використовувати методи згладження, наприклад ковзаюче середнє.

Часовий ряд – це зібрані в різні моменти часу статистичні дані про значення будь-яких параметрів (в найпростішому випадку одного) досліджуваного процесу. Виявлення структури часового ряду необхідно для того, щоб побудувати математичну модель того явища, яке є джерелом часового ряду, що аналізується. Прогноз майбутніх значень часового ряду використовується для ефективного прийняття рішень та може бути застосований у багатьох галузях, наприклад: кліматологія, океанологія, геофізика, техніка, обробка зображень, медицина, економетрика та багато інших.

2.1 Модель ціноутворення капітальних активів

Модель ціноутворення капітальних активів, Модель оцінки довготривалих активів, Capital Asset Pricing Model (CAPM) – теорія оцінки акцій котра припускає, що премія за ризик зростає пропорційно β -коефіцієнту акції або інвестиційного портфелю. Має наступний вигляд:

$$E(R_i) = R_f + \beta_i(E(R_m) - R_f), \quad (2.1)$$

де $E(R_i)$ – очікувана ставка дохідності на довготривалий актив;

R_f – безризикова ставка дохідності;

β_i – коефіцієнт чутливості активу до змін ринкової дохідності R_m , виражений як коваріація дохідності активу R_i з дохідністю всього ринку R_m по відношенню до дисперсії дохідності всього ринку $\sigma^2(R_m)$, рівний:

$$\beta_i = \frac{\text{cov}(R_i, R_m)}{\sigma^2(R_m)}, \quad (2.2)$$

де β - коефіцієнт для ринку в цілому завжди рівний одиниці;

$E(R_m)$ – очікувана ставка дохідності ринкового портфелю;

$E(R_m) - R_f$ - премія за ризик вкладення у акції, рівна різниці ставок ринкової та безстрокової дохідностей.

β -коефіцієнт акції є мірою ринкового ризику акції, показуючи мінливість прибутковості акції до прибутковості на ринку в середньому (застосовується для оцінки ризику вкладень в цінні папери).

За безризикову ставку доходу на американському і міжнародному ринках прийнято купувати цінних паперів, випущені американським урядом (T-bills). У деяких випадках також приймають ставки з цінних паперів Великобританії. З 2011 року йде обговорення того, чи будь-який дохід по відсотку є безризиковим насправді [20].

Як було зазначено вище, багато вихідних положень CAPM повністю або частково не виконуються на реальних ринках.

2.2 Авторегресійні моделі

2.2.1 Авторегресійна модель (AR)

Авторегресійна модель, autoregressive model (AR) – модель часових рядів, в якій значення часового ряду в даний момент лінійно залежать від попередніх значень цього ж ряду. Авторегресійний процес порядку p визначається наступним чином:

$$X_t = c + \sum_{i=1}^p a_i X_{t-i} + \varepsilon_t, \quad (2.3)$$

де a_1, \dots, a_p – параметри моделі (коефіцієнти авторегресії);

c – константа;

ε_t – білий шум.

2.2.2 Модель ковзаючого середнього

Модель ковзаючого середнього порядку q – модель часового ряду, виду:

$$X_t = c + \sum_{j=0}^q b_j \varepsilon_{t-j}, \quad (2.4)$$

де b_0, \dots, b_q – параметри моделі ;

c – константа;

ε_t – білий шум.

2.2.3 Модель авторегресії - ковзаючого середнього (ARMA)

Модель авторегресії ковзаючого середнього, Autoregressive integrated moving average (ARMA) - математична модель аналізу та прогнозування стаціонарних часових рядів, є узагальненням моделі авторегресії та моделі ковзаючого середнього.

Моделлю $ARMA(p, q)$, де p і q – цілі числа, що задають порядок моделі, називається наступний процес генерації тимчасового ряду X_t :

$$X_t = c + \varepsilon_t + \sum_{i=1}^p a_i X_{t-i} + \sum_{j=0}^q b_j \varepsilon_{t-j}, \quad (2.5)$$

де a_1, \dots, a_p – коефіцієнти авторегресії;

b_0, \dots, b_q – коефіцієнти ковзаючого середнього;

c – константа;

ε_t – білий шум.

Така модель може інтерпретуватися як лінійна модель множинної регресії, в якій в якості пояснюючих змінних виступають минулі значення

самої залежною змінною, а в якості регресійного залишку – ковзаючі середні з елементів білого шуму. ARMA-процеси мають більш складну структуру в порівнянні зі схожими за поведінкою AR- або MA-процесами в чистому вигляді, але при цьому ARMA-процеси характеризуються меншою кількістю параметрів, що є однією з їхніх переваг.

Для побудови моделі ARMA по серії спостережень необхідно визначити порядок моделі (числа p і q), а потім і самі коефіцієнти. Для визначення порядку моделі може застосовуватися дослідження таких характеристик часового ряду, як його автокореляційна функція і приватна автокореляційна функція. Для визначення коефіцієнтів застосовуються такі методи, як метод найменших квадратів і метод максимальної правдоподібності.

2.2.4 Модель Бокса-Дженкінса (ARIMA)

Модель Бокса-Дженкінса, Autoregressive integrated moving average (ARIMA) – інтегрована модель авторегресії - ковзаючого середнього, модель і методологія аналізу часових рядів. Є розширенням моделей ARMA для нестационарних часових рядів, які можна зробити стаціонарними взяттям різниць деякого порядку від вихідного часового ряду (так звані інтегровані або різницево-стаціонарні тимчасові ряди). Має наступний вигляд:

$$\Delta^d X_t = c + \varepsilon_t + \sum_{i=1}^p a_i \Delta^d X_{t-i} + \sum_{j=0}^q b_j \varepsilon_{t-j}, \quad (2.6)$$

де a_1, \dots, a_p – коефіцієнти авторегресії;

b_0, \dots, b_q – коефіцієнти ковзаючого середнього;

c – константа;

ε_t – білий шум.

Δ^d – оператор різниці часового ряду порядку d (послідовне взяття d раз різниць першого порядку – спочатку від тимчасового ряду, потім від отриманих різниць першого порядку, потім від другого порядку і т.д.)

Також дана модель інтерпретується як $ARMA(p + d, q)$ – модель з d одиничними корінням. При $d = 0$ маємо звичайні $ARMA$ моделі.

Підхід ARIMA до часових рядах полягає в тому, що в першу чергу оцінюється стаціонарність ряду. Різними тестами виявляються наявність поодиноких коренів і порядок інтегрованості тимчасового ряду (зазвичай обмежуються першим або другим порядком). Далі при необхідності (якщо порядок інтегрованості більше нуля) ряд перетворюється взяттям різниці відповідного порядку і вже для перетвореної моделі будується деяка $ARMA$ -модель, оскільки передбачається, що отриманий процес є стаціонарним, на відміну від вихідного нестаціонарного процесу (різницево-стаціонарного або інтегрованого процесу порядку d).

2.3 Аналіз сингулярного спектру (SSA)

Аналіз сингулярного спектру, Singular spectrum analysis (SSA) – метод аналізу часових рядів, заснований на перетворенні одновимірного часового ряду в багатовимірний ряд з подальшим застосуванням до отриманого багатомірного тимчасового ряду методу головних компонент.

Спосіб перетворення одновимірного ряду в багатовимірний є «згорткою» часового ряду в матрицю, що містить фрагменти тимчасового ряду, отримані з деяким зсувом. Загальний вигляд процедури зсуву нагадує «гусеницю», тому сам метод нерідко так і називають – «Гусениця»: довжина фрагмента називається довжиною «гусениці», а величина зсуву одного фрагмента щодо іншого кроком «гусениці».

SSA може бути використаний без попереднього завдання моделі ряду для аналізу довільних, в тому числі, нестаціонарних, рядів. Основна мета

SSA – розкласти ряд в суму інтерпретованих компонент, таких як тренд, періодичні компоненти, шум. При цьому знання параметричної форми цих компонент не вимагається.

Алгоритм SSA можна представити у вигляді 4 кроків.

Крок 1. Розглянемо матеріально-значний ряд $\mathbb{X} = (x_1, \dots, x_n)$ довжини N . Нехай L ($1 < L < N$)- деяке ціле число, зване *довжина вікна*, і $K = N - L + 1$.

Будується $L \times K$ траекторна матриця ряду \mathbb{X} наступним чином:

$$\mathbf{X} = [X_1 : \dots : X_K] = (x_{ij})_{i,j=1}^{L,K} = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_K \\ x_2 & x_3 & x_4 & \dots & x_{K+1} \\ x_3 & x_4 & x_5 & \dots & x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & x_{L+2} & \dots & x_N \end{bmatrix} \quad (2.7)$$

де $X_i = (x_i, \dots, x_{i+L-1})^T$ ($1 \leq i \leq K$) – вектори вкладення довжини L .

Матриця \mathbf{X} є ганкелевою.

Крок 2. Виконується сингулярне розкладання траекторної матриці. Покладемо $\mathbf{S} = \mathbf{X}\mathbf{X}^T$ і позначимо $\lambda_1, \dots, \lambda_L$ власні числа \mathbf{S} , взяті в незростаюча порядку ($\lambda_1 \geq \dots \geq \lambda_L \geq 0$), і U_1, \dots, U_L ортонормована система власних векторів матриці \mathbf{S} , відповідних власним числам.

Покладемо $d = \text{rank } \mathbf{X} = \max\{i : \lambda_i > 0\}$ (зауважимо, що в реальності, як правило, $d = L$) і $V_i = \mathbf{X}^T U_i / \sqrt{\lambda_i}$ ($i = 1, \dots, d$). У цих позначеннях сингулярне розкладання траекторної матриці \mathbf{X} може бути записано як

$$\mathbf{X} = \mathbf{X}_1 + \dots + \mathbf{X}_d, \quad (2.8)$$

де $\mathbf{X}_i = \sqrt{\lambda_i} U_i V_i^T$ мають ранг 1 і називаються елементарними матрицями. Набір $(\sqrt{\lambda_i}, U_i, V_i^T)$ називається i -ю власною трійкою (коротко - ET від eigentriple) сингулярного розкладу. Вектори U_i та V_i називаються

лівими і правими, відповідно, сингулярними векторами матриці \mathbf{X} , числа $\sqrt{\lambda_i}$ - сингулярні числа (вони складають сингулярний спектр \mathbf{X} , що і дало назву методу Singular Spectrum Analysis), вектори $\sqrt{\lambda_i}V_i = \mathbf{X}^T U_i$, за аналогією з аналізом головних векторів, називаються векторами головних компонент.

Крок 3. Виконаємо групування власних трійок. Множина всіх індексів $\{1, \dots, d\}$ розбивається на m підмножин I_1, \dots, I_m , що не перетинаються.

Нехай $I = \{i_1, \dots, i_p\}$. Тоді результуюча матриця \mathbf{X}_I , що відповідає групі I , визначається як $\mathbf{X}_I = \mathbf{X}_{i_1} + \dots + \mathbf{X}_{i_p}$. Результуючі матриці обчислюються за групами $I = I_1, \dots, I_m$ і згруповане SVD розкладання матриці \mathbf{X} може бути записано як $\mathbf{X} = \mathbf{X}_{I_1} + \dots + \mathbf{X}_{I_m}$.

Крок 4. Виконаємо діагональне усереднення. Кожна матриця \mathbf{X}_{I_j} згрупированного розкладання ганкелізується (усереднюється по анти-діагоналям) і потім отримана ганкелева матриця трансформується в новий часовий ряд довжини N на основі взаємно-однозначної відповідності між ганкелевими матрицями і тимчасовими рядами.

Діагональне усереднення, застосоване до кожної результуючої матриці \mathbf{X}_{I_k} , виробляє відновлені ряди $\tilde{\mathbf{X}}^{(k)} = (\tilde{x}_1^{(k)}, \dots, \tilde{x}_N^{(k)})$. Таким чином, початковий ряд x_1, \dots, x_N розкладається в суму m відновлених рядів:

$$x_n = \sum_{k=1}^m \tilde{x}_n^{(k)} \quad (n = 1, 2, \dots, N). \quad (2.9)$$

Даний розклад є головним результатом алгоритму SSA для аналізу часового ряду. Це розкладання має сенс, якщо кожна з його компонент може бути інтерпретована як тренд, або коливання (періодики), або шум.

Непараметричний метод сингулярного спектрального аналізу (Гусениця) дозволяє отримати результати трохи менш точні, ніж більшість параметричних методів при аналізі часових рядів із відомою моделлю. Якщо ж модель невідома або ряд не стаціонарен, як у випадку з фондовим ринком,

застосування методу сингулярного спектрального аналізу дає результати набагато більш точні, ніж відомі статистичні параметричні методи [19].

2.4 Пошук прихованих шаблонів (hidden T-patterns)

Приховані, або не очевидні закономірності в часових рядах, представляють не абиякий інтерес для фінансових аналітиків, оскільки знаходження таких закономірностей допоможе у розробці більш якісних фінансових прогнозів. Такі закономірності можуть бути представлені у вигляді шаблону – певної послідовності значень[20].

Нехай час спостереження розбитий на N_t інтервалів. У кожен момент періоду спостереження $[1, N_t]$ може статися деяка подія з множини допустимих подій ε . Відповідно, кожному типу подій зіставляється множина моментів часу $TS(A)$

$$TS(A) = \{T_{A,1}, \dots, T_{A,N_A}\}, A \in \varepsilon, 0 \leq T_{A,i} \leq N_t \quad (i = 1, \dots, N_A) \quad (2.10)$$

Під час пошуку закономірностей в даних, нас цікавлять відносини між розподілом окремих подій. Якщо припустити, що у вихідних даних не існує ніяких закономірностей, то кожна подія має з'являтися незалежно, від інших; тобто розподіл компонентів повинен бути незалежним. Але шаблон характеризується появою своїх компонентів в однаковому порядку, більш того, тимчасові інтервали, що розділяють компоненти, повинні бути приблизно однаковими. Назвемо такий шаблон - часовий шаблон (ЧШ).

Будемо говорити, що події **A** і **B** пов'язані ставленням критичного інтервалу, якщо, після появи події **A** в момент часу t , існує інтервал $[t + d_1, t + d_2]$, ($d_2 \geq d_1 \geq 0$), що містить **B**, частіше, ніж це очікується з припущення про незалежність подій. Даний взаємо-зв'язок будемо позначати, як **A** $[d_1, d_2]$ **B**, або, коротше, (**AB**).

Далі розкриємо поняття “частіше ніж це очікується”. Нехай N_A і N_B - кількість виникнень **A** і **B**, відповідно, протягом $[1, N_t]$.

$P(A) = N_A/N_t$ – ймовірність появи події **A** в певний момент часу;

$P(\neg A) = 1 - P(A)$. $P(\neg A)^d$ – ймовірність, що **A** чи не з'явиться протягом будь-якого інтервалу $[d_1, d_2]$, довжини d ($d = d_1 - d_2 + 1$). Імовірність спостерігати **A** на інтервалі довжини d один або більше разів, дорівнює $1 - P(\neg A)^d$. Зафіксуємо події **A**, **B**, і довжину інтервалу d . Приймаючи гіпотезу про незалежний розподілі подій, подія **B** міститься один і більше раз в інтервалі довжини d , після події **A**, $N_A * (1 - P(\neg B))^d$ раз.

$\rho = P(\geq N_{AB}) = 1 - P(< N_{AB})$ – апіорна ймовірність того, що N_{AB} з N_A інтервалів містять входження **B**. Очевидно, що $P(< N_{AB})$ розподілено по біноміальному закону, де N_A - кількість “випробувань”, $1 - P(\neg B)^d$ - ймовірність “успіху”. Тобто

$$\rho = P(\geq N_{AB}) = 1 - \sum_{i=0}^{N_{AB}-1} C_{N_A}^i (1 - P(\neg B)^d)^i P(\neg B)^{N_A-i} \quad (2.12)$$

Отримана ймовірність ρ порівнюється з граничним значенням α , що є структурним параметром пошуку: якщо $\rho \leq \alpha$, то заданий інтервал позначається критичним. Зауважимо, що ρ залежить від N_A, N_B, N_t, N_{AB}, d :

$$\rho = p(N_A, N_b, N_t, N_{AB}, d) \quad (2.13)$$

Дамо рекурсивне визначення ЧШ. Домовимося називати кожну допустиму подію псевдошаблоном. Тоді ЧШ **Q** можна визначити як:

$$\mathbf{Q} = \mathbf{X}_1[dL_1, dR_1] \mathbf{X}_2[dL_2, dR_2] \dots \mathbf{X}_i[dL_i, dR_i] \mathbf{X}_{i+1} \dots \mathbf{X}_m, \quad (2.14)$$

де \mathbf{X}_i , ($i = 1 \dots m$) – ЧШ або псевдошаблон. Для шаблонів відношення критичного інтервалу $\mathbf{Q}_L[dL, dR] \mathbf{Q}_R$ вводиться, як і для подій, з урахуванням того, що інтервал $[dL, dR]$ відраховується від останнього елемента \mathbf{Q}_L , і

входження Q_R визначається його першим елементом. Будемо називати подвійними серіями шаблон Q , множину $\{\{Left_i, Right_i\}_{i=1...N_Q}\}$, де:

N_Q – кількість появ шаблону Q ;

$Left_i, Right_i$ – індекси початкової і кінцевої подій i -ї появи шаблону Q .

Алгоритм пошуку ЧШ, описаний вище, полягає в ітеративному повторі двох стадій: конструювання нових шаблонів і видалення неповних шаблонів. На виході алгоритм видає множину ЧШ з їх супутніми характеристиками: критичним інтервалом, рівнем значущості, частотою появи.

Нехай D_i – множина шаблонів, виявлених до i -ї ітерації. Фактично, множина $D_m \setminus E$, де m – номер останньої ітерації, і буде результатом роботи алгоритму.

Крок 1. Пошук та конструювання. На даному етапі, для будь-якої впорядкованої пари

$$(Q', Q''): Q', Q'' \in D_i, \quad (2.15)$$

перевіряється існування критичного зв'язку. Якщо критичний зв'язок $[dL, dR]$ був знайдений, і шаблон (Q', Q'') зустрічається частіше, ніж N_{\min} раз, то в множину D_{i+1} додається новий шаблон $Q'[dL, dR]Q''$. Однак такі дії призводять до того, що один і той же шаблон **ABCD** може бути сконструйований різними способами (наприклад як $(A(BCD))$ і $((AB)(CD))$), що веде до заповнення множини D зайвими шаблонами. Для уникнення даної проблеми пропонується наступне рішення: перед додаванням, шаблон порівнюється з вже існуючими шаблонами і перевіряється, чи не є він дублікатом.

Крок 2. Видалення неповних шаблонів. На даній стадії алгоритм прагне видалити знайдені шаблони, які є меншими частинами, або неповними версіями інших виявлених шаблонів. Для їх визначення можна застосовувати різні евристичні методи. Нижче опишемо умову, взятую зі статті [20].

Отже, шаблон Q_x вважається менш повним, ніж Q_y , якщо Q_x і Q_y з'являються однаково часто, і всі події виникають в Q_x , також виникають в Q_y .

2.4.3 Формалізований опис

Введемо позначення:

E – допустимі події,

N_t – тривалість спостереження,

α – мінімальний рівень значущості,

N_{\min} – мінімальна кількість появ шаблону.

Так як для завдання пошуку ЧШ не вводиться визначення функціоналу якості знайдених шаблонів, то алгоритм вимагає ручного налаштування параметрів α і N_{\min} . Вибір значень параметрів повинен ґрунтуватися на специфіці спостережуваних процесів і очікуваних результатів.

```

1:  $D_1 = 0$  //Ініціалізація
2:  $D_0 = E$ 
3:  $t = 0$ 
4: поки  $D_t = D_{t-1}$  :
5:      $t = t + 1$ 
6:      $D_t = D_{t-1}$ 
7:     для  $P_L \in D_{t-1}$  : //Крок 1
8:         для  $P_R \in D_{t-1}$  :
9:             для всіх  $d_L, d_R \in [1, N_t]$  таких, що  $d_L \leq d_R$  do
10:                якщо  $\sigma(N_{PL}, N_{PR}, N_t, N_{PLPR}, d_R - d_L + 1) < \alpha$ :
11:                    якщо  $\text{isUnique}(P_L \cup P_R, D_t) < \alpha$ :
12:                         $D_t = D_t \cup \{P_L \cup P_R\}$ 
13:     для  $P_L \in D_{t-1} \setminus E$  : //Крок 2
14:         для  $P_R \in D_{t-1}$  :
15:             якщо  $P_L \cup P_R$  та  $|\text{DS}(P_L)| = |\text{DS}(P_R)|$  та  $\text{isIntersect}(P_L, P_R)$ :
16:                 видалити  $P_L$  з  $D_t$ 
17: function  $\text{isUnique}(Q, D)$  //Чи є шаблон унікальним
18:     для  $P \in D$ :
19:         якщо  $\text{DS}(Q) = \text{DS}(P)$  та  $\text{Events}(Q) = \text{Events}(P)$ :
20:             повернути хибність
21:     повернути істина
22: function  $\text{isIntesect}(P_L, P_R)$ : //Чи перетинаються два шаблони
23:     для  $i = 1 \dots |\text{DS}(P_L)|$ :

```

- 24: якщо $DS_{i,Left}(P_L) > DS_{i,Right}(P_R)$ або $DS_{i,Right}(P_L) < DS_{i,Left}(P_R)$:
 25: повернути хибність
 26: повернути істина

2.5 Нейромережеві методи

Нейромережеві методи це методи або математичні, що ґрунтуються на штучних нейронних мережах. Нейромережеві моделі можна розглядати як прості математичні моделі, що визначають функцію $f : X \rightarrow Y$, або розподіл над X , або над X та Y . Іноді моделі тісно пов'язують з певним правилом навчання.

Важливим є завдання побудова мережі. Це питання вирішується в два етапи:

- вибір конфігурації мережі;
- навчання (підбір ваг) мережі.

На першому етапі слід, власне, підібрати конфігурацію нейромережі, вирішити наступні питання:

- які нейрони ми хочемо використовувати (число входів, передавальні функції);
- яким чином слід з'єднати їх між собою;
- що взяти в якості входів і виходів мережі.

Це завдання на перший погляд здається складним, але необов'язково придумувати нейромережу – існує кілька десятків різних нейромережевих архітектур, причому ефективність багатьох з них доведена математичною статистикою. Найбільш популярні і вивчені архітектури - це багатошаровий перцептрон, нейромережа із загальною регресією, мережі Кохонена, мережі Хопфілда, мережі Хеммінга та інші.

На другому етапі слід навчити обрану мережу, тобто підібрати такі значення її ваг, щоб мережа працювала належним чином. У використовуваних на практиці нейромережах кількість ваг може становити кілька десятків

тисяч, тому навчання це дійсно складний процес. Для багатьох архітектур розроблені спеціальні алгоритми навчання, які дозволяють налаштувати ваги мережі певним чином. Навчання може відбуватися у вигляді контрольованого навчання (коли результат роботи мережі постійно порівнюється з очікуваним результатом), та неконтрольоване (не використовують зовнішніх впливів для коректування своїх ваг і внутрішньо контролюють свою ефективність, шукаючи регулярність або тенденції у вхідних сигналах та здійснюють адаптацію відповідно до навчальної функції). Для прогнозування використовують моделі з контрольованим навчанням.

2.5.1 Штучна нейронна мережа

Штучна нейронна мережа – це мережа простих елементів, званих нейронами, які отримують вхід, змінюють свій внутрішній стан (збудження) відповідно до цього входу, і виробляють вихід, залежний від входу та збудження. Мережа утворюється з'єднанням виходів певних нейронів зі входами інших нейронів з утворенням орієнтованого зваженого графу. Ваги, як і функції, що обчислюють збудження, можуть змінюватися процесом, званим навчанням, який керується правилом навчання.

В поширених реалізаціях ШНМ сигнал на з'єднанні між штучними нейронами є дійсним числом, а вихід кожного штучного нейрону обчислюється нелінійною функцією суми його входів. Штучні нейрони та з'єднання зазвичай мають вагу, яка підлаштовується в перебігу навчання. Вага збільшує або зменшує силу сигналу на з'єднанні. Штучні нейрони можуть мати такий поріг, що сигнал надсилається лише якщо сукупний сигнал перетинає цей поріг. Штучні нейрони зазвичай організовано в шари. Різні шари можуть виконувати різні види перетворень своїх входів. Сигнали проходять від першого (вхідного) до останнього (вихідного) шару, можливо,

після проходження шарами декілька разів (наприклад у рекурсивних нейронних мережах).

Для опису нейромереж використовують кілька усталених термінів, які в різних джерелах можуть мати різне трактування, зокрема:

- структура нейромережі – спосіб зв'язків нейронів у нейромережі;
- архітектура нейромережі – структура нейромережі та типи нейронів;
- парадигма нейромережі – спосіб навчання та використання, іноді містить поняття архітектури.

Сукупність даних параметрів (структура, архітектура, парадигма) будемо називати конфігурацією нейромережі, тобто конфігурація нейромережі – це параметри що визначають:

- кількість входів;
- кількість виходів;
- кількість проміжних шарів;
- інформацію що подається на вхід;
- інформацію яка отримується на виходів;
- функції поширення;
- зв'язки між шарами;
- правило навчання;
- вибір інформації для входу та виходу.

Визначення конфігурації нейронної мережі яка буде ефективно розв'язувати поставлену задачу це довготривалий та складний процес.

На базі однієї архітектури може бути реалізовано різні парадигми нейромережі і навпаки. Оскільки всі штучні нейронні мережі базуються на концепції нейронів, з'єднань та передатних функцій, існує подібність між різними структурами або архітектурами нейронних мереж.

За способом організації зв'язків виділяють два основні види нейронних мереж:

- нейронна мережа прямого поширення;
- рекурентні нейронні мережі.

Нейронна мережа прямого поширення – це вид нейронної мережі, в якій сигнали поширюються в одному напрямку, починаючи від вхідного шару нейронів, через приховані шари до вихідного шару і на вихідних нейронах отримується результат опрацювання сигналу. В мережах такого виду немає зворотніх зв'язків.

Рекурентні нейронні мережі – це мережі, у яких з'єднання між вузлами утворюють орієнтований цикл. Це створює внутрішній стан мережі, що дозволяє їй проявляти динамічну поведінку в часі. На відміну від нейронних мереж прямого поширення, РНМ можуть використовувати свою внутрішню пам'ять для обробки довільних послідовностей входів. Це робить їх застосовними до таких задач, як розпізнавання несегментованого неперервного рукописного тексту та розпізнавання мовлення.

2.5.2 Штучний нейрон

Штучний нейрон являє собою одиницю обробки інформації у нейронній мережі. Поведінка нейрона будується наступним чином: нехай є $m + 1$ входів, значення яких дорівнюють $x_0, x_1, x_2, \dots, x_m$, а значення їх ваг рівні, w_0, w_1, \dots, w_m , при цьому перший вхідний елемент, як правило, являє собою фіксоване значення зміщення $x_0 = 1$. Графічно нейрон та його зв'язки зображені на рис. 1.

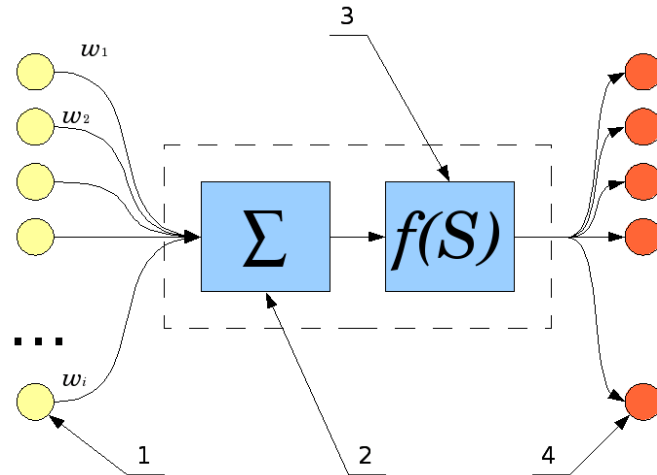


Рисунок 2.1 – Схема штучного нейрону[27]

Тоді вихідне значення нейрона є значення функції активації від зваженої суми його вхідних значень:

$$y = f(S), \quad S = \sum_{i=0}^m \omega_i x_i, \quad (2.16)$$

де x_i – входи (1), ω_i – ваги, S – суматор (2), f – передатна функція (функція активації) (3), y – вихід/виходи штучного нейрону (4).

В програмних реалізаціях штучні нейрони називають «елементами обробки» або «процесорами» і вкладають в них більше можливостей, ніж в базовому штучному нейроні, що описаний вище.

Функція суматора може бути складною, наприклад, вибір мінімуму, максимуму, середнього арифметичного, добутку або обчислюватися за іншим алгоритмом. Багато програмних реалізацій використовують власні функції суматора.

Результат функції суматора перетворюється у вихідний сигнал через передатну функцію. В передатній функції для визначення виходу нейрона загальна сума порівнюється з деяким порогом (зазвичай, це діапазон $[0, 1]$ або $[-1, 1]$ або інше) за допомогою певного алгоритму.

Переважно застосовують нелінійну передатну функцію, оскільки лінійні (прямолінійні) функції є обмеженими і вихід є пропорційним до входу. Застосування лінійних передатних функцій було проблемою у ранніх моделях мереж, і їх обмеженість та недоцільність була доведена [24].

В існуючих нейромережах як передатну функцію використовують сигмоїду, синус, гіперболічний тангенс та і.н..

2.5.3 Перцептрон

Першою моделлю нейромереж вважають перцептрон Розенбалата. Теорія перцептронів є основою для багатьох типів штучних нейромереж прямого поширення і вони є класикою для вивчення.

Одношаровий перцептрон здатний розпізнавати найпростіші образи. Окремий нейрон обчислює зважену суму сигналів вхідних елементів, віднімає значення зсуву і пропускає результат через жорстку порогову функцію, вихід якої дорівнює +1 чи -1. В залежності від значення вихідного сигналу приймається рішення:

Алгоритм навчання одношарового перцептрона можна навести у вигляді 5 кроків.

Крок 1. Ініціалізація синаптичних ваг і зсуву: синаптичні ваги приймають малі випадкові значення.

Крок 2. Пред'явлення мережі нового вхідного і бажаного вихідного сигналів: вхідний сигнал $x=(x_1, x_2, \dots, x_n)$ пред'являється нейрону разом з бажаним вихідним сигналом d .

Крок 3. Обчислення вихідного сигналу нейрона:

$$y(t) = f\left(\sum_{i=1}^N w_i(t)x_i(t) - b\right), \quad (2.17)$$

де $y(t)$ – вихідний сигнал, $w_i(t)$ - ваги на поточній ітерації, b - зсув (поріг).

Крок 4. Налаштовування значень ваг:

$$w_i(t+1) = w_i(t) + r[d(t) - y(t)]x_i(t), \quad i=1, \dots, N, \quad (2.18)$$

де $w_i(t)$ - вага зв'язку від i -го елемента вхідного сигналу до нейрона в момент часу t , r - швидкість навчання (менше 1); $d(t)$ - бажаний вихідний сигнал.

Якщо мережа приймає правильне рішення, синаптичні ваги не модифікуються.

Крок 5. Перехід до кроку 2.

Багатошарові перцептрони дають можливість будувати складні поділяючі поверхні і є більш поширеними. До недоліків даного підходу можна віднести примітивність розділяючих поверхонь (гіперплощин), які як наслідок, обмежені у застосуванні лише для розв'язання відносно простих задач. В той же час до переваг можна віднести простоту реалізації та швидкість навчання.

2.5.4 Нейромережа зворотного поширення похибки

Архітектура мережі зі зворотнім поширенням похибки була розроблена на початку 1970-х років декількома незалежними авторами: Вербор (Werbor); Паркер (Parker); Румельгарт (Rumelhart), Хінтон (Hinton) та Вільямс (Williams) [28][29]. На даний час, така парадигма є популярною, ефективною та легкою модель навчання для складних, багатошарових мереж. Вона використовується в різних типах застосувань і породила великий клас нейромереж з різними структурами та методами навчання.

Така мережа має вхідний шар, вихідний шар та принаймні один прихований шар. Теоретично, обмежень відносно числа прихованих прошарків не існує.

Нейрони організовано у структуру у вигляді шарову з прямою передачею (вперед) сигналу. Кожний нейрон мережі продукує зважену суму своїх входів, пропускає цю величину через передатну функцію і видає вихідне значення. Мережа може моделювати функцію практично будь якої складності, причому число шарів і число нейронів у кожному шарі визначають складність функції.

Кількість входів та виходів мережі визначаються кількістю вхідних та вихідних параметрів досліджуваного об'єкту, явища, процесу, тощо. На відміну від зовнішніх шарів, число нейронів прихованих шарів обирається емпіричним шляхом. Після того, як визначено число прошарків і число нейронів в кожному з них, потрібно знайти значення для синаптичних ваг і порогів мережі, які спроможні мінімізувати похибку спродукованого результату. Саме для цього існують алгоритми навчання, де відбувається підгонка моделі мережі до наявних навчальних даних. Похибка для конкретної моделі мережі визначається шляхом проходження через мережу всіх навчальних прикладів і порівняння спродукованих вихідних значень з бажаними значеннями. Множина похибок створює функцію похибок, значення якої можна розглядати, як похибку мережі. В якості функції похибок найчастіше використовують суму квадратів похибок.

Алгоритм навчання можна навести у вигляді 4 кроків.

Крок 1. Ініціалізація мережі: вагові коефіцієнти і зсуви мережі приймають малі випадкові значення.

Крок 2. Визначення елемента навчальної множини: (вхід - вихід).
Входи (x_1, x_2, \dots, x_N) , повинні розрізнятися для всіх прикладів навчальної множини.

Крок 3. Обчислення вихідного сигналу:

$$w S_{i_m} = \sum_{i_{m-1}}^{N_{m-1}} w_{i_m j_{m-1}} - b_{i_m}, \quad y_{i_m} = f(S_{i_m}), \quad i_m = 1, \dots, N_m, \quad m = 1, \dots, L, \quad (2.19)$$

де S – вихід суматора, w – вага зв'язку, y – вихід нейрона, b – зсув, i – номер нейрона, N – число нейронів у прошарку, m – номер прошарку, L – число шарів, f – передатна функція.

Крок 4. Налаштування синаптичних ваг:

$$w_{ij}(t + 1) = w_{ij}(t) + r g_j x'_i, \quad (2.20)$$

де w_{ij} – вага від нейрона i або від елемента вхідного сигналу i до нейрона j у момент часу t , x'_i – вихід нейрона i , r – швидкість навчання (крок), g_j – значення похибки для нейрона j .

Якщо нейрон з номером j належить останньому шару, тоді:

$$g_j = y_j(1 - y_j)(d_j - y_j), \quad (2.21)$$

де d_j – бажаний вихід нейрона j , y_j – поточний вихід нейрона j .

Якщо нейрон з номером j належить одному з прошарків з першого по передостанній, тоді:

$$g_i = x'_j(1 - x'_j) \sum_k g_k w_{jk}, \quad (2.22)$$

де k пробігає всі нейрони прошарку з номером на одиницю більше, ніж у того, котрому належить нейрон j .

Зовнішні зсуви нейронів b налаштовуються аналогічним чином.

У мережах такого типу зазвичай використовують сигмоїдальну функцію активації, модифікації алгоритму зворотного поширення зазвичай полягають

у використанні різних підходів для визначення напрямку та величини кроку (швидкості навчання). До недоліків даного підходу можна віднести низьку швидкість навчання. Серед переваг – ефективний та популярний алгоритм для вирішення численних практичних задач.

2.5.5 Мережа Кохонена

Мережа розроблена Тойво Кохоненом [25] на початку 1980-х рр. і принципово відрізняється від розглянутих вище мереж, оскільки використовує неконтрольоване навчання і навчальна множина складається лише із значень вхідних змінних.

Мережа розпізнає кластери в навчальних даних і розподіляє дані до відповідних кластерів. Якщо в наступному мережа зустрічається з набором даних, несхожим ні з одним із відомих зразків, вона відносить його до нового кластеру. Якщо в даних містяться мітки класів, то мережа спроможна вирішувати задачі класифікації. Мережі Кохонена можна використовувати і в задачах, де класи відомими – перевага буде у спроможності мережі виявляти подібність між різноманітними класами.

Такі мережі чудово підходять для задач кластерезації, але не прогнозування, тому детально розглядатися вони не будуть.

2.5.6 Нейронна мережа Гопфілда

Нейронна мережа Гопфілда [26] – це тип рекурентної, повнозв'язної, штучної нейронної мережі з симетричною матрицею зв'язків. У процесі роботи динаміка таких мереж сходиться (конвергує) до одного з положень рівноваги. Ці положення рівноваги є локальними мінімумами функціоналу, що називається енергія мережі (у найпростішому випадку – локальними мінімумами негативно певної квадратичної форми на n -вимірному кубі). Така мережа може бути використана як авто-асоціативна пам'ять, як фільтр, а

також для розв'язання деяких завдань оптимізації. На відміну від багатьох нейронних мереж, що працюють до отримання відповіді через певну кількість тактів, мережі Хопфілда працюють до досягнення рівноваги, коли наступний стан мережі дорівнює попередньому.

Мережа Хопфілда використовує три прошарки: вхідний, прошарок Хопфілда та вихідний прошарок. Кожен прошарок має однакову кількість нейронів. Виходи нейронів вхідного прошарку надходять до входів відповідних нейронів прошарку Хопфілда. Тут, зв'язки мають фіксовані вагові коефіцієнти. Виходи прошарку Хопфілда під'єднуються до входів всіх нейронів прошарку Хопфілда, за винятком самого себе, а також до відповідних елементів у вихідному прошарку. Під час навчання, мережа скеровує дані з вхідного прошарку до прошарку Хопфілда.

Навчання мережі Хопфілда вимагає, щоб навчальний образ був представлений на вхідному та вихідному прошарках одночасно. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань. Для правильного навчання мережі відповідні пари "вхід-вихід" мають відрізнитися між собою.

Алгоритм навчання мережі Хопфілда можна представити у вигляді 4 кроків.

Крок 1. На стадії ініціалізації мережі вагові коефіцієнти синапсів встановлюються таким чином

$$w_{ij} = \left\{ \begin{array}{l} \sum_{k=1}^m x_i^k x_j^k, i \neq j \\ 0, i = j \end{array} \right\}, \quad (2.23)$$

де i і j – індекси, відповідно, предсинаптичного і постсинаптичного нейронів; x_i^k , x_j^k - i -ий та j -ий елементи вектора k -го зразка.

Крок 2. На входи мережі подається невідомий сигнал (t - номер ітерації). Його поширення безпосередньо встановлює значення виходів:

$$y_i(0) = x_i, i = 0 \dots n - 1, \quad (2.24)$$

тому позначення на схемі мережі вхідних сигналів у явному виді носить чисто умовний характер. Нуль у скобці справа від y_i означає нульову ітерацію в циклі роботи мережі.

Крок 3. Обчислюється новий стан нейронів

$$S(t + 1) = \sum_{i=1}^n w_{ij} y_j(t), j = 0 \dots n - 1, \quad (2.25)$$

і нові значення виходів

$$y_j(t+1) = f | S_j(t+1) |, \quad (2.26)$$

де f - передатна функція.

Крок 4. Перевіряємо чи змінилися вихідні значення виходів за останню ітерацію. Якщо так – перехід до пункту 2, інакше (якщо виходи стабілізувались) – кінець. При цьому вихідний вектор являє собою зразок, що найкраще відповідає вхідним даним.

Іноді мережа не може провести розпізнавання і видає на виході неіснуючий образ. Це пов'язано з проблемою обмеженості можливостей мережі. Для мережі Хопфілда число збережених образів m не повинно перевищувати $0.15 * n$ (n – кількість нейронів вихідного прошарку) Крім того, якщо два образи А и Б сильно схожі, вони, можливо, будуть викликати в мережі перехресні асоціації, тобто пред'явлення на входи мережі вектора А призведе до появи на її виходах вектори Б и навпаки.

2.6 Не математичні методи

Одним з найбільш поширених методів аналізу ринкових котирувань, що застосовують фінансові аналітики є технічний аналіз - побудова технічних

індикаторів та пошук фігур технічного аналізу. Детальний огляд нематематичних методів було наведено вище. Технічний аналіз ґрунтується на припущеннях, що:

- рух цін на ринку враховує всю інформацію;
- рух цін підпорядкований тенденціям;
- події, що відбувалися у минулому, будуть потроюватися у майбутньому.

Власне технічний індикатор – математична функція побудована на основі цін або об'ємів попиту. Індикатори можна розділити на дві групи:

- Індикатори тенденцій (підтверджують тенденції);
- Осцилятори (підказують розвороти трендів).

Хоча побудова – це застосування математичної функції до числового ряду і отримання нового числового ряду, його інтерпретація це завжди людський фактор.

Фігура технічного аналізу – певний шаблон поведінки часового ряду фінансових котирувань, який повторюється з часом, причому може повторюватися з іншою швидкістю. Хоч даний підхід не є математичним, проте прецеденти ефективності такого підходу існують [14].

2.7 Аналіз

Для порівняння ефективності прогнозування та обрання подальшого напрямку дослідження було обрано такі методи:

- Авторегресія – найпростіший метод статистичного аналізу;
- ARIMA – спосіб застосування авторегресії для нестационарних рядів;
- Аналіз сингулярного спектру – дозволяє розкласти нестационарний ряд на складові: шум, тренд, періодичні компоненти.

- Неймережа на прикладі багат шарового перцептрону – найпоширеніший представник надзвичайно великої групи методів прогнозування, що ґрунтуються на використанні штучний нейронних мереж. Архітектуру мережі зображено на рис. 2.2.

Застосування методу пошуку прихованих шаблонів не є доцільним, оскільки зможе виконувати прогнозування тільки в межах знайдених шаблонів, а не всього ряду. Застосування нематематичних методів не розглядалося через неможливість формалізації і, як наслідок, алгоритмічного кодування.

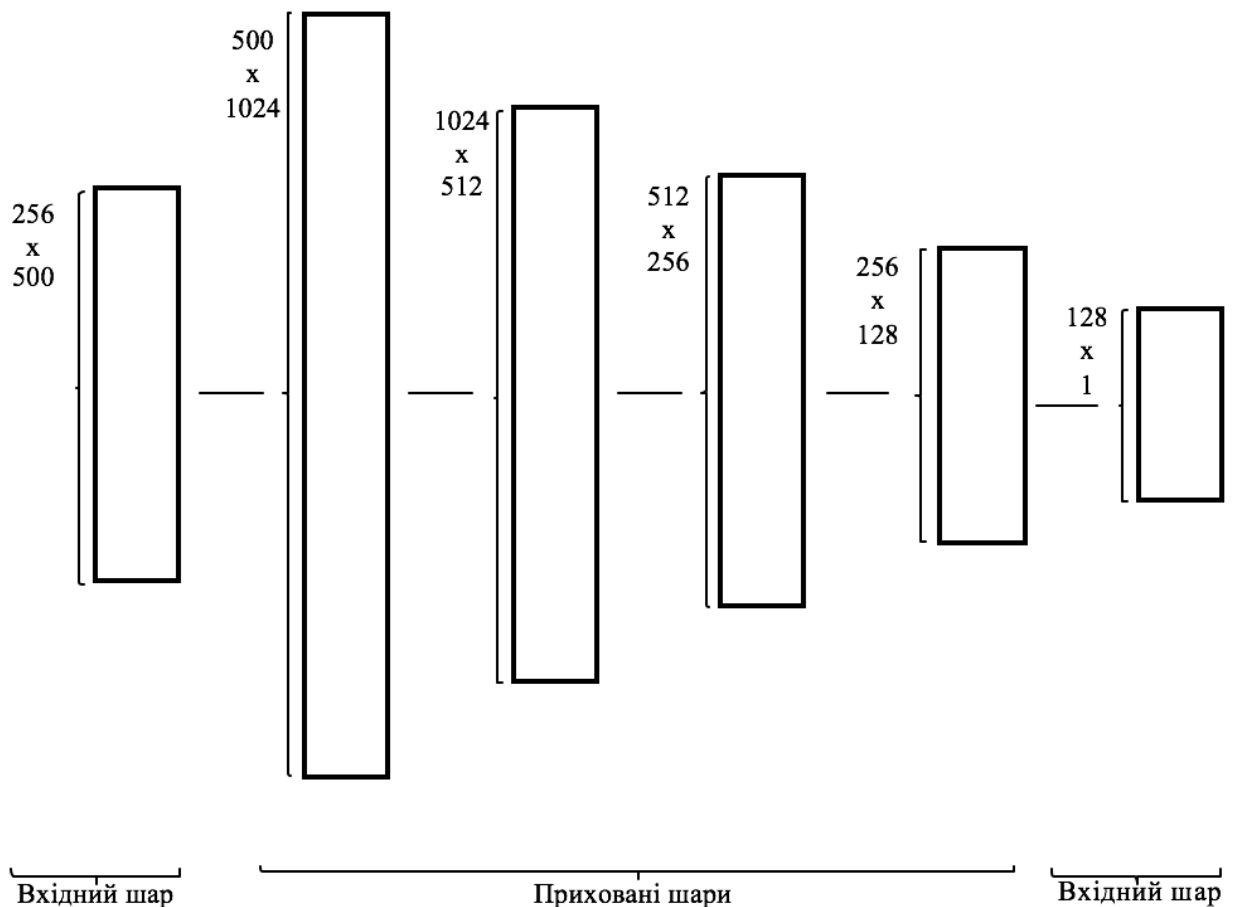


Рисунок 2.2 – Архітектура нейронної мережі

Ефективність перелічених методів розглядатиметься за критерієм найменшої квадратичної помилки на прикладі таких часових рядів:

- синусоїда – один з найпростіших прикладів числового ряду який легко подати у вигляді часового;
- денна мінімальна температура [30] – приклад нестационарного ряду, що має сезонні складові котрі легко визначити графічно;
- кількість пасажирів авіалінії [31] – приклад нестационарного ряду з явно вираженими сезонними складовими та трендом;
- ринковий індекс S&P 500 [32] – один з об'єктів дослідження даної роботи.

Застосувавши моделі до обраних часових рядів, отримали значення середньоквадратичних помилок, що наведені у таблиці 2.1.

Таблиця. 2.1 – Середньоквадратичні похибки прогнозів моделей

Часовий ряд	Кількість спостережень	Модель	Середньоквадратич на похибка
Синусоїда	720	Авторегресія	0.342
		ARIMA	0.021
		Аналіз сингулярного спектру	0.009
		Нейромережа	0.002
кількість пасажирів авіалінії	145	Ауторегресія	4342.589
		ARIMA	29.087
		Аналіз сингулярного спектру	7.513
		Нейромережа	0.029
денна мінімальна температура	3648	Ауторегресія	23.075
		ARIMA	8.212
		Аналіз сингулярного спектру	1.139

		Нейромережа	0.005
ринковий індекс S&P 500	395x500	Ауторегресія	41.006
		ARIMA	11.786
		Аналіз сингулярного спектру	259.292
		Нейромережа	46.183

Порівняно велика точність нейромережі пов'язана з її архітектурою, проте, це також може бути пов'язано з перенавчанням, що потребує додаткового аналізу. Найбільшу похибку має ряд ринкового індексу S&P 500.

Найбільш цікавими є нестационарні ряди, одним з прикладів яких є дена температура. Графічні результати прогнозування ряду денної температури наведено на рис. 2.3. Графічні результати прогнозування ряду ринкового індексу S&P 500 наведено на рис. 2.4, графік до пунктирної межі – кінець навчальної вибірки. Різні показники шкал пов'язані з предобробкою (масштабуванням) вибірок.

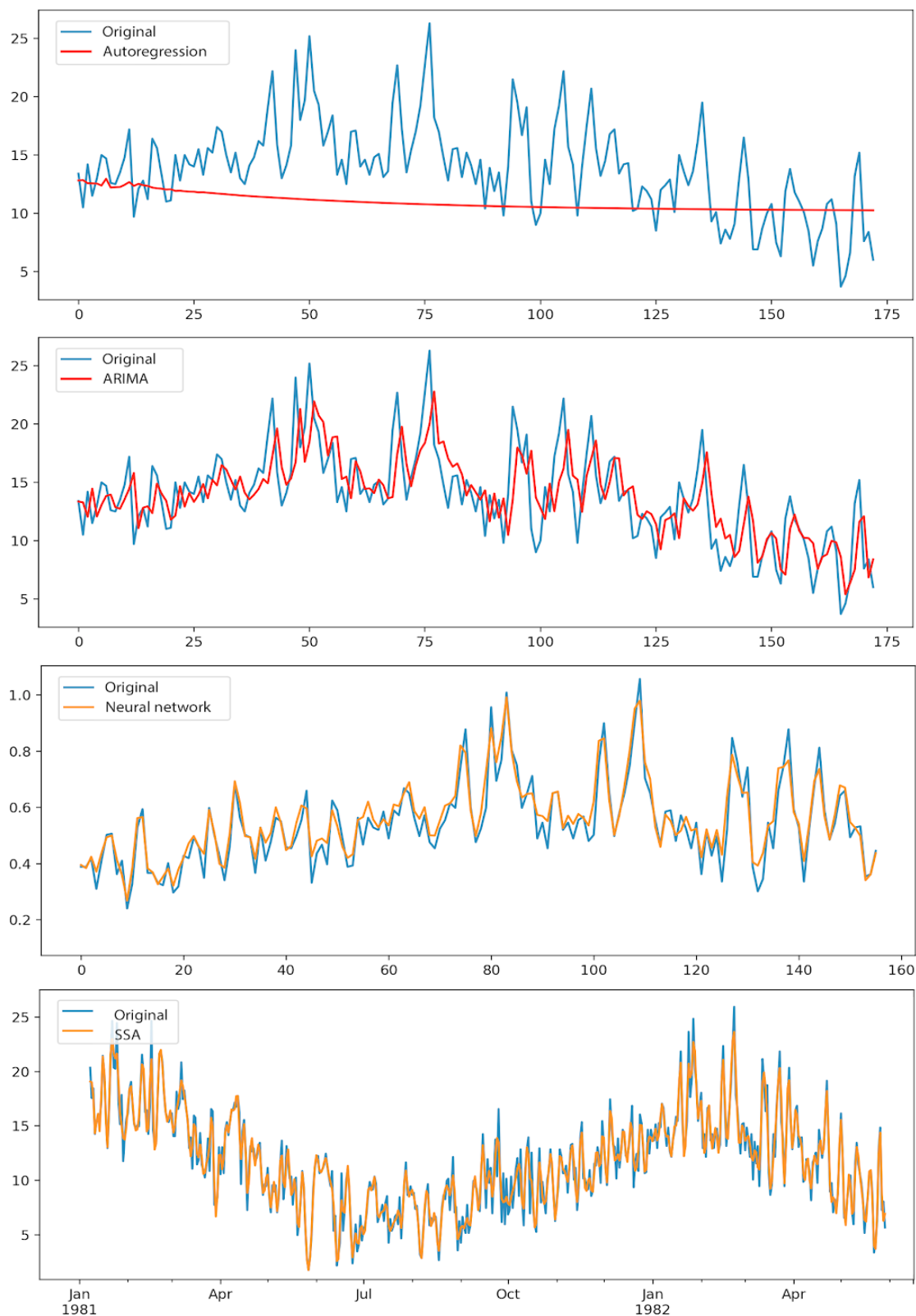


Рисунок 2.3 – Результати застосування моделей до числового ряду значень показників температури за 1981-1982 роки

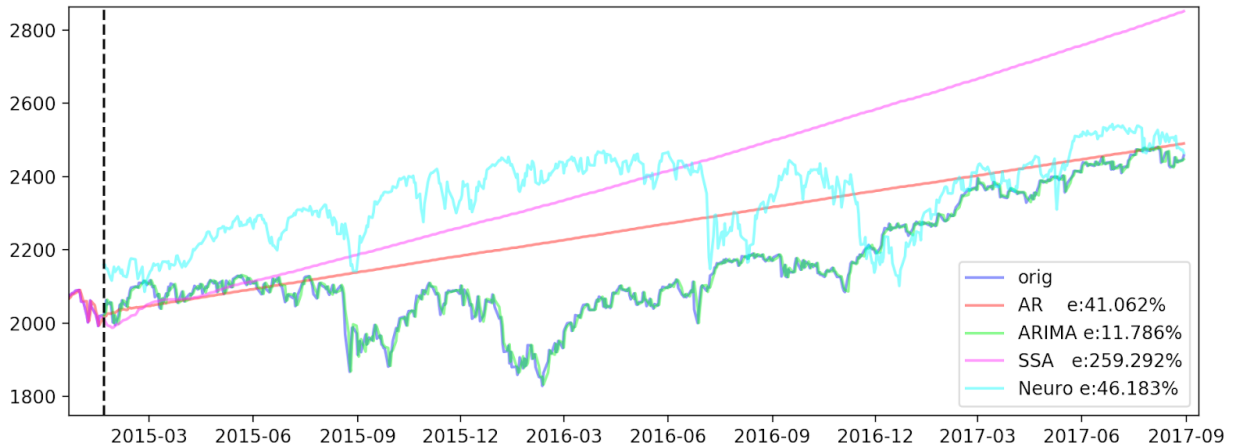


Рисунок 2.4 – Результати застосування моделей до ряду ринкового індексу S&P 500

Проаналізувавши отримані дані, можна визначити область та ефективність даних методів.

Авторегресія ефективно може застосовуватися для стаціонарних рядів оскільки відображає лише лінійну залежність величин, та не відображає їх функціональної зв'язності.

Модель Бокса-Дженкінса (ARIMA) має найменшу похибку, але детальніше розглянувши графік можна зробити висновок, що цей метод повторює оригінальний ряд з певним запізненням, а тому використання його в якості прогнозування не має жодного сенсу.

Аналіз сингулярного спектру (SSA) гарно показує себе на нестационарних рядах в яких можна виділити тренд, періодичні компоненти, шум. В той час як на ряді ринкового індексу S&P 500 здатен відобразити тільки тренд.

Багатошаровий перцептрон – показав не найкращу ефективність, одна графічно дуже схожий результат, це одна з причин чому їх було обрано для дослідження. Додатковим позитивним моментом є можливість використання одночасно декількох джерел даних у якості вхідних даних, що теоретично

дозволить підвищити ефективність. Проте виникає наступна проблема – вибір кращої конфігурації нейронної мережі.

2.8 Висновок

Напрямок, що дозволяє вирішити поставлену проблему є нейромереві методи, однак існує проблема підбору ефективної конфігурації нейромереві.

Розв'язання задачі прогнозування фінансових котирувань представлених у вигляді часових рядів значно покращить ефективність фінансових аналітиків.

При проведенні аналізу літературних джерел були оцінені основні сучасні методи розв'язку задачі прогнозування часових рядів і виявлені їх недоліки, основними з яких є:

- низька точність при прогнозуванні нестационарних рядів без явних періодичних компонент;
- велика зашумленість початкових даних;
- необхідність постійного вдосконалення та зміни моделі.

Зазначені недоліки є досить вагомими і роблять практично неможливим ефективне застосування даних методів без залучення людини для остаточного прийняття рішення.

Основні завдання, які мають бути виконані:

- розробити алгоритм визначення ефективної конфігурації штучних нейронних мереж;
- програмно реалізувати розроблений алгоритм;
- провести порівняння розробленого алгоритму з розглянутими вище методами.

3 ВИЗНАЧЕННЯ ЕФЕКТИВНОЇ КОНФІГУРАЦІЇ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ

Для вирішення поставленої задачі прогнозування необхідно визначити ефективну конфігурацію нейронної мережі з урахуванням особливостей часового ряду, що аналізується. Конфігурація нейронної мережі задається багатьма параметрами, серед яких:

- кількість входів;
- кількість виходів;
- кількість проміжних шарів;
- інформацію що подається на вхід;
- інформацію яка отримується на виходів;
- функції поширення;
- зв'язки між шарами;
- правило навчання;
- вибір інформації для входу та виходу.

Ефективність нейронної мережі оцінюється за значенням середньої квадратичної похибки або середньої абсолютної похибки. Таким чином оригінальну задачу можна представити у вигляді оптимізації багатопараметричної функції, оскільки про часовий ряд зазвичай нічого невідомо окрім його значень, то доцільним буде використання евристичного підходу, а саме – ідей генетичного алгоритму (ГА) [34].

3.1 Вхідні дані

В якості вхідних даних для штучної нейронної мережі можуть виступати як зміни цін одного типу, наприклад цін закриття, так і комбінації змін різних типів цін в межах певного інтервалу - вікна. Недостатньо широке вікно не здатне надати всю необхідну інформацію, що, знижує ефективність

передбачення. З іншого боку, розширення вікна до таких значень, коли захоплюються далекі значення ряду, підвищує розмірність мережі, що в свою чергу, призводить до зниження точності передбачення через розростання розміру мережі. Виходом з цієї ситуації є альтернативні способи кодування минулого поведінки ряду. Альтернативним поданням вхідної інформації можна вважати значення технічних індикаторів, побудовані на основі цінової динаміки.

Технічний індикатор – функція, побудована на значеннях статистичних показників торгів (ціни, обсяг торгів і т. д.), аналіз поведінки якої покликаний відповісти на питання зміниться або збережеться поточна тенденція на ринку. Іншими словами, це певна функція яка певним чином оброблює дані, наприклад:

- ковзаюче середнє;
- перетворення значення ціни у зміну відносно попереднього значення.

Очевидні плюси такого підходу:

- значення кожного з індикаторів залежить від певного числа значень часового ряду в минулому, таким чином використання сукупності кількох індикаторів дозволяє охопити ринок широким поглядом і подивитися на ринкову ситуацію в минулому з різних точок зору;
- численність індикаторів ускладнює їх використання, тоді як кожен з них може виявитися корисним в застосуванні до конкретного фінансового ряду;
- вибірка з індикаторами зазвичай порівняно невелика відносно оброблених нею даних, і, відповідно, кількість вхідних нейронів мережі не велика.

Отже для нейронної мережі вхідні дані – це значення числових рядів

Для задачі конфігурації нейронної мережі вхідні дані – це можливі параметри її конфігурації.

3.2 Застосування генетичного алгоритму для вирішення задачі конфігурації нейронної мережі

Для кожного параметру конфігурації нейронної мережі, що досліджується задаються можливі значення та кодуються побудова нейромережі на основі заданих периметрів. В контексті ГА, гени – параметри конфігурації ШНМ, особа – навчена ШНМ. Нижче наведена узагальнена схема застосування генетичного алгоритму до вирішення поставленої задачі:

Крок 1. Задається множина нежиттєздатних осіб, як порожня.

Крок 2. Генерується початкове покоління (набори) параметрів конфігурацій нейронної мережі.

Крок 3. Виконується схрещення та мутації у поколінні, таким чином що нові особи мають бути відсутні у множині нежиттєздатних осіб.

Крок 4. Для кожного набору параметрів будується та навчається нейронна мережа (особа).

Крок 5. Для кожної навченої нейронної мережі, визначається її життєздатність, як значення середньої квадратичної похибки або як значення середньої абсолютної похибки (чим менше значення, тим дана особа – набір параметрів – життєздатніша).

Крок 6. Найменш нежиттєздатна особа прибирається з покоління, та додається у множину нежиттєздатних осіб. На даному кроці, найбільш життєздатну особу вже можна використовувати для прогнозування, замінюючи її на біль життєздатну, якщо така знайдеться у наступних поколіннях.

Крок 7. Якщо поточне покоління пусте ми перебрали усі можливі варіанти. Якщо ми перевищили можливу кількість ітерацій – слід припинити роботу. Інакше, перейти до кроку 4, і починається нова ітерація.

3.3 Математична постановка задачі

Нехай s – часовий ряд, що досліджується; N_{pc} – кількість параметрів конфігурації нейромережі; конкретний набір параметрів конфігурації представимо у вигляді вектора $\mathbf{x}=(x_1, \dots, x_{N_{pc}})$, де $x_i \in X_i$, X_i – множина можливих значень параметру конфігурації i , $i=1\dots N_{pc}$, $X = X_1 \times \dots \times X_{N_{pc}}$; $f_s(\mathbf{x})=f(x_1, \dots, x_{N_{pc}})$ – нейромережа що має конфігурацією \mathbf{x} ; $z(f_s(\mathbf{x}))$ – функція що обернено визначає ефективність (життєздатність) нейромережі $f_s(\mathbf{x})$ (тобто найменше $z(f_s(\mathbf{x}))$ відповідатиме найбільш життєздатній конфігурації). Необхідно знайти, такий \mathbf{x} , який мінімізує наступний функціонал:

$$z(f_s(x)) \rightarrow \min \quad (3.1)$$

та $\mathbf{x} \in X$

Алгоритм розв'язання задачі

Крок 1. Довільним чином, сформуємо початкову множину Y векторів $\mathbf{y} = (y_1, \dots, y_n)$, де $y_i \in X_i$. Позначимо множину D , як порожню множину векторів – хибних рішень (нежиттєздатна популяція).

Крок 2. Виконаємо операції схрещення та мутації векторів \mathbf{y} .

Схрещення, довільним чином оберемо $t = 1..n$, довільним чином оберемо вектори $\mathbf{a} \in Y$ та $\mathbf{b} \in Y$. Додамо до множини Y , вектор $(a_1, \dots, a_p, b_{t+1}, b_n) \notin D$ та вектор $(b_1, \dots, b_p, a_{t+1}, a_n) \notin D$.

Мутація, довільним чином оберемо $t = 1..n$, довільним чином оберемо вектор $\mathbf{a} \in Y$. Додамо до множини Y , вектор $(a_1, \dots, a_{t-1}, c, a_{t+1}, a_n) \notin D$, де $c \in X_t$.

Вилучимо з множини Y хибні рішення:

$$Y = Y \setminus D, \quad (3.2)$$

Крок 4. Визначимо найменш життєздатний вектор d :

$$d = \max_y \{z(f_s(y))\}, y \in Y \quad (3.3)$$

Визначимо найбільш життєздатний вектор x' :

$$x' = \min_y \{z(f_s(y))\}, y \in Y \quad (3.4)$$

Крок 5. Оновимо множини векторів:

$$\begin{aligned} Y &= Y \setminus \{d\}, \\ D &= D \cup \{d\}. \end{aligned} \quad (3.5)$$

Крок 6. Якщо множина Y порожня або кількість ітерацій перевищила встановлену, то x' – шуканий ефективний розв'язок, інакше – перейти до кроку 2 (почати нову ітерацію).

У результаті ми отримуємо конфігурацію x нейронної мережі яка достатньо якісно виконує поставлену задачу.

3.4 Практичні аспекти застосування алгоритму

Ідея застосування генетичного алгоритму для визначення ефективної конфігурації штучної нейронної мережі полягає у наступному: сформувати множини можливих параметрів нейронної мережі (X_i) та застосувати покроковий алгоритм наведений вище. Гарною властивістю є можливість використання працюючої нейромережі вже після першої ітерації, і у процесі роботи алгоритму – віднаходження більш ефективних нейромереж, використовувати їх. Слід додатково зауважити формування множин X_i . Кожна множина X_i містить можливі значення певного параметру конфігурації нейронної мережі. Наприклад, кількість входів, тобто об'єм вхідної інформації, від 10 записів до 100 $X_i = \{10 \dots 100\}$, або ж фіксовані $X_i = \{10, 25, 100\}$. Доцільним може бути обрання декількох критеріїв, що

описують вхідні дані, наприклад: X_1 це останні $\{10...100\}$ значень з джерела 1, а X_2 – предостанні $\{15,25,50\}$ значень з другого джерела. Також, для відсіювання зайвих параметрів, доцільним буде додати 0 до їх можливих значень, наприклад: кількість вхідних значень, що надходять з другого джерела, $X_2 = \{0,15,25,50\}$. Окремими параметром можна закодувати базову архітектуру використовуваної нейромережі (багатошаровий перцептрон, рекурсивна нейромережа та ін.), і додатково об'єднувати їх у одну нейромережу з'єднуючи входи та виходи, однак це може викликати додаткові складнощі при кодуванні такої системи, проте цього можна уникнути дотримуючись високої модульності та низької зв'язності при кодуванні системи.

Через перебір великої кількості конфігурацій нейронних мереж та їх навчання, виконання алгоритму потребує значного часу, проте вже після першої ітерації ми матимемо нейронну мережу яку можна використовувати для прогнозування.

3.5 Висновок

У даному розділі був описаний алгоритм знаходження ефективної конфігурації штучних нейронних мереж. Описано формування вхідних даних, спосіб задання параметрів конфігурації для навчання ШНМ.

Також описано можливість практичного застосування, переваги та недоліки розробленого методу.

4 ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ПРОДУКТУ

4.1 Загальні вимоги до архітектури програмного забезпечення

Вимоги до архітектури програмного забезпечення складені з урахуванням потреби в побудові зручного та простого в користуванні додатку. Також враховувалось те, що для роботи алгоритму необхідна постійна побудова нейромереж з різними конфігураціями, це призводить до потреби в модульності системи. Для вирішення поставленого завдання, система повинна надавати можливість виконувати наступні функції:

- імпорт даних часового ряду;
- запуск роботи основного алгоритму з заданими параметрами;
- перегляд результатів роботи алгоритму.

Важливою додатковою вимогою є можливість під час роботи основного алгоритму переглядати прогрес та ставити інші задачі: імпортувати інші дані, запускати алгоритм з іншими налаштуваннями, переглядати історію навчання нейромереж.

Для забезпечення виконання цих вимог було вирішено розмежувати систему на 3 сервіси:

- сервіс взаємодії з користувачем;
- сервіс задач – приймає в роботу довготривалі завдання (навчання нейромереж, робота алгоритму, імпорт даних) та виконує їх, сповіщаючи про це сервіс взаємодії з користувачем;
- сховище даних – база даних.

Дані вимоги призвели до вибору наступних засобів розробки:

- основна мова програмування Python [35];

- платформа Django [36] та фреймворк Bootstrap [37] для інтерфейсу користувача;
- бібліотека Celery [38] для виконання довготривалих завдань, що дозволяє використовувати для довготривалих задач та сервісу взаємодії з користувачем одну кодову базу;
- брокер повідомлень (message broker) RabbitMQ [39] для взаємодії між інтерфейсом користувача та довготривалими завданнями;
- засіб автоматизації розгортання та керування застосунками – Docker [40];
- база даних PostgreSQL [41].

Архітектура рішення та організація компонентів зображена на рис. 4.1.

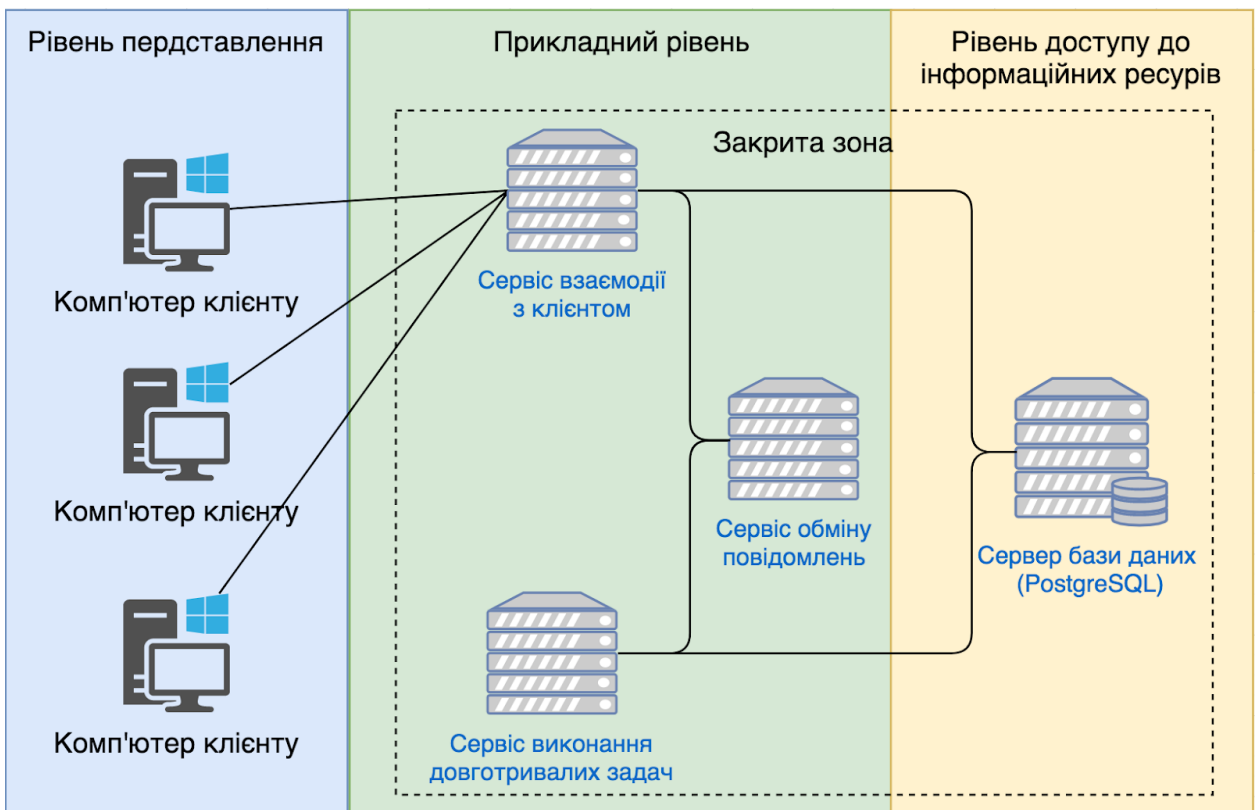


Рисунок 4.1 – Архітектура створеного рішення та організація компонентів
 Дана архітектура є класичною трирівневою архітектурою, у ній сервери є незалежними один від одного. Клієнти також функціонують паралельно і незалежно один від одного. Немає жорсткої прив'язки клієнтів до серверів.

Більш ніж типовою є ситуація, коли один сервер одночасно обробляє запити від різних клієнтів; з іншого боку, клієнт може звертатися то до одного сервера, то до іншого. Клієнти мають знати про доступні сервери, але можуть не мати жодного уявлення про існування інших клієнтів.

Відокремлюється окремий програмний рівень, на якому зосереджується прикладна логіка застосування. Програми проміжного рівня можуть функціонувати під управлінням спеціальних серверів застосувань, але запуск таких програм може здійснюватися і під управлінням звичайного веб-сервера. Нарешті, управління даними здійснюється сервером даних.

Для роботи з системою користувач використовує стандартне програмне забезпечення — звичайний браузер. Це позбавляє його необхідності завантажувати та встановлювати спеціальні програми (хоча інколи така необхідність все-таки виникає). Але користувачеві слід надати інтерфейс, який дозволяв би йому взаємодіяти з системою і формувати запити до неї. Форми, що визначають цей інтерфейс, розміщуються на веб-сторінках та завантажуються разом з ними.

Для розробки інтерфейсу користувача було вирішено використовувати архітектурний шаблон Модель-Вид-Контролер(MVC) [42]. За даним шаблоном система поділяється на 3 частини, їх організація зображена на Рис. 4.2. До обов'язків компоненту Модель (Model) входить зберігання даних і забезпечення інтерфейсу до них. Вигляд (View) відповідальний за представлення цих даних користувачеві. Контролер (Controller) керує компонентами, отримує сигнали у вигляді реакції на дії користувача, і повідомляє про зміни компоненту Модель. Така внутрішня структура в цілому поділяє систему на самостійні частини і розподіляє відповідальність між різними компонентами.

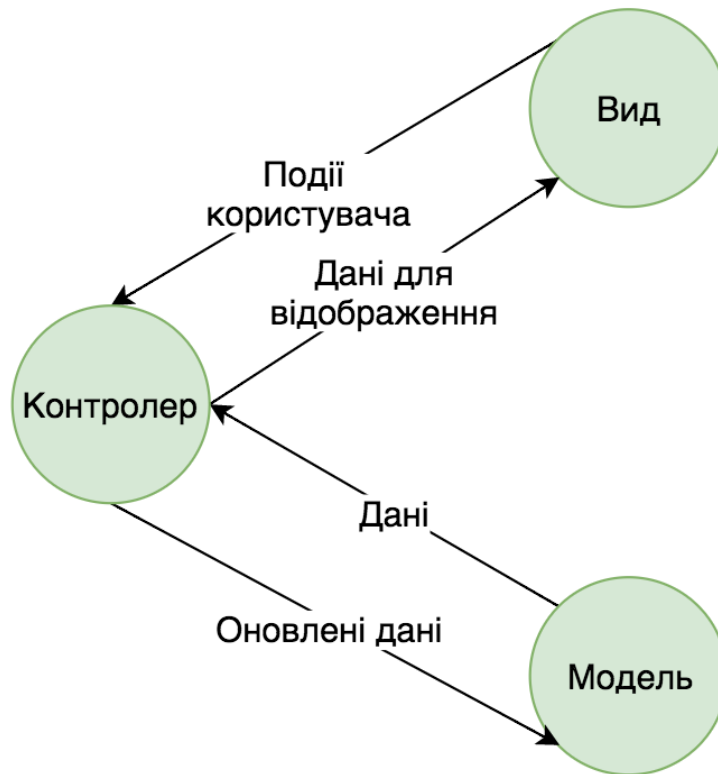


Рис 4.2 – шаблон Модель-Вид-Контролер

Як зазначалося раніше, для комунікації між сервісом інтерфейсу користувача та сервісом виконання довготривалих задач використано брокер повідомлень – систему обміну повідомленнями, один з підходів реалізації взаємодії компонентів і систем, що використовується в паралельних обчисленнях, об'єктно-орієнтованому програмуванні, також – одна з форм взаємодії між процесами в операційних системах, в мікроядерних операційних системах підхід використовується для обміну інформацією між одним з ядер і одним або більше виконуючих блоків. RabbitMQ – платформа, що реалізує систему обміну повідомленнями між компонентами програмної системи на основі стандарту AMQP (Advanced Message Queuing Protocol)[43].

RabbitMQ створений на основі випробуваної Open Telecom Platform[44], що забезпечує високу надійність і продуктивність промислового рівня і написаний на мові Erlang. Підтримується горизонтальне масштабування для побудови кластерної архітектури.

Оскільки система має багато сервісів, то для автоматизації розгортання було вирішено використовувати Docker та Docker-Compose [45].

Docker – програмне забезпечення для автоматизації розгортання і управління застосунками в середовищі віртуалізації на рівні операційної системи. Дозволяє «упакувати» застосунки з усім його оточенням і залежностями в контейнер, який може бути перенесений на будь-яку Linux-систему, а також надає середовище з управління контейнерами. З 2015 року застосовував власну бібліотеку, що абстрагує віртуалізаційних можливості ядра Linux. З появою Open Container Initiative [46] почався перехід від монолітної до модульної архітектури.

Docker-compose – це інструмент для визначення та керування багатьма контейнерами Docker. Для налаштування сервісів використовується спеціальний файл. Потім, за допомогою однієї команди, створюються і запускаються всі сервіси з конфігурації. Особливості Compose, які роблять його ефективним, є:

- кілька ізольованих середовищ на одному хості;
- перестворення лише тих контейнерів(сервісів), які змінилися;
- можливість використання змінних при визначенні конфігурації сервісів
- можливість переміщення композиції між середовищами.

4.2 Вхідні та вихідні дані

Вхідними даними до програми є:

- часові ряді;
- множина можливих конфігурацій нейромережі.

Вихідними даними є:

- конфігурація ефективної нейромережі.

4.3 Структура класів програмного забезпечення

Для опису даних системи було використано наступні класи(моделі даних):

- SeriesInfo – описує завантажені користувачем дані
- LearningTask – описує можливі параметри конфігурації ШНМ розробленого алгоритму;
- LpTechnicalIndicator – модель, що описує технічний індикатор, який обробляє дані до відправки як вхідні чи вихідні дані у ШНМ. Дозволяє виконати динамічне зв'язування коду для пред-обробки даних.
- LpLayer – описує параметри шару ШНМ;
- LearningHistory – містить дані про конкретну “особу” створену підчас роботи основного алгоритму в межах LearningTask.

Структуру перелічених класів наведено на рис 4.3.

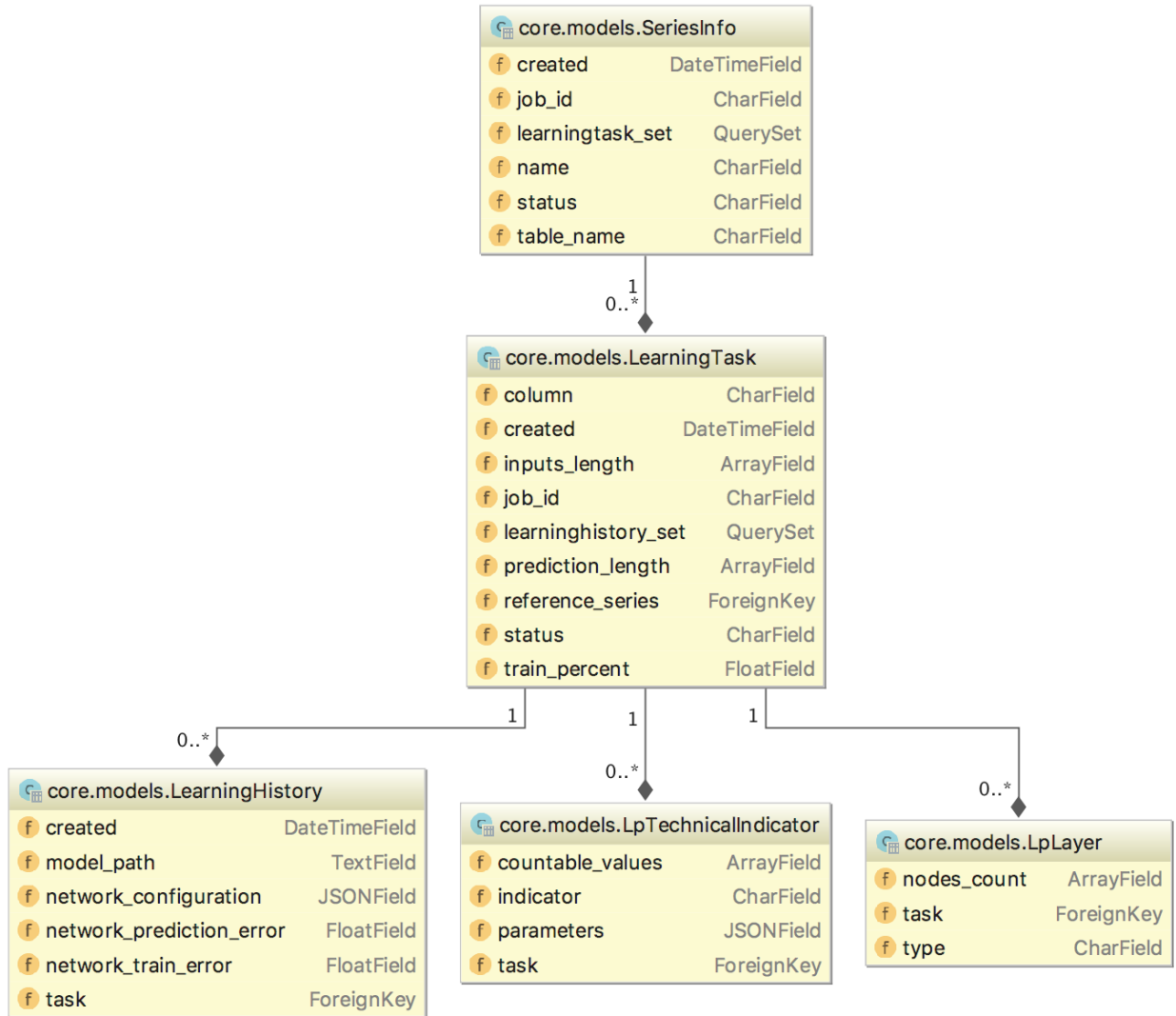


Рисунок 4.3 – Діаграма класів моделей даних

Повну діаграму класів наведену у додатку А (“Діаграма класів”).

4.4 Керівництво користувача

У даному розділі наведено опис інтерфейсу користувача та опис роботи розробленої системи. Для початку, користувачу необхідно завантажити досліджуваний часовий ряд, використовуючи форму зображену на рис. 4.4.

Upload series
Uploaded series
Start train
Train history

Series name

Table name

Csv file

Choose File

SP500.csv

Date format

Submit

Рисунок 4.4 – Форма імпорту даних

Після завантаження файлу, та попередньої валідації, система створить задачу для внесення завантажених до бази даних. Імпорт файлу розміром 287 мегабайт, що містить 4433010 записів, кожен по 5 кортежів (похвилинно: часова мітка, ціна відкриття, ціна закриття, найвища ціна, найшижча ціна) займає 15 хвилин. Переглянути прогрес та статус імпорту даних можна використовуючи інтерфейс зображений на рис 4.5.

Upload series
Uploaded series
Start train
Train history

Id	Date added	Name	Table name	Total rows	Status	Columns	Job info
10	May 13, 2018, 12:12 p.m.	S&P 500	sp500	2198	COMPLETED	['date_time', 'sp_0']	{'result': 'Ok, rows processed 2198, rows skipped 0, time spend 0:00:00.470912', 'state': 'SUCCESS'}
9	May 8, 2018, 2:36 p.m.	EUR/USD	eur_usd	4857142	COMPLETED	['date_time', '_open_0', '_high_1', '_low_2', '_close_3', '_volume_4']	{'result': 'Ok, rows processed 4857142, rows skipped 0, time spend 0:14:59.803647', 'state': 'SUCCESS'}
5	May 8, 2018, 2:15 p.m.	daily temp	daily_temp	507	COMPLETED	['date_time', 'temperatur_0']	{'result': 'Ok, rows processed 507, rows skipped 0', 'state': 'SUCCESS'}

Рисунок 4.5 – Інтерфейс перегляду задач імпорту даних

Наступним кроком буде створення задачі на виконання основного алгоритму – задання можливих параметрів конфігурації ШНМ, для цього використовується інтерфейс зображений на рис 4.6.

Upload series
Uploaded series
Start train
Train history

Reference series	10 S&P 500 sp500
Column	sp_0
Train percent	90
Prediction length	5,10,20
Input length	10,20,50
Layers	
#0	
Type	Perceptron
Nodes count	100,300,500
	Remove
#1	
Type	Perceptron
Nodes count	200,400,600
	Remove
#2	
Type	Perceptron
Nodes count	0,50,100
	Remove
	Add Layer
Technical indicators	
#0	
Indicator	Moving Average
Countable values	5,10,20
Window size	10,20
	Remove
#1	
Indicator	Diff
Countable values	5,20
	Remove
	Add Technical indicator
	Submit

Рисунок 4.6 – Створення задачі навчання

Перебіг навчання та інформацію застосування отриманої моделі можна переглянути використовуючи форми зображені на рис 4.7 та рис 4.8.

Created	Series	Column	Prediction length	Input length	Train percent	Best	Learned models	Status
May 13, 2018, 10:56 p.m.	10 S&P 500	sp_0	[5, 10, 20]	[10, 20, 50]	80.0	0.0883326381444931	85	{'result': {'msg': 'Processed gens 85/2916'}, 'state': 'PROGRESS'}

Рисунок 4.7 – Історія задач навчання

Created	Configuration	Train mse	Forecast mse
May 14, 2018, 12:27 a.m.	{'layers': [{'type': 'pr', 'nodes_count': 500}, {'type': 'pr', 'nodes_count': 600}, {'type': 'pr', 'nodes_count': 50}], 'is_valid': True, 'indicators': [{'type': 'ma', 'window_size': 10, 'countable_value': 10}, {'type': 'diff', 'countable_value': 5}], 'input_length': 10, 'prediction_length': 20}	0.118263587355614	0.0880122408270836
May 14, 2018, 12:17 a.m.	{'layers': [{'type': 'pr', 'nodes_count': 500}, {'type': 'pr', 'nodes_count': 400}, {'type': 'pr', 'nodes_count': 50}], 'is_valid': True, 'indicators': [{'type': 'ma', 'window_size': 10, 'countable_value': 5}, {'type': 'diff', 'countable_value': 20}], 'input_length': 10, 'prediction_length': 20}	0.0946006402373314	0.0883326381444931
May 14, 2018, 12:16 a.m.	{'layers': [{'type': 'pr', 'nodes_count': 100}, {'type': 'pr', 'nodes_count': 200}], 'is_valid': True, 'indicators': [{'type': 'ma', 'window_size': 20, 'countable_value': 20}, {'type': 'diff', 'countable_value': 5}], 'input_length': 10, 'prediction_length': 20}	0.0679208561778069	0.0892160832881927
May 14, 2018, 12:13 a.m.	{'layers': [{'type': 'pr', 'nodes_count': 500}, {'type': 'pr', 'nodes_count': 600}, {'type': 'pr', 'nodes_count': 100}], 'is_valid': True, 'indicators': [{'type': 'ma', 'window_size': 20, 'countable_value': 10}, {'type': 'diff', 'countable_value': 20}], 'input_length': 10, 'prediction_length': 20}	0.0835744440555573	0.089500218629837

Рисунок 4.8 – Інформація по “покоління” створених в ході виконання задачі навчання

4.5 Висновок

В даному розділі були описано роботу системи та продемонстровано роботу інтерфейсу користувача.

5 АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

5.1 Вхідні дані

Для дослідження експериментальних даних, було обрано ряд S&P 500 [27]. В якості вхідних даних алгоритму було використано наступні значення:

- а) досліджуваний ряд S&P 500;
- б) інтервал передбачення – значення 5, 10 або 20;
- в) кількість вхідних даних – значення 10, 20 або 50;
- г) технічні індикатори:
 - 1) різниця попереднього – кількість врахованих значень 5 або 20;
 - 2) ковзне середнє:
 - 2.1) кількість врахованих значень 5 або 20;
 - 2.2) розмір віка 10 або 20;
- д) шари нейронної мережі:
 - 1) перший – перцептрон, з кількістю вузлів: 100, 300 або 500;
 - 2) другий – перцептрон, з кількістю вузлів: 200, 400 або 600;
 - 3) третій – перцептрон, відсутній або містить 50 або 100 вузлів.

5.2 Інтерпретація отриманих результатів

Найкращі 3 моделі наведено у таблиці 5.1 та найгірші 3 моделі наведено у таблиці 5.2. Розгорнуту таблицю випробувань наведено у додатку А (“Результати експерименту”).

Таблиця. 5.1 – 3 кращих моделі

Конфігурація мережі	Інтервал передбачення	Середньо квадратична похибка навчання	Середньо квадратична похибка передбачення
Вхідні дані – 10 значень; Середне-ковзне 20 значень, вікно – 20; Різниця попереднього – 20; Тришаровий перцептрон 100, 200, 50;	20	0.0949	0.0663
Вхідні дані – 50 значень; Середне-ковзне 20 значень, вікно – 20; Різниця попереднього – 20; Тришаровий перцептрон 100, 200, 50;	20	0.0929	0.0666
Вхідні дані – 10 значень; Середне-ковзне 20 значень, вікно – 20; Різниця попереднього – 5; Двошаровий перцептрон 100, 200;	20	0.0892	0.0679

Таблиця. 5.2 – 3 гірших моделі

Конфігурація мережі	Інтервал передбачення	Середньо квадратична похибка навчання	Середньо квадратична похибка передбачення
Вхідні дані – 10 значень; Середне-ковзне 10 значень, вікно – 20; Різниця попереднього – 5; Тришаровий перцептрон 500, 200, 100;	10	885015.93	2801583.75
Вхідні дані – 50 значень; Середне-ковзне 20 значень, вікно – 20; Різниця попереднього – 20; Тришаровий перцептрон 100, 400, 50;	10	7452.63	28947.29
Вхідні дані – 10 значень; Середне-ковзне 5 значень, вікно – 10; Різниця попереднього – 5; Тришаровий перцептрон 500, 400, 100;	20	6108.01	18584.27

Як видно з отриманих результатів, найкращий прогноз можна отримати коли перший та останній шари багат шарового перцептронну мають середню кількість вузлів, а окрім самого числового ряду також враховано ковзне середнє, й вхідний ряд також представлено у вигляді різниць попередніх значень. В найкращому випадку отримано середньо-квадратичну похибку 0.0663, що значно краще за початкові результати нейронної мережі отриманої у розділі 2.7. Однак, надзвичайно мала похибка може свідчити про перетренованість моделі. Найгірший прогноз отримано коли перший шар має непомірно велику кількість вузлів.

5.3 Висновки

В даному розділі було розглянуто роботу алгоритму на прикладі мережі даних часового ряду S&P 500. Також наведено інтерпретацію практичних результатів та зроблено висновки щодо ефективності розробленого методу.

ВИСНОВКИ

У даній магістерській дисертації було розглянуто поширені методи прогнозування часових рядів, проведено їх порівняльний аналіз та визначення їх ефективності при прогнозуванні фінансових часових рядів. На основі виконаного порівняльного аналізу було визначено напрямок дослідження та остаточно сформульовано задачу. Було розроблено та закодовано алгоритм, що поєднує в собі генетичний алгоритм та штучні нейромережі для визначення ефективної архітектури нейронної мережі з метою її застосування для прогнозування фінансових часових рядів. Розроблений алгоритм дозволяє наближено визначити необхідні і достатні параметри для оцінки стану ринку (виконання прогнозу фінансових часових рядів) та постійно навчається для використання нових даних.

Для вирішення завдання модифікації генетичного алгоритму та визначення оптимальної структури нейронної мережі, а також визначення необхідної кількості інформації про ці параметри для оперативного і точного рішення, в роботі були розглянуті основні принципи застосування генетичних алгоритмів, а також опрацьований метод взаємодії генетичних алгоритмів спільно з нейронними мережами.

Програмна реалізація системи написана мовою програмування Python 3 з використанням фреймворку Tensorflow SDK для моделювання нейронних мереж.

Таким чином в результаті проведеної роботи була досягнута поставлена мета дослідження.

За напрямком дослідження було опубліковано такі матеріали:

- огляд методів аналізу фінансових часових рядів [47];
- порівняльний аналіз методів прогнозування стаціонарних та нестаціонарних часових рядів [48];

- застосування генетичного алгоритму для визначення ефективної конфігурації штучних нейронних мереж [49].

Напрямами подальшого дослідження можуть бути:

- збільшення різноманітності даних, що подаються на вхід системи.
- розширення можливостей системи, шляхом вибору інших вихідних даних (наприклад: середня ціна).
- використання інших архітектур нейронних мереж окрім багат шарового перцептрон.
- підвищення швидкості обчислень, за рахунок використання технологій паралельних обчислень.

ПЕРЕЛІК ПОСИЛАНЬ

1. Гипотеза эффективного рынка [Электронный ресурс] // Режим доступа: <https://postnauka.ru/longreads/62491>
2. Петерс Эдгар Хаос и порядок на рынках капитала [Текст] / Эдгар Петерс // М.: Мир, 2000. – 336 с.
3. George Routledge Gibson. The stock exchanges of London, Paris, and New York: a comparison [Текст] / George Routledge Gibson // New York; London : G.P. Putnam, 1889. – 155 с.
4. Мэрфи Джон Дж. Технический анализ фьючерсных рынков. Теория и практика [Текст] / Джон Дж. Мэрфи // «Альпина Паблишер», 2011. – 616 с.
5. Efficient Market Hypothesis: Is The Stock Market Efficient? [Электронный ресурс] // Режим доступа: <https://www.investopedia.com/articles/basics/04/022004.asp>
6. Швагер Джек. Технический анализ. Полный курс [Текст] / Джек Швагер. // «Альпина Паблишер», 2001. – 805 с.
7. No such thing as risk free [Электронный ресурс] // Режим доступа: <https://www.economist.com/blogs/freeexchange/2011/11/financial-market>
8. Подробно про технический анализ [Электронный ресурс] // Режим доступа: <http://berg.com.ua/tech/technical-analysis-in-detail/>
9. Что такое Фундаментальный анализ [Электронный ресурс] // Режим доступа: <http://allfi.biz/glossary/eng/F/fundamentalanalysis.php>
10. Хегстром Роберт Дж. Как 5 долларов превратить в 50 миллиардов. Простые правила великого инвестора [Текст] / Роберт Дж. Хегстром, Уоррен Баффетт // Манн, Иванов и Фербер, 2015. – 352 с.
11. Black F. Noise [Текст] / Black F. // Journal of Finance. 1986. Vol. 41 – с. 529-543.

12. De Long J. B. Noise Trader Risk in Financial Markets [Текст] / De Long J. B., Shleifer A. M., Summers L. H., Waldmann R. J. // Journal of Political Economy. 1990. Vol. 98 – с. 703-738.
13. Shiller R. Do Stock Prices Move Too Much to be Justified by Subsequent Changes in Dividends? [Текст] / Shiller R. // American Economic Review. 1981. Vol. 71. – с. 421-436.
14. Barber B. M. Trading is hazardous to your wealth: The common stock investment performance of individual investors [Текст] / Barber B. M., Odean T. // Journal of Finance. 2000. Vol. 55. № 2. – с. 773-806.
15. T. Plummer. Psychology of Technical Analysis Hardcover [Текст] / T. Plummer. // Irwin Professional Publishing, 1993. – 302 с.
16. Проблемы прогнозирования финансовых рынков [Электронный ресурс] // Режим доступа: <https://geektimes.ru/post/209504/>
17. Линия рынка ценных бумаг [Электронный ресурс] // Режим доступа: <http://allfi.biz/financialmanagement/RiskAndReturns/linija-rynka-cennyh-bumag.php>
18. Фундаментальный анализ Forex [Электронный ресурс] // Режим доступа: <http://enc.fxeuroclub.ru/2/>
19. Нові методи аналізу та прогнозування часових рядів на фінансових ринках [Электронный ресурс] // Режим доступа: http://www.rusnauka.com/13_EISN_2013/Economics/4_136384.doc.htm
20. Lefevre E. Reminiscences of a Stock Operator [Текст] / E. Lefevre // J CreateSpace Independent Publishing Platform; 1 edition (July 15, 2014) – 236 с.
21. Прогнозирование финансовых временных рядов с MLP в Keras [Электронный ресурс] // Режим доступа: <https://habrahabr.ru/post/327022/>

22. Discovering hidden time patterns in behavior: T-patterns and their detection [Електронний ресурс] // Режим доступу: <https://link.springer.com/article/10.3758/BF03200792#citeas>
23. КУРС ЛЕКЦІЙ "НЕЙРОННІ МЕРЕЖІ" [Електронний ресурс] // Режим доступу: <http://www.victoria.lviv.ua/library/students/nn/lecture.html>
24. Minsky P. Perceptrons: an introduction to computational geometry [Текст] / Marvin Minsky, Seymour Papert // – 1969.
25. Hecht-Nielsen R. Neurocomputing [Текст] / R. Hecht-Nielsen // Reading, MA: Addison-Wesley, – 1990
26. Hopfield J. J. Neural networks and physical systems with emergent collective computational abilities [Текст] / J. J. Hopfield // Proceedings of National Academy of Sciences, vol. 79 no. 8 pp. 2554–2558, – April 1982.
27. Штучний нейрон [Електронний ресурс] // Режим доступу: https://uk.wikipedia.org/wiki/%D0%A8%D1%82%D1%83%D1%87%D0%BD%D0%B8%D0%B9_%D0%BD%D0%B5%D0%B9%D1%80%D0%BE%D0%BD
28. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. [Текст] / Ф. Уоссермен // М.: «Мир», – 1992.
29. Хайкин С. Нейронные сети: Полный курс. [Текст] / Пер. с англ. Н. Н. Куссуль, А. Ю. Шелестова. // 2-е изд., испр. М.: Издательский дом Вильямс, 1103 с – 2008
30. Daily minimum temperatures in Melbourne, Australia, 1981-1990 [Електронний ресурс] // Режим доступу: <https://datamarket.com/data/set/2324/daily-minimum-temperatures-in-melbourne-australia-1981-1990#!ds=2324&display=line>
31. Дані ринкового індексу S&P 500 [Електронний ресурс] // Режим доступу: <http://files.statworx.com/sp500.zip>

32. Air Passengers per month. Workshop dataset [Електронний ресурс] // Режим доступу: <https://www.kaggle.com/rakannimer/air-passengers/data>
33. Профессиональный информационно-аналитический ресурс, посвященный машинному обучению, распознаванию образов и интеллектуальному анализу данных. [Електронний ресурс] // Режим доступу: <http://www.machinelearning.ru>
34. Субботін С. О., Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей: Монографія / Під заг. ред. С. О. Субботіна. Олійник А. О., Олійник О. О. // Запоріжжя: ЗНТУ, 2009. – 375 с.
35. About Python [Електронний ресурс] // <https://www.python.org/about/>
36. Why Django? [Електронний ресурс] // Режим доступу: <https://www.djangoproject.com/start/overview/>
37. Introduction to Bootstrap [Електронний ресурс] // Режим доступу: <https://v4-alpha.getbootstrap.com/getting-started/introduction/>
38. Celery – Distributed Task Queue [Електронний ресурс] // Режим доступу: <http://www.celeryproject.org/>
39. RabbitMQ – Messaging that just works [Електронний ресурс] // Режим доступу: <https://www.rabbitmq.com/>
40. What is Docker? [Електронний ресурс] // Режим доступу: <https://www.docker.com/what-docker>
41. PostgreSQL: The world's most advanced open source database [Електронний ресурс] // Режим доступу: <https://www.postgresql.org/>
42. Model-View-Controller [Електронний ресурс] // Режим доступу: <https://ru.wikipedia.org/wiki/Model-View-Controller>
43. Advanced Message Protocol [Електронний ресурс] // Режим доступу: <https://ru.wikipedia.org/wiki/AMQP>

44. Open Telecom Platform [Електронний ресурс] // Режим доступу:
https://ru.wikipedia.org/wiki/Open_Telecom_Platform
45. Overview of Docker Compose [Електронний ресурс] // Режим доступу:
<https://docs.docker.com/compose/overview/>
46. Open Container Initiative [Електронний ресурс] // Режим доступу:
<https://www.opencontainers.org/>
47. Перерва А.С. “Огляд методів аналізу фінансових часових рядів” [Текст] / А.С. Перерва // Актуальные научные исследования в современном мире, Выпуск 3(35) Часть 3, 2018, 98 – 105 сс.
48. Перерва А.С. “Порівняльний аналіз методів прогнозування стаціонарних та нестаціонарних часових рядів” [Текст] / А.С. Перерва, Т. В. Ковалюк // Науковий огляд №3(46), 2018, 101 – 111 сс.
49. Перерва А.С. “Застосування генетичного алгоритму для визначення ефективної конфігурації штучних нейронних мереж” [Текст] / А.С. Перерва, Т. В. Ковалюк // Науковий огляд №3(46), 2018, прийнято до друку.
50. Матеріали науково-практичної конференції «Інформатика та обчислювальна техніка-ІОТ-2018». – м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 23-24 квітня 2018 р.

ДОДАТОК А
Графічний матеріал