

Ганчук А. А., Соловйов В. М., Чабаненко Д. М.

МЕТОДИ
ПРОГНОЗУВАННЯ
навчальний посібник

МЕТОДИ
ПРОГНОЗУВАННЯ
навчальний посібник

ЧЕРКАСИ, 2012

Методи прогнозування

Ганчук Андрій Анатолійович
Соловйов Володимир Миколайович
Чабаненко Дмитро Миколайович

1 квітня 2012 р.

Зміст

1	Вступ	5
2	Подяка	7
3	Класичні методи прогнозування	9
4	Трендові моделі прогнозування	17
4.1	Лабораторна робота № 1. Трендові моделі	32
4.2	Лаб.2. Нелін. тренд. моделі	38
5	Авторегресійні методи прогнозування.	45
5.1	Лаб.3. Моделі ARIMA	56
6	Нейромережеве прогнозування	69
6.1	Лаб.3. Нейромережеві моделі	82
7	Приховані марківські моделі	97
7.1	Приховані марківські моделі	100

8	Методика СЛМ	119
8.1	Технологія прогнозування	119
8.2	Стани у СЛМ	122
8.3	Алгоритм багатокрокового прогнозу	129
8.4	СЛМ та вплив пам'яті. Порядок СЛМ	130
8.5	Ієрархія приростів часу та процедура склеювання	138
8.6	Вибір оптимальних параметрів	146
8.7	Прогнозування часових рядів з точним періодом (синусоїда)	148
8.8	Прогнозування рядів динамічного хаосу	152
8.9	Багатосценарний підхід	154
8.10	Лаб.5. Методи СЛМ	159
8.11	Лаб.6. Багатосценарний СЛМ	170
	8.11.1 Паралельне прогнозування	175
9	Дискретне Фур'є-продовження	179
9.1	Лаб.6. Дискретне Фур'є-продовження	186

Розділ 1

Вступ

Прогнозування стану об'єктів будь-якої природи є дуже актуальною, не розв'язаною на даний час задачею. Велика кількість методів прогнозування, алгоритмів та їх програмних реалізацій [1, 2] вказують на зацікавленість світових науковців даною задачею. Насьогодні відомо понад 200 різноманітних методів прогнозування [1, 2, 3], також з'являються нові підходи, що означає існування проблем, не розв'язаних на даний момент.

Читачу доступна значна кількість підручників [4, 5, 6, 7] та монографій [8, 9, 10, 11], присвячених прогнозуванню. Більшість підручників широко розглядають класичні методи та Microsoft Excel як популярне середовище побудови прогнозних моделей, але рідко містять сучасних методів прогнозування та інструкцій щодо їх застосування. Наукові моно-

графії розкривають передові досягнення сучасних методів, але далеко не завжди описані методи зручно і легко повторити самостійно навіть підготовленому старшокурснику або аспіранту. Тому ціллю даної книги є огляд методів прогнозування, а також комп'ютерних та програмних засобів побудови і використання прогнозних моделей.

Для студентів економічних спеціальностей, аспірантів та викладачів, зацікавлених у практичному використанні прогнозних моделей.

Розділ 2

Подяка

Автори даної роботи висловлюють щирю вдячність Бахрушину Володимиру Євгеновичу, Бідюку Петру Івановичу, Зайченку Юрію Петровичу, Кириченко Людмилі Олександрівні, Савченко Євгенії Анатоліївні, Сапціну Володимиру Михайловичу, Снитюку Віталію Євгеновичу, Степашку Володимиру Семеновичу за конструктивну критику наших досліджень та цікаві бесіди, результатом якої з'явилася дана робота.

Розділ 3

Класичні методи прогнозування

Існує два головних напрями аналізу фінансових ринків: фундаментальний та технічний аналіз. Фундаментальний аналіз полягає у вивченні факторів, які впливають на стан економіки, секторів промисловості та окремих компаній. Наприклад, для передбачення майбутніх цін на окремий вид акцій, фундаментальний аналітик поєднує дослідження стану економіки, секторів промисловості та окремих компаній для визначення “справедливої” ціни (фундаментального значення), яка відображає реальну вартість підприємства та надає можливість передбачення майбутніх змін цієї вартості. Якщо фактична ціна акцій відхиляється від фундаменталь-

10 РОЗДІЛ 3. КЛАСИЧНІ МЕТОДИ ПРОГНОЗУВАННЯ

ного значення, фундаментальні аналітики стверджують, що акції або недооцінені, або переоцінені, а ринкова ціна коливається навколо фундаментального значення.

Існують деякі недоліки, притаманні фундаментальному аналізу: значний час збору даних та розрахунків, що спричинює відставання прогнозу від найсвіжішої інформації, яка постійно з'являється, а також завжди має місце суб'єктивність думок аналітиків.

Технічний аналіз базується на дослідженні динаміки змін ціни в минулому для передбачення майбутнього її значення. В даному дослідженні цей підхід є основним, а прогнози базуються на динаміці денних цін акцій.

Теоретичні основи технічного аналізу базуються на твердженні, що зміни ціни відображають усю важливу інформацію, яка доступна на ринку. Цей факт впливає з припущення неефективності фінансових ринків. Гіпотеза ефективного ринку розглянута в роботах [12, 13]. Невдовзі після появи, гіпотеза ефективного ринку була піддана гострій критиці як теоретиками, так і емпіричними дослідниками.

Паралельно фундаментальному аналізу в даний час отримав розвиток технічний аналіз - метод вивчення рухів цін, головним інструментом якого служать графіки. Своє коріння сучасний технічний аналіз бере з теорії Чарльза Доу, впливаючи з неї прямо або побічно. Технічний аналіз увібрав у себе такі принципи і поняття, як напрям руху цін, "ціни враховують усю відому інформацію", підтвердження і розбіжність, об'єм як дзеркало цінових змін і підтримка/опір.

Розглянемо деякі підходи до класифікації сучасних методів прогнозування.

Підручники з економічного прогнозування [14, 5] традиційно пропонують класифікацію, в якій, за джерелами прогнозної інформації, усі методи прогнозування поділяються на інтуїтивні та формалізовані. Для кожного методу з групи формалізованих будується математична модель, властивості якої використовуються при прогнозуванні. Пропонується поділ формалізованих методів на методи екстраполяції та методи моделювання [14, 5]. Інтуїтивні рекомендується використовувати для довгострокового прогнозування, коли закономірності розвитку важко формалізувати у вигляді математичної або комп'ютерної (алгоритмічної, імітаційної) моделі. Джерелом прогнозної інформації для інтуїтивних методів прогнозування є досвід експертів, тому інтуїтивні методи також варто називати експертними [5].

В посібнику [3] розглядається структурний та формалізований підхід до побудови прогнозних моделей. Під структурним підходом мається на увазі дослідження закономірностей безпосередньо об'єкта, який продукує часові ряди, а під формалізованим - побудова моделей часових рядів, які є похідними певного процесу, суть якого невідома (концепція чорного ящика).

Недоліками загальноприйнятих класифікацій треба зазначити наступні:

1. відсутність поділу моделей на лінійні та нелінійні.
2. Ігнорування сучасних підходів до прогнозування часових рядів, таких як нейронні мережі, теорія нечітких множин, адаптивні фільтри тощо.

12 РОЗДІЛ 3. КЛАСИЧНІ МЕТОДИ ПРОГНОЗУВАННЯ

Нами пропонується удосконалена класифікація методів прогнозування, яка наведена на рисунку 3.1.

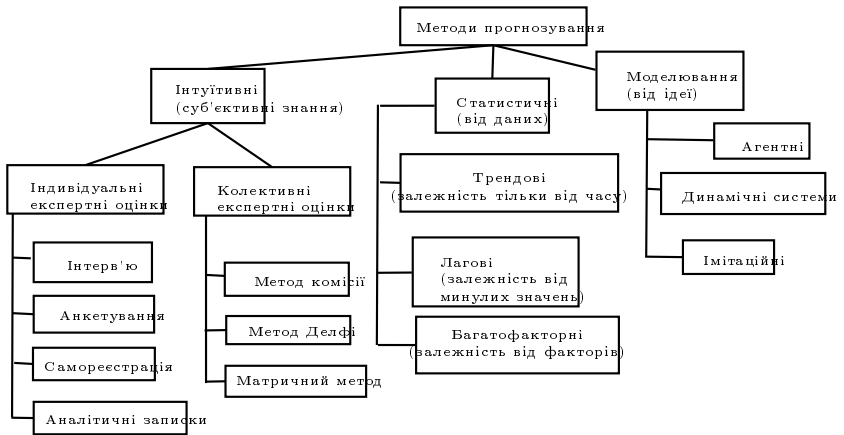


Рис. 3.1: Класифікація методів прогнозування

За основу класифікації взято об'єкт, вихідні дані чи модель якого дозволяє побудувати прогноз. В першій групі методів (експертні методи) таким об'єктом є експерти, які опитуються чи анкетуються. Основна ідея даних методів полягає у проведенні опитування та статистичному опрацюванні суб'єктивних прогнозів експертів. Оскільки формально джерелами знань є експерти, які володіють власними, здебільшого інтуїтивними правилами прийняття рішень, а експер-

тні методи не будують ніяких моделей міркування експертів, а лише збирають дані, то експертні методи не є предметом дослідження даної книги. Але ідея дослідження колективної думки, оцінювання ваги, актуальності, “правильності”, компетентності думок кожного експерта є дуже плідною. Майбутній розвиток цієї ідеї ми вважаємо у побудові системи видобування знань з джерел новин мережі Інтернет, аналізу правдоподібності будь-якої інформації, включаючи гіпотези та чутки, саме засобами інформаційних технологій. В рамках даної роботи даний напрям розглядатись не буде, через його складність та значну кількість невирішених проблем.

Основна увага в даній роботі надається формалізованим методам прогнозування. Згідно з запропонованою нами класифікацією (див. рис. 3.1) дані методи поділяються на статистичні методи та моделювання. Варто зазначити, що термін “моделювання” тут розуміється у вузькому смислі, як побудова моделей, які описують безпосередньо функціонування досліджуваного об’єкту. Статистичні методи тут треба розуміти як методи, які досліджують тільки похідні показники досліджуваного об’єкту, представлені у вигляді часових рядів.

Методи прогнозування з групи “Моделювання” базуються на моделях об’єкту дослідження. До них можна віднести імітаційне моделювання, яке традиційно розуміється як генерування послідовності випадкових подій чи реалізації випадкового процесу із заданими статистичними властивостями (розподілом ймовірності, автоковаріаційною функцією тощо) та оцінювання ризику за допомогою імітування критичних ситуацій в досліджуваній системі. Динамічні си-

14 РОЗДІЛ 3. КЛАСИЧНІ МЕТОДИ ПРОГНОЗУВАННЯ

стеми, в яких закономірності розвитку системи моделюються на основі диференціального чи різницевого рівняння. Якщо розуміти поняття “динамічні системи” тільки як розв’язки диференціальних рівнянь (не звертаючи уваги на закономірності, які описують динамічні системи), то формально дані моделі можна віднести до статистичних, зокрема трендових. У цьому розумінні, розв’язок диференціального рівняння апроксимує досліджуваний ряд. Але в даному випадку акцент робиться на системні закономірності, на яких основана побудова диференціального рівняння, що описує динаміку системи. Тому динамічні системи, на нашу думку, доцільно вважати моделями самої системи.

Потужними можливостями для прогнозування складних систем, зокрема, фінансово-економічних, є агентні моделі [15, 16, 17, 18, 19]. Ідея їх побудови аналогічна моделям молекулярної динаміки у фізиці. Агентна модель є програмною системою, яка включає модель агентів, як окремих відносно простих підсистем досліджуваної системи, та моделі середовища їх взаємодії. Прикладом може бути система підприємств певної галузі, розташованих на певній території, які певним чином взаємодіють між собою, множина трейдерів, які торгують на фінансовому ринку тощо. При взаємодії великої кількості агентів, з’являються емерджентні властивості побудованої системи, які безпосередньо не впливають із властивостей окремих агентів. Тому навіть прості моделі поведінки агента можуть пояснювати емерджентні явища в сучасних складних системах. На шляху ефективної побудови та використання агентних моделей існують певні проблеми, такі як недостатня складність у поведінці агентів, неможли-

вість точного копіювання усіх учасників системи, проблеми калібровки, налаштування невизначених параметрів алгоритму поведінки агента та інші. Тому це питання не входить до компетенції даної роботи.

Основна увага нашої роботи направлена на удосконалення статистичних методів прогнозування. Даний термін треба розуміти у широкому сенсі, як будь-які моделі одновимірних чи багатовимірних часових рядів. Традиційно в даній групі згадуються трендові, економетричні, авторегресійні моделі. Загальноприйнятою для статистичних методів моделлю, яка описує залежність, вважається лінійна, або ті, які зводяться до лінійної. Але дослідження показують, що, по-перше, нелінійні аналітичні моделі, а, по-друге, сучасні нейромережеві, нечіткі моделі показують значно кращі можливості у виявленні та оцінюванні складних залежностей. Тому всі методи групи “Статистичні” піддаються додатковій класифікації залежно від моделі залежності.

Означення 3.1. *Часовим рядом* називається послідовність рівнів ряду

$$\{y_i\} = y_1, y_2, \dots, y_n \quad (3.1)$$

де i – натуральні числа, які відповідають моментам чи проміжкам вимірювання досліджуваної величини. Вважаємо, що вимірювання проходить через рівні проміжки часу. y_i – рівні ряду.

Усі методи прогнозування можуть бути подані у вигляді

16 РОЗДІЛ 3. КЛАСИЧНІ МЕТОДИ ПРОГНОЗУВАННЯ

наступної математичної моделі:

$$y_{t+1} = f(t, y_t, \dots, y_{t-m}, x_t^{(1)}, \dots, x_{t-m_1}^{(1)}, x_t^{(2)}, \dots, x_{t-m_2}^{(2)}, x_t^{(k)}, \dots, x_{t-m_k}^{(k)}), \quad (3.2)$$

де y_t - величина показника, що прогнозується в момент часу t ; $x_t^{(i)}$ - величина i -го фактора в момент часу t , який впливає на y ; m, m_1, \dots, m_k - довжини "пам'яті" ряду, які використовуються при прогнозуванні. Відмінності різних методів будуть стосуватися структури вхідних даних x та виду функції $f(\dots)$. Таким чином, можна виділити такі види залежностей, які використовуються у сучасних методів прогнозування:

1. лінійна,
2. поліноміальна,
3. нейромережева,
4. нечітка,
5. випадкова.

В наступних розділах даної роботи будуть розглядатись вищезазначені моделі залежностей у вигляді методів прогнозування.

Розділ 4

Перспективи використання трендових моделей при побудові сучасних методів прогнозування

Технічний аналіз [20, 21, 22, 23] полягає у вивченні минулих цін з метою визначення ймовірного напрямку їх розвитку в майбутньому. Поточна динаміка цін (тобто поточні очікування) порівнюється із зіставною динамікою цін у минулому, за

18 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

допомогою чого досягається більш менш реалістичний прогноз.

Незалежно від того, якому типу аналізу віддає перевагу трейдер [23] – фундаментальному або технічному, при складанні ринкових прогнозів він звертається до поняття “Лінія тренду” (trendline). Поняття “Лінія тренду” часто трактується неоднозначно і непослідовно. Враховуючи фрактальність, на різних часових рівнях може існувати декілька ліній тренду, які адекватно описують динаміку, що сформувалася. Тому при дослідженні трендів обов’язково необхідно включати поняття “Горизонт прогнозування” та “інтервал трендостійкості” тощо.

Посібники по технічному аналізу пропонують методику вибору двох критичних точок, через які рекомендується проведення лінії тренду [23]. Графічний аналіз ліній тренду втратив минулу суб’єктивність і перетворився на чисто механічну процедуру. З’явилися чіткі критерії проривів лінії тренду і можливість легко розрахувати цінові орієнтири, що є достатнім для створення повноцінних торгівельних систем. На рисунках 4.1 і 4.2 представлено 2 лінії тренду: пропозиція низхідною лінією, попит - висхідною.

Можна звернути увагу, що на рисунку 4.1 поступове зниження цін відображене низхідною лінією “пропозиції”; цінові списи і западини також послідовно знижуються.

На рисунку 4.2 поступове підвищення цін відображене висхідною лінією “попиту”; цінові списи і западини також послідовно підвищуються. Складність полягає у виборі особливих точок, через які проходять ці прямі лінії (див. рисунок 4.2). Як правило, аналітик привносить неабияку частку

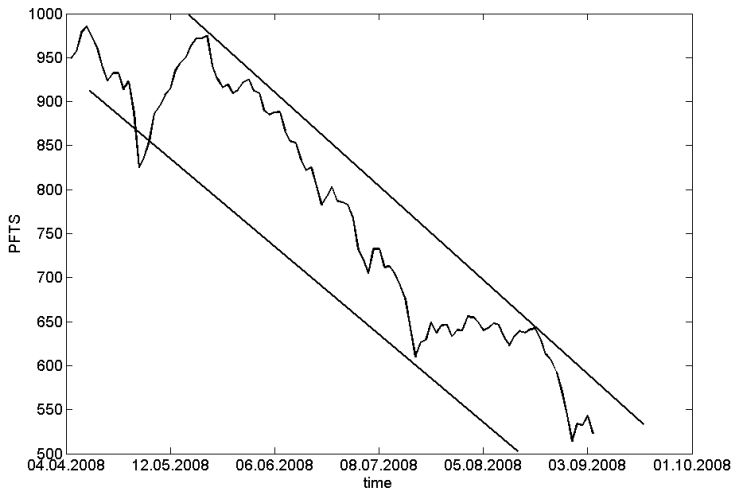


Рис. 4.1: Графік лінії тренду (низхідна)

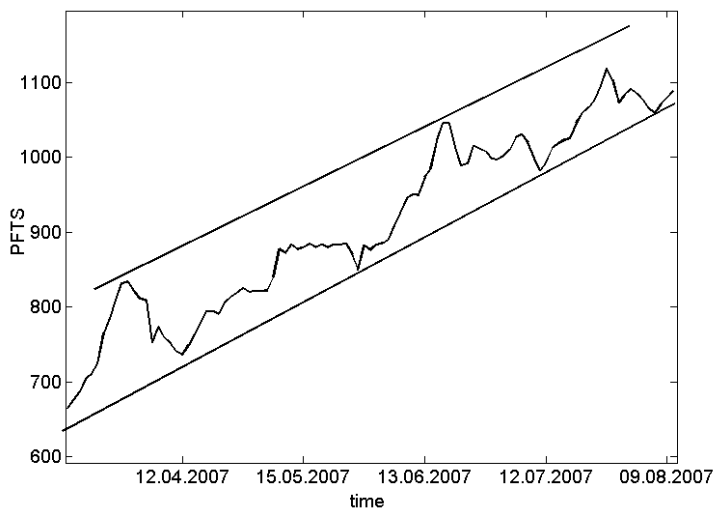


Рис. 4.2: Графік лінії тренду (висхідна)

суб'єктивізму в побудову ліній тренду. Так, рух цін на ринку прийнято розглядати у ретроспективі - від минулого до майбутнього, тому і дати на графіку відкладаються зліва направо.

Загальноприйнята процедура побудови множини ліній тренду, з яких одна неухильно вважається вірною, також грішить неточністю та певною неувагою до дрібних деталей та фрактальних властивостей часового ряду. Тим часом успішне застосування ліній тренду вимагає від аналітика підвищеної уважності і послідовності.

Слід зазначити, що найбільш популярні класичні методи, які використовуються у світі для аналізу і прогнозування фінансового стану підприємств, зв'язані із застосуванням економетричних моделей, кожен змінну яких оцінює певний статистичний індикатор з тією чи іншою точністю, який вимірює певну сторону господарської діяльності досліджуваного об'єкту. У ряді випадків вони дозволяють виявити конкретні кількісні взаємозв'язки економічних процесів, що протікають на підприємствах-емітентах, з індикативними показниками, доступними для інвесторів. Проте в сучасних умовах використання розроблених на основі західної практики моделей за очевидних причин ускладнене.

Моделі часового ряду, в яких математичне сподівання ряду описується аналітичною залежністю тільки від часу називаються трендовими моделями [24, 4, 5]. Більшість робіт [24, 4, 5] під трендовими моделями розуміють лінійні моделі тренду або ті, що до них зводяться.

Означення 4.1. *Трендовою моделлю* будемо називати ана-

22 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

літичну функцію виду

$$y = f(t, \vec{a}), \quad (4.1)$$

де t — час, заданий в певних одиницях (для даного дослідження - днях); $f(t, \vec{a})$ - аналітична функція, \vec{a} - параметри моделі, яка описує залежність досліджуваного ряду $\{y_i\}$ від часу t та забезпечує мінімальні відхилення фактичних та змодельованих рівнів

$$F(\vec{a}) = \|y_t - f(t, \vec{a})\| \quad (4.2)$$

на певному інтервалі часу $t, t_{first} < t < t_{last}$.

Зауважимо, що функція $f(t, \vec{a})$ може бути не тільки традиційною аналітичною (поліномом), але й гармонічною [25, 26, 27], складною нейронною мережею тощо.

Для побудови даних моделей здебільшого використовуюється метод найменших квадратів [5], або його модифікації для більш складних випадків.

До недоліків традиційних підходів варто віднести зведення до методу найменших квадратів, ігноруючи інші методи нелінійної чи стохастичної оптимізації, які у випадку багатьох екстремумів є більш ефективними. Також необхідно зазначити:

1) апроксимація окремо параметрів тренду та гармонічних складових. Наші роботи показують, що оптимізація (мінімізація функціоналу невязки) для усіх параметрів разом, крім того декількох гармонік одночасно, дозволяють де-що покращити результати як апроксимації, так і прогнозу.

Важливим розвитком таких моделей є методи, які враховують зміну частот;

2) Динамічні системи як трендові моделі. Розв'язок диференціальних рівнянь, заданих моделлю динамічної системи у вигляді залежності $y = f(t)$ може вважатись трендовою у випадку, коли цей розв'язок є функцією тільки від одного параметру - часу.

Для сучасних методів прогнозування трендові моделі відіграють функцію нормалізації ряду, що дозволяє збільшити повторюваність навчальної вибірки. Розглянемо застосування методу найменших квадратів, який часто використовується для оцінювання параметрів трендових моделей та для обчислення довірчих інтервалів прогнозних значень. Строго обґрунтування методу дано Марковим та Колмогоровим.

Для одновимірного випадку

$$y'(t) = \alpha + \beta \cdot x_t + u_t; L = \sum_{t=1}^T (y_t' - y_t)^2;$$

Величини параметрів α та β оцінюються за формулами:

$$\beta = \frac{T \sum_{i=1}^T x_i y_i - \sum_{i=1}^T x_i \sum_{i=1}^T y_i}{T \sum_{i=1}^T x_i^2 - \sum_{i=1}^T x_i \sum_{i=1}^T x_i};$$

$$\alpha = \frac{\sum_{i=1}^T x_i^2 \sum_{i=1}^T y_i - \sum_{i=1}^T x_i \sum_{i=1}^T x_i y_i}{T \sum_{i=1}^T x_i^2 - \sum_{i=1}^T x_i \sum_{i=1}^T x_i},$$

24 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

або

$$\bar{y} = \frac{1}{T} \sum_{i=1}^T y_i, \bar{x} = \frac{1}{T} \sum_{i=1}^T x_i; \beta = \frac{\sum_{i=1}^T (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^T (x_i - \bar{x})^2}; \alpha = \bar{y} - \beta \cdot \bar{x}.$$

Для багатовимірного випадку:

$$y = a_0 + a_1 \cdot x_1 + a_2 \cdot x_2 + a_3 \cdot x_3 + \dots + a_k \cdot x_k + u_k; Y = X \cdot A + U;$$

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} & x_{13} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & x_{23} & \dots & x_{2k} \\ 1 & x_{31} & x_{32} & x_{33} & \dots & x_{3k} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & x_{n1} & x_{n2} & x_{n3} & \dots & x_{nk} \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \dots \\ a_k \end{pmatrix} + \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \dots \\ u_n \end{pmatrix}.$$

Вектор невідомих параметрів $A = (a_0, a_1, a_2, \dots, a_k)$ оцінюється за формулою:

$$A = (X^T \cdot X)^{-1} \cdot X^T \cdot Y.$$

В підручниках [5, 14, 4] даний метод використовується для побудови багатофакторних та авторегресійних прогнозних моделей.

Передумови використання даного методу базуються на твердженнях

- незміщеності оцінок параметрів;
- відсутності мультиколінеарності векторів вхідних параметрів;

- відсутності автокорельованості залишків;

Для подолання таких труднощів рекомендуються процедури нормалізації і стандартизації даних. Метод найменших квадратів може бути також застосований для деяких видів нелінійних трендових моделей, які описані в підручнику [5]. Їх можна звести до лінійних за допомогою заміни змінних. Розглянемо деякі з них:

Варто зауважити, що оцінювання коефіцієнту детермінації та статистичної значущості результатів здійснюється не для вихідних змінних, а для лінеаризованих, що обов'язково необхідно враховувати. Обчислення відхилень, довірчих інтервалів тощо, для трендових моделей.

Для нормального розподілу $N(\mu, \sigma)$ з невідомими параметрами μ та σ процес відповідає наступній допоміжній статистиці [28] :

$$\frac{X_{n+1} - \bar{X}_n}{S_n \sqrt{1 + 1/n}} \sim T^{n-1}$$

Така проста комбінація можлива у випадку, коли вибірккові середня та дисперсія прогнозних значень розподілені за нормальним законом.

Оцінений розподіл для майбутнього значення X_{n+1} є розподілом прогнозного значення

$$\bar{X}_n + S_n \sqrt{1 + 1/n} \cdot T^{n-1}.$$

Ймовірність значення X_{n+1} знаходиться в заданому інтервалі, наступна:

$$\Pr \left(\bar{X}_n - T_a S_n \sqrt{1 + (1/n)} \leq X_{n+1} \leq \bar{X}_n + T_a S_n \sqrt{1 + (1/n)} \right) = p$$

26 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

Табл. 4.1: Нелінійні трендові моделі

Вид трендової моделі	Формула залежності	Лінеаризація
Лінійна	$y(t) = a_0 + a_1 t$	-
Поліном n-го степеня	$y(t) = a_0 + a_1 t + a_2 t^2 + \dots + a_n t^n$	$y(t) = a_0 + a_1 z_1 + \dots + a_n z_n,$ $z_i = t^i$
Експонента	$y(t) = a_0 e^{a_1 t}$	$\ln(y(t)) = \ln(a_0) + a_1 t$
Логарифмічна крива	$y(t) = a_0 + a_1 \ln(t)$	$y(t) = a_0 + a_1 t^*, t^* = \ln(t)$
Обернена логарифмічна крива	$y(t) = \frac{1}{a_0 + a_1 \ln(t)}$	$Y(t) = \frac{1}{y(t)},$ $t^* = \ln(t)$
Степенева	$y(t) = a_0 e^{a_1 t}$	$Y(t) = \ln(y(t)), t^* = \ln(t)$
Гіперболічна I	$y(t) = a_0 + \frac{a_1}{t}$	$t^* = \frac{1}{t}$
Гіперболічна II	$y(t) = \frac{1}{a_0 + a_1 t}$	$Y(t) = \frac{1}{y(t)}$
Гіперболічна III	$y(t) = \frac{t}{a_0 + a_1 t}$	$Y(t) = \frac{1}{y(t)},$ $a = a_0, b = a_1,$ $t^* = \frac{1}{t},$ $Y(t) = at^* + b$
Модифікована експонента	$y(t) = a + bc^t$	$c = \frac{(n-1) \sum_{t=0}^{n-1} y_t y_{t+1} - \sum_{t=1}^{n-1} y_t \sum_{t=1}^{n-1} y_{t+1}}{(n-1) \sum_{t=1}^{n-1} y_t^2 - \left(\sum_{t=1}^{n-1} y_t \right)^2},$ $b = \frac{n \sum_{t=1}^n c^t y_t - \sum_{t=1}^n c^t \sum_{t=1}^n y_t}{n \sum_{t=1}^n c^{2t} - \left(\sum_{t=1}^n c^t \right)^2},$ $a = \frac{\sum_{t=1}^n y_t - b \sum_{t=1}^n c^t}{n}$
крива Гомперця	$y(t) = a_0 a_1^{a_2^t}$	$\ln(y(t)) = \ln(a_0) + a_2^t \ln(a_1)$
Логістична	$y(t) = \frac{1}{a_0 + a_1 a_2^t}$	$Y(t) = \frac{1}{y(t)}, t^* = \frac{1}{t},$ $Y = a_0 + a_1 a_2^t$

де T_a - $100((1+p)/2)$ -на процентиль t -розподілу Стьюдента з $n - 1$ ступенями свободи. Тому величини

$$\bar{X}_n \pm T_a S_n \sqrt{1 + (1/n)}$$

є граничними точками $100\rho\%$ -го прогнозного інтервалу для значення X_{n+1} .

На нашу думку, до трендових моделей необхідно віднести також лінійно-гармонічну модель, наведену у монографії [25]. Розглянемо її.

$$y(t) = c_0 + c_1 t + \sum_{i=1}^n \left(a_i \cos \frac{2\pi}{T_i} t + b_i \sin \frac{2\pi}{T_i} t \right), \quad (4.3)$$

де $t = 1, 2, \dots, n$ - час, c_0, c_1, a_i, b_i, T_i - параметри моделі. Для оцінювання параметрів пропонується сканування діапазону можливих періодів T_i та оцінювання значень c_0, c_1, a_i, b_i параметрів багатofакторної моделі з факторами $(t, \cos \frac{2\pi}{T_i} t, \sin \frac{2\pi}{T_i} t)$ методом найменших квадратів. З множини перебраних періодів T_i вибирається той, який забезпечує мінімальне значення функціоналу нев'язки

$$Z(c_0, c_1, a_i, b_i) = \sum_{t=T_{min}}^{T_{max}} \left[Y(t) - c_0 - c_1 t - a_i \cos \frac{2\pi}{T_i} t + b_i \sin \frac{2\pi}{T_i} t \right]^2 \rightarrow \min. \quad (4.4)$$

Даний метод показав достатньо високу якість прогнозів для рядів урожайності зернових культур України. До недоліків даної моделі можна віднести наступні:

- повний перебір частот не є оптимальним, такий підхід не дає точно оцінити період коливань

- неможливість оптимізувати модель за параметром T_i
- можливі проблеми з визначенням параметрів за МНК (мультиколінеарність, гетероскедастичність тощо).

Удосконалення даного методу розглядається в розділі 9 даної роботи.

Відомим методом аналізу частотного спектру часового ряду є метод “Гусениця”, більш відомий в західних публікаціях як метод сингулярного спектрального аналізу (Singular Spectrum Analysis, SSA) [29, 30, 31]. Суть його полягає у конструюванні траєкторної матриці з лагових змінних часового ряду та аналізу головних компонент отриманої траєкторної матриці. Розглянемо суть даного методу.

Нехай задано часовий ряд $\{y_t\}$. Метод “Гусениця” складається з наступних кроків.

1. Нормалізація вхідного ряду

$$y_t^* = \frac{y_t - \bar{y}_t}{\sigma},$$

або

$$y_t^* = \frac{y_t - \bar{y}_t}{\sigma}.$$

2. Побудова траєкторної матриці

$$X = \begin{pmatrix} y_1^* & y_2^* & \cdots & y_m^* \\ y_2^* & y_3^* & \cdots & y_{m+1}^* \\ \cdots & \cdots & \cdots & \cdots \\ y_{N-m}^* & y_{N-m+1}^* & \cdots & y_N^* \end{pmatrix},$$

де m - розмірність вектору вкладення. Матриця X називається ганкелевою. 3. Побудова кореляційної матриці

$$C = \frac{1}{m} X^T X,$$

4. Обчислення власних векторів та власних значень матриці C . Нехай $v^{(1)}, v^{(2)}, \dots, v^{(m)}$ - власні вектори матриці C , а $\lambda_1, \lambda_2, \dots, \lambda_m$ - її власні значення. 5. Вибір мінімального базису власних векторів. З множини власних векторів пропонується вибрати підмножину, які описують тільки суттєву варіацію часового ряду. Для цього Упорядкуємо пари $(\lambda_i, v^{(i)})$ у порядку спадання власних значень. Нехай отримані номери власних векторів є наступними: k_1, k_2, \dots, k_m . Способи вибору підмножини власних векторів для аналізу часових рядів, детально обговорюється в [32]. Розглянемо найпростіший метод. Визначемо τ як кількість найбільших власних векторів, питома вага власних значень яких більша, наприклад, за 75% ваги усіх власних значень.

Запишемо матрицю V_x у вигляді отриманих власних векторів-стовбців:

$$V_x = \left(v^{(k_1)}, v^{(k_2)}, \dots, v^{(k_\tau)} \right) = \begin{pmatrix} v_1^{(k_1)} & v_1^{(k_2)} & \dots & v_1^{(k_\tau)} \\ v_2^{(k_1)} & v_2^{(k_2)} & \dots & v_2^{(k_\tau)} \\ \dots & \dots & \dots & \dots \\ v_m^{(k_1)} & v_m^{(k_2)} & \dots & v_m^{(k_\tau)} \end{pmatrix}$$

6. Відображення матриці X в новому базисі власних векторів:

$$U_x = V_x^T X = (U_1 : \dots : U_\tau).$$

30 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

7. Відновлення вихідної матриці по першим r головним компонентам:

$$\tilde{X} = \left(v^{(k_1)}, v^{(k_2)}, \dots, v^{(k_r)} \right) \begin{pmatrix} U_1^T \\ \dots \\ U_r^T \end{pmatrix}.$$

8. Діагональне усереднення (ганкелізація) матриці \tilde{X} : Оскільки відновлена матриця не буде ганкелевою (Елементи діагоналей, паралельні побічній діагоналі, не рівні між собою), необхідно здійснити зведення матриці \tilde{X} до такого виду. Елементи нової матриці матимуть значення:

$$| f_s = \begin{cases} \frac{1}{s} \sum_{i=1}^s \tilde{x}_{i,s-i+1}, 1 \leq s \leq \tau, \\ \frac{1}{\tau} \sum_{i=1}^{\tau} \tilde{x}_{i,s-i+1}, \tau \leq s \leq n, \\ \frac{1}{N-s+1} \sum_{i=1}^{N-s+1} \tilde{x}_{i+s-n,n-i+1}, \tau \leq s \leq N, \end{cases}$$

Оригінальний метод “Гусениця” був розроблений для виявлення та аналізу прихованих періодичностей в часових рядах, але не містив моделі часового ряду, яку можна використати для побудови прогнозів. У роботі [32] пропонується декілька підходів побудови прогнозних моделей на основі вище запропонованого сингулярного розкладу. Не всі вони можуть бути застосовані до реальних часових рядів, так як базуються на понятті строго детермінованої послідовності. Розглянемо підхід, який найбільш ефективно може використовуватись для аналізу часових рядів зі стохастичними складовими.

Класична економетрія включає багато робіт з прогнозування часових рядів [4, 33, 34, 35]. Головна ідея полягає в наступному: наблизити часовий ряд до лінійного вигляду за допомогою лінеаризації. Далі знаходимо лінійну регресію для передбачення майбутніх значень, $y_t = a + b \cdot y_{t-1} + u$, де a та b – регресійні параметри, u – залишки. Моделюючи відхилення на основі t -статистики, можна оцінити довірчі інтервали отриманого точкового прогнозу. Даний метод є більш грубий, але працює для детермінованих лінійних трендів.

4.1 Лабораторна робота № 1. Трендові моделі

Як відомо з теоретичної частини, трендовими прийнято називати моделі, в яких використовується залежність показника, що прогнозується, від часу.

$$y = f(t),$$

де t - параметр часу. Розглянемо процес побудови трендової моделі. Візьмемо, наприклад, одну з трендових моделей, яку можна звести до лінійної. Візьмемо поліном 3-го степеня $y(t) = a_3t^3 + a_2t^2 + a_1t + a_0$, де a_3, a_2, a_1, a_0 - параметри моделі, які необхідно знайти. Дану модель можна звести до лінійної багатofакторної, здійснивши заміни $x_3(t) = t^3, x_2(t) = t^2, x_1(t) = t$. Таким чином, можна використати метод найменших квадратів для оцінювання значень параметрів. Модель матиме вигляд $y(t) = a_3x_3(t) + a_2x_2(t) + a_1x_1(t) + a_0$.

Розглянемо практичну побудову такої моделі в середовищі табличного процесора Microsoft Excel. Для цього необхідно:

Здійснити розмітку листа на секцію дано, секцію таблиць даних та секцію графіків (див. рис. 4.3).

Секцію “Дано” зручно використовувати для введення та зміни параметрів моделі та параметрів розрахунку. До таких параметрів необхідно віднести:

початкове значення t_0 ; крок часу Δt ; Також секцію “Дано” знизу зручно доповнити розрахунковими результатами прогнозу моделі. Наприклад:

4.1. ЛАБОРАТОРНА РОБОТА № 1. ТРЕНДОВІ МОДЕЛІЗЗ

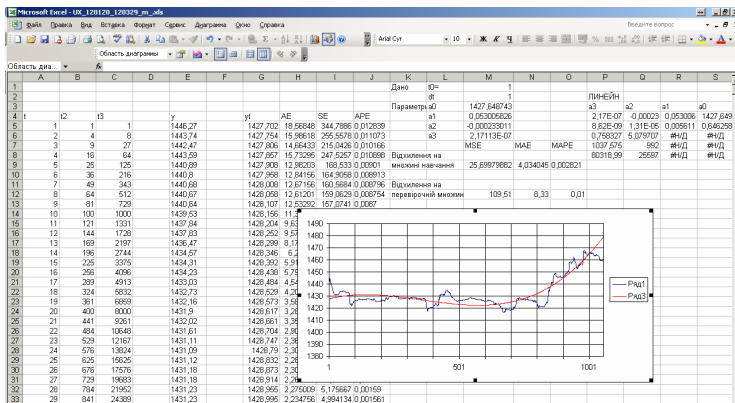


Рис. 4.3: Побудова трендової моделі в Excel

34 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

1. параметри моделі a_3, a_2, a_1, a_0
2. середнє лінійне відхилення (MAE) на навчальній вибірці
3. середньоквадратичне відхилення (MSE) на навчальній вибірці
4. дисперсія процесу
5. коефіцієнт детермінації
6. середнє лінійне відхилення (MAE) на перевіірчній вибірці
7. середньоквадратичне відхилення (MSE) на перевіірчній вибірці

Секція “Таблиця даних” обов’язково повинна містити наступні колонки:

1. час t ,
2. допоміжна колонка квадрат часу t^2 ,
3. допоміжна колонка куб часу t^3 ,
4. можливі інші допоміжні колонки (в залежності від моделі та виду лінеаризації).
5. показник, що прогнозується
6. можливі похідні від показника, що прогнозується (в залежності від моделі та виду лінеаризації).

4.1. ЛАБОРАТОРНА РОБОТА № 1. ТРЕНДОВІ МОДЕЛІЗ5

7. Прогнозний ряд $y(t)$,
8. модулі відхилень
9. квадрати відхилень.

В загальному вигляді для побудови нелінійної трендової моделі, необхідно здійснити лінеаризацію моделі, обчислити допоміжні перетворення незалежних та залежної змінної, додавши необхідні стовбці даних, та виконати оцінювання параметрів лінійної моделі, наприклад вбудованою в Excel функцією ЛИНЕЙН.

Функція ЛИНЕЙН приймає наступні параметри: Масив значень незалежних змінних (X) Масив значень залежної змінної (Y) Константа (1 або 0, ИСТИНА або ЛОЖЬ в залежності від того, чи обчислювати значення параметру а0 (істина, 1), чи присвоїти йому значення 0 (Ложь, 0).) Статистика (1 або 0, ИСТИНА або ЛОЖЬ в залежності від того, чи необхідно повертати рядки статистики оцінювання параметрів).

Розглянемо більш детально хід роботи при побудові поліноміальної трендової моделі.

1. Задати початкові параметри у секції “Дано” (початкове значення t_0 та крок Δt ;
2. Заповнити послідовність значень колонки t . Перше значення повинне містити значення t_0 із секції “Дано” (для прикладу з рис. 4.3, формула EXCEL для комірки початкового значення A5 матиме вигляд =M1), а кожен наступний більший попереднього на величину кроку

36 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

Δt (для комірки A6 формула матиме вигляд $=A5+\$M\2).
Усі наступні формули можна отримати копіюванням другої формули (значення комірки A6) у всі наступні комірки.

3. Додаткові комірки (значення квадратів і кубів для t у якості факторів багатофакторної моделі) можуть бути обчислені простим піднесенням до степені відповідного степеня значення першого стовбця. Тобто для комірки B5 з листа прикладу формула матиме вигляд $=A5^2$, а для комірки C5 - $=A5^3$. Важливо, що порядок стовбців додаткових факторів, які приймають участь у лінеаризованій моделі, зберігається при пошуку значень параметрів (виклик ЛИНЕЙН). Необхідно розташувати додаткові фактори саме в тому порядку, в якому вони знаходяться у лінеаризованій моделі. У випадку поліноміальної моделі 3-го степеня, порядок наступний: t, t^2, t^3 .
4. Заповнити вихідні значення ряду, що буде прогнозуватись (колонка E у прикладі на рис.4.3). Радимо залишати декілька значень для післяпрогновної перевірки навіть у випадках прогнозування майбутнього. Це дозволить отримати апріорні оцінки можливих похибок реального прогнозу.
5. Обчислити додаткові перетворення величини, що прогнозується, необхідні для побудови лінеаризованої моделі. У випадку нелінійних трендових моделей із таблиці 4.1, яка наведена у попередній главі, частіше такими

4.1. ЛАБОРАТОРНА РОБОТА № 1. ТРЕНДОВІ МОДЕЛІЗ

перетвореннями можуть бути логарифмування ($Y(t) = \ln(y(t))$) або обчислення обернених значень ($Y(t) = 1/y(t)$).

6. Оцінити коефіцієнти лінеаризованої моделі. Для цього необхідно оформити форматкуванням діапазон комірок, куди будуть повернені параметри (Розмірність масиву має бути $m+1$ рядків та 5 стовбців, де m - кількість факторів). Можна підписати відповідні стовбчики, наприклад як комірки P3-S3 у прикладі. Порядок коефіцієнтів починаючи з найстаршого a_m (множник при останньому факторі), останній коефіцієнт a_0 - вільний член. Після оформлення, виділіть комірки для результату функції ЛИНЕЙН, викличіть функцію ЛИНЕЙН, задайте параметри (значення залежної змінної Y , значення незалежних змінних - масив X , константа =1, статистика=1). Після заповнення параметрів, необхідно натиснути комбінацію клавіш CTRL+SHIFT+Enter. Після обчислення параметрів, для зручності їх значення можна записати у відповідні комірки у секції “Дано” за допомогою ссилки Excel.
7. Обчислити значення трендової функції $y(t)$ для оцінених параметрів та для усіх значень t . У вищенаведеному прикладі, формула для комірки G5 матиме вигляд: $=\$M\$3+\$M\$4*A5+\$M\$5*A5^2+\$M\$6*A5^3$. Дану формулу можна скопіювати у наступні комірки стовбця, обчисливши значення трендової моделі як для навчальної і перевіркової вибірки, так і екстрапольовавши дану функцію у майбутнє (продовження можна здійснити, скопіювавши значення останніх комірок для фа-

кторів та для теоретичної трендової моделі нижче у таблиці. Побудувати графіки отриманих прогнозів.

8. Обчислити абсолютні та квадратичні похибки для навчальної та прогновної вибірки. Для цього необхідно знайти модулі (наприклад, комірка H5 містить формулу $=ABS(G5-E5)$) та містить модуль відхилення фактичного та теоретичного значення прогнозованого показника. Для квадратичної похибки, наприклад, комірка I5 вищенаведеного прикладу, містить наступну формулу $=(G5-E5)^2$. Значення даних стовбців необхідно усереднити окремо для вибірки навчання (яка використовувалась функцією ЛИНЕЙН для оцінювання параметрів моделі методом найменших квадратів), та перевіркою вибірки (продовження часового ряду, дані якого не використовувались при побудові моделі. Порівнявши похибку на навчальній та перевірочній (прогнозній) вибірці, робимо висновок щодо адекватності отриманого прогнозу.

4.2 Лабораторна робота № 2. Методи оцінювання параметрів нелінійних трендових моделей

Згідно алгоритму, розглянутого у попередній лабораторній роботі, можна побудувати будь-яку трендову модель з таблиці 4.1, яка дозволяє зведення моделі до лінійного виду.

Далеко не усі трендові моделі просто можуть бути зведені до лінійного виду. Нами пропонується метод оцінювання параметрів будь-якої трендової моделі, який базується на нелінійній оптимізації.

Нехай нелінійна трендова модель задана у вигляді функції $y = f(t, a_0, a_1, \dots, a_m)$, де a_0, a_1, \dots, a_m - параметри моделі, t - час. Нехай задано навчальну вибірку, яка складається з точок $\{(t_i, y_i)\}$. Тоді функціонал невязки може бути записаний наступним чином:

$$L(a_0, a_1, \dots, a_m) = \sum_{i=1}^n |y_i - f(t_i, a_0, a_1, \dots, a_m)|^k, \quad (4.5)$$

де k - параметр чутливості до відхилень. При $k = 1$ мають місце абсолютні похибки, при $k = 2$ - квадратичні похибки, при $k \rightarrow \infty$ міра L відповідає максимальному відхиленню серед навчальної вибірки. Необхідно знайти значення вектору параметрів a_0, a_1, \dots, a_m , при яких вибрана Функція L набуває свого мінімального значення.

Використовуючи табличний процесор Excel та надстройку "Поиск решения", можна реалізувати пошук параметрів, при яких функціонал невязки буде приймати мінімальне значення. Для цього необхідно повторити наступні кроки (деякі з них можна використати з минулої лабораторної роботи, зробивши копію відповідного листа. В середовищі Excel 2003 для цього достатньо натиснути на ім'я листа правою кнопкою миші, виберіть опцію "Переместить/скопировать", поставивши галочку "Создать копию").

Хід роботи:

40 РОЗДІЛ 4. ТРЕНДОВІ МОДЕЛІ ПРОГНОЗУВАННЯ

1. Задати початкові параметри у секції “Дано” (початкове значення t_0 та крок Δt ;
2. Заповнити послідовність значень колонки t . Перше значення повинне містити значення t_0 із секції “Дано” (для прикладу з рис. 4.3, формула EXCEL для комірки початкового значення A5 матиме вигляд =M1), а кожен наступний більший попереднього на величину кроку Δt (для комірки A6 формула матиме вигляд =A5+\$M\$2). Усі наступні формули можна отримати копіюванням другої формули (значення комірки A6) у всі наступні комірки.
3. Заповнити вихідні значення ряду, що буде прогнозуватись (колонка E у прикладі на рис.4.3). Радимо залишати декілька значень для післяпрогновної перевірки навіть у випадках прогнозування майбутнього. Це дозволить отримати апріорні оцінки можливих похибок реального прогнозу.
4. Задати початкові наближення для параметрів моделі. В якості початкових наближень можна брати значення параметрів, знайдених з попередньої роботи за допомогою лінеаризації. У випадку коли лінеаризація неможлива, можна лінеаризувати спрощений варіант моделі, припустивши певні початкові значення параметрів, які не вдалося звести до лінійних залежностей. Зауважимо, що для методів локальної нелінійної оптимізації початкове значення грає значну роль. Тому необхідно продумати не одне, а певну множину для перебору по-

чаткових наближень для запобігання попаданню у локальний мінімум.

5. Обчислити значення трендової функції $y(t)$ для оцінених параметрів та для усіх значень t . У вищенаведеному прикладі, формула для комірки G5 матиме вигляд: $=M3+M4*A5+M5*A5^2+M6*A5^3$. Дану формулу можна скопіювати у наступні комірки стовбця, обчисливши значення трендової моделі як для навчальної і перевіркової вибірки, так і екстраполювавши дану функцію у майбутнє (продовження можна здійснити, скопіювавши значення останніх комірок для факторів та для теоретичної трендової моделі нижче у таблиці. Побудувати графіки отриманих прогнозів.
6. Обчислити абсолютні та квадратичні похибки для навчальної та прогнозової вибірки. Для цього необхідно знайти модулі (наприклад, комірка H5 містить формулу $=ABS(G5-E5)$) та містить модуль відхилення фактичного та теоретичного значення прогнозованого показника. Для квадратичної похибки, наприклад, комірка I5 вищенаведеного прикладу, містить наступну формулу $=(G5-E5)^2$). Значення даних стовбців необхідно усереднити окремо для вибірки навчання (яка використовувалась функцією ЛИНЕЙН для оцінювання параметрів моделі методом найменших квадратів), та перевіркової вибірки (продовження часового ряду, дані якого не використовувались при побудові моделі. Середнє значення похибки для вибірки навчання буде грати роль цільової функції, яку будемо мінімізувати.

7. Застосуємо функцію “Поиск решения”, яка доступна з меню “Сервис-Поиск решения”. (див. рис. 4.4.) У випадку, коли такої функції нема у меню, необхідно її встановити, скориставшись опцією меню “Сервис-Надстройки” і встановивши відповідну галочку навпроти надстройки “Поиск решения”. Для встановлення знадобиться інсталяційний диск Microsoft Office. Вказавши цільову комірку та комірки параметрів, застосуйте пошук та збережіть знайдене рішення.
8. У деяких нелінійних трендових моделей (наприклад тих, що мають гармонічні складові), може мати місце декілька локальних екстремумів. Для пошуку глобального, на жаль, засобами Excel ми вимушені перебрати початкові наближення з деякої множини. Тому на даному етапі здійсніть пошук для усіх вибраних початкових наближень та зафіксуйте знайдені параметри та значення цільової функції. Найкращим вважається вектор параметрів, при якому цільова функція приймає найменше з можливих значень.

Книгу Excel з прикладами, розглянутими в лабораторних, можна завантажити за веб-адресою
http://prognoz.ck.ua/predict_xls.rar

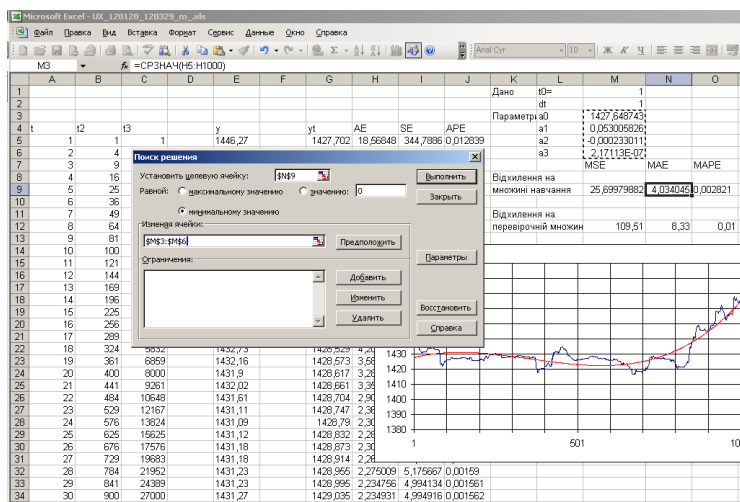


Рис. 4.4: Побудова нелінійної трендової моделі в Excel

Розділ 5

Авторегресійні методи прогнозування.

Означення 5.1. Процесом з авторегресією $AR(p)$ називається послідовність рівнів ряду $\{y_i\}$, яка задовольняє рівняння:

$$y_{n+1} = a_0 + a_1 y_n + a_2 y_{n-1} + \dots + a_{k-1} y_{k-2} + a_k y_{k-1} + e, \quad (5.1)$$

де $\{a_i\}$ — параметри авторегресії, k - порядок авторегресійної моделі, y_i - рівні ряду.

У монографіях [3, 36, 37, 38] пропонується метод знаходження аналітичного розв'язку авторегресійного рівняння 5.3 відносно параметру часу t , що дозволяє звести авторегресійну модель часового ряду до трендової. Дане перетворення

46 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

базується на пошуку часткових розв'язків авторегресійного рівняння виду $y_i^h(k) = A_i \cdot (\alpha_i)^k$. Проілюструємо даний метод на прикладі авторегресійної моделі другого порядку

$$y(k) = 2,1 + 0,8y(k-1) - 0,15y(k-2)$$

Сформуємо однорідне рівняння

$$\lambda^2 - 0,8 \cdot \lambda + 0,15 = 0.$$

Воно має два цілих розв'язки $\lambda_1 = 0,5$ та $\lambda_2 = 0,3$, тому часткові розв'язки авторегресійного рівняння мають вигляд: $y_1^h(k) = A_1 \cdot (0,5)^k$ та $y_2^h(k) = A_2 \cdot (0,3)^k$. Знаходимо частковий розв'язок для однорідних збурень:

$$y^p = 2,1 + 0,8y^p - 0,15y^p \Rightarrow y^p = 6,0$$

Об'єднуємо однорідний розв'язок з частковими:

$$y(k) = 6,0 + A_1 \cdot (0,5)^k + A_2 \cdot (0,3)^k,$$

де A_1, A_2 - довільні константи, які визначаються при заданих початкових умовах. Нехай задані початкові умови $y(0) = 1,0$; $y(1) = 2,0$, то

$$\{1,0 = 6,0 + A_1 + A_2 \cdot 2,0 = 6,0 + 0,5 \cdot A_1 + 0,3 \cdot A_2\}$$

звідки $A_1 = -12,5$; $A_2 = 7,5$. Остаточний розв'язок прийме вигляд:

$$y(k) = 6,0 - 12,5 \cdot (0,5)^k + 7,5 \cdot (0,3)^k.$$

Для рівнянь вищого порядку можна виконати аналогічні перетворення, але виникають труднощі, зв'язані з розв'язанням алгебраїчного рівняння високої степені. На нашу думку, даний метод не покращує прогностні можливості авторегресійних моделей у порівнянні з ітераційним обчисленням прогнозованих значень ряду, але розв'язок авторегресійного рівняння відносно невідомої функції $y(t)$ дозволяє провести класифікацію можливих режимів авторегресійних процесів [3, 11].

Також варто зауважити, що авторегресійні випадкові процеси у ряді робіт [5, 39] називаються марківськими, так як описують процеси з короткою пам'ятю, тобто лінійну залежність майбутньої величини від ряду попередніх. Оскільки під дискретними ланцюгами Маркова розуміють випадкові процеси з пам'яттю один крок, то марківськими процесами у роботах [5, 39] називаються авторегресійні моделі порядку 1.

На нашу думку, клас марківських процесів не обмежується лінійними авторегресійними, так як залежність може бути нелінійною та записуватись у вигляді алгоритму абстрактної машини або клітинного автомату [10].

Означення 5.2. *Процесом ковзного середнього $MA(q)$ називається послідовність рівнів ряду $\{y_i\}$, яка задовольняє рівняння:*

$$y_{n+1} = b_0 + b_1 e_n + b_2 e_{n-1} + \dots + b_{k-1} e_{k-2} + b_k e_{k-1}, \quad (5.2)$$

де $\{b_i\}$ - параметри авторегресії, k - порядок авторегресійної моделі, e_i - залишки.

48 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

Для моделювання та прогнозування частіше використовуються комбіновані $ARMA(p, q)$ -моделі.

Означення 5.3. *Процесом авторегресії - ковзного середнього $ARMA(p, q)$ називається послідовність рівнів ряду $\{y_i\}$, яка задовольняє рівняння:*

$$y_{n+1} = a_0 + a_1 y_n + a_2 y_{n-1} + \dots + a_{k-1} y_{k-2} + a_k y_{k-1} + b_0 + b_1 e_n + b_2 e_{n-1} + \dots + b_{k-1} e_{k-2} + b_k e_{k-1}, \quad (5.3)$$

де $\{a_i\}, \{b_i\}$ — параметри авторегресії, p - порядок авторегресійної складової моделі, q - порядок складової ковзного середнього, e_i - залишки.

Прогнозування на базі ARIMA-моделей. ARIMA-моделі охоплюють досить широкий спектр часових рядів, а незначні модифікації цих моделей дозволяють також досить точно описувати часові ряди з сезонністю. Говорячи про ARIMA-моделі, матимемо на увазі, що окремими випадками ARIMA є AR-, MA- і ARMA-моделі. Крім того, виходитимемо з того, що вже здійснено підбір моделі для аналізованого часового ряду, включаючи ідентифікацію вибраної моделі.

Прогнозуватимемо невідоме значення, x_{t+1} , $l > 1$ вважаючи, що x_t - останнє за часом спостереження аналізованого часового ряду, наявне в нашому розпорядженні і позначимо його через x_t^l . Відмітимо, що хоча x_t^l і x_{t-1}^{l+1} позначають прогноз одного і того ж невідомого значення x_{t+1} , але обчислюються вони по-різному, оскільки є вирішеннями різних завдань.

Ряд x_τ , що аналізується в рамках ARIMA (p, k, q) -моделі, представимо (при будь-якому $\tau > k$) у вигляді

$$\begin{aligned} (1 - \alpha_1 L - \dots - \alpha_p L^p) \cdot \sum_{j=0}^k (-1)^j C_k^j x_{\tau-1} &= \\ = \delta_\tau - \theta_1 \delta_{\tau-1} - \dots - \theta_q \delta_{\tau-q} \end{aligned} \quad (5.4)$$

де L - оператор зсуву функції часу на один часовий такт назад.

Зі співвідношення (5.4) можна виразити x_τ для будь-якого $\tau = t - q, \dots, t-1, t+1, \dots, t+1$. Отримуємо

$$\begin{aligned} x_\tau &= \left(\sum_{j=1}^p \alpha_j L^j \right) x_\tau + \left(1 - \sum_{j=1}^p \alpha_j L^j \right) \cdot \left(\sum_{i=0}^k (-1)^i C_k^i x_{\tau-i} \right) + \\ &+ \left(1 - \sum_{j=1}^q \theta_j L^j \right) \delta_\tau \end{aligned} \quad (5.5)$$

Правими частинами цих співвідношень є лінійні комбінації з $p + k$ попередніх (по відношенню до лівої частини) значень аналізованого процесу x_τ , доповнені лінійними комбінаціями поточного і q попередніх значень випадкових залишків δ . Причому коефіцієнти, за допомогою яких ці лінійні комбінації підраховуються, відомі, оскільки виражаються в термінах уже оцінених параметрів моделі. Цей факт і дає можливість використовувати співвідношення (5.5) для побудови прогнозних значень часового ряду на l тактів часу вперед. Теоретичну базу такого підходу до прогнозування забезпечує відомий результат, відповідно до якого найкращим (за середньоквадратичною похибкою) лінійним прогнозом у

50 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

момент часу t з випередженням l є умовне математичне сподівання випадкової величини x_{t+l} , обчислене за умови, що всі значення x_τ до моменту часу t . Цей результат є окремим випадком загальної теорії прогнозування [40, 41, 42].

Таким чином, визначається наступна процедура побудови прогнозу по відомій до моменту траєкторії часового ряду:

1) за формулами (5.5) обчислюються ретроспективні прогнози $x_{t-q-l}^l \dots x_{t-1}^l$ на базі попередніх значень часового ряду;

2) використовуючи формули (5.5) для $\tau > t$, підраховуються умовні математичні сподівання для обчислення прогнозних значень.

Метод експоненціального згладжування. Досить ефективним і надійним методом прогнозування є метод експоненціального згладжування [42, 5]. Основні переваги методу полягають у можливості оцінювання вагів початкової інформації, у простоті обчислювальних операцій, в гнучкості опису різних динамічних процесів. Метод експоненціального згладжування дає можливість отримати оцінку параметрів тренду, що характеризує не середній рівень процесу, а тенденцію, що склалася до моменту останнього спостереження. Після появи робіт Р. Брауна найбільше застосування метод знайшов для реалізації короткострокових і середньострокових прогнозів. Для методу експоненціального згладжування основним і найбільш важким моментом є вибір параметра згладжування α , початкових умов і степеня прогнозуючого полінома [5, 40].

Нехай початковий динамічний ряд описується функцією

$$y_1 = a_0 + a_1 t + \frac{a_2}{2} t^2 + \dots + \frac{a_p}{p!} t^p + \varepsilon_t \quad (5.6)$$

Метод експоненціального згладжування, що є узагальненням методу ковзного середнього, дозволяє побудувати такий опис процесу (5.6), при якому новим спостереженням додаються більші ваги в порівнянні зі старими, причому ваги спостережень спадають за експонентою. Вираз

$$S_t^{[k]}(y) = \alpha \sum_{i=0}^n (1 - \alpha)^i S_{t-1}^{[k-1]}(y)$$

називається експоненціальною середньою до n -го порядку для ряду y_t , де α - параметр згладжування.

У розрахунках для визначення експоненціальної середньої користуються рекурентною формулою [43]

$$S_t^{[k]}(y) = \alpha S_t^{[k-1]}(y) + (1 - \alpha) S_{t-1}^{[k]}(y)$$

Як було відмічено, важливу роль в методі експоненціального згладжування грає вибір оптимального параметра згладжування α , оскільки саме він визначає оцінки коефіцієнтів моделі, а отже, і результати прогнозу. Вибір параметра α доцільно пов'язувати з точністю прогнозу, тому для більш обґрунтованого вибору α можна використовувати процедуру узагальненого згладжування, яка дозволяє отримати співвідношення, що зв'язують дисперсію прогнозу і параметр згладжування [42]. Істотним для практичного використання є питання про вибір порядку прогнозуючого полінома,

52 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

що багато в чому визначає якість прогнозу. У роботі [42] показано, що перевищення другого порядку моделі не приводить до істотного збільшення точності прогнозу, але значно ускладнює процедуру розрахунку.

Відзначимо, що даний метод є одним з найбільш популярних і широко застосовується в практиці прогнозування. Враховуючи, що метод експоненціального згладжування є узагальненням методу найменших квадратів, можна сподіватися, що він удосконалюватиметься і далі як в теоретичному, так і в прикладному аспекті. Один з найбільш перспективних напрямів розвитку даного методу є метод різницевого прогнозування, в якому саме експоненціальне згладжування розглядається як окремий випадок [42].

Головний недолік цього методу полягає в тому, що він розглядає часові ряди ізольовано від інших явищ. Крім того, точність прогнозу помітно падає при довгостроковому прогнозуванні [42].

На увагу заслуговує метод сингулярного спектрального аналізу “Тусениця” (Singular Spectrum Analysis, далі - SSA) [30, 31, 29] та методи прогнозування, які базуються на ньому. Розглянемо суть методу SSA.

Як відмічено раніше, вузьким місцем усіх адаптивних методів, і методів експоненціального згладжування зокрема, є підбір відповідного до даного конкретного завдання параметра згладжування (адаптації) α . Навіть при оптимальному підборі параметра модель Брауна поступається в точності прогнозу ARIMA (0,1,1) -моделі.

Огляд літератури дозволив виявити ряд недоліків загальновідомих методів прогнозування і завдань, які необхідно

вирішити в даній області.

Недоліками методу найменших квадратів є використання процедури оцінки що передбачає обов'язкове задоволення цілої ряду передумов, невиконання яких може привести до значних помилок:

1) випадкові похибки повинні мати нульову середню, скінченні дисперсії і коваріації;

2) кожен вимір випадкової похибки характеризується нульовим середнім, не залежним від значень спостережуваних змінних;

3) дисперсії кожної випадкової похибки є постійними, їх величини незалежні від значень спостережуваних змінних (гомоскедастичність);

4) відсутність автокореляції похибок, тобто значення похибок різних спостережень незалежні один від одного;

5) випадкові похибки повинні відповідати нормальному закону розподілу;

6) значення ендогенної змінної x не містять похибок вимірювання і мають скінченні середні значення та дисперсії.

Вибір моделі у кожному конкретному випадку здійснюється за цілим рядом статистичних критеріїв, наприклад за дисперсією, кореляційним відношенням і ін. Класичний метод найменших квадратів передбачає рівноцінність початкової інформації в моделі. У реальній практиці майбутня поведінка процесу значно більшою мірою визначається пізніми спостереженнями, ніж ранніми. Ця властивість вказує на зменшення цінності ранньої (більш віддаленої у часі, старої) інформації.

Важливим моментом отримання прогнозу за допомогою

54 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

методу найменших квадратів є оцінка адекватності отриманого результату. Для цієї мети використовується цілий ряд статистичних характеристик:

- оцінка стандартної помилки;
- середня відносна помилка оцінки;
- середнє лінійне відхилення;
- кореляційне відношення для оцінки надійності моделі;
- оцінка достовірності вибраної моделі через значущість індексу кореляції за Z -критерієм Фішера;
- оцінка достовірності моделі за F -критерієм Фішера;
- перевірка на наявність автокореляцій за критерієм Дарбіна - Уотсона.

Недоліки методу найменших квадратів обумовлені жорсткою фіксацією тренду і надійний прогноз при цьому можна отримати на невеликий період упередження. Тут мова йде про обмежені можливості методів математичної статистики, теорії розпізнавання образів, теорії випадкових процесів, нечітких методів та ін., оскільки багато реальних процесів не можуть адекватно бути описані за допомогою традиційних статистичних моделей, оскільки є істотно нелінійними і мають або хаотичну, або квазіперіодичну, або змішану основу.

Недоліком методу адаптивного згладжування є те, що тільки при дуже довгих рядах можна отримати надійний прогноз на інтервал більший, ніж при звичайному експоненціальному згладжуванні. На жаль, для даного методу немає строгої процедури оцінки необхідної або достатньої довжини початкової інформації, для скінчених рядів немає конкретних умов оцінки точності прогнозу. Тому для скінчених рядів існує ризик отримати вельми наближений прогноз, тим

більше що в більшості випадків в реальній практиці зустрічаються ряди, що містять не більше 20-30 точок.

Проблеми з методом Бокса - Дженкінса (моделі авторегресії – ковзного середнього) зв'язані, перш за все, з неоднорідністю часових рядів і складністю практичної реалізації методу. В цілому результати застосування традиційних технологій оцінювання і прогнозування фінансового стану компаній, а також реальної вартості пакетів їх цінних паперів (акції, облігації), які вільно продаються і купуються на фондовому ринку, можна назвати обмеженими. Обмеженість цих методів полягає у їх залежності від початкових умов і відсутності гнучкості. Жорсткі статистичні припущення про властивості часових рядів обмежують можливості методів математичної статистики, теорії розпізнавання образів, теорії випадкових процесів, нечітких методів і ін. Вони не здатні враховувати те, що відносна значущість окремих показників фінансової звітності та чинників, що їх визначають, на практиці змінюються з часом, іноді дуже різко і непередбачувано. Окрім цього традиційні підходи характеризуються обмеженою інформативністю, оскільки призначені для опису якісних чинників або закономірностей в кількісних термінах. Таким чином, на зміну традиційним технологіям повинні прийти нові підходи, ефективніші в умовах структурної нестабільності світової економіки [44]. Останнім часом усе більше уваги приділяється дослідженню і прогнозуванню фінансових часових рядів з використанням теорії динамічних систем, теорії хаосу. Це досить нова область, яка є популярним розділом математичних методів економіки, що активно розвивається [45, 46, 43, 47, 48].

5.1 Лабораторна робота № 3. Моделі авторегресії-ковзного середнього

Для побудови прогнозів на основі авторегресійних моделей реалізовані процедури у багатьох статистичних пакетах (Statistica, Matlab та багато інших). В даній лабораторній роботі обрано Excel як найбільш наочний засіб, який дозволяє студентам, не знайомих з суттю авторегресійних моделей, більш глибоко усвідомити їх суть і можливості при побудові прогнозів.

Пропонуємо розробити лист Excel наступним чином (див. рис.5.1). В секції “Дано” необхідно задавати наступні параметри:

1. значення початкового моменту навчальної вибірки t_0 . При зміні значення цього параметра буде змінюватись фрагмент вибірки для побудови прогновної моделі.
2. значення кроку дискретизації Δt . Даний параметр містить ціле число - крок дискретизації в одиницях інтервалу дискретизації вхідного ряду. Задає розрідження вхідного ряду.

Секцію “Дано” доповнимо наступними розрахунковими параметрами:

1. параметри моделі a_0, a_1, \dots, a_m ;
2. середнє лінійне відхилення (MAE) на навчальній вибірці

3. середньоквадратичне відхилення (MSE) на навчальній вибірці
4. дисперсія процесу
5. коефіцієнт детермінації
6. середнє лінійне відхилення (MAE) на перевірочній вибірці
7. середньоквадратичне відхилення (MSE) на перевірочній вибірці

The screenshot displays an Excel spreadsheet with a data table starting from row 2 and column A. The columns are labeled A through S. The data includes time-series observations for variables X, Y, and Z, as well as calculated regression coefficients (a1 to a6) and various statistical metrics such as MAE, MSE, and R-squared for different parts of the dataset.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
2	2010/2/4	103100	1446,27															
3	2010/2/4	103200	1445,74															
4	2010/2/4	103300	1442,47															
5	2010/2/4	103400	1443,59															
6	2010/2/4	103500	1440,09															
7	2010/2/4	103600	1440,09															
8	2010/2/4	103700	1440,09															
9	2010/2/4	103800	1440,67															
10	2010/2/4	103900	1440,64															
11	2010/2/4	104000	1439,53															
12	2010/2/4	104100	1437,04															
13	2010/2/4	104200	1437,03															
14	2010/2/4	104300	1436,47															
15	2010/2/4	104400	1434,57															
16	2010/2/4	104500	1434,23															
17	2010/2/4	104600	1434,23															
18	2010/2/4	104700	1433,03															
19	2010/2/4	104800	1432,73															
20	2010/2/4	104900	1432,16															
21	2010/2/4	105000	1431,9															
22	2010/2/4	105100	1432,02															
23	2010/2/4	105200	1431,61															
24	2010/2/4	105300	1431,11															
25	2010/2/4	105400	1431,09															
26	2010/2/4	105500	1431,12															
27	2010/2/4	105600	1431,18															
28	2010/2/4	105700	1431,23															
29	2010/2/4	105800	1431,27															
30	2010/2/4	105900	1431,29															
31	2010/2/4	106000	1431,29															
32	2010/2/4	106100	1431,27															
33	2010/2/4	106200	1431,27															
34	2010/2/4	106300	1431,25															
35	2010/2/4	106400	1431,27															
36	2010/2/4	106500	1431,26															
37	2010/2/4	106600	1431,26															
38	2010/2/4	106700	1431,27															

Рис. 5.1: Побудова авторегресійної моделі в Excel

Секція “Таблиця даних” пропонується розділити на 2 частини. Перша буде містити вихідні дані (дати та значення

58 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

показника, що прогнозується), а друга - формується з першої у вигляді векторів лагових змінних із заданими початковим значенням t та кроком Δt :

1. час t ,
2. лаговий вектор $x_t = (y_{t-(m-1)\Delta t}, \dots, y_{t-\Delta t})$, що відповідає моменту часу t . Значення y_t є останньою координатою лагового вектора
3. Наступне значення, що відповідає моменту часу $t + \Delta t$. Використовується як фактичні значення відгуку авторегресійної моделі при заданому вхідному векторі.
4. Прогнозний ряд $y(x_t)$,
5. модулі відхилень
6. квадрати відхилень.

Розглянемо приклад листу Ексел для побудови авторегресійної моделі. Стовбець часу t задає послідовність індексів, що відповідають часу поточного лагового вектору. Тому значення t_0 в секції “Дано” повинне бути задане з відступом, щоб даних вистачило для першого вектору, враховуючи заданий крок Δt . Перше значення стобця часу необхідно взяти рівним t_0 з секції “Дано”. На листі, наведеного як приклад, формула для комірки D4, матиме вигляд =Q1. Наступні комірки, починаючи з D5, заповнюються значеннями, що збільшуються з кроком 1, це дозволить розширити об’єм навчальної вибірки, в порівнянні з розрідженням. Для комірки D5

формула матиме вигляд $y_t = D4 + 1$. Таким чином, компоненти лагового вектора будуть формуватись зі значень вихідного ряду через інтервал Δt , а між лаговими векторами інтервал буде рівний одному кроку вихідної дискретизації.

Вибір даних з заданим кроком може бути реалізований через функцію ІНДЕКС, яка повертає значення масиву з заданими координатами. Вибравши як орієнтир значення колонки t , можна побудувати такі лагові вектори, а також сформовані величини лагів над таблицею (у прикладі - комірки E2-K2). Значення даних комірок розраховуються таким чином: В комірку J2, що відповідає останній координаті лагового вектору, записується 0. У комірку I2, яка знаходиться зліва від J2, записується формула $=J2+Q2$, при якій минула величина лагу збільшується на Δt . Усі формули зліва від розглянутих отримуються копіюванням формули в комірку I2 у комірки E2... H2. Для значення відгуку системи (комірка K2) формула матиме вигляд $=J2-Q2$. Таким чином можна побудувати лагову модель не тільки для розмірності лагового вектору 6, але і для будь-якого. При цьому необхідно правильно спланувати розміщення лагових векторів на листу. При зміні значення Δt , усі компоненти перерахуються.

Далі необхідно сформувати значення лагових векторів у таблиці. Наприклад, для комірки J4, яка містить останню координату першого лагового вектору, формула матиме вигляд $=\text{ІНДЕКС}(\$C\$2:\$C\$13510;\$D4-J2;1)$. Формули для попередніх координат лагових векторів можна отримати, копіюванням комірки J4 в усі комірки E4... K4. Формула для майбутнього значення (відгуку системи, наприклад для комірки

60 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

К4) також може мати аналогічний вигляд, так як абсолютні та відносні посилки налаштовані на можливість копіювання як у вертикальному, так і в горизонтальному напрямі. Усі інші формули таблиці заповнені аналогічно. Після заповнення першого рядку, формули інших рядків можуть бути скопійовані з першого.

Після заповнення таблиці даних до моменту початку фактичного прогнозу, можна здійснити оцінювання параметрів моделі. За аналогією з попередньої роботи, можна запропонувати два методи оцінювання параметрів: метод найменших квадратів, реалізованих функцією ЛИНЕЙН та методи нелінійної оптимізації, реалізовані надстройкою “Поиск решения”. Перевагою методів нелінійної оптимізації можна назвати можливість використання метрики абсолютних відхилень, на відміну від квадратичної метрики у МНК. Після оцінювання параметрів, знайдені оптимальні значення повинні бути записані в секцію “Параметри” (комірки Q3-Q9 на листі прикладу). Використовуючи їх значення, необхідно побудувати ряд теоретичних $y(x_t)$, продовження яких фактично і будуть прогнозами.

Після настання моменту прогнозу, подальше використання минулої формули для формування лагового вектору стає неможливим, так як недоступні нові дані для побудови нових лагових векторів. Тому лагові вектори будуємо, використовуючи обчислені прогнози (див. рис.5.2). Таким чином можна реалізувати багатокроковий прогноз. Формула для лагових векторів має брати дані не з колонки реальних значень (колонка С в прикладі), а з колонки прогнозів (колонка К). Зауважимо, що значення колонки К зміщені відносно С

1	Дата	Время	Загр															Дано	т ₀
2	20120124	103100	1446,37																
3	20120124	103200	1443,74																
992	20120126	125900	1465,05	994	1465,64	1465,63	1465,05	1463,66	1463,45	1463,73	1463,87	1463,72	0,091916	0,009587					
993	20120126	130000	1463,66	995	1465,63	1465,05	1463,66	1463,45	1463,73	1463,87	1463,87	1463,87	0,002009	0,04E-06					
994	20120126	130100	1463,45	996	1465,05	1463,66	1463,45	1463,73	1463,87	1463,87	1464,52	1463,89	0,061225	0,437219					
995	20120126	130200	1463,73	997	1463,66	1463,45	1463,73	1463,87	1463,87	1464,52	1465,25	1464,722	0,523003	0,27895					
996	20120126	130300	1463,87	998	1463,45	1463,73	1463,87	1463,87	1464,52	1465,25	1465,28	1465,536	0,256833	0,065451					
997	20120126	130400	1463,87	999	1463,73	1463,87	1463,87	1464,52	1465,25	1465,28	1465,21	1465,384	0,174366	0,030403					
998	20120126	130500	1464,52	1000	1463,87	1463,87	1464,52	1465,25	1465,28	1465,21	1465,31	1466,17	0,0554	0,003069					
999	20120126	130600	1465,25	1001	1463,87	1464,52	1465,25	1465,28	1465,21	1465,31	1466,17	1465,408	0,762202	0,590952					
1000	20120126	130700	1465,28	1002	1464,52	1465,25	1465,28	1465,21	1465,31	1466,17	1468,16	1466,45	0,256833	0,003883					
1001	20120126	130800	1465,21	1003	1464,722	1465,536	1465,384	1465,256	1465,408	ИНДЕКС(\$L\$4:\$L\$1231;\$D1001-\$J\$2-\$Q\$1;1)									
1002	20120126	130900	1465,31	1004	1465,536	1465,384	1465,256	1465,408	1466,45	1466,775	1466,11	1466,965	0,065481	0,731848					
1003	20120126	131000	1466,17	1005	1465,384	1465,256	1465,408	1466,45	1466,775	1466,965	1466,12	1467,079	0,959252	0,320164					
1004	20120126	131100	1466,16	1006	1465,256	1465,408	1466,45	1466,775	1466,965	1467,079	1465,38	1467,205	1,816099	3,298216					
1005	20120126	131200	1466,05	1007	1465,408	1466,45	1466,775	1466,965	1467,079	1467,206	1465,43	1467,321	1,890741	3,574901					
1006	20120126	131300	1466,11	1008	1466,45	1466,775	1466,965	1467,079	1467,206	1467,321	1465,56	1467,406	1,845996	3,4077					
1007	20120126	131400	1466,12	1009	1465,775	1466,965	1467,079	1467,206	1467,321	1467,406	1465,43	1467,475	2,045196	4,103626					
1008	20120126	131500	1465,39	1010	1465,965	1467,079	1467,206	1467,321	1467,406	1467,475	1465,53	1467,536	2,030065	4,024698					
1009	20120126	131600	1465,43	1011	1467,079	1467,206	1467,321	1467,406	1467,475	1467,536	1465,53	1467,595	2,062656	4,265376					

Рис. 5.2: Побудова прогнозу за авторегресійною моделлю в Excel

на t_0 , кроків. Тобто, якщо використати вищенаведений приклад, перший рядок прогнозного ряду у таблиці відповідає 6-му значенню вихідного ряду. Таким чином, прогнозне значення 1001-го рядку таблиці лагових змінних, яке відповідає початку прогнозу, відповідає 1007-му значенню вихідного ряду. Якщо змінити формулу утворення лагових змінних, щоб дані брались з рядку прогнозу, індекси мають бути зменшені на величину t_0 . Формулу для відтворення реальних майбутніх значень (колонка K), залишимо без зміни. Вона буде використовуватись при обчисленні похибок прогнозу. А комірки лагового вектору (починаючи з E1001-J1001) змінимо. Наприклад, нова формула для комірки J1001 матиме вигляд: =ИНДЕКС(\$L\$4:\$L\$1231;\$D1001-\$J\$2-\$Q\$1;1). Формули для інших координат та продовження лагових ве-

62 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

кторів у часі можна отримати, скопіювавши вміст комірки J1001. Формули для обчислення теоретичних (прогнозних) значень та похибок залишаться без зміни.

Для обчислення середніх похибок достатньо обчислити середні значення відповідних колонок абсолютних чи квадратичних похибок для вибірки навчання та перевірконої (прогнозної) вибірки. Як показують дослідження, даний метод може бути застосований тільки для короткострокового прогнозування. У цьому можна переконавшись, здійснивши декілька прогнозних експериментів.

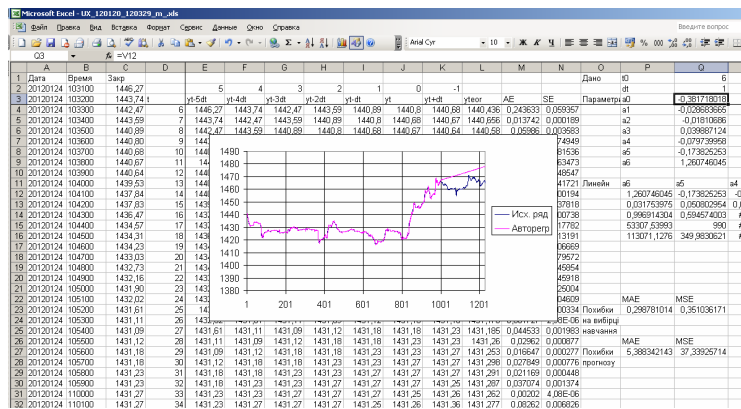


Рис. 5.3: Результат прогнозування за авторегресійною моделлю в Excel

Після виконання усіх вищезазначених розрахунків, мо-

жна зробити копію отриманого листа та розрахувати прогнози для інших інтервалів дискретизації Δt . Але при цьому доведеться змінити початок фрагменту t_0 та, можливо, момент початку прогнозу (рядок, в якому відбувається заміна усіх наступних формул побудови лагових векторів, використовуючи прогнозні значення).

Як відомо з теоретичної частини, прогнозні можливості авторегресійних моделей обмежені декількома режимами. Такі режими більш адекватно можуть описувати процеси, стаціонарні у статистичному сенсі (зі сталим математичним сподіванням та дисперсією). Одним зі способів переходу до стаціонарного ряду є перехід до різниць сусідніх значень. Такі моделі називаються авторегресійними з першим порядком інтегрованості. Для побудови таких моделей можна застосувати вищезазначену технологію не до вихідного ряду, а до його перших різниць $r_i = y_i - y_{i-1}$, $i = 1..N$. Після прогнозування ряду різниць, пропонується зворотне перетворення від ряду прогнозних різниць до прогнозних значень вихідного ряду, обчисленням визначеного інтегралу $y_t = \int_{t_N}^t r(t)dt$. В дискретному випадку це може бути звичайна сума спрогнозованих різниць.

Розглянемо приклад побудови такої прогнозної моделі в Excel (див. рис.5.4). Для цього додамо параметр Δt_{integr} для кроку обчислення різниць, а значення Δt задамо 1.

Для формування ряду різниць і виконання оберненого перетворення, необхідно додати ще 3 стовбця після стовбця даних: стовбець індексів розріджених даних, стовбець розріджених даних та стовбець прогнозованих вихідних даних (для процедури відновлення прогнозу різниць, див. рис.5.5).

64 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

Прогноз	MSE	...
1	0.0000	1443.74
2	0.0000	1442.47
3	0.0000	1443.99
4	0.0000	1440.89
5	0.0000	1440.89
6	0.0000	1440.69
7	0.0000	1440.67
8	0.0000	1440.64
9	0.0000	1439.53
10	0.0000	1437.84
11	0.0000	1437.89
12	0.0000	1436.47
13	0.0000	1434.97
14	0.0000	1434.31
15	0.0000	1434.23
16	0.0000	1433.03
17	0.0000	1432.73
18	0.0000	1432.18
19	0.0000	1431.90
20	0.0000	1431.90
21	0.0000	1431.23
22	0.0000	1431.61
23	0.0000	1431.11
24	0.0000	1431.09
25	0.0000	1431.12
26	0.0000	1431.18
27	0.0000	1431.18
28	0.0000	1431.23
29	0.0000	1431.23
30	0.0000	1431.23

Рис. 5.4: Прогнозування за авторегресійною моделлю в Excel з 1-м порядком інтегрованості

Необхідно заповнити стовбець індексів наступним чином. Перше значення (комірка D2 містить значення 1). Наступна комірка D3 містить формулу $=D2+\$U\3 . Усі комірки, що знаходяться нижче D3, заповнюються копіюванням вмісту (формули) з комірки D3.

Різниці заповнюються, використовуючи функцію ИНДЕКС. Наприклад, комірка E3 містить формулу $=\text{ИНДЕКС}(\$C\$2:\$C\$13510;D3;1)-\text{ИНДЕКС}(\$C\$2:\$C\$13510;D2;1)$.

Усі комірки, що знаходяться нижче E3, заповнюються копією формули з E3.

Після цього необхідно замінити формули для формування лагових векторів. Формула для комірки I4 має вигляд: $=\text{ИНДЕКС}(\$E\$3:\$E\$13510;\$H4-I\$2;1)$. Усі комірки діапазо-

ну I5:01000 можна замінити, скопіювавши формулу з комірки I4 у ці комірки. Формули на прогностичній частині можна не змінювати. Після зміни, авторегресійна модель буде прогнозувати ряд різниць першого порядку. Тепер необхідно перейти від ряду спрогнозованих різниць до значень вихідного ряду. Для цього у комірку F1000 записується значення останнього відомого значення перед прогнозом (формула має вигляд =C999) Наступне значення цього стовбчика (комірка F1001) обчислюється як сума попереднього значення та спрогнозованої різниці (для комірки F1001 формула має вигляд =F1000+P1001). Кожне наступне значення знаходиться аналогічно, тому комірки, що знаходяться нижче комірки F1001, можна скопіювати з F1001. Колонка відхилень обчислюється аналогічно розглянутим в попередніх роботах.

В	С	Д	Е	Ф	Г	Н	І	К	Л	М	О	Р	Q	R	S
1	Время	Знач	D (Integer)	Разность	Упрогн	MSE									Дано
2	103100	1446,27	1												
3	103200	1443,74	-2	-2,53											
997	130300	1463,97	996	0,14											
998	130400	1463,97	996	0,00											
999	130500	1464,52	997	0,65											
999	130600	1465,25	998	0,73											
1000	130700	1465,25	999	0,00	1465,25	0,00	1001	0,16	0,73	0,05	-0,07	0,16	0,06	0,002202	0,006936
1001	130800	1465,21	1000	-0,07	=F1000+P1001		1002	0,1539	0,267	0,1	0,0371	0,082	0,28	-0,11	0,110276
1002	130900	1465,31	1001	0,10	1465,44	0,02	1003	0,267	0,1005	0,037	0,0821	0,282	0,11	0,06	0,079663
1003	131000	1466,17	1002	0,86	1465,49	0,46	1004	0,1038	0,0371	0,082	0,2822	0,11	0,08	0,01	0,054273
1004	131100	1466,16	1003	-0,01	1465,56	0,38	1005	0,0371	0,0821	0,282	0,1103	0,08	0,05	-0,73	0,052056
1005	131200	1466,05	1004	-0,11	1465,59	0,21	1006	0,0821	0,2822	0,11	0,0797	0,054	0,05	0,04	0,047115
1006	131300	1466,11	1005	0,06	1465,64	0,22	1007	0,2822	0,1103	0,08	0,0543	0,052	0,05	0,13	0,046894
1006	131400	1466,12	1006	0,01	1465,68	0,20	1008	0,1103	0,0797	0,054	0,0521	0,047	0,05	-0,13	0,039634

Рис. 5.5: Результат прогнозування за авторегресійною моделлю з 1-м порядком інтегрованості

Оцінювання параметрів АRІМА-моделі та прогнозування часового ряду можна виконати за допомогою Garch Toolbox,

66 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

яка включена в Econometrics Toolbox системи Matlab. Розглянемо та прокоментуємо просту matlab-функцію для прогнозування за допомогою Garch Toolbox:

```
function garchpred_try2(infile, outfile);

%infile='GSPC_close.txt';
%outfile='GSPC_closeArma.txt';
y=dlmread(infile); % завантаження ряду
%y=y(2000:1:2400); %вибір фрагменту
n=length(y); % визначення довжини
% задаємо параметри моделі ARMA(R,M)+GARCH(P,Q)
Spec=garchset('R',10,'M',2,'C',0,'P',1,'Q',1);
%здійснюємо оцінювання параметрів моделі
[coeff,errors,LLF,innovations,sigmas]=garchfit(Spec,y);
% виконуємо прогнозування
[sigma,mean]=garchpred(coeff,y,5000);
% визначаємо коефіцієнти авторегресійної моделі
arcoeffs=coeff.AR;
% зберігаємо усі змінні робочої області
outfile_data=strrep(outfile,'.txt','.mat');
save(outfile_data)
% зберігаємо ряд середніх значень прогнозу
outfile_mean=strrep(outfile,'.txt','_mean.txt');
dlmwrite(outfile_mean,[y;mean]);
% зберігаємо ряд верхньої межі прогнозу
outfile_max=strrep(outfile,'.txt','_max.txt');
dlmwrite(outfile_max,[y;mean+sigma]);
% зберігаємо ряд нижньої межі прогнозу
```

```
outfile_min=strrep(outfile, '.txt', '_min.txt');
dlmwrite(outfile_min, [y; meany - sigmay]);
% будемо графік прогнозного ряду та довірчих інтервалів
figure;
plot([y; meany], 'r');
hold on;
plot([y; meany + sigmay]);
plot([y; meany - sigmay]);
```

В даній лабораторній роботі розглядається можливість табличного процесору Excel та системи комп'ютерної математики matlab для прогнозування на основі авторегресійних моделей. Для першого знайомства ми радимо почати з варіанту в таблицях Excel, хоча він, на відміну від програми Matlab, не дозволяє гнучко змінювати усі параметри. Книгу Excel з прикладами, розглянутими в лабораторних, можна завантажити за веб-адресою

http://prognoz.ck.ua/predict_xls.rar

68 РОЗДІЛ 5. АВТОРЕГРЕСІЙНІ МЕТОДИ ПРОГНОЗУВАННЯ.

Розділ 6

Використання нелінійних функцій, нейронних мереж при побудові прогностичних моделей.

Багато реальних процесів, у тому числі і показники ринку цінних паперів, не можуть адекватно бути описані за допомогою традиційних статистичних моделей, оскільки по суті є істотно нелінійними і мають або хаотичну, або квазіперіоди-

чну, або змішану (стохастика + хаос-динаміка+детермінізм) основу [49, 47, 48]. У даній ситуації адекватним апаратом для вирішення завдань аналізу і прогнозування ринку цінних паперів можуть служити штучні нейронні мережі, що реалізуються на базі концепцій штучного інтелекту. Програмно-математичній основою цих методів є нейронні мережі, що самоорганізуються [50, 51, 52, 53, 54, 55].

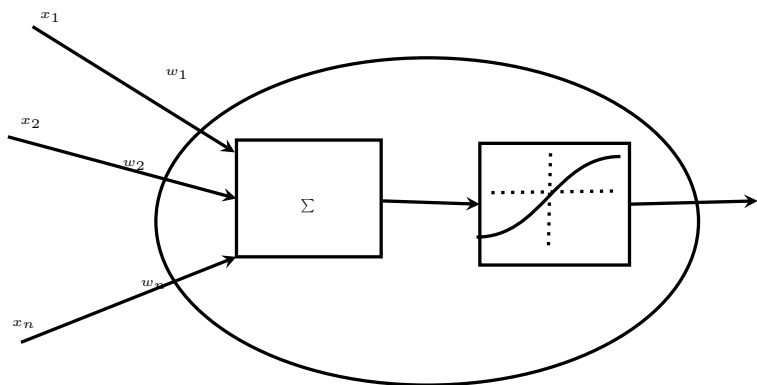


Рис. 6.1: Математична модель нейрона

Нейронна мережа - це сукупність нейронних елементів і зв'язків між ними 6.2. Основний елемент нейронної мережі - це формальний нейрон 6.1, що здійснює операцію нелінійного перетворення суми добутків вхідних сигналів на вагові

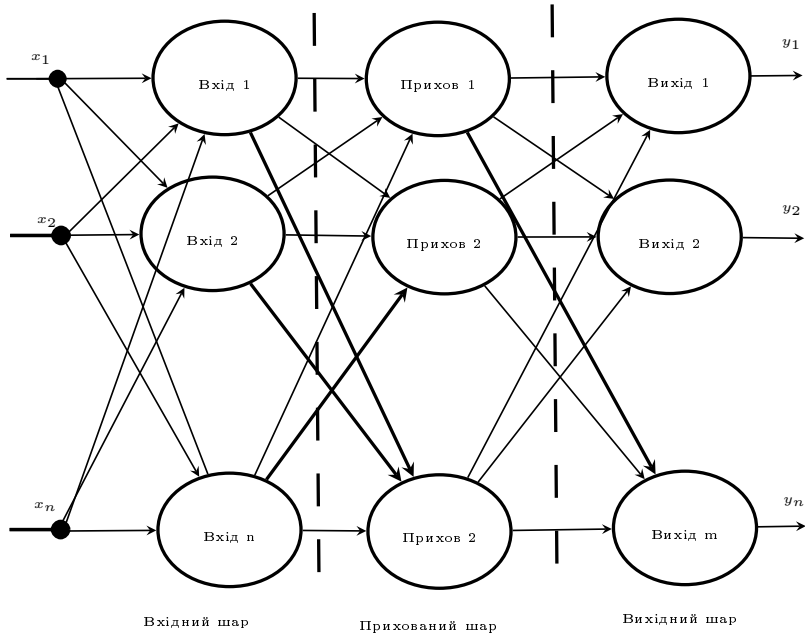


Рис. 6.2: Приклад нейронної мережі - багатошаровий перцептрон

коефіцієнти

$$Y = F \left(\sum_{i=1}^n w_i \cdot x_i \right) \cdot F(WX), \quad (6.1)$$

де $X = (x_1, x_2, \dots, x_n)^T$ – вектор вхідного сигналу; $W = (w_1, w_2, \dots, w_n)$ – ваговий вектор; F – оператор нелінійного перетворення.

В даний час методи штучних нейронних мереж вже довели свою високу ефективність в області економіки і фінансів. Штучні нейронні мережі є апаратом з області нейрокомп'ютингу (neural computing), області обчислювальних технологій, що швидко розвивається останнім часом. Обчислювальні операції в таких мережах виконуються великим числом порівняно простих процесорних елементів (processing element). Структура мережі (network) тотожна математично певній структурі обчислювальної системи, в якій всі операції виконуються в певних вузлах, а потік інформації відображається направленими ребрами графа. Кожен вузол (нейрон) мережі представлений процесорним елементом, нейронною клітиною, яка спільно з багатьма іншими процесорними елементами утворює нейронну обчислювальну мережу. Аналогом такого вузла у фізіологічній нервовій системі є нервова клітина мозку [44]. У загальному випадку штучна нейронна мережа є адаптивною нелінійною динамічною системою. За допомогою рівноважних станів такої мережі можна вирішувати математичні або обчислювальні задачі, такі як апроксимація функцій, класифікація, кластеризація, прогнозування та багато інших.

Існує два класи нейронних мереж:

- мережі, що навчаються з учителем;
- мережі, що навчаються без учителя.

Нейронні мережі, що навчаються з учителем, є засобами для видобування з набору даних інформації про взаємозв'язки між входами і виходами нейромережі. Ці взаємозв'язки можуть бути записані математичними рівняннями, які можна використовувати для прогнозування або прийняття управлінських рішень. “Учителем” в даному випадку є набір параметрів, який дослідник поміщає на виході мережі. На вхід мережі при цьому подається відповідний даному виходу вхідний набір даних. Мережа навчається встановлювати залежність між початковою інформацією і результатами адаптивного ітераційного процесу [44].

Алгоритм роботи мережі, що навчається без учителя, ґрунтується на навчанні змаганням. Алгоритм забезпечує таку поведінку нейронів мережі, що при подаванні чергового набору даних на вхід вони як би “змагаються” один з одним за якнайкращу відповідність вихідному набору згідно вибраним критеріям. В результаті змагання визначається нейрон переможець, після чого структура мережі піддається корекції. Клас нейронних мереж, що самоорганізуються, навчаються без вчителя, позначається терміном адаптивні нейронні мережі (АНМ). Важливою відмінністю методу АНМ є те, що він є чисельним, а не символічним методом обробки даних. Однією з унікальних особливостей АНМ є те, що вона надає внутрішньо властиві їй точні і прості механізми для поділу обчислювального завдання на окремі потоки, що дозволяє проводити обчислення з високим ступенем паралельності.

Навчання без учителя дає можливість виявляти у вхідних наборах даних невідомі раніше структури або закономірності, що відображає здатність АНМ до узагальнення (generalization) на основі вхідних прикладів. Цю властивість дозволяють узагальнювати великі набори багатовимірних даних, якими є фінансово-економічні показники підприємств, виявляти і демонструвати структури, що містяться в них, а також виявляти нові образи і взаємозв'язки в таких наборах даних. Нейронні мережі є потужним інструментом прогнозування. Закладені в них генетичні алгоритми, еволюціонуючи природним чином, дозволяють виявити правила і стратегії, переслідуючі множинні цілі.

Перші кроки в області штучних нейронних мереж зробили в 1943 р. Уорен Мак-Калох та Уолтер Пітс. Вони показали, що за допомогою порогових нейронних елементів можна реалізувати числення будь-яких логічних функцій [53]. У 1949 р. Хебб запропонував правило навчання, яке стало математичною основою для навчання ряду нейронних мереж [52]. У 1957-1962 рр. Ф. Розенблатт запропонував і досліджував модель нейронної мережі, яку він назвав перцептроном [54]. У 1959 р. В. Відроу и М. Хофф запропонували процедуру навчання для лінійного адаптивного елемента - AD ALINE. Процедура навчання отримала назву "Дельта правило" [53]. У 80-і роки значно розширюються дослідження в області нейронних мереж. Д. Хопфілд в 1982 р. дав аналіз стійкості нейронних мереж із зворотними зв'язками і запропонував використовувати їх для вирішення задач оптимізації. Т. Кохонен розробив і досліджував нейронні мережі, що самоорганізуються. Ряд авторів запропонували алгоритм зво-

ротного поширення помилки, який став потужним засобом для навчання багатошарових нейронних мереж [52, 53, 54]. В даний час розроблено велике число нейросистем, які використовуються в різних областях: прогнозуванні фінансових показників, управлінні, діагностиці в медицині і техніці, розпізнаванні образів і так далі [56, 57].

Для навчання мережі використовуються різноманітні алгоритми навчання і їх модифікації [58, 59, 60, 61, 62].

Методам аналізу і підвищення якості навчальних вибірок для нейромережових прогнозних моделей присвячені роботи [63]. Запропоновані міри несуперечливості та повторюваності даних навчальної вибірки, а також методи підвищення якості навчальних вибірок, згідно запропонованим мірам. Методи покращення полягають у попередньому перетворенні даних (ППД) першого рівня, яке збільшує повторюваність даних (нормуванні, згладжуванні, заміні ряду абсолютними та відносними приростами, вибору множини вхідних сигналів), та ППД другого рівня, що забезпечує однозначність вибору розміру опису образу навчальних вимірів. Для цього здійснюється пошук розміру образу, при якому часовий ряд розбивається на ділянки різної довжини, складність на яких однакова. Для кожної ділянки формулюється стислий опис за допомогою спільної апроксимуючої функції. Порядок та клас апроксимуючої функції задає складність кожної ділянки. Пропонується модифікований метод формування навчальної вибірки, який, на відміну від методу "вікон", дозволяє зняти неоднозначність вибору розміру розпізнаваного образу навчальних наборів.

Авторами [64, 65] розроблений алгоритм навчання мере-

жі, який мінімізує середньоквадратичну помилку нейронної мережі за рахунок використання адаптивного кроку навчання $\tau(t)$. Пропонується використовувати логарифмічну функцію активації для вирішення завдань прогнозування, яка дозволяє отримати прогноз значно точніше. Аналіз багат шарових нейронних мереж авторів [64, 65] і алгоритмів їх навчання дозволив виявити ряд недоліків і виникаючих проблем:

- невизначеність у виборі числа шарів і кількості нейронних елементів в шарі;
- повільна збіжність градієнтного методу з постійним кроком навчання;
- складність вибору відповідної швидкості навчання;
- неможливість визначення точок локального і глобального мінімуму, оскільки градієнтний метод їх не розрізняє;
- вплив випадкової ініціалізації вагових коефіцієнтів НС на пошук мінімуму функції середньоквадратичної помилки.

Використання нейронних мереж для обробки фінансово-економічних даних підприємств або компаній інколи є недостатньо гнучким. Наприклад, в процесі самонавчання нейромережі не допускається додавання нових нейронів. Складність використання АНМ також обумовлена тим, що розмірність плоскості вихідних параметрів має бути визначена до початку навчання. У таких випадках метод АНМ доцільно доповнити генетичними алгоритмами.

Прогнозування з використанням інструментарію генетичних алгоритмів (machine learning) вперше було запропоновано в 70-х роках ХХ століття [66, 51, 67, 68]. У другій половині 1980-х до цієї ідеї повернулися у зв'язку з навчанням

нейронних мереж. Вони дозволяють вирішувати завдання прогнозування (останнім часом найширше генетичні алгоритми навчання використовуються для банківських прогнозів), класифікації, пошуку оптимальних варіантів, і абсолютно незамінні в тих випадках, коли в звичайних умовах рішення задачі засноване на інтуїції або досвіді, а не на строгому (у математичному сенсі) її описі. Використання механізмів генетичної еволюції для навчання нейронних мереж здається природним, оскільки моделі нейронних мереж розробляються за аналогією з мозком і реалізують деякі його особливості, що з'явилися в результаті біологічної еволюції [69].

Важливим недоліком генетичних алгоритмів є складність інтерпретації розв'язку та програмна реалізація. Проте перевагою є ефективність в пошуку глобальних мінімумів адаптивних рельєфів, оскільки в них досліджуються великі області допустимих значень параметрів нейронних мереж. Генетичні алгоритми дають можливість оперувати дискретними значеннями параметрів нейронних мереж, що може привести до скорочення спільного часу навчання.

Особливо відзначимо, що в 70-і роки увагу фахівців в області прогнозування привертала методи індуктивного моделювання. [70, 71, 72]. На відміну від дедуктивного методу пізнання, основна ідея індуктивного моделювання полягає у побудові моделей реальних процесів на основі часових рядів, а не на основі теоретичних знань про закономірності функціонування об'єкту.

Ідея методу групового урахування аргументів запропонована академіком О.Г. Івахненком [73], полягає у перекладен-

ні на ЕОМ задачі відшукування моделі, яка найкраще описує процес, що досліджується. Пошук організовано, базуючись на еволюційному принципі, починаючи з найпростіших моделей, які ґрунтуються на опорній функції, яка найчастіше має 2 аргументи, наприклад:

$$\begin{aligned}
 y &= a_0 + a_1x_ix_j, \\
 y &= a_0 + a_1x_i + a_2x_j, \\
 y &= a_0 + a_1x_i + a_2x_j + a_3x_ix_j, \\
 y &= a_0 + a_1x_i + a_2x_j + a_3x_i^2 + a_4x_j^2 + a_5x_ix_j.
 \end{aligned} \tag{6.2}$$

В загальному випадку, використовується поліном Колмогорова-Габора:

$$y = a_0 + \sum_{i=1}^n a_ix_i + \sum_{i=1}^n a_{ij}x_ix_j + \sum_{i<j<k} a_{ijk}x_ix_jx_k. \tag{6.3}$$

Відомо, що при збільшенні порядку полінома точність наближення ним множини експериментальних даних спочатку зростає, а потім спадає. Тому алгоритм МГУА спрямований на пошук моделі оптимальної складності. Розглянемо більш детально алгоритм МГУА.

Усі доступні дані $\{x_1(t_i)\}, \{x_2(t_i)\} \dots \{x_m(t_i)\}$ та $\{y(t_i)\}, i = 1..N$ поділяють на навчальну та контрольну вибірки. Необхідно побудувати модель процесу наступного виду

$$y(t) = F(x_1(t), x_2(t), \dots, x_m). \tag{6.4}$$

Пошук моделі оптимальної складності виконується у декілька етапів селекції. На кожному етапі перебираються усі пари вхідних параметрів $x_i(t)$ та $x_j(t)$ та будується модель вихідної величини $y(t + \Delta t) = f(x_i(t), x_j(t))$, де Δt - період упередження. У якості функції f вибирається одна з опорних функцій, наприклад, задані рівняннями 6.3 або 6.3. Якщо вибрати внутрішнім критерієм якості моделей критерій мінімальності середньоквадратичного відхилення прогнозів $f(x_i(t_k), x_j(t_k))$ від фактичних майбутніх значень $y(t_{k+1})$, то параметри опорної функції оцінюються методом найменших квадратів, хоча для більш складних опорних функцій можливий вибір іншого критерію та використання нелінійних методів оптимізації для оцінювання параметрів.

Множина отриманих моделей оцінюються згідно вибраного зовнішнього критерію. Таким критерієм може бути мінімальність середньоквадратичного відхилення прогнозів від фактичних майбутніх значень $y(t_{k+1})$ для даних контрольної вибірки. У наступний етап селекції проходять моделі з найкращими показниками зовнішнього критерію. Кількість моделей, які залишаються, задається дослідником і дозволяють обмежити кількість моделей для послідуного перебору.

На наступному етапі у якості вхідних величин використовуються відгуки моделей $y_k(x_i, x_j)$, вибраних на попередньому етапі селекції. Будуються моделі виду $z(t) = f(y_i(t), y_j(t))$. Найкращі за зовнішнім критерієм моделі переходять у наступний етап і т.д. Критерієм завершення алгоритму МГУА є досягнення найкращої точності моделей, після чого точність почне падати.

У монографії [8] запропоновано нечіткий МГУА. У ро-

80 РОЗДІЛ 6. НЕЙРОМЕРЕЖЕВЕ ПРОГНОЗУВАННЯ

ботах [30-32,46, zauchenko] розглянута лінійна інтервальна модель регресії

$$Y = A_0x_0 + A_1z_x + \dots + A_nx_n, \quad (6.5)$$

де $A_i = (a_i, c_i)$ - нечіткі числа трикутного виду, де a_i - центр інтервала, c_i - його ширина. $c_i \geq 0$, x_1, x_2, \dots, x_n - вхідні змінні ($x_i \in \mathbb{R}$). Тоді відгук Y моделі 6.5 буде теж нечітким числом з центром інтервалу

$$a_y = \sum a_i z_i = a^T \cdot z, \quad (6.6)$$

а ширина інтервала

$$c_y = \sum c_i |z_i| = c^T \cdot |z|. \quad (6.7)$$

Щоб дана модель коректно описувала навчальну вибірку $\{(y_i, x_1^{(i)}, x_2^{(i)}, \dots, x_n^{(i)})\}$, $i = 1..n$, необхідно знайти такі значення нечітких параметрів $A_i = (a_i, c_i)$ моделі регресії 6.5, щоб межі нечітких відгуків моделі $Y(x_1^{(i)}, x_2^{(i)}, \dots, x_n^{(i)})$ включали реальні значення y_i та б) сумарна ширина нечітких відгуків моделі була б мінімальною. Таким чином, задача нечіткої лінійної регресії зводиться до задачі лінійного програмування:

$$\begin{aligned} \min & (c_0 \cdot n + c_1 \sum_{k=1}^n |x_i^{(k)}| + c_2 \sum_{k=1}^n |x_j^{(k)}| + c_3 \sum_{k=1}^n |x_j^{(k)} \cdot x_j^{(k)}| + \\ & + c_4 \sum_{k=1}^n |x_i^{(k)}|^2 + c_5 \sum_{k=1}^n |x_j^{(k)}|^2), \end{aligned} \quad (6.8)$$

ПРИ УМОВАХ

$$\begin{aligned}
 & a_0 + a_1 x_i^{(k)} + a_2 x_j^{(k)} + a_3 x_i^{(k)} \cdot x_j^{(k)} + a_4 \left(x_j^{(k)} \right)^2 + \left(a_5 x_{kj} \right)^2 - \\
 & \left. \begin{aligned}
 & - \left(c_0 + c_1 | x_i^{(k)} | + c_2 | x_j^{(k)} | + c_3 | x_i^{(k)} \cdot x_{kj} | + a_4 | \left(x_j^{(k)} \right)^2 | + a_5 | \left(x_j^{(k)} \right)^2 | \right) \leq y_k \\
 & - \left(c_0 + c_1 | x_i^{(k)} | + c_2 | x_j^{(k)} | + c_3 | x_i^{(k)} \cdot x_{kj} | + a_4 | \left(x_j^{(k)} \right)^2 | + a_5 | \left(x_j^{(k)} \right)^2 | \right) \leq y_k \\
 & - \left(c_0 + c_1 | x_i^{(k)} | + c_2 | x_j^{(k)} | + c_3 | x_i^{(k)} \cdot x_{kj} | + a_4 | \left(x_j^{(k)} \right)^2 | + a_5 | \left(x_j^{(k)} \right)^2 | \right) \leq y_k
 \end{aligned} \right\} \\
 & a_0 + a_1 x_i^{(k)} + a_2 x_j^{(k)} + a_3 x_i^{(k)} \cdot x_{kj} + a_4 \left(x_j^{(k)} \right)^2 + a_5 \left(x_j^{(k)} \right)^2 + \\
 & + \left(c_0 + c_1 | x_i^{(k)} | + c_2 | x_j^{(k)} | + c_3 | x_i^{(k)} \cdot x_j^{(k)} | + a_4 | \left(x_j^{(k)} \right)^2 | + a_5 | \left(x_j^{(k)} \right)^2 | \right) \geq y_k
 \end{aligned}
 \tag{6.9}$$

де k - номер точки вхідних даних $k = 1, 2, \dots, n$. Оскільки для застосування симплекс-методу необхідно мати вимогу невід'ємності шуканих параметрів, то пропонується перейти до двоїстої задачі, розв'язавши двоїсту задачу, відновити значення шуканих параметрів.

В монографії [8] також наведено методи нечіткої регресії типу (6.5) для нечітких чисел гаусового та дзвоноподібного виду, а також для регресії у вигляді ортогональних поліномів Чебишева та Лагерра, а також тригонометричних поліномів. Зведення регресії до нечіткого виду дозволяє подолати відомі проблеми регресійного аналізу, як мультиколінеарність, гетероскедастичність. На увагу заслуговує метод МГУА для нечітких входів та нечіткого виходу, так як дані про ціни сучасних фінансових ринків мають невизначеність і задані інтервальним способом.

6.1 Лабораторна робота № 3. Нейромережві моделі засобами Matlab.

Як зазначено вище, нейронні мережі є потужним інструментом моделювання відображень входів та виходу у тому випадку, коли залежність чітко не визначена. Модель нейронних мереж, як модель типу “Чорна скринька”, аналогічно біологічній пам’яті, здатна виявити залежності у даних. Щодо механізмів або методик формування наборів даних для побудови нейромережових моделей прогнозування та здійснення прогнозів, то такі процедури часто описуються поверхнево у вітчизняних наукових публікаціях і нерідко є відсутніми навіть у сучасних реалізаціях нейромережового програмного забезпечення. Тому нами пропонуються деякі підходи і алгоритми до побудови вибірок навчання саме для прогнозування часових рядів фінансово-економічної динаміки.

Найчастіше нейронні мережі описують лагову залежність виду $y_{t+1} = f(y_t, y_{t-1}, y_{t-2}, \dots, y_{t-m})$. Тому можна скористатись табличним процесором Excel, сформувавши вибірку, аналогічну описаній у попередній лабораторній роботі “Авторегресійні моделі”. Після підготовки таблиць координат лагових векторів та відгуків системи, дані таблиці необхідно зберегти у окремі текстові файли для імпорту в системі Matlab.

Розглянемо текст функції `neural_prepare.m`, яка призначена для побудови вибірки навчання та

```
function neural_prepare(infilename,nprognos,dt,k)
```

```
y=dlmread(infile);% Завантаження ряду з текстового файлу.
%%%%%%%%%%
if (size(y,2)>size(y,1))
    y=transpose(y);
end
k=k+2;
y_all=y;
y_init=y;%(10000:15200);
n=length(y); % n - довжина вхідного ряду.
nprognoz=nprognoz/dt;

%%%%%%%%%%5
n=length(y); % n - довжина вхідного ряду.
x=1:n;

%y_max=max(y);
%y_min=min(y);
%y=(y-min(y))/(max(y)-min(y)); % нормалізація

%y=y(2:n)-y(1:n-1);n=n-1; % абс. прирости
%y=(y(2:n)-y(1:n-1))./y(1:n-1); n=n-1; % відн. прирости

mas_vhod=[]; % Масиви входів: лагові змінні
mas_vyhod=[]; % Масив виходів
n=length(y);
for i=1:n-dt*k % Цикл формування масиву вибірки навчання.
    y_vhod=y(i:dt:i+dt*(k)); % Масив вхідних параметрів
    dy_vhod=(y_vhod(2:k)-y_vhod(1:k-1))./y_vhod(1:k-1);
    y_vyhod=y(i+dt*k); % Масив вихідних параметрів
```

84 РОЗДІЛ 6. НЕЙРОМЕРЕЖЕВЕ ПРОГНОЗУВАННЯ

```
dy_vyhod=dy_vhod(k-1);
dy_vhod=dy_vhod(1:k-2);
mas_vhod=[mas_vhod,dy_vhod]; %додаємо новий запис
mas_vyhod=[mas_vyhod,dy_vyhod];
end

% Після запуску даної програми, в робочій області
% з'являться змінні mas_vhod та mas_vyhod, які є
% навчальною вибіркою для нейромережі. Після цього
% можна запустити nntool та імпортувати ці дані
% в нейромережу

% network_type=0; % завантажити параметри навченої нейромережі
% network_type=1; % перцептрон
% network_type=2; % радіально-базисна нейромережа
network_type=0;
switch (network_type)
    case 1
        net = newp(mas_vhod,mas_vyhod,'purelin','learnp');
        %net.trainParam.epochs = 20;
        net=train(net,mas_vhod,mas_vyhod);
        %net=init(net);
    case 2%rb
        net=newrb(mas_vhod,mas_vyhod,0.0,1.0,100);
        %net=train(net,mas_vhod,mas_vyhod);
    case 3
        load('net_.mat');
    case 0
        uiload;
```

```
net=network1;

end

y_vhod=mas_vhod(:,size(mas_vhod,2));%y(n-(k-1)*dt:dt:n);

% покрокове прогнозування
y_progn=y;
for i=1:nprognoz
    xnnext=sim(net,y_vhod);
    ynext=(1+xnnext)*y_progn(length(y_progn));
    yprev=y_progn(length(y_progn));
    dy=(ynext-yprev)/dt;
    yadd=yprev+dy:dy:ynext;
    y_progn=[y_progn; yadd'];
    y_vhod=[y_vhod(2:k-2); xnnext];
end
y_vidn=y_init;
y_progn_vidn = y_progn ;%*(y_max-y_min)+y_min;

% побудова графіків прогнозу
figure;
plot(y_progn_vidn,'r');hold on; plot(y_vidn);

%збереження прогнозного ряду
outfilename=strrep(infile,'.txt',['_dt_' num2str(dt) ...
    '_k_' num2str(k) '.txt']);
dlmwrite(outfilename,y_progn_vidn,'\n');
```

```

outfile=strep(outfile, '.txt', '.mat');
save(outfile);

end

```

Вищенаведена функція приймає 4 аргументи (вхідний файл, довжина прогнозу, інтервал дискретизації Δt та кількість лагових змінних r (довжина пам'яті)). Наприклад, підготовка навчальної вибірки для прогнозування часового ряду, збереженого у файлі `ux.txt` на 100 кроків вперед з дискретизацією 1 день та довжиною лагового вектора 10, може бути викликана наступною командою:

```
neural_prepare('dj.txt',100,1,10);
```

Після запуску даної команди, система запропонує зберегти `mat`-файл з даними згенерованої навчальної вибірки. Після збереження, пропонується створити та навчити нейронну мережу в інтерактивному режимі. Для початку наберіть команду `nntool`, після чого з'явиться діалогове вікно створення нейронних мереж (див. рис.6.3).

Необхідно імпортувати дані в дану систему. Для цього необхідно натиснути кнопку "Import". З'явиться вікно імпорту (показано на рис. 6.3). Реалізовано 2 способи імпорту: з робочої області змінних `matlab` (Import from Matlab workspace) та з `mat`-файлу (Import from disk file). Виберіть другий спосіб, натисніть кнопку `Browse` та вкажіть шлях до `mat`-файлу, у який було збережено навчальну вибірку. Виберіть спочатку змінну `mas_vhod` та укажіть, що це вхідні дані (Input data). Для імпорту натисніть кнопку `Load` справа вниз. Після імпорту, з цього ж вікна можна імпортувати вихідний масив.

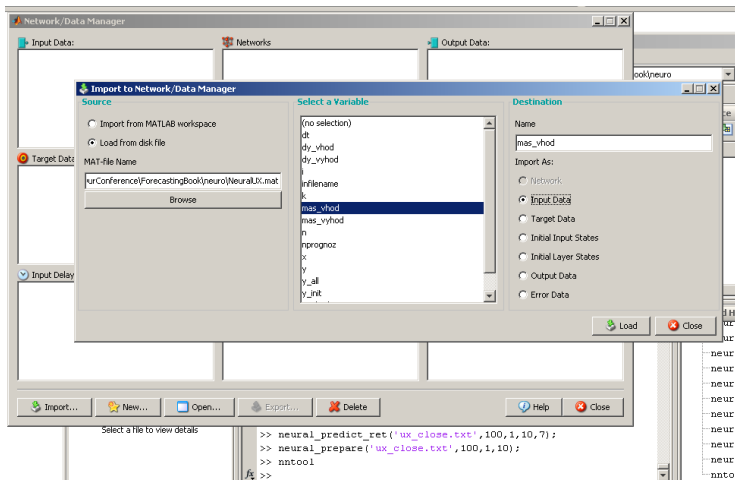


Рис. 6.3: Інтерфейс створення нейронної мережі

Укажіть змінну `mas_vyhod`, вкажіть, що це вихідна змінна (Target data) та імпортуйте дані, натиснувши кнопку Load. Після цього діалогове вікно імпорту можна закрити. В загальному випадку можна завантажити і інші дані, але це - у якості власних експериментів.

Після імпорту необхідно створити нейромережу. Це здійснюється кнопкою “New” на головному вікні програми nntool. З'явиться вікно створення нейронної мережі (рис.6.4).

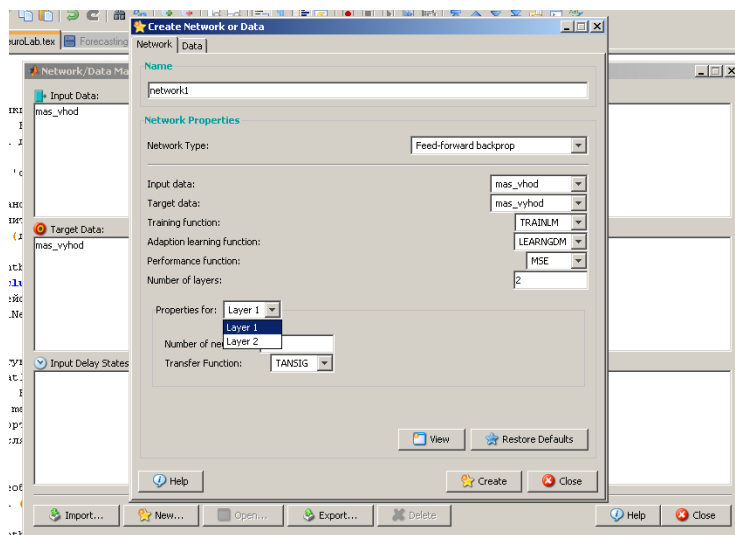


Рис. 6.4: Інтерфейс створення нейронної мережі

Спершу необхідно вказати архітектуру нейронної мережі (список Network Type), вказати завантажені вхідні (Input data) та вихідні (Target data) масиви. Також задається функція навчання (Training function) та адаптації (Adaptation function) та міра якості апроксимації (Perfomance function). Можна залишити значення по замовчуванню. Кількість шарів нейронів задається у полі “Number of layers” (залишимо значення 2. При бажанні змінити, треба ввести нове значення та натиснути ентер). Для редагування налаштувань кожного шару (кількість нейронів та активаційна функція), необхідно вибрати номер шару у списку Properties for: та задати відповідні параметри шару. Для останнього шару кількість нейронів задається рівним розмірності вихідного вектору.

Після завдання усіх параметрів нейронної мережі, для її створення натисніть кнопку “Create”, після повідомлення про успішне створення нейромережі, діалогове вікно створення можна закрити. Нова нейронна мережа матиме назву network1 (не змінюйте її при створенні) і виведена в середньому списку. Подвійно натиснувши мишкою на ім'я нейронної мережі, Ви активуєте вікно роботи з нею. (рис.6.5). На першій закладці зображена архітектура нейронної мережі. Нам необхідно перейти до закладки навчання (Train, див. рис. 6.6)

У закладці налаштування навчання необхідно задати вхідний та вихідний масиви, після чого натиснути кнопку “Train network”. Після цього звичайно з'являється вікно, яке демонструє процес навчання нейронної мережі (рис.6.7). Процес навчання часто займає значний час. Тому під час навчан-

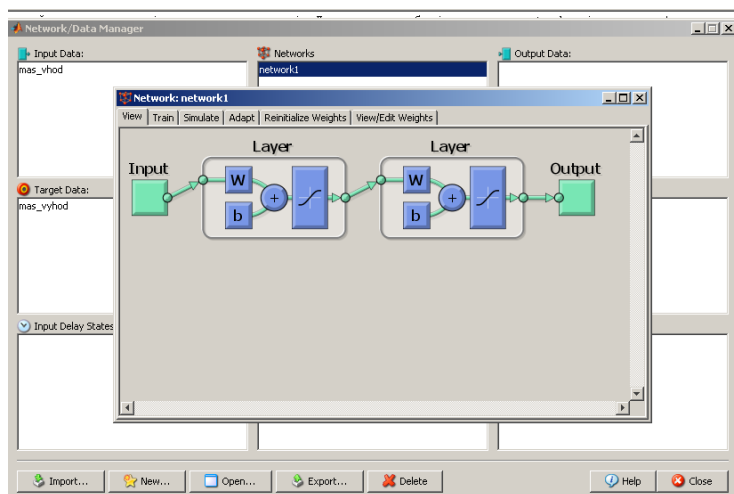


Рис. 6.5: Налаштування нейронної мережі

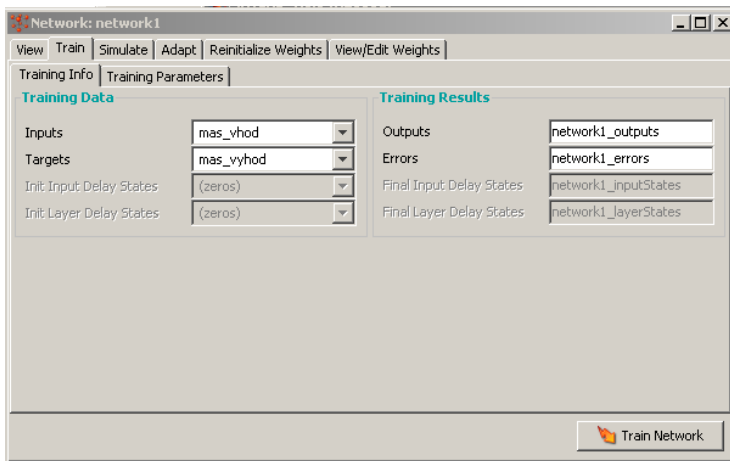


Рис. 6.6: Налаштування навчання нейронної мережі

92 РОЗДІЛ 6. НЕЙРОМЕРЕЖЕВЕ ПРОГНОЗУВАННЯ

ня можна або дочекатись досягнення заданої точності, або, пересвідчившись, що точність покращується дуже повільно, допустимо перервати процес навчання кнопкою “Stop training”. При цьому у нейронній мережі збережуться найкращі параметри, які вдалося знайти до перервання навчання.

Після навчання, необхідно зберегти дані навченої нейронної мережі в окремий mat-файл. Для цього виділимо нейронну мережу у середньому списку та скористаємося кнопкою “Export”. Вказавши у списку змінну “network1”, натисніть кнопку “Save”. Після збереження нейронної мережі можна закрити усі вікна, включаючи головне вікно nntool.

Наступним етапом буде побудова багатокрокового прогнозу. Для цього призначена функція neural_predict.m, текст якої наведено нижче.

```
function neural_predict;
uiload;
network_type=0;
switch (network_type)
    case 1
        net = newp(mas_vhod,mas_vyhod,'purelin','learnp');
        %net.trainParam.epochs = 20;
        net=train(net,mas_vhod,mas_vyhod);
        %net=init(net);
    case 2%rb
        net=newrb(mas_vhod,mas_vyhod,0.0,1.0,100);
        %net=train(net,mas_vhod,mas_vyhod);
    case 3
        load('net_.mat');
```

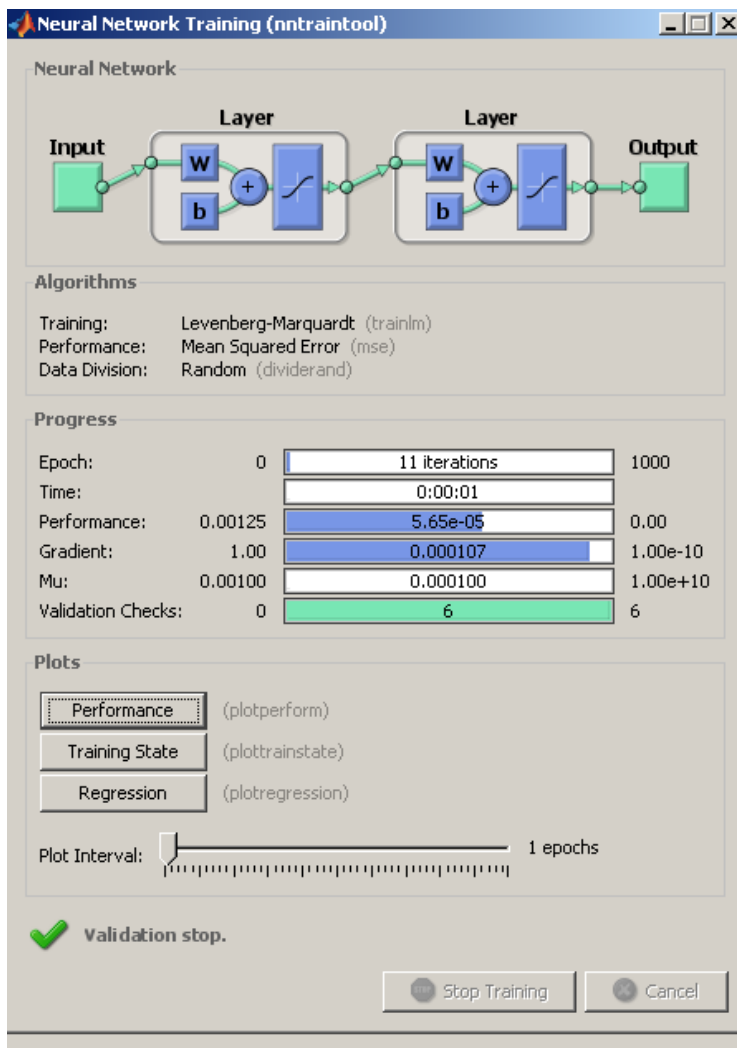


Рис. 6.7: Процес навчання нейронної мережі

94 РОЗДІЛ 6. НЕЙРОМЕРЕЖЕВЕ ПРОГНОЗУВАННЯ

```
case 0
    uiload;
    net=network1; % переименовать network1 если необходимо
end

y_vhod=mas_vhod(:,size(mas_vhod,2));%y(n-(k-1)*dt:dt:n);

% покрокове прогнозування
y_progn=y;
for i=1:nprognoz
    xnext=sim(net,y_vhod);
    ynext=(1+xnext)*y_progn(length(y_progn));
    yprev=y_progn(length(y_progn));
    dy=(ynext-yprev)/dt;
    yadd=yprev+dy:dy:ynext;
    y_progn=[y_progn; yadd'];
    y_vhod=[y_vhod(2:k-2); xnext];
end
y_vidn=y_init;
y_progn_vidn = y_progn ;%*(y_max-y_min)+y_min;

% побудова графіків прогнозу
figure;
plot(y_progn_vidn,'r');hold on; plot(y_vidn);

%збереження прогнозного ряду
outfilename=strrep(infile,'.txt',['_dt_' num2str(dt) '_k_' n
```

```
dlmwrite(outfilename,y_progn_vidn,'\n');  
outfilename=strrep(outfilename,'.txt','.mat');  
save(outfilename);  
  
end
```

Запустивши дану функцію, програма запитає завантажити спочатку mat-файл, збережений при створенні вибірки навчання (запуск `neural_prepare`), а потім навченної мережі (збережена програмою `ntool` після навчання). Після завантаження файлів, буде обчислено остаточний прогностичний ряд та виведено у вихідний файл та на екран у вигляді графіків. Графіки можна зберегти, хоча не обов'язково, так як ряди зберігаються автоматично у текстовому форматі.

Даний алгоритм дає можливість змінювати будь-які налаштування у конфігурації нейронних мереж і дозволяє провести багато експериментів з ними. Нами вибрано 7 нейронних мереж та розроблена функція `neural_predict_ret.m`, яка автоматично здійснює формування навчальної вибірки, створення та навчання нейронної мережі, багатокрокове прогнозування та збереження результатів. Функція `neural_predict_ret` містить наступні параметри:

```
neural_predict_ret(infile, nprogn, dt, k, network_type)  
infile - ім'я вхідного файла, рядок символів  
nprogn - кількість кроків прогнозування, ціле число  
dt - крок дискретизації при формуванні лагових векторів  
k - розмірність лагового вектору (довжина пам'яті)  
network_type - тип нейронної мережі (ціле число від 1 до
```

7).

96 РОЗДІЛ 6. НЕЙРОМЕРЕЖЕВЕ ПРОГНОЗУВАННЯ

Коди програм, розглянутих у даній лабораторній роботі, можна завантажити за адресою:

<http://prognoz.ck.ua/neuro.rar>

Розділ 7

Випадкові процеси та методи побудови марківських моделей при прогнозуванні.

У даному підрозділі розглядаються моделі для прогнозування фінансових рядів, які базуються на ланцюгах Маркова. У представленій роботі, на відміну від аналогічних досліджень з фінансового прогнозування [74], не будуються моделі регресійних рівнянь, а пропонується модель машинного навчання (приховані марківські моделі) для аналізу фінан-

сових часових рядів.

Структура роботи наступна. Спочатку розкриваються основні концепції фінансових часових рядів, складнощі точного їх прогнозування, обмеженість регресійних моделей, а також мета та основні методи навчання прихованих марківських моделей. проводиться аналіз попередніх робіт, розкриваються недоліки розроблених раніше моделей та пропонуються шляхи удосконалення, зокрема, використання неперервних сумішів функцій розподілу Гауса та відповідна модифікація класичного алгоритму максимізації правдоподібності. Далі в [75] пропонується подвійно зважений алгоритм максимізації правдоподібності, основна мета якого полягає у подоланні однієї з проблем класичних прихованих марківських моделей – однакової значущості всіх фрагментів даних для прогнозу. Оскільки при прогнозуванні фінансових рядів нові фрагменти змін ціни мають більшу значущість, така модифікація алгоритму навчання моделі дозволяє покращити прогнозні можливості саме для фінансово-економічних часових рядів.

У розділі 2 висвітлюється методика генетично складних ланцюгів Маркова, основні концепції якої вперше опубліковані в роботі [39] та були удосконалені одним із авторів даної роботи.

Зазначимо три основні проблеми задач прогнозування часових рядів. *По-перше*, характерні властивості відомих фінансових рядів є динамічними, тобто не існує єдиної моделі, яка б працювала весь час. *По-друге*, важко визначитись між довготривалим трендом та короточасними флуктуаціями під час торгівлі. Іншими словами, ефективна система

повинна уточнювати чутливість з часом. *По-третє*, важко визначити корисність інформації. Інформація, яка вводить в оману, повинна бути ідентифікована та вилучена.

Робота [75] націлена на подолання вищезазначених проблем за допомогою системи передбачення, яка базується на прихованих марківських моделях (Hidden Markov Model, НММ, далі - ПММ) [76, 77, 78, 79]. Модель ПММ включає суміш функцій розподілу для кожного стану, як генератор прогнозу. Суміші функцій Гауса мають бажані властивості, завдяки чому можливо моделювати часові ряди, розподіл яких не відповідає єдиному розподілу Гауса. Потім параметри прихованої марківської моделі доповнюються при кожній ітерації за допомогою алгоритму **максимізації правдоподібності (МП)**. В кожний дискретний момент часу параметри суміші розподілів, вага кожного компоненту розподілу на виході, та матриці перетворення динамічно оновлюються для забезпечення нестационарності результуючого часового ряду.

Новий експоненційно зважений алгоритм МП дає можливість подолати другу та третю з вищезазначених проблем. Даний алгоритм надає більшого значення більш свіжим даним, налаштовуючи вагові коефіцієнти на зміну волатильності. Волатильність обчислюється як відхилення індексу S&P 500 від індексу Dow Jones. Таким чином, додаємо нову інформацію, яка корисна для перевірки достовірності результатів.

Баланс між довгими та короткими рядами знайдено в процесі комбінованої торгівлі та передбачення.

Базуючись на експоненційно зваженому алгоритму МП, пропонується подвійно зважений алгоритм МП. Даний ал-

горитм здатний подолати недолік надчутливості експоненціально зваженого методу максимізації правдоподібності.

Показано, що зважений алгоритм МП може бути записаний у формі експоненціальних ковзних середніх від змінних моделі ПММ. Це дозволяє отримати перевагу над існуючими економетричними методами у генеруванні рівнянь переоцінки, основаних на максимізації правдоподібності.

Використовуючи технології з теореми про МП, доведено збіжність запропонованих алгоритмів.

Експериментальні результати показують, що модель ПММ стабільно показує кращі результати, ніж прибутковість індексу змішаних фондів протягом п'яти 400-денних тестових періодів на проміжку від 1994 до 2002 року. Запропонована модель також стабільно показує кращі результати, ніж відомі алгоритми прогнозування для індексу S&P 500 за показником Шарпа.

7.1 Приховані марківські моделі

Почнемо з визначення властивостей марківського процесу, після чого розглянемо характеристики прихованих марківських моделей. В роботах Рабінера [76] наведено детальний огляд марківських процесів та прихованих марківських моделей та відображені основні проблеми прихованих марківських моделей. Дані роботи є основоположними при побудові прихованих марківських моделей для прогнозування фінансових часових рядів.

Марківські моделі використовуються для задач аналізу

даних, таких як розпізнавання мови, дослідження варіацій температури, біологічних часових рядів та ін. У марківській моделі кожна послідовність спостережень залежить від попередніх елементів у даній послідовності. Розглянемо систему, в якій задано множину станів, $S = \{1, 2, \dots, S\}$. У кожний дискретний проміжок часу t , система здійснює перехід з одного стану в інший, згідно із заданими ймовірностями переходів. Позначимо стан системи в момент часу t як s_t .

В багатьох випадках прогноз наступного стану та зв'язана з ним послідовність спостережень залежить тільки від теперішнього стану, це означає те, що ймовірності переходів між станами не залежать від усієї передісторії системи. Такий процес називається *марківським процесом першого порядку*.

Властивості таких процесів можна описати наступними рівняннями:

$$P(X_{t+1} = s_k | X_1, \dots, X_t) = P(X_{t+1} = s_k | X_t) \text{ (обмежений горизонт) (2.1)}$$

$$= P(X_2 = s_k | X_1) \text{ (часова інваріантність) (2.2)}$$

Оскільки переходи між станами інваріантні у часі, можна записати матрицю ймовірностей переходів A з елементами:

$$a_{ij} = P(X_{t+1} = s_j | X_t = s_i) \text{ (2.3)}$$

a_{ij} є ймовірностями, отже:

$$a_{ij} \geq 0;$$

$$\forall i, j \sum_{j=1}^N a_{ij} = 1 \quad (7.1)$$

Також необхідно зафіксувати ймовірність кожного стану

на початку певного часового інтервалу та початковий розподіл ймовірностей:

$$\pi_i = P(X_1 = s_i) \quad (2.5)$$

При

$$\sum_{i=1}^N \pi_i = 1 \quad (7.2)$$

У явній марківській моделі стани, між якими здійснюються переходи та функції ймовірностей є відомими, тому можна розглядати послідовність станів як вихідні дані.

Ланцюги Маркова можна застосувати для моделювання рухів ціни акцій, ввівши множину дискретних станів системи для моделювання зміни ціни певного виду акцій. Ланцюг Маркова можна зобразити у вигляді графу переходів між станами. Стани в ланцюгу Маркова можна зв'язати зі зміною ціни: “значне збільшення”, “невелике збільшення”, “незмінність”, “невеликий спад”, “значний спад”.

Марківська модель, яка описана в попередньому параграфі, є обмеженою, тому ми розширюємо її для покращення можливостей в приховану марківську модель (далі ПММ). В ПММ процес, який генерує розглядувану послідовність, є невідомим. Кількість станів, ймовірності переходів та початковий стан також невідомі.

Замість квантування кожної зміни детермінованого виходу набором дискретних станів, (таких як велике зростання, незначний спад тощо), кожний стан прихованої марківської моделі зв'язаний з імовірнісною функцією. В момент часу t ,

послідовність станів o_t генерується імовірнісною функцією $b_j(o_t)$, яка зв'язана зі станом j , та має такий вигляд:

$$b_j(o_t) = P(o_t | X_t = j) \quad (3.1)$$

Розглянемо приклад прихованої марківської моделі, яка описує діяльність інвестора з передбачення змін ринкової ціни, основу на множині стратегій інвестування. Припустимо, що існує 5 стратегій інвестування, кожна з яких дає відмінний прогноз ринкової ціни на наступний день (великий спад, невеликий спад, без змін, невелике збільшення, значне збільшення). Відомо, що інвестор здійснює прогноз кожен день лише згідно однієї стратегії.

Яким чином професіонал здійснює прогноз – невідомо. Також невідомо, якою стратегією він буде користуватись кожного наступного дня.

Для відповіді на перше запитання створимо приховану марківську модель як модель фінансового експерту. Кожна з п'яти стратегій відповідає прихованому стану в ПММ. Множина змін ціни (явних станів) містить усі символи (коди) змін ціни, та кожна зміна зв'язана з певним станом. Зі стратегіями зв'язані різні розподіли ймовірностей символів змін ціни. Кожний стан зв'язаний з іншими станами за допомогою функції ймовірностей переходів.

На відміну від марківських ланцюгів, прихована марківська модель передбачає вибір трейдером окремої стратегії після використання в минулий раз однієї з інших, тільки залежно від ймовірності послідовності спостережень. Шлях до удосконалення ефективності ПММ полягає у тому, що вибирається найкраща послідовність стратегій на основі наявної послідовності станів. Вводячи функції розподілу ймовірності

стей для станів ПММ, збільшуємо можливості моделі у презентабельності, порівняно з явною моделлю, при якій зв'язки “стратегія – стан” є фіксованими.

Алгоритм генерації послідовності прогнозних станів включає наступні кроки:

1. Вибрати початковий стан (початкову стратегію) $X_1 = i$, згідно з початковим розподілом станів Π .

2. Задати номер проміжку часу $t = 1$, з якого відраховується послідовність станів.

3. Отримати прогноз, згідно зі стратегією даного стану i , тобто $b_{ij}(o_t)$. Таким чином, ми маємо прогноз для кожного наступного дня.

4. Здійснити перехід до нового стану X_{t+1} , згідно розподілу ймовірностей переходів між станами.

5. Задати час $t = t+1$ та перейти до кроку 3, якщо $t \leq T$, інакше завершити роботу.

Очевидно, що існує реальна відповідність між аналізом послідовностей даних та ПММ. Особливо у випадку аналізу фінансових часових рядів та прогнозування, коли дані можуть бути далекі від генерованих певним стохастичним процесом, який невідомий публіці. Аналогічно вищезазначеному прикладу, якщо кількість стратегій професіонала та їх суть невідомі, можна взяти уявного професіонального інвестора і вважати його зовнішньою силою, яка рухає ціну вгору чи вниз. Як показано в прикладі, ПММ має більш широкі можливості, ніж звичайні ланцюги Маркова. Ймовірності переходів, як і функції розподілу ймовірностей появи послідовності станів, уточнюються в процесі “навчання” алгоритму.

Крім схожості властивостей ПММ та даних часового ряду, алгоритм максимізації правдоподібності (МП) представляє собою клас ефективних методів “навчання” моделі. Маючи значні об’єми даних, які є результатом деякого прихованого процесу, можна створити ПММ, а алгоритм МП дає можливість оцінити параметри моделі, що найкраще відповідають відомим даним торгівлі.

Загальний підхід ПММ включає технологію некерованого навчання, яка дозволяє знайти нові характерні риси досліджуваного процесу. Алгоритм навчання здатен приймати вхідні послідовності різної довжини, без необхідності зводити навчання до “шаблону”. Удосконалений авторами роботи [75] алгоритм МП дає можливість збільшити “вагу” більш свіжих даних при навчанні, також відкривається можливість налаштувати баланс між вимогами до чутливості та точності.

ПММ застосовуються у багатьох галузях: розпізнання мови, біоінформатиці та аналізі різних часових рядів, таких як дані погоди, дефекти напівпровідників та ін. Вайгенд та Ши вперше застосували ПММ в аналізі фінансових часових рядів.

Можна виділити три фундаментальних задачі ПММ, які є важливими для прихованих марківських моделей:

1. Для відомої моделі $\mu = (A, B, \Pi)$, та послідовності явних станів $O = (o_1, \dots, o_T)$, обчислити ймовірність певної послідовності явних станів моделі. Тобто обчислити ймовірність $P(o_j)$.

2. Для відомої моделі та послідовності явних станів O , визначення прихованої послідовності станів $(1, \dots, N)$, що

найкраще “пояснює” дану послідовність явних станів.

3. Для відомої послідовності O , та простору можливих моделей визначити параметри та знайти модель, яка максимізує ймовірність $P(O_j)$.

Перша задача може використовуватись для визначення, які з “навчених” моделей найбільш адекватно описують послідовність явних станів, яка спостерігається. У випадку фінансових часових рядів для відомої історії змін ціни або індексу, яка модель найкраще пояснює дану історію. Друга задача полягає в знаходженні прихованого шляху. Звернувшись до вищенаведеного прикладу зі стратегіями, задача полягає у визначенні послідовності стратегій, які щоденно використовує інвестор-професіонал. Третя задача є найбільш цікавою, тому що стосується “навчання” моделі. В аналізі реальних фінансових часових рядів, розглядається навчання моделі на основі історії минулих рухів ціни. Тільки після “навчання” ми можемо використовувати модель для подальшого прогнозування.

В загальному вигляді ПММ є кортеж (S, K, Π, A, B) , де:

1. $S = \{1, \dots, N\}$ – множина станів. Стан системи в момент часу t позначимо s_t .

2. $K = \{k_1, \dots, k_M\}$ – вихідний алфавіт. У випадку дискретних станів, M є кількістю дискретних станів. У вищезгаданому прикладі M дорівнює кількості можливих змін ціни.

3. Початковий розподіл станів $= \{\pi_i\}$, де $i \in S$, π_i визначається як

$$\pi_i = P(s_1 = i) \quad (7.3)$$

4. Розподіл ймовірностей переходів між станами $A = \{a_{ij}\}$; $i, j \in S$.

$$a_{ij} = P(s_{t+1}|s_t); 1 \leq i, j \leq N \quad (7.4)$$

(2.14)

5. Розподіл ймовірностей явних станів $B = b_j(o_t)$. Функція ймовірності для кожного стану задається наступним чином:

$$b_j(o_t) = P(o_t | s_t = j) \quad (7.5)$$

Після формалізації задачі у вигляді ПММ та припущення, що модельовані дані, згенеровані за допомогою ПММ, необхідно обчислити ймовірності послідовностей станів системи та можливу послідовність прихованих станів. Також проводиться “навчання” параметрів моделі, яке базується на відомих даних та отримується більш точна модель. Далі “навчену” модель можна використовувати для передбачення нових даних. Алгоритм симуляції прихованої марківської моделі включає наступні кроки:

Виберемо початковий стан x_1 по розподілу π .

Для усіх t від 1 до :

а) Виберемо спостережувану d_t за розподілом

$$b_j(k) = p(v_k | x_j)$$

б) Виберемо наступний стан за розподілом

$$a_{ij} = p(q_{t+1} = x_j | q_t = x_i).$$

Алгоритм знаходження ймовірності послідовності спостережень у даній моделі:

Ініціалізація $\alpha_1(i) = \pi_i b_i(d_1)$.

Крок індукції: $\alpha_{t+1}(j) = \left[\sum_{i=1}^n \alpha_t(i) a_{ij} \right] b_j(d_{t+1})$.

Після T ітерацій, підрахуємо приховану ймовірність:

$$p(D|\lambda) = \sum_{i=1}^n \alpha_T(i).$$

Алгоритм знаходження найбільш ймовірних послідовностей станів:

Ініціалізація $\delta_1(i) = \pi_i b_i(d_1)$, $\psi_1(i) = \{ \}$.

Крок індукції:

$$\delta_t(j) = \max_{1 \leq i \leq n} [\delta_{t-1}(i) a_{ij}] b_j(d_t),$$

$$\psi_t(j) = \arg \max_{1 \leq i \leq n} [\delta_{t-1}(i) a_{ij}].$$

Після T ітерації визначити:

$$p^* = \max_{1 \leq i \leq n} \delta_T(i), \quad q_T^* = \arg \max_{1 \leq i \leq n} \delta_T(i).$$

4. Визначити послідовність: $q_t^* = \psi_{t+1}(q_{t+1}^*)$.

Розглянутий ітераційний алгоритм дозволяє уточнити параметри моделі та послідовність прихованих станів для даного набору спостережень.

У даному підрозділі теоретично описується загальна форма ПММ, та матеріал ілюструється за допомогою прикладів.

Формалізується марківська властивість, яка є основою даних моделей. Далі розглядаються три основні задачі, зв'язані з ПММ. Розглянута загальна структура моделі та введені розширення даної базової моделі, описані технології розв'язання поставлених задач. Для організації “навчання” системи передбачення використовуються алгоритми Вітербі та Баум-Велча.

ПММ для аналізу фінансових часових рядів

Проаналізуємо попередній досвід застосування прихованих марківських моделей для прогнозування часових рядів. Першою спробою застосувати ПММ для прогнозування в фінансах здійснили вчені А. Вайгенд та Ш. Ши [78]. Ними була запропонована модель експертів для прогнозування динаміки індексу S&P 500. Під експертами, які були закодовані дискретними станами ПММ мались на увазі три нейронних мережі з різними налаштуваннями. Перша нейромережа була налаштована на прогнозування даних з невеликою волатильністю, друга була навчена на високоволатильних даних, а третя відповідала за “викиди”. Під “викадами” мається на увазі виняткова частина даних, які не увійшли до двох вищезазначених груп.

Під час прогнозування створювався торговий пул, який являв собою середовище з обмеженими у часі даними про динаміку ціни. Даний пул оновлювався за допомогою додавання свіжих даних про ціну та вилучення з пулу старих. Моделі в даному пулі проходили “навчання” з ціллю генерації прогнозу на наступний торговий день. Навчання експерта, оснований на ПММ являло собою коригування значень матриці ймовірностей переходів A .

Модель Вайгенда та Ши має наступні недоліки. По-перше, в торговому пулі значення кожного фрагменту були однаковими, що при великій кількості даних призводить до проблеми перенавчання. Для вирішення цієї проблеми запропоновано експоненціально зважений алгоритм МП, у якому для свіжих даних надається більша значущість. Наступною проблемою є те, що при “навчанні” для трьох моделей використовується однаковий алгоритм оцінювання коефіцієнтів, не враховуючи специфічні можливості кожного виду моделі. Навіть якщо класифікувати фрагменти даних по показнику волатильності, такий поділ буде суб’єктивним та ряди з проміжними значеннями волатильності не обов’язково демонструють однакові властивості. В роботі [75] наводяться моделі з сумішами розподілів, в яких налаштовуються окремі параметри кожного компоненту суміші.

В попередніх підрозділах формули для адаптації параметрів моделі були викладені для випадку дискретної послідовності спостережень. Але зміни ціни не є дискретними значеннями, процедура квантифікації неперервних значень дискретизованими станами призводить до втрати інформації. Тому пропонується використовувати суміші функцій розподілу Гауса як модель неперервних спостережень. В загальному випадку суміш функцій розподілу Гауса має такий вигляд:

У випадку неперервної функції розподілу ПММ, функція $b_j(o_t)$ є неперервною функцією розподілу або суміш неперервних функцій розподілу:

$$b_j(o_t) = \sum_{k=1}^M w_{jk} b_{jk}(o_t) \quad j = 1, \dots, N \quad (7.6)$$

де M – кількість сумішей та w є вага кожної суміші. Ваги сумішей мають наступні обмеження:

$$\sum_{k=1}^M w_{jk} = 1, j = 1, \dots, N; w_{jk} \geq 0; j = 1, \dots, N; k = 1, \dots, M. \quad (7.7)$$

Кожна функція $b_{jk}(o_t)$ є D -вимірною логарифмічно ввігнутою функцією розподілу з середнім вектором μ_{jk} та матрицею коваріацій Σ_{jk} :

$$b_{jk}(o_t) = N(o_t, \mu_{jk}, \Sigma_{jk}) \quad (7.8)$$

Основна задача “навчання” – визначення параметрів складових функцій розподілу ймовірності. Таке D -вимірне навчання вимагає вибору найбільш корельованих часових рядів у якості вхідних даних. Наприклад, для прогнозування ціни на акції ІВМ, є сенс включити у вхідні дані ціни акцій Microsoft та інших високотехнологічних компаній.

Звичайна функція розподілу Гауса має наступний вигляд:

$$b_{jk}(o_t) = N(o_t, \mu_{jk}, \Sigma_{jk}) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (7.9)$$

де o_t – дійсне число. Багатовимірна функція розподілу Гауса запишеться так:

$$\begin{aligned} b_{jk}(o_t) &= N(o_t, \mu_{jk}, \Sigma_{jk}) = \\ &= \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma_{jk}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} (o_t - \mu_{jk})^T \Sigma_{jk}^{-1} (o_t - \mu_{jk})\right) \end{aligned} \quad (7.10)$$

де o_t – вектор явних значень прогнозованого ряду. Пропонується модифікація алгоритму Баума-Велча [75] для випадку багатовимірних часових рядів з дійсними значеннями. Вводяться наступні допоміжні змінні:

$$\delta_i(t) = \sum_{m=1}^M \delta_{im}(t) = \sum_{m=1}^M \frac{1}{P} a_i(t-1) a_{ji} w_{im} b_{im}(o_i) \beta_i(t) \quad (7.11)$$

Далі необхідна змінна для позначення ймовірності знаходження у стані j в момент часу t з k -м компонентом суміші, яка пояснює даний набір спостережень o_t :

$$\nu_{j,k}(t) = \frac{\alpha_j(t) \beta_j(t)}{\sum_{j=1}^N \alpha_j(t) \beta_j(t)} \cdot \frac{w_{jk} N(o_i, \mu_{jk}, \sigma_{jk})}{\sum_{k=1}^M w_{jk} N(o_i, \mu_{jk}, \sigma_{jk})} \quad (7.12)$$

З новими допоміжними змінними запишемо модифіковані формули уточнення параметрів:

$$\mu'_{im} = \frac{\sum_{t=1}^T \delta_{im}(t) o_t}{\sum_{t=1}^T \delta_{im}(t)} \quad (7.13)$$

$$\sigma'_{im} = \frac{\sum_{t=1}^T \delta_{im}(t) (o_t - \mu'_{im}) (o_t - \mu'_{im})'}{\sum_{t=1}^T \delta_{im}(t)} \quad (7.14)$$

$$w'_{im} = \frac{\sum_{t=1}^T \delta_{im}(t) o_t}{\sum_{t=1}^T \delta_i(t)} \quad (7.15)$$

$$a'_{ij} = \frac{\sum_{t=1}^T a_i(t) \alpha_i(t) a_{ij} b_j (o_{t+1}) \beta_i(t+1)}{\sum_{t=1}^T \alpha_i(t) \beta_i(t)} \quad (7.16)$$

В даному підрозділі створено ПММ для прогнозування фінансових часових рядів, зокрема для індексу S&P 500. Розглянуті питання про вибір функції щільності розподілу приростів досліджуваної величини, правило коригування параметрів моделі, яке базується на максимізації правдоподібності. Кожен стан ПММ зв'язаний з сумішню функцій розподілу Гауса з різними параметрами, які дають кожному стану різні прогнозні переваги. Деякі стани більш чутливі до низьковолатильних даних, інші більш точні на високоволатильних проміжках часу. Модернізований алгоритм МП приділяє більше уваги свіжим даним. Індекс Dow Jones включений в експерименти як допоміжний індикатор для визначення волатильності ряду S&P 500.

Зважений алгоритм максимізації правдоподібності для ПММ

Розглянемо більш детально традиційний алгоритм МП з точки зору конструювання функції ймовірності та Q-функції.

У попередніх роботах не приділяється достатньої уваги випадку затухання значення фрагменту даних з часом. Прийнято, що кожен фрагмент даних у послідовності навчання має однакову важливість та остаточне оцінювання базується на припущенні цієї рівної важливості, не звертаючи увагу, наскільки велика вибірка даних для навчання та проміжок часу, який описують дані. Такий підхід є ефективним при невеликих фрагментах навчання та нечутливості, або розмірність часу не є важливою. Традиційні моделі ПММ, основані на максимізації правдоподібності, показали високі результати в обробці мови та відео, де послідовність фрагментів даних не має великого значення.

Але фінансові часові ряди відрізняються від вищезгаданих сигналів своїми унікальними характеристиками. Необхідно більше звернути увагу на свіжі дані, не лишаячи старих, віддалених в часі фрагментів, але зі зменшеною довірою. Усвідомлюючи, що зміна важливості даних з часом та мотивована цим концепція експоненційного ковзного середнього (далі ЕКС) фінансових часових рядів, в роботі [75] пропонується модернізація експоненційно зваженого алгоритму МП. Показано, що остаточне перевизначення параметрів ПММ та функцій розподілу може бути виражене у формі комбінації експоненційних ковзних середніх (ЕКС) для змінних, які містять проміжні стани. Для знаходження кращого балансу між короткочасними та довготривалими трендами, автори пізніше пропонують подвійно зважений алгоритм МП, який має властивості налаштовувати чутливість за допомогою включення інформації про волатильність.

Доведено збіжність експоненціально зваженого алгори-

тму МП за допомогою зваженої версії теореми про максимізацію правдоподібності.

Нехай функція розподілу ймовірності $p(X|\Theta)$, яка показує ймовірність послідовності спостережень $X = \{x_1, \dots, x_N\}$ для даного параметру Θ . В моделі прихованих марківських гаусових сумішів (далі ПМГС) Θ включає множину середніх та коваріацій гаусових сумішей, зв'язаних з кожним станом, так же як матриця ймовірностей переходу та ймовірність кожного компоненту суміші.

Вважаючи, що дані є незалежними та рівномірно розподіленими за законом p , ймовірність спостерігати всю послідовність даних обчислюється наступним чином:

$$P(X|\Omega) = \prod_{i=1}^N p(x_i|\Omega) = L(\Omega|X) \quad (7.17)$$

Функція L називається функцією правдоподібності для параметрів при заданому наборі даних. Ціль алгоритму МП полягає в знаходженні вектору параметрів Θ^* , який відповідає максимальному значенню L :

$$\Omega^* = \arg \max_{\Omega} L(\Omega|X) \quad (7.18)$$

Далі алгоритм МП знаходить математичне сподівання для логарифмічної функції правдоподібності $\log p(X; Y|\Theta)$ відносно невідомих значень Y для даної явної частини X та параметрів останньої оцінки. Виберемо Q -функцію в якості позначення для цього очікування:

$$\begin{aligned}
 Q(\Theta, \Theta^{(i-1)}) &= E[\log p(X, Y|\Theta) | X, \Theta^{(i-1)}] = \\
 &= \int_{y \in \gamma} \log p(X, y|\Theta) f(y|X, \Theta^{(i-1)}) dy
 \end{aligned}
 \tag{7.19}$$

Іе де $\Theta^{(i-1)}$ є оцінки з минулої ітерації алгоритму МП. Базуючись на $\Theta^{(i-1)}$ намагаємось знайти нові оцінки Θ , які оптимізують функцію Q . γ є простір можливих значень для змінної y .

При обчисленні функції Q вважається, що кожен фрагмент даних має рівну важливість при оцінці нових параметрів. Цей момент є важливим у задачах розпізнавання мови та візуальних сигналів, але фінансові часові ряди мають характеристики, для яких параметр часу стає дуже важливим. З плином часу, вираженість старих фрагментів затухає, з'являються нові риси часового ряду. Якщо локальна послідовність даних, яка використовується у “навчанні” є достатньо великою, нові риси або тренди, що виникають, неминуче втратять свою чіткість.

Тому необхідно створити новий алгоритм МП, який надає більше важливості для більш свіжих даних у процедурі “навчання”. Для розв'язання цієї задачі, необхідно по-перше перетворити Q -функцію. Нагадаємо, що Q -функція є мірою “правдоподібності” повного набору даних відносно прихованих даних Y для даної послідовності явних станів X та оцінки останніх параметрів Θ . Пропонується включити часовозалежні вагові коефіцієнти η , очікуючи відобразити інформацію про часовий параметр.

Нова форма функції правдоподібності Q^* матиме наступний вигляд:

$$\begin{aligned} Q^*(\Theta, \Theta^{(i-1)}) &= E [\log \eta p(X, Y|\Theta) | X, \Theta^{(i-1)}] = \\ &= \int_{y \in \gamma} \log \eta p(X, y|\Theta) f(y|X, \Theta^{(i-1)}) dy \end{aligned} \quad (7.20)$$

Аналогічно формулам (7.13)-(7.16) виводяться ітераційні формули уточнення параметрів моделі при зваженому врахуванні фрагментів даних.

Розділ 8

Методика прогнозування на основі складних ланцюгів Маркова

8.1 Технологія прогнозування на основі складних ланцюгів Маркова

Розглянемо поняття ланцюгів Маркова. Припустимо існує послідовність дискретних станів певної системи. З цієї по-

слідовності можна визначити ймовірності переходу з одного стану в інший. Простим ланцюгом Маркова є випадковий процес, в якому ймовірності наступного стану залежать тільки від попереднього стану та не залежать від усіх інших станів. На відміну від простого, складним ланцюгом Маркова називають випадковий процес, в якому ймовірності наступного стану залежать не тільки від наявного, а від послідовності декількох попередніх станів (передісторії) [80, 81]. Кількість станів в передісторії є порядком ланцюга Маркова.

Теорія простих ланцюгів Маркова широко викладена в літературі, наприклад в [80, 81], що стосується ланцюгів Маркова вищих порядків, то в літературі даний термін використовується рідко. Наприклад, в статті [82] автор використовує термін “ланцюги Маркова” для процесів з пам'яттю об'ємом декілька попередніх значень.

Ланцюг Маркова порядку вище 1-го можна звести до простого ланцюга Маркова за допомогою узагальнення поняття стану: “наявний стан”, як ряд послідовних станів [39]. В цьому випадку апарат простих ланцюгів Маркова може бути застосований до складних.

Припустимо, що процес, динамічний ряд якого досліджується, породжений певною моделлю детермінованого хаосу, що означає існування причинно-наслідкової залежності наступних станів від передісторії. Для точного оцінювання ймовірностей наступного стану, необхідно врахувати кожен з попередніх, оцінивши ймовірності переходів при даній передісторії. Оскільки знання про причинно-наслідкову залежність обмежена даною реалізацією ряду, то ймовірності можуть бути оцінені тільки наближено. Основна задача су-

часних методів прогнозування максимально використати інформацію з відомого відрізка ряду для побудови моделі ряду та використати побудовану модель, виокремивши з можливих сценаріїв продовження ряду найбільш ймовірні.

Ряд вихідних значень, призначений для прогнозування, необхідно перевести в ряд дискретних номерів станів. Необхідно зафіксувати кількість станів з дискретного ланцюга Маркова, кожен з яких зв'язуємо з абсолютною або відносною зміною величини вихідного сигналу за певний проміжок часу Δt . Найпростішим прикладом може бути 2 стани: зростання (стан 1) та спадання (стан 2). В загальному вигляді усі можливі прирости вихідного ряду необхідно класифікувати на s класів [83]. Способи розбиття будуть розглянуті у наступних підрозділах даної роботи.

Далі здійснюється прогнозування ряду дискретизованих станів. При заданому порядку ланцюга Маркова та останньому узагальненому стані вибирається найбільш ймовірний стан в якості наступного. У випадках неоднозначності при визначенні найбільш ймовірного стану застосовується алгоритм, який дозволяє зменшити кількість можливих сценаріїв прогнозу. Таким чином, маємо ряд прогнозованих станів, які для відомого останнього значення ряду можуть бути перетворені на дискретизований ряд прогнозних значень.

Обчислення приростів, прогнозування та послідуєчне відновлення можна здійснити для різних приростів часу Δt . Для ефективного використання інформації, представленої в наявному часовому ряді, прогнозування здійснюється для різних приростів часу $\Delta t = 1, 2, 4, 8, \dots$, або більш складної ієрархії приростів, яка розглянута в даній роботі в підрозділі

8.5, та послідовного “склеювання” результатів прогнозування, отриманих при різних інтервалах дискретизації Δt .

Процедура прогнозування та склеювання є ітераційною та проводиться, починаючи з менших приростів, додаючи на кожному кроці прогноз з більшим приростом часу.

Оскільки при збільшенні кроку дискретизації часу Δt зменшується статистика для визначення ланцюгів Маркова, найбільший крок дискретизації, який приймає участь у прогнозуванні, обмежується. Для доповнення прогнозу низькочастотною складовою використовується наближення нульового порядку у вигляді лінійного тренду [39], або комбінації лінійного тренду та гармонійних коливань [84]. Дана процедура розглядається в розділі 9 даної роботи.

8.2 Стани в складних ланцюгах Маркова та підходи до їх визначення.

Як зазначено вище, стани дискретного ланцюга Маркова визначаються величинами абсолютних або відносних приростів величини, що прогнозується. Пропонується декілька алгоритмів класифікації відхилень на стани. Алгоритми класифікації виділяють межеві прирости для s класів, чим визначають, до якого класу відноситься той чи інший приріст у вибірці навчання. Розглянемо алгоритми класифікації:

1. Розподіл, рівномірний за кількістю представників кожного класу;
2. Розподіл, рівномірний за величиною відхилень;

3. Комбінований розподіл (крайні лівий та правий стан визначається рівномірними за кількістю, усі інші - рівномірні за відхиленням)
4. Комбінований розподіл при можливості регулювати кількість представників крайніх станів.

Розподіл, рівномірний за кількістю представників, проводиться за наступною схемою. Обчислюємо прирости ряду з даною дискретизацією, упорядковуємо їх та ділимо відсортований масив на рівні частини. Таким чином, ми досягаємо рівномірності по кількості представників в станах. Проблема може створити велика кількість однакових станів, що спричинює однакові межі декількох сусідніх станів. Це створює номери станів без жодного представника, що робить необхідним корекцію розбиття з ціллю досягнути найбільш можливої рівномірності розподілу частот представників станів.

Другий спосіб розбиття на стани полягає в тому, щоб розбити інтервал від мінімального (від'ємного) до максимального відхилення на рівні по відхиленню частини. В цьому випадку рівномірність по кількості представників в станах не зберігається. Це далеко не всі можливі способи розбиття.

Для кожного стану вибирається середнє значення, яке буде використовуватись при відновленні ряду по прогнозованим станам.

Оскільки реальна причинно-наслідкова залежність в процесі, який представлений динамічним рядом, невідома, то пропонується декілька способів розбиття на стани, при яких, по-перше, переходи між станами були представлені достатньою кількістю прецедентів, а по-друге, усереднення від-

хилень всередині станів не вплинуло б суттєво на точність одержаного прогнозу.

Для перевірки ефективності розбиття пропонується провести процедуру дискретизації та класифікації приростів на стани на кожному рівні Δt ієрархії приростів, а потім по відомим станам для кожного Δt відновити значення ряду та провести процедуру склеювання. Оскільки ряди станів повністю відповідають вихідному ряду, то отримується крива, відхилення якої від вихідної спричинені лише похибкою усереднення всередині станів (похибка квантування). Таким чином, прийнявши певне значення кількості станів s , та провівши процедуру дискретизації, відновлення та склеювання (без прогнозування), одержуємо абсолютну похибку дискретизації (квантування).

Розглянемо процес дискретизації та відновлення дискретизованих значень функції $y = \sin(x)$ для $s = 3, 6, 9, 12$ у випадку простої та складної ієрархії.

На рис. 8.1-8.4 видно, що складна ієрархія дає більш точне відновлення, так як більш густа сітка дискретизації часу виправляє похибки квантування, які виникають на інших рівнях Δt ієрархії приростів часу. Також зі збільшенням кількості станів точність відновлення збільшується, але необхідно мати на увазі, що вибір кількості рівнів квантування обмежений тим, що для прогнозування необхідне визначення ймовірностей переходів з достатньою точністю, для чого необхідна достатня кількість переходів між виокремленими станами.

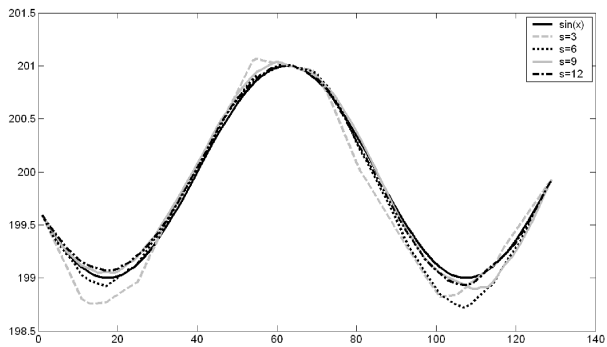


Рис. 8.1: Дискретизація та відновлення $y = \sin(x)$. Проста ієрархія.

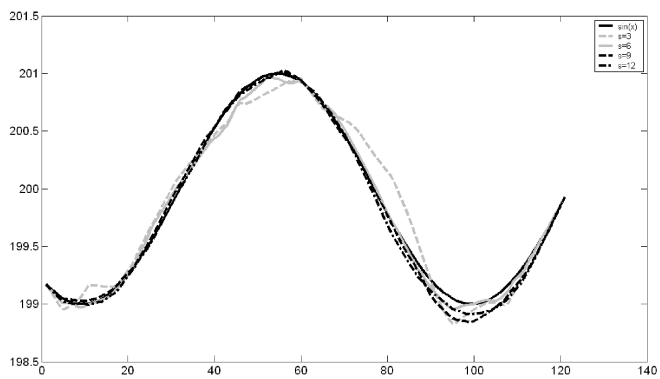


Рис. 8.2: Дискретизація та відновлення $y = \sin(x)$. Складна ієрархія.

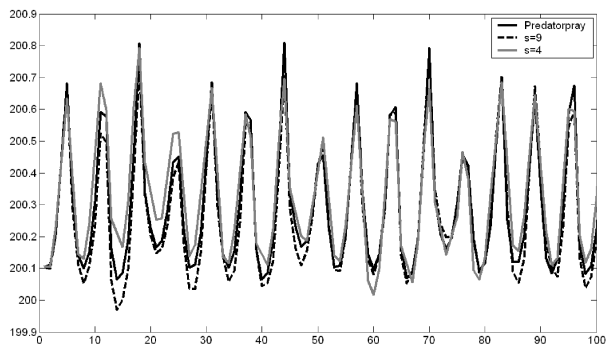


Рис. 8.3: Дискретизація та відновлення для моделі Хижак-жертва. Проста ієрархія.

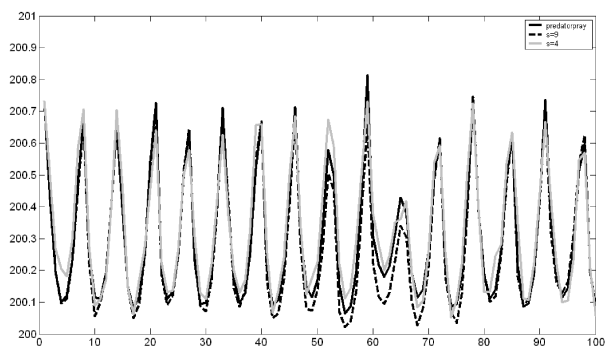


Рис. 8.4: Дискретизація та відновлення для моделі Хижак-жертва. Складна ієрархія.

8.3 Алгоритм побудови багатокрокового прогнозу

Розглянемо послідовність операцій, необхідних для побудови прогнозу вихідного ряду. Для цього необхідно задати наступні параметри:

1) Вид ієрархії приростів часу (проста – степені двійки, складна – добуток степенів перших простих чисел).

2) Величини s – кількість станів та r – порядок ланцюга Маркова. Ці параметри можна зробити індивідуальними для кожного рівня дискретизації, знаходження оптимальних параметрів буде обговорюватись у підрозділі 8.6.

3) Величина порогу δ , та мінімальна кількість N_{min} .

Отже, алгоритм побудови прогнозу:

1. Визначити ієрархію приростів часу - послідовність величин Δt , максимальне з яких повинне відповідати довжині навчальної вибірки N .
2. Для кожного приросту часу Δt у порядку зростання приростів, провести прогнозування станів та відновити за цими станами ряд. Для цього:
 - (а) обчислити прирости з дискретизацією Δt ;
 - (б) перевести ряд приростів у ряд номерів станів $\{1 \dots s\}$;
 - (с) обчислити ймовірності переходів для узагальнених станів;

- (d) побудувати ряд прогнозних станів за допомогою застосування процедури визначення найбільш ймовірного наступного стану;
 - (e) відновити ряд прогнозних значень з ряду подій з дискретизацією Δt ;
 - (f) провести процедуру склеювання результату з рядом, який отримався в результаті склеювання попередніх шарів (з меншим кроком Δt). У випадку, якщо даний ряд є першим, повернути його без змін у якості результату склеювання.
3. Останній склеєний ряд склеїти з продовженням лінійного тренда, побудованого по усім попередньо відомим точкам.

Ряд, склеєний з лінійним трендом, є результатом прогнозування.

8.4 Складні ланцюги Маркова та вплив передісторії (пам'яті). Порядок ланцюга Маркова.

У випадку простого ланцюга Маркова на основі ряду станів обчислюються ймовірності переходів з кожного стану у всі інші. Ці ймовірності прийнято записувати у вигляді квадратної матриці розмірністю s . Наприклад, звязавши рядки матриці зі станами, з якого відбувається перехід, а стовпці

(елементи цих рядків) – зі станами, у які визначається ймовірності переходу.

Наприклад, для трьох станів можна записати матрицю ймовірностей переходів A

$$A = \begin{pmatrix} 0,0967 & 0,0462 & 0,0366 & 0,0648 & 0,113 \\ 0,1628 & 0,1354 & 0,1317 & 0,1865 & 0,1477 \\ 0,4765 & 0,5621 & 0,6658 & 0,5639 & 0,4521 \\ 0,1612 & 0,1937 & 0,133 & 0,1386 & 0,1799 \\ 0,1048 & 0,0627 & 0,033 & 0,0462 & 0,1073 \end{pmatrix}$$

в якій, наприклад $A(1, 2) = 0,0366$ – ймовірність переходу із стану 1 в стан 2.

Якщо відомий вектор $p_t = (1, 0, 0, 0, 0)$, ймовірностей виникнення станів у поточний момент t то, згідно теоремі Байєса, вектор ймовірностей наступного стану обчислюється за формулою

$$p_{t+1} = Ap_t,$$

а вектор ймовірностей станів через k кроків обчислюється за формулою

$$p_{t+k} = A^k p_t,$$

У випадку складного ланцюга Маркова, ймовірність наступного стану залежить не тільки від попереднього стану, а й від послідовності r станів, які відбулись перед даним. У цьому випадку, необхідно обчислити ймовірності переходів з

послідовності r станів у стан $r+1$. Формально ці ймовірності можна було б записати в прямокутну таблицю з розмірністю (r^s, s) .

Можна звести ланцюги Маркова порядку r до ланцюга порядку 1, узагальнивши поняття “наявний стан”, включивши у нього послідовність з r станів, які передують стану, ймовірність появи якого обчислюється. Таким чином, ймовірності переходів можна записати в квадратну матрицю розмірністю (r^s, r^s)

Наведемо приклад матриці ймовірностей переходів A для ланцюга Маркова порядку $r = 2$ та кількості станів $s = 3$. Відповідність послідовностей станів введеним узагальненим станам наступна:

Послідовність станів	Узагальнений стан
11	1
12	2
13	3
21	4
22	5
23	6
31	7
32	8
33	9

Тоді матриця A ймовірностей переходів для узагальнених станів матиме розмірність 9×9 , наприклад:

$$A = \begin{pmatrix} 0,22 & 0 & 0 & 0,19 & 0 & 0 & 0,26 & 0 & 0 \\ 0,49 & 0 & 0 & 0,6 & 0 & 0 & 0,52 & 0 & 0 \\ 0,29 & 0 & 0 & 0,21 & 0 & 0 & 0,22 & 0 & 0 \\ 0 & 0,19 & 0 & 0 & 0,15 & 0 & 0 & 0,24 & 0 \\ 0 & 0,27 & 0 & 0 & 0,7 & 0 & 0 & 0,57 & 0 \\ 0 & 0,23 & 0 & 0 & 0,15 & 0 & 0 & 0,19 & 0 \\ 0 & 0 & 0,21 & 0 & 0 & 0,22 & 0 & 0 & 0,29 \\ 0 & 0 & 0,52 & 0 & 0 & 0,60 & 0 & 0 & 0,46 \\ 0 & 0 & 0,27 & 0 & 0 & 0,18 & 0 & 0 & 0,25 \end{pmatrix}.$$

Процес прогнозування полягає в наступному: вибирається останній стан (у випадку ланцюга Маркова порядку $r > 1$ береться послідовність r останніх станів). Визначається ймовірність переходу з даного стану у всі можливі. Из усіх можливих наступних станів вибирається стан з максимальною ймовірністю. У випадку декількох станів з максимальною ймовірністю процес прийняття рішення описаний далі.

Вибраний найбільш ймовірний стан приймається, як наступний прогнозований стан та процедура повторюється з наступним (доданий як останній) станом. Таким чином, ми отримуємо ряд прогнозних станів для даної величини дискретизації Δt .

Далі по одержаному ряду станів та відомому початковому значенню u_N відбувається відновлення ряду для даної дискретизації Δt . При цьому кожний стан являє собою Δt точок ряду. З кожним станом зв'язаний середній приріст, який додається до значення останньої точки ряду та обчислюється наступна дискретизована точка. Проміжні точки заповнюються, як лінійна інтерполяція двох відомих сусідніх точок.

Процедура покрокового прогнозування. Визначення найбільш ймовірного стану на наступному кроці, сценарії про-

гнозу.

При прогнозуванні в якості наступного прогнозного стану вибирається стан з найбільшою ймовірністю при даних умовах. Для цього ми використовуємо матрицю ймовірностей переходів із стану в стан. При цьому необхідно враховувати, що ймовірності можуть обчислюватись тільки наближено. Для підвищення точності оцінювання ймовірностей необхідно збільшити об'єм вибірки можливих переходів між станами. Але дана вибірка обмежена конкретною реалізацією ряду, що прогнозується, тому пропонується ряд алгоритмічних правил, які дозволяють врахувати принципову наближеність оцінених ймовірностей переходів при прогнозуванні майбутніх станів дискретного ланцюга Маркова.

Розглянемо конкретний випадок розподілу ймовірностей наступного стану та проілюструємо алгоритм вибору прогнозного стану при заданих умовах.

Як видно з рисунку, для того, щоб не допустити втрати станів, ймовірності яких обчислені з похибкою (наприклад, стани 6 на рис.8.5), необхідно до стану з максимальною ймовірністю додати сусідні стани, ймовірність яких не відрізняється від максимального більше ніж на δ . Параметр δ очевидно залежить від похибки оцінювання ймовірностей, але його необхідно експериментально апробувати.

Якщо прийняти $\delta=0,01$ (рис.8.5), то до множини найбільш ймовірних станів увійдуть наступні: $\{2, 3, 6, 7\}$. Декілька сусідніх станів з максимальною ймовірністю будемо називати кластером. У даному випадку це стани 2 та 3. Якщо прийняти $\delta = 0,02$, тоді множина найбільш ймовірних станів буде $\{1, 2, 3, 4, 6, 7\}$. Для прогнозування необхідно вибра-

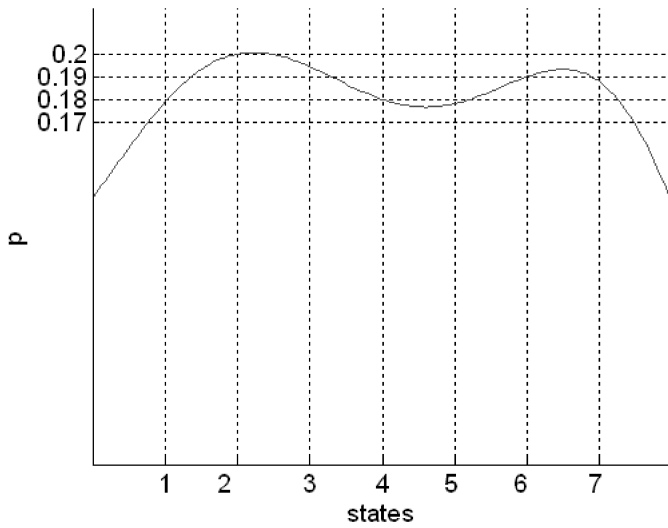


Рис. 8.5: Розподіл ймовірностей на певному кроці

ти один або два найбільш ймовірних стани. Вибираючи два стани з найбільш ймовірних, метод дозволяє розглянути два найбільш ймовірних сценарії можливого продовження ряду. Для вибору наступного стану пропонуються наступні правила:

1. Якщо рівні (квантовані прирости) з максимальною ймовірністю створюють декілька кластерів (кластер - група з декількох сусідніх рівнів – елементів кластера, мінімальний кластер – один ізольований рівень), то вибираємо найбільший за розміром кластер.
2. Якщо число елементів кластера непарне, то вибираємо центральний елемент у якості k_{max} .
3. Якщо число елементів в кластері парне, то розглядаємо два центральні елементи кластеру, та вибираємо в якості k_{max} той з них, який ближче до центру розподілу.
4. Якщо два центральних елементи кластера знаходяться на однаковій відстані від центру розподілу, то необхідно розглянути обидва, як можливі варіанти 1 та 2 значень k_{max} (точка біфуркації).
5. Якщо кластерів з максимальним розміром декілька, то розглядаємо їх, як нові елементи, які, в свою чергу, можуть формувати кластери, з якими необхідно вчинити аналогічно пунктам 1–4.

В цей принцип відбору закладені наступні міркування:

1. Якщо існують два однакових сусідні стани з максимальною ймовірністю, то краще взяти той, який ближче до центра розподілу, щоб звести до мінімуму ризик відображення у прогнозі хибних трендів.
2. Якщо рівні з максимальною ймовірністю не сусідні (багатомодовий розподіл), то необхідно розглядувати як мінімум два варіанти, оскільки це може бути зв'язане з біфуркаціями, які теж хотілось не упустити з розгляду.
3. Реалізуючи прогноз по варіанту 1) (на усіх етапах ієрархії), отримується в певному наближенні нижня границя прогнозу, а реалізуючи прогноз за варіантом 2) – його верхня границя.

У прикладі розподілу (рис. 8.5) у випадку $\delta=0,02$ кластерами найбільш ймовірних станів будуть множини $\{1, 2, 3, 4\}$ та $\{6, 7\}$. Згідно правилам 1 та 3, рекомендується вибрати середній стан найбільшого (у нашому випадку, першого) кластеру - стан 3 (згідно правилу 4, із двох середніх 2 та 3 найближчий до центра розподілу - стану 4).

Якщо прийняти ($\delta=0,01$), кластерами найбільш ймовірних станів будуть множини $\{2, 3\}$ та $\{6, 7\}$. Якщо прийняти проміжок між станами 4 та 5, як центр розподілу (стан, що відповідає нульовому відхиленню, мав би знаходитись між станами 4 та 5), то відстані від центра розподілу до двох вибраних кластерів є однаковими. В такому випадку має місце бімодальний розподіл (обидва кластери є однаково ймовірними), тому для адекватного моделювання майбутніх сцена-

ріїв, необхідно розглянути 2 варіанти. Перший сценарій буде починатись зі стану 3, (згідно правилу 3 як найближчий до центра розподілу у кластері з парної кількості станів) а другий - зі стану 6 (за аналогічним правилом 3).

Отже, у випадку невизначеності майбутнього стану, обмежуємось розглядом одного чи двох варіантів (сценаріїв) прогнозу: один з варіантів є верхньою границею прогнозу, а другий – нижньою.

8.5 Ієрархія приростів часу та процедура склеювання

Прогнози часового ряду необхідно обчислювати з різними кроками дискретизації часу. Наприклад, аналогічно з дискретним перетворенням Фур'є, розглядаємо прирости величиною степенів двійки. Спочатку обчислюємо прирости, як різницю двох сусідніх значень ряду, потім здійснюємо розрідження ряду з кроками 2, 4, 8, 16 і т.д. Позначимо поточну різницю між вимірами у часі через Δt .

Для кожного Δt виконуємо перетворення ряду приростів у ряд станів, здійснюємо прогнозування майбутньої послідовності станів, потім відновлюємо ряд із заданою дискретизацією по спрогнозованому ряду станів.

Ряди, отримані при відновленні для різних Δt , проходять ітераційну процедуру склеювання, в результаті якої прогнози з більшим кроком Δt коригують значення ряду, отриманих при прогнозуванні з меншими значеннями інтервалу Δt .

8.5. ІЄРАРХІЯ ПРИРОСТІВ ЧАСУ ТА ПРОЦЕДУРА СКЛЕЮВАННЯ 139

Таким чином, вибирається ієрархія приростів, кожен з яких відповідає за свою частоту, на якій відбувається прогнозування та поєднання прогнозів на різних Δt в один ряд у процесі склеювання.

Суть процесу склеювання полягає в наступному. Процедура склеювання є ітераційною, ряд з кожною наступною (з більшим кроком) дискретизацією коригує, підтягуючи до своїх вузлових точок прогноз, отриманий після склеювання на попередніх, менших Δt . Перетворення, які виконуються в процесі склеювання, можна записати у вигляді наступних обчислень:

Нехай задана ієрархія відносних (квантованих) приростів: $y_{\Delta t_k}^k, y_{2\Delta t_k}^k, \dots, y_N^k$; $k = 1, 2, \dots, K_{max}$ для кроків дискретизації $\Delta t_1 = 1; \Delta t_2, \dots, \Delta t_{K_{max}}$ відповідно.

Нехай процедура склеювання проведена для сукупності кроків $\Delta t_1 = 1; \Delta t_2; \dots; \Delta t_{K_{max}}$ та отриманий ряд значень $X_0^k, X_1^k, \dots, X_N^k$ з інтервалом дискретизації Δt_k . Тоді продовження процедури склеювання та перехід до ряду значень $X_0^{k-1}, X_1^{k-1}, \dots, X_N^{k-1}$ відбувається за алгоритмом:

початкове значення:

$$x_0^{k+1} = x_0^k = x_0;$$

для проміжку $0 < t \leq \Delta t_{k+1}$:

$$\begin{aligned}
x_1^{k+1} &= x_0^{k+1} + (x_1^k - x_0^k) + 1 \cdot \frac{x_0^{k+1} y_{\Delta t_{k+1}}^{k+1} - (x_{\Delta t_{k+1}}^k - x_0^k)}{\Delta t_{k+1}}; \\
x_2^{k+1} &= x_0^{k+1} + (x_2^k - x_0^k) + 2 \cdot \frac{x_0^{k+1} y_{\Delta t_{k+1}}^{k+1} - (x_{\Delta t_{k+1}}^k - x_0^k)}{\Delta t_{k+1}}; \\
&\dots \\
x_{\Delta t_{k+1}-1}^{k+1} &= x_0^{k+1} + (x_{\Delta t_{k+1}-1}^k - x_0^k) + \\
&+ (\Delta t_{k+1} - 1) \cdot \frac{x_0^{k+1} y_{\Delta t_{k+1}}^{k+1} - (x_{\Delta t_{k+1}}^k - x_0^k)}{\Delta t_{k+1}}; \\
x_{\Delta t_{k+1}}^{k+1} &= x_0^{k+1} + (x_{\Delta t_{k+1}}^k - x_0^k) + \Delta t_{k+1} \cdot \frac{x_0^{k+1} y_{\Delta t_{k+1}}^{k+1} - (x_{\Delta t_{k+1}}^k - x_0^k)}{\Delta t_{k+1}} \equiv \\
&\equiv x_0^{k+1} \left(1 + y_{\Delta t_{k+1}}^{k+1} \right);
\end{aligned}$$

для проміжку $\Delta t_{k+1} < t \leq 2\Delta t_{k+1}$:

$$\begin{aligned}
x_{\Delta t_{k+1}+1}^{k+1} &= x_{\Delta t_{k+1}}^{k+1} + \left(x_{\Delta t_{k+1}+1}^k - x_{\Delta t_{k+1}}^k \right) + \\
&+ 1 \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2 \cdot \Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}}; \\
x_{\Delta t_{k+1}+2}^{k+1} &= x_{\Delta t_{k+1}}^{k+1} + \left(x_{\Delta t_{k+1}+2}^k - x_{\Delta t_{k+1}}^k \right) + \\
&+ 2 \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2 \cdot \Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}}; \\
&\dots \\
x_{2\Delta t_{k+1}-1}^{k+1} &= x_{\Delta t_{k+1}}^{k+1} + \left(x_{2\Delta t_{k+1}-1}^k - x_{\Delta t_{k+1}}^k \right) + \dots \\
&+ (\Delta t_{k+1} - 1) \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2 \cdot \Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}}; \\
x_{2\Delta t_{k+1}}^{k+1} &= x_{\Delta t_{k+1}}^{k+1} + \left(x_{2\Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k \right) + \\
&+ \Delta t_{k+1} \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2 \cdot \Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}} \equiv \\
&\equiv x_{\Delta t_{k+1}}^{k+1} \left(1 + y_{2\Delta t_{k+1}}^{k+1} \right);
\end{aligned}$$

для останнього проміжку $N - \Delta t_{k+1} < t \leq N$:

$$\begin{aligned}
x_{N-\Delta t_{k+1}+1}^{k+1} &= x_{N-\Delta t_{k+1}}^{k+1} + \left(x_{N-\Delta t_{k+1}+1}^k - x_{N-\Delta t_{k+1}}^k \right) + \\
&+ 1 \cdot \frac{x_{N-\Delta t_{k+1}}^{k+1} y_N^{k+1} - \left(x_N^k - x_{N-\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}}; \\
x_{N-\Delta t_{k+1}+2}^{k+1} &= x_{N-\Delta t_{k+1}}^{k+1} + \left(x_{N-\Delta t_{k+1}+2}^k - x_{N-\Delta t_{k+1}}^k \right) + \\
&+ 2 \cdot \frac{x_{N-\Delta t_{k+1}}^{k+1} y_N^{k+1} - \left(x_N^k - x_{N-\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}}; \\
&\dots \\
x_{N-1}^{k+1} &= x_{N-\Delta t_{k+1}}^{k+1} + \left(x_{N-1}^k - x_{N-\Delta t_{k+1}}^k \right) + \\
&+ (\Delta t_{k+1} - 1) \cdot \frac{x_{N-\Delta t_{k+1}}^{k+1} y_N^{k+1} - \left(x_N^k - x_{N-\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}}; \\
x_N^{k+1} &= x_{N-\Delta t_{k+1}}^{k+1} + \left(x_N^k - x_{N-\Delta t_{k+1}}^k \right) + \\
&+ \Delta t_{k+1} \cdot \frac{x_{N-\Delta t_{k+1}}^{k+1} y_N^{k+1} - \left(x_N^k - x_{N-\Delta t_{k+1}}^k \right)}{\Delta t_{k+1}} \equiv \\
&\equiv x_{N-\Delta t_{k+1}}^{k+1} (1 + y_N^{k+1}).
\end{aligned}$$

Цю процедуру проводимо для $k = 1, 2, 3, \dots, K_{\max}$, починаючи з $k = 1$. Початковий ряд (при $k = 1$ для $\Delta t_1 = 1$) $x_0^1, x_1^1, \dots, x_N^1$ знаходимо за алгоритмом:

$$\begin{aligned}
x_0^1 &= x_0; \\
x_1^1 &= x_0^1 (1 + y_1^1); \\
x_2^1 &= x_1^1 (1 + y_2^1); \\
&\dots \\
x_N^1 &= x_{N-1}^1 (1 + y_N^1).
\end{aligned}$$

Розглянемо процедуру склеювання часового ряду на прикладі фрагменту індексу PFTS:

8.5. ІЄРАРХІЯ ПРИРОСТІВ ЧАСУ ТА ПРОЦЕДУРА СКЛЕЮВАННЯ 143

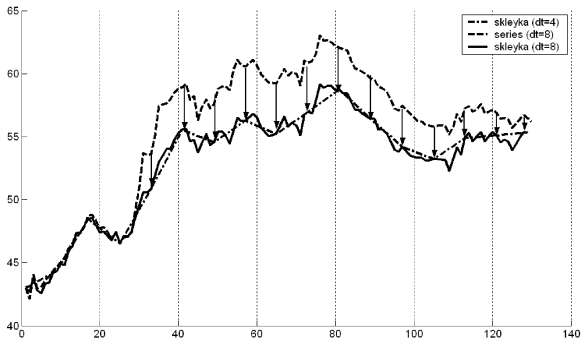


Рис. 8.6: Процес склеювання двох ієрархій часового ряду

На рис.8.6 зображено процес склеювання прогнозів з інтервалами дискретизації $\Delta t=1,2,4$ до нового ряду з дискретизацією $\Delta t = 8$. Як видно з рисунка, точки ряду, склеєного з $\Delta t=1,2,4$ (рис.1, пунктирна лінія) “підтягуються” до точок нового ряду з дискретизацією $\Delta t=8$ (рис. 1, штрихпунктирна лінія). В результаті отримується ряд, склеєний з $\Delta t 1,2,4,8$ (рис. 8.6, суцільна лінія). Як видно з рисунку, характерні риси прогнозу при $\Delta t = 4$, зберігаються після склеєння з прогнозним рядом при $\Delta t = 8$.

Ієрархія природів дискретизацій, як степені числа 2 – не єдиний спосіб побудови ієрархії дискретизацій. Для покращення прогнозу можна використовувати більш часту ієрархію дискретизацій, наприклад, добуток степенів простих чисел.

Висувається гіпотеза, що при більшій кількості прогнозів з різними проміжками дискретизації Δt , отриманий після склеювання прогноз буде більш точно відображати характер процесу, що прогнозується. Тому у якості альтернативної до множини значень $\Delta t_i = 2^i$, пропонується множина $\Delta t_i = p_1^{m_1} \cdot p_2^{m_2} \cdot \dots \cdot p_n^{m_n}$, де $\{p_k\}$ - підмножина множини простих чисел, $\{m_k\}$ - спеціально вибрана множина показників степенів. Множина простих чисел $\{p_k\}$ та відповідна множина їх степенів $\{m_k\}$ вибираються з розрахунку, щоб кількість різних значень Δt_i було максимальним. У цьому випадку, згідно припущення, можна покрити фрагмент часового ряду, взятий як навчальна вибірка, більш густою сіткою природів Δt_i , таким чином використавши більше інформації, наявної у часовому ряді.

Припустимо, що довжина ряду, який необхідно розкла-

8.5. ІЄРАРХІЯ ПРИРОСТІВ ЧАСУ ТА ПРОЦЕДУРА СКЛЕЮВАННЯ 145

сти на ієрархії – N . Взявши послідовність перших простих чисел p_1, p_2, \dots, p_n та послідовність їх степенів, при яких $p_1^{m_1} \cdot p_2^{m_2} \cdot \dots \cdot p_n^{m_n} = N_1 < N$. Число N_1 повинне бути якомога ближчим до N , щоб збільшити кількість точок, які використовуються при прогнозуванні. Прості числа і їх степені слід вибирати таким чином, щоб кількість дільників цього числа було максимальним. Це забезпечить максимальну щільність сітки точок дискретизацій. Оскільки N_1 є добутком заданих степенів простих чисел, то кількість дільників числа N_1 можна обчислити за наступною формулою: $(m_1+1) \cdot (m_2+1) \cdot \dots \cdot (m_n+1)$. Ця кількість приростів Δt_i повинне бути максимізоване.

Задача полягає в тому, щоб знайти послідовність простих чисел та відповідні їх степені, для яких по-перше, добуток $p_1^{m_1} \cdot p_2^{m_2} \cdot \dots \cdot p_n^{m_n} = N_1$ був би близьким до N , а по-друге – кількість дільників цього числа була б максимальною. Для розв'язання цієї задачі можна використовувати повний перебір варіантів, оскільки цей процес не займає значного обчислювального часу, а також проводиться лише один раз на початку прогнозування. Розглянемо приклад:

Для $N=5000$ необхідно підібрати такий набір простих чисел $2, 3, 5, 7, \dots, q_k$, при якому число кроків дискретизації буде близьким до максимально можливого. Максимальне k та відповідне йому q_k визначається умовою:

$$\frac{\ln N}{k \ln q_k} \approx 1; \quad \Rightarrow \quad k \approx \frac{\ln N}{\ln q_k}.$$

$$\begin{aligned}
 N &= 5000; & q_1 &= 2; & q_2 &= 3; & q_3 &= 5; & q_4 &= 7; & q_5 &= 11; \\
 k &= 3; & \frac{\ln 5000}{\ln 5} &= 5,29; & k &= 4; & \frac{\ln 5000}{\ln 7} &= 4,37; \\
 k &= 5; & \frac{\ln 5000}{\ln 11} &= 3,55.
 \end{aligned}$$

Вибираємо $k = 4$:

$$\begin{aligned}
 m_2 &= \frac{\ln 5000}{4 \ln 2} = 3,07; & m_3 &= \frac{\ln 5000}{4 \ln 3} = 1,94; & ; \\
 m_7 &= \frac{\ln 5000}{4 \ln 7} = 1,09; & \Rightarrow & \\
 2^3 3^2 5^1 7^1 &= 2520; & \Rightarrow & N = 2^4 3^2 5^1 7^1 = 5040; & \Rightarrow \\
 K_{\max} &= (m_2 + 1)(m_3 + 1)(m_5 + 1)(m_7 + 1) = 5 \cdot 3 \cdot 2 \cdot 2 = 60; & \Rightarrow \\
 \Delta t &= 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 15, 16, 18, 20, 21, 24, 28, 30, \\
 &35, 36, 40, 42, 45, 48, 56, 60, 63, 70, 72, 80, 84, 90, 112, 120, 126, \\
 &140, 144, 168, 180, 210, 240, 252, 280, 315, 336, 360, 420, 504, 560, \\
 &630, 720, 840, 1008, 1260, 1680, 2520, 5040.
 \end{aligned}$$

Таким чином, замість 12 приростів у випадку простої ієрархії ($2^{12}=4096$), маємо 60 приростів, що дозволить створити більш густу сітку часових приростів, та за висунутим припущенням, отримати більш точний прогноз.

8.6 Оптимізація алгоритму прогнозування, вибір оптимальних параметрів алгоритму.

Описаний вище алгоритм має параметри, які необхідно підбирати експериментально. Для цього проводиться процедура

оптимізації параметрів, проводячи тестові прогнози на відому частину ряду та оцінюючи результат.

Вихідний ряд розбивається на 2 частини, першу приймаємо як відому, а іншу як тестову. Будемо проводити тестові прогнози при різних параметрах та оцінимо відхилення прогнозу від реального значення.

Оскільки висунуто припущення, що для кожного приросту часу необхідно окремо підбирати параметри r та s , то пропонується наступний алгоритм оптимізації параметрів:

1. Будуємо перше наближення – лінійний тренд;
2. Для кожного приросту часу Δt (дискретизації) в порядку спадання, проводимо підбір оптимальних параметрів. Для цього:
 - (а) Для кожної пари параметрів (r, s) , при яких будуть перевірятись прогнози, та для декількох початкових точок виконуємо наступні дії:
 - i. Здійснюємо прогноз на даній ієрархії та здійснюємо склеювання для усіх раніше побудованих прогнозів з більшим кроком Δt , використовуючи вже підібрані на минулих кроках оптимальні r та s .
 - ii. Порівнюємо прогноз на величину Δt з реальними значеннями (тестовий проміжок). Обчислюємо похибку, як суму квадратів відхилень відповідних точок прогнозу та реальних точок.

- (b) Таку похибку обчислюємо для декількох прогнозів з різних точок. Знаходимо середнє значення похибок для даної пари r та s .
- (c) Із усіх пар r та s вибираємо ту, яка в середньому дає найменшу похибку.
- (d) Приймаємо кращу пару для даного кроку приросту часу Δt та переходимо до наступного рівню приросту в порядку спадання.

Таким чином, для кожного рівня приросту часу знаходимо оптимальні параметри, які будуть використовуватись в остаточному прогнозі.

Припускається, що для певних рядів існує закономірність в оптимальних параметрах. Для виявлення цієї закономірності проводиться експериментальна робота, мета якої - отримати оптимальний набір параметрів для ряду без повного перебору варіантів.

8.7 Прогнозування часових рядів з точним періодом (синусоїда)

При експериментах на рядах з точним періодом, наприклад, залежності $y(t)=\sin(at+b)$, оптимальний порядок ланцюга Маркова r прямує до максимально можливого, оскільки ряд станів містить абсолютно періодичні фрагменти. Але цей параметр обмежений тим, що для визначення ймовірностей переходів кожен елемент матриці в середньому повинен бути

8.7. ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ З ТОЧНИМ ПЕРІОДОМ (СИНУСОІДИ)

представлений хоча б декількома представниками. Для цього необхідно, щоб кількість елементів матриці була б одного порядку з розміром вибірки станів. Якщо припустити, що в середньому кількість станів на комірку матриці повинні бути k , розмір вибірки станів дорівнює N , тоді оптимальне співвідношення для параметрів r та s можна записати рівнянням:

$$k(r^{s+1}) = N$$

Якщо прийняти факт, що для рядів з точним періодом порядок ланцюга Маркова r повинен бути максимальним, тоді можна здійснити перебір тільки для параметрів s та k . Експерименти показують, що параметр k для синусоїд не повинен перевищувати кількості коливань, представлених у вибірці навчання.

Розглянемо прогнози синусоїди з різним періодом.

Неточності прогнозування, які видні на рис. 8.8 обумовлені похибкою квантування на Δt більших періоду синусоїди. На рис. 8.7 у вибірці навчання представлена достатня кількість періодів для відновлення на великих ієрархіях, а на рис. 8.8 цих представників недостатньо. Не зважаючи на це, перше коливання синусоїди з періодом 60 точок було спрогнозоване. З цього можна зробити висновок, що максимальний горизонт прогнозу може бути не більше 10-15 % від довжини навчальної вибірки.

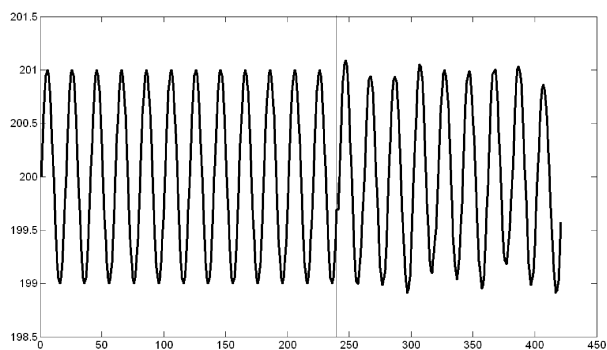


Рис. 8.7: Прогнозування $y = \sin\left(\frac{2\pi}{20}t\right)$ при $k=5$, $s=9$

8.7. ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ З ТОЧНИМ ПЕРІОДОМ (СИНУС)

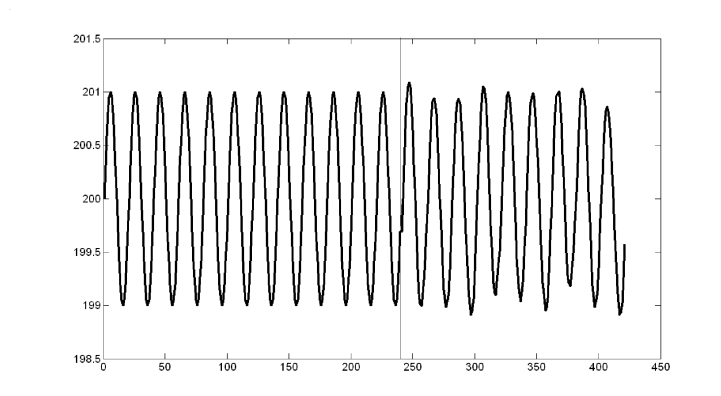


Рис. 8.8: Прогнозування $y = \sin\left(\frac{2\pi}{60}t\right)$ при $k = 2$, $s = 9$

8.8 Прогнозування рядів динамічного хаосу

На наступних рисунках зображений прогноз рядів динаміки моделі “Хижак-жертва”

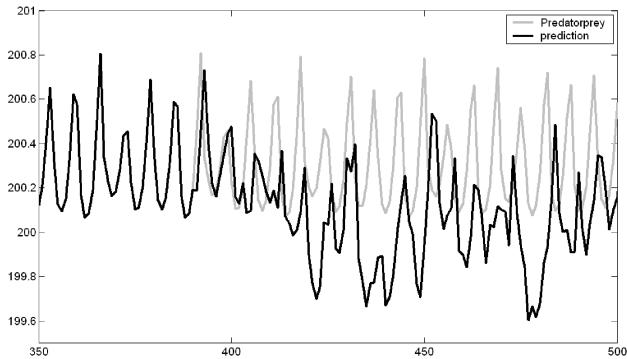


Рис. 8.9: Прогнозування ряду моделі “Хижак-жертва” $s=5$

На другому рисунку зображений прогноз при динамічному зростанні s зі зростанням Δt . Причиною такого експерименту стало те, що для великих приростів часу необхідна велика точність, при похибках на такому рівні виникають биття, які помітні на обох графіках. Очевидно, що в другому випадку вдалося покращити прогноз для перших прогно-

8.8. ПРОГНОЗУВАННЯ РЯДІВ ДИНАМІЧНОГО ХАОСУ 153

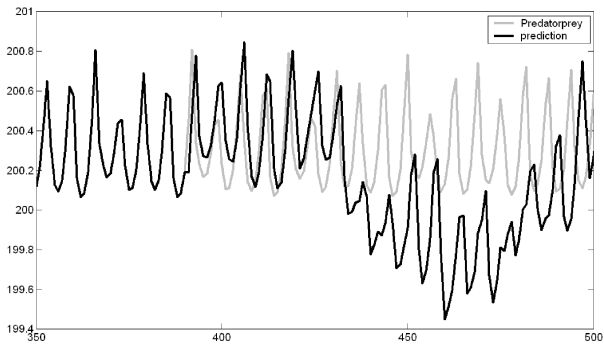


Рис. 8.10: Прогнозування ряду моделі “Хижак-жертва” $s = \sqrt{\Delta t}$

зних значень.

Експериментальна робота з рядами динамічного хаосу ще не закінчена, продовжується аналіз кривих, одержаних із різних моделей з різними параметрами. Результати, отримані на рядах динамічного хаосу, підтверджують гіпотезу про горизонт прогнозу 10%, хоча величина цього горизонту уточнюється.

8.9 Багатосценарний підхід до прогнозування рядів фінансово-економічної динаміки

У даному розділі будуть представлені результати прогнозування світових фондових індексів. Дані фондових індексів вільнодоступні на сайті <http://finance.yahoo.com>. Помічено, що при різній довжині навчальної вибірки, прогнозні криві дещо відрізняються (див. рис.8.11 та 8.12). Ряди прогнозів для індексу Dow Jones Industrial Average (далі DJI) зображено на рис.8.11 є більш корельовані між собою, ніж прогнози індексу FTSE 100, зображеного на рис. 8.12. На графіку b) справа вищезазначені графіки усереднено та відкладено одне середньоквадратичне відхилення вверх та вниз від середнього значення. Дані криві можуть інтерпретуватись як довірчі інтервали прогнозу. Час початку прогнозування зображено точкою 1000 та відповідає даті 1 грудня 2011 року.

Для порівняння індексів різних країн та зображення та-

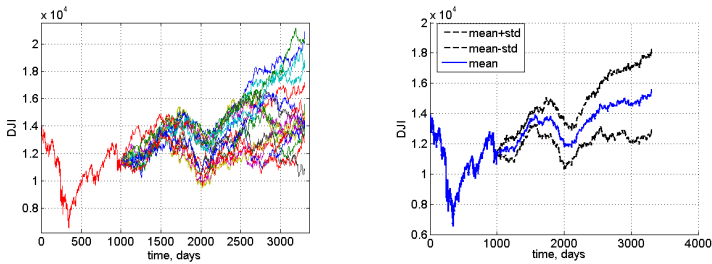


Рис. 8.11: Dow Jones Industrial Average - DJI (USA). а) Ряд прогнозів, обчислених з різною довжиною навчальної вибірки. б) Середнє значення та стандартне відхилення прогнозних кривих.

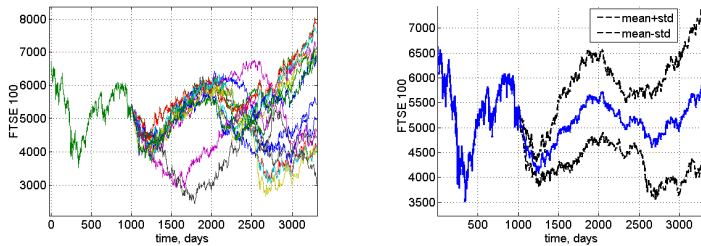


Рис. 8.12: Прогноз індексу FTSE 100. а) Ряд прогнозів, обчислених з різною довжиною навчальної вибірки. б) Середнє значення та стандартне відхилення прогнозних кривих.

ких прогнозів на одному графіку пропонується процедура нормалізації. Нормалізовані значення ряду будуть знаходитись у інтервалі $[0...1]$ та обчислюється за наступною формулою:

$$y_n(t) = \frac{y(t) - \min(y(t))}{\max(y(t)) - \min(y(t))}. \quad (8.1)$$

Нормалізовані прогнозні ряди показані на рис.8.13 (Американські ринки), рис.8.14 (Європа, ринки розвинутих країн), рис.8.15 (Європа, ринки країн PIIGS), fig.8.16 (азіацькі країни). Усі рисунки містять ряд усередненого прогнозу, які зображені чорною лінією та є середньозваженими з вагами ВВП відповідних країн.

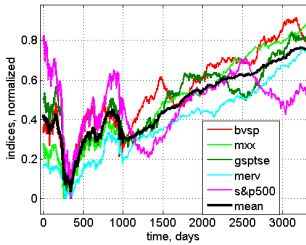


Рис. 8.13: Нормалізовані усереднені прогнози американських фондових індексів. Бразилія (BVSP), Мексика (MXX), Канада (GSPTSE), Аргентина (MERV), США (S&P 500)

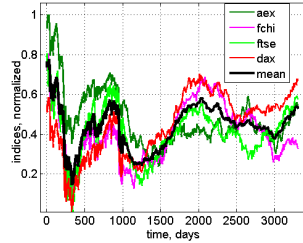


Рис. 8.14: Нормалізовані усереднені прогнози європейських фондових індексів. Розвинуті країни: FTSE (Великобританія), DAX (Німеччина) FCHI (Франція), AEX (Нідерланди).

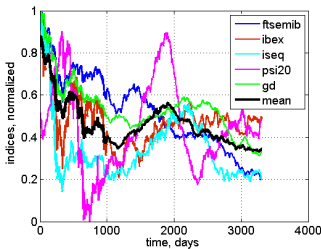


Рис. 8.15: Нормалізовані усереднені прогнози європейських фондових індексів. Португалія (PSI20), Італія (FTSEMIB), Ірландія (ISEQ), Греція (GD) та Іспанія (IBEX).

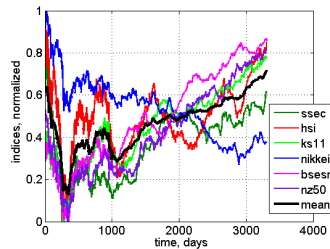


Рис. 8.16: Нормалізовані усереднені прогнози фондових індексів Азії. Китай (SSEC, HSI), Корея (KS11), Японія (NIKKEI), Індія (BSESN), Нова Зеландія (NZ50).

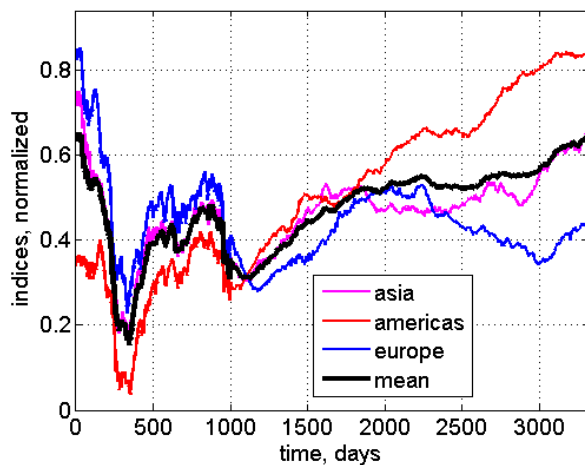


Рис. 8.17: Усереднені та нормалізовані прогнози індексів світових фондових ринків.

8.10 Лабораторна робота № 5. Прогнозування часових рядів за технологією складних ланцюгів Маркова

Дана лабораторна робота базується на використанні авторського програмного забезпечення MarkovChains, яке працює в системі Matlab. Найновішу версію програми можна завантажити на сайті <http://prognoz.ck.ua> або <http://kafek.at.ua>

Для запуску пакету для прогнозування часового ряду MarkovChains, необхідно відкрити Матлаб, перейти в робочу папку пакету MarkovChains та запустити пакет командою MarkovChains. Після запуску Ви побачите робоче вікно програми (рис. 8.18)

Програма приймає в якості вхідних даних текстові файли, які містять ряд чисел, розділених пробілом або символами кінця рядку. Завантажити в програму вхідний файл можна, натиснувши на кнопку "Обзор" в верхній частині вікна. При цьому вміст вхідного файлу буде завантажений в програму та з'являться значення в полях "Длина входного ряда", "Начало" "Конец", які задають відповідно початок і кінець відрізка, на якому буде вестись навчання. Переглянути графік проміжку навчання можна, натиснувши кнопку "Просмотреть график" (рис. 8.19).

Значення поля "Точка начала" задає точку, починаючи з якої здійснюється прогнозування. Ця точка задається відно-

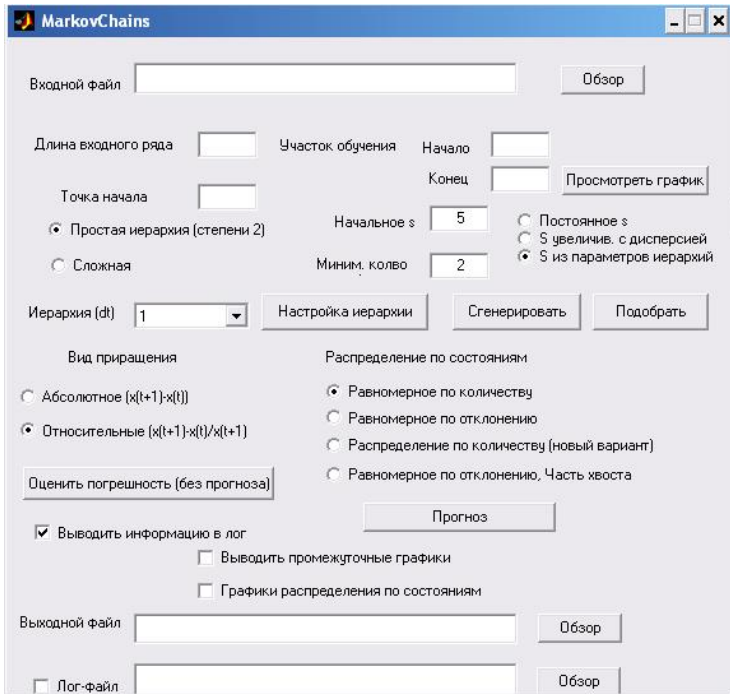


Рис. 8.18: Рабочее вікно програми MarkovChains

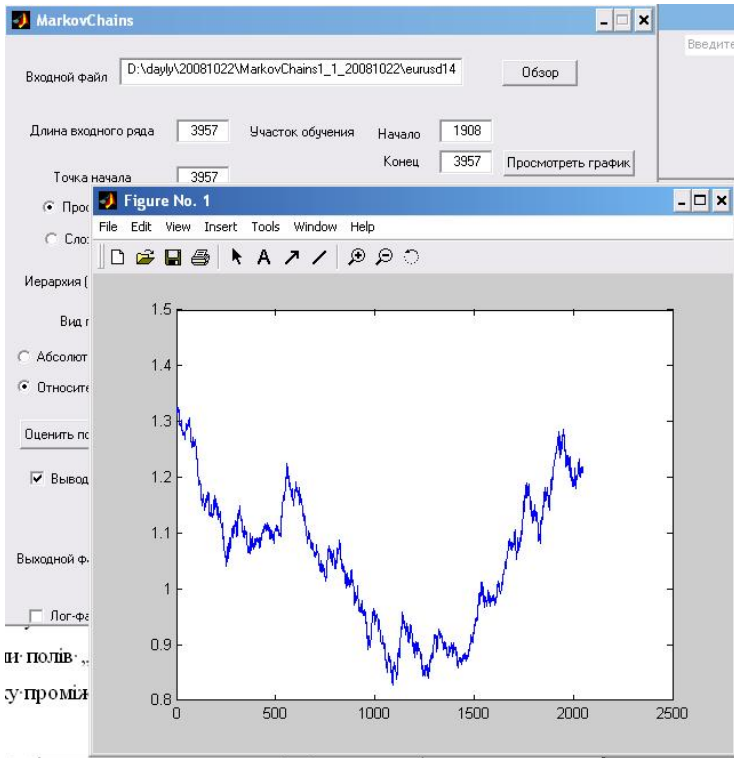


Рис. 8.19: Рабочее вікно програми MarkovChains

сно відрізка навчання, заданого значеннями полів "Начало" та "Кінець". Це значення краще всього визначити по графіку проміжку навчання.

Мережу приростів часу можна згенерувати 2-ма способами: проста ієрархія (степені числа 2) та складна (степені перших простих чисел). Спосіб генерації ієрархії приростів часу можна задати в головному вікні. Як відомо з теоретичного матеріалу, всі менші прирости dt повинні бути кратними найбільшому приросту dt . Найбільший приріст вибирається так, щоб його довжина була менша проміжку навчання, але максимально до нього наближалась (максимально використовувався вирізаний ряд). При зміні значень полів "Начало" и "Кінець" здійснюється генерація ієрархій та вирізування проміжку навчання, який стає рівним максимальному dt . При цьому значення поля "Кінець" не змінюється, а значення "Начало" збільшується, задаючи довжину проміжку навчання $N1 = dtmax$. Пам'ятайте, що значення кінцю проміжку незмінне, а значення початку проміжку може змінюватись, це необхідно враховувати при заданні проміжку навчання.

Під час автоматичної генерації ієрархії приростів часу кожному приросту dt задаються налаштування: кількість станів s , порядок ланцюгу Маркова r та мінімальна кількість повторень k . Ці параметри генеруються автоматично, але є можливість під коректувати їх для кожного приросту часу (кнопка "Налаштування ієрархії") або згенерувати з іншими значеннями (кнопка "Сгенерувати"). Процес налаштування параметрів кожного приросту буде описаний далі.

Як відомо з теоретичної частини, часовий ряд перетворюється на ряд приростів (різниць сусідніх значень). Ці різниці

можуть бути абсолютними або відносними. Вибір абсолютних або відносних різниць здійснюється у головному вікні програми за допомогою перемикачів "Абсолютные приращения" "Относительные приращения".

Також реалізовано 4 способи розподілення приростів на стани: рівномірний по станам (стани визначаються так, щоб кількість представників кожного стану були приблизно рівними), рівномірним по відхиленню (ділиться проміжок відхилень від максимального додатного до мінімального від'ємного на рівних s частин), "Равномерное по количеству (новый вариант)" – попереджує можливу ситуацію, коли в ряді приростів присутньо дуже багато представників одного значення (наприклад, 0 при незмінності ряду); "Равномерное по отклонению (часть хвоста)" – дозволяє відділити "хвости" розподілу в 2 окремих стани, а проміжні стани діляться за відхиленням. Більш детально про можливості розподілу ряду на стани див. теоретичну частину. Вибрати спосіб розділення на стани можна в головному вікні за допомогою перемикачів "Равномерные по состояниям", "Равномерные по количеству", та інші.

Для перегляду та редагування налаштувань кожного приросту часу необхідно скористатися кнопкою "Настройка иерархии" на головному вікні програми. Після її натискання, Ви побачите наступне вікно (рис. 8.20).

З лівого боку вікна знаходиться список всіх приростів, які згенеровані при даних налаштуваннях. Натискаючи на кожне значення (1, 2, 4, 8 і т.д.) в останній частині вікна зображуються параметри даного приросту: кількість станів s , порядок ланцюга Маркова r , мінімальна кількість повто-

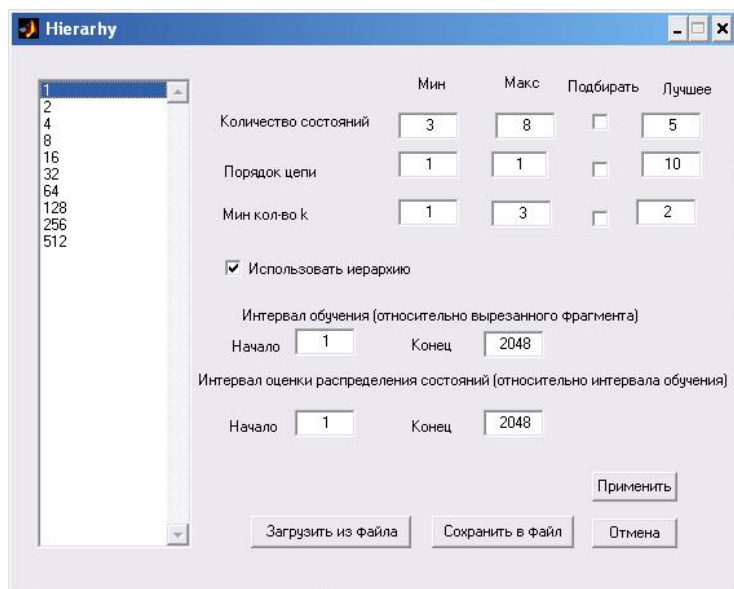


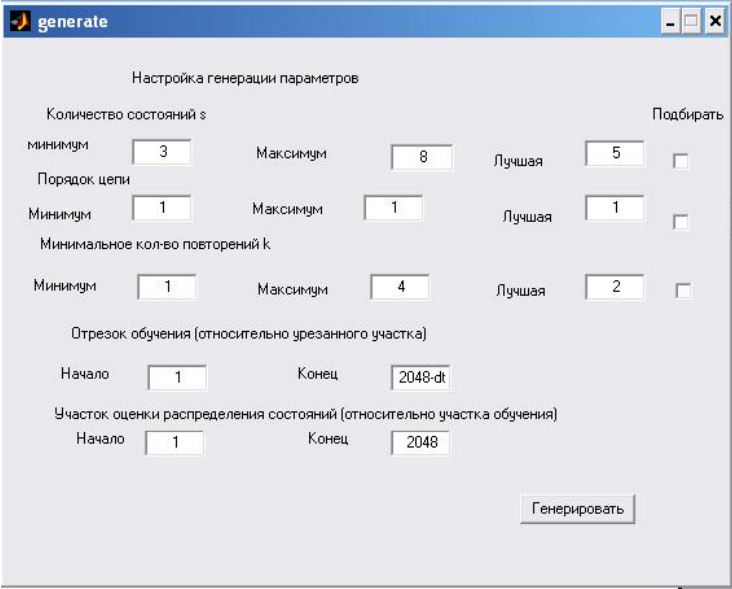
Рис. 8.20: Налаштування параметрів ієрархії приростів часу

рень k . Оптимальні значення цих параметрів можна підібрати, для цього необхідно відмітити галочку "Подобрать" навпроти відповідного параметру. Значення в полях введення "Макс" та "Мін" задають інтервал, в якому здійснюється підбір значень параметру. Найкраще значення – це значення, яке буде використовуватися під час прогнозування. Також для кожного приросту часу можна окремо задати проміжок навчання. Оскільки довжина ряду станів, в який перетворюється ряд при різних приростів часу, може відрізнятись, керуючи довжиною проміжку навчання для кожного приросту дає можливість уникнути так званих "крайових ефектів". Також за допомогою галочки "Использовать иерархию" можна виключити певні прирости часу з процесу прогнозування.

Кнопка "Применить" виконує зміни, які користувач задав у вікні для даного приросту часу. Якщо здійснити зміни якихось параметрів та не натиснути "Применить", то зміни проігноруються. На лаштування ієрархії можна зберегти в файл на диску комп'ютера за допомогою кнопки "Сохранить файл", або завантажити раніш збережені параметри за допомогою кнопки "Загрузить из файла".

По замовчуванню значення параметрів ієрархії беруться із полів "Начальное s " та "Миним. кол-во k ". Якщо Ви хочете згенерувати параметри, які відміні від тих, що по замовчуванні, то можна скористатись кнопкою "Сгенерировать". Після натискання Ви побачите вікно параметрів генерування, яке зображене на рис. 8.21.

В кожне поле введення необхідно задати значення параметру, яке буде застосоване для всіх приростів ієрархії.



The screenshot shows a window titled "generate" with a standard Windows-style title bar. The main content area is titled "Настройка генерации параметров" (Parameter Generation Settings). It contains several sections of input fields:

- Количество состояний s** (Number of states s): Includes "минимум" (minimum) set to 3, "Максимум" (maximum) set to 8, and "Лучшая" (best) set to 5. A "Подбирать" (optimize) checkbox is present.
- Порядок цепи** (Chain order): Includes "Минимум" (minimum) set to 1, "Максимум" (maximum) set to 1, and "Лучшая" (best) set to 1. A "Подбирать" (optimize) checkbox is present.
- Минимальное кол-во повторов k** (Minimum number of repetitions k): Includes "Минимум" (minimum) set to 1, "Максимум" (maximum) set to 4, and "Лучшая" (best) set to 2. A "Подбирать" (optimize) checkbox is present.
- Отрезок обучения (относительно урезанного участка)** (Training segment (relative to the truncated section)): Includes "Начало" (start) set to 1 and "Конец" (end) set to 2048- dt .
- Участок оценки распределения состояний (относительно участка обучения)** (Evaluation section (relative to the training section)): Includes "Начало" (start) set to 1 and "Конец" (end) set to 2048.

A "Генерировать" (Generate) button is located at the bottom right of the window.

Рис. 8.21: Генерация параметров иерархий приростів часу

Можна використовувати вирази з параметром dt , як, наприклад, кінець проміжку навчання. Значення цього параметру буде обчислене для кожного приросту та застосоване окреме значення. Після натискання кнопки "Генерировать" буде здійснення генерування нових параметрів. Для відміни генерації достатньо закрити вікно рис. 8.21.

Дослідження представлення кодованого ряду

Для перевірки адекватності представлення ряду в вигляді послідовності станів на різних ієрархіях приростів часу можна проробити наступні дії: закодувати ряд на всіх рівнях ієрархії та відновити їх значення по реальним послідовностям станів (без прогнозування) і проробити процедуру зшивання. Таким чином можна порівняти оригінальний і відновлений ряди і з цього судити про адекватність представлення ряду.

Для проведення вищеописаної процедури достатньо налаштувати ієрархію та натиснути кнопку "Оценить погрешность (без прогноза)". При цьому буде виведене вікно з двома графіками, один з яких – оригінальний, а інший – відновлений після кодування станами та виконання процедури зшивання. На рис. 8.22. зображений результат цієї процедури.

Результати виконання процедури виводяться текстовими повідомленнями на консоль Матлаба, який можна звичайним способом виділити і скопіювати в буфер обміну. На рис. 8.22. справа можна побачити фрагмент таких записів. Після процесу відновлення останнім рядком виводиться середнє значення квадратів відхилень кодованого ряду від оригінального. Це значення можна вважати точністю представ-

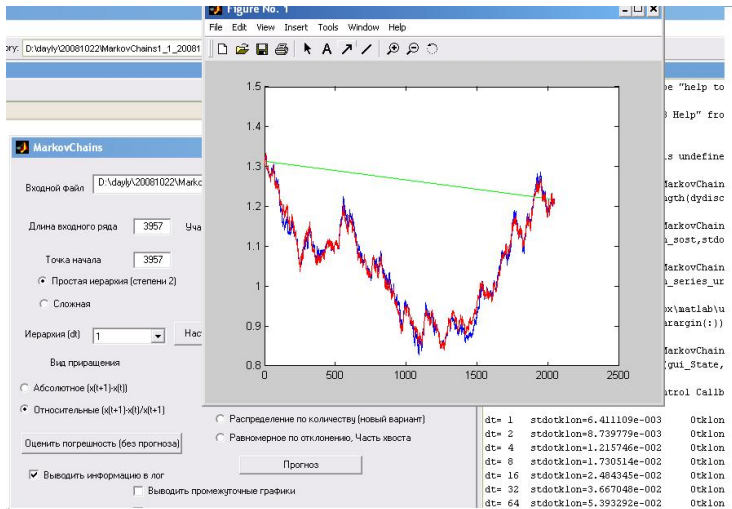


Рис. 8.22: Процедура проверки представления (без прогнозирования)

лення.

Пакет має можливість виводити текстовий лог не на консоль, а в будь-який файл. Для цього слугують нижнє поле введення та кнопка "Обзор", яка дає можливість вибрати файл для логу. Якщо файл не вибраний і вміст поля введення порожній (імя лог-файлу не задане) – то виведення інформації здійснюється на консоль.

В процесі як прогнозування, так і вищєрозглянутого тестування є можливість виводити проміжні графіки в процесі виконання зклеювання. Виведення цих графіків включається вибором опції "Выводить промєжуточные графики", яка знаходиться в нижній частині головного вікна.

Прогнозування

Для прогнозування достатньо задати параметри ієрархії (або залишити по замовчуванні) та натиснути відповідно кнопку "Прогноз" зліва внизу на головному вікні. Одразу ж почнеться процес прогнозування з менших приростів часу. Для контролю проходження прогнозування, на екран виводиться прогрес-бар, який показує процент здійснення прогнозування для даного приросту ієрархії часу. Якщо відкнута опція виводу проміжних графіків, то в кінці прогнозу кожної наступної ієрархії виводиться графік прогнозу даної ієрархії і результат зклеювання з попередніми ієрархіями.

8.11 Лабораторна робота № 6. Багатосценарне прогнозування методами складних ланцюгів Маркова

У даній лабораторній роботі буде розглянуто методику та технічні особливості побудови багатосценарних прогнозів засобами складних ланцюгів Маркова. У минулій лабораторній роботі розглядався візуальний інтерфейс програмного продукту `MarkovChains`. Цей продукт також має текстовий інтерфейс, який значно зручніше використовувати у випадках, які вимагають автоматизації або програмування.

Текстовий інтерфейс тулбоксу `MarkovChainsCmd` доступний за допомогою команди `MarkovChainsCmd`. Набір вхідних параметрів даної команди наступний:

`MarkovChainsCmd(infile, predictlength, outfile);`
`infile` - ім'я вхідного файлу, текстовий рядок. Може містити як ім'я файлу (у цьому випадку система `Matlab` намагатиметься відкрити файл у своїй поточній папці), або повний шлях до файлу (у цьому випадку система буде намагатися знайти файл по заданому шляху). Файл має бути текстовим та через символ `Enter` містити значення часового ряду у текстовому представленні. Може бути створений, збереженням листу `Excel`, який містить стовпець значень для прогнозування. Зауважимо, що вхідний текстовий файл не повинен містити ніякого зайвого тексту, окрім чисел ряду.

`predictlength` - довжина прогнозу, ціле додатне число. Бу-

де округлене до найближчого степеня числа 2. `outfile` - ім'я вихідного файлу. Має бути текстовим (*.txt). Після виконання команди даний файл буде містити значення навчальної вибірки та продовження (прогноз) як один ряд.

Також, окрім текстового вихідного файлу, програма створює `.mat`-файл, який зберігає усі проміжні розрахунки при прогнозуванні. Там збережено вид розподілу на стани, ряди прогнозів з різними Δt та інша інформація. Для детальної інформації можна розглянути вихідний код програми, вільно доступний за адресою http://prognoz.sk.ua/MarkovChains1_2_20111201_I_usually_use.rar.

Усі інші параметри, які можуть бути змінені у методі, задані значеннями, які дозволяють отримати найкращі прогнози. Якщо Ви бажаєте змінити певні параметри, можете прочитати вихідний файл `MarkovChainsCmd.m` (введіть в командному рядку `edit MarkovChainsCmd` і у редакторі відкриється потрібний файл). Доречі ідея варіювати іншими параметрами (не тільки довжиною вхідної вибірки), після чого розглядати і усереднювати отримані прогнозні ряди, є цікавою і частково нами перевірена.

Для генерації та виконання прогнозування з різною довжиною навчальної вибірки розроблена програма `MarkovChainsUsredn`, код якої доступний за адресою <http://prognoz.sk.ua/MarkovUsrednenie.rar>. Архів містить три файли:

1. `MarkovChainsUsredn.m` - програма для генерації завдань, виконання прогнозів та усереднення результатів.
2. `NormalizeFigs.m` - програма для нормалізації отриманих прогнозів до інтервалу $[0...1]$.

3. usred.m - програма для усереднення нормалізованих прогнозів індексів різних країн в один ряд.

Детальна інструкція по використанні даних програм буде наведена далі.

1. Для зручної роботи заведіть окрему папку, в яку запишіть 3 вищенаведені файли та скопіюйте ряд (або декілька рядів), які необхідно спрогнозувати.
2. Запустіть систему Matlab та перейдіть у підготовлену папку.
3. Введіть команду MarkovChainsUsred в командному рядку Matlab.
4. Після появи діалогового вікна відкриття файлу, укажіть ім'я файлу для прогнозування.
5. Введіть максимальну, мінімальну довжини навчальної вибірки (система підкаже довжину ряду для введення максимальної. Якщо Ви погоджуєтесь з цим вибором, можна просто натиснути Enter). Для мінімальної довжини та шляху також задані значення по замовчуванню. Вони підібрані для прогнозування денних значень рядів фондових ринків.
6. Після задання параметрів, програма створить у поточній папці файли-фрагменти часового ряду для прогнозування та файл commands_markov_predict.m, де записані виклики команди MarkovChainsCmd з відповідними вхідними параметрами. Після створення усіх

необхідних файлів, програма попередить про початок прогнозування і попросить натиснути Enter для початку.

7. Після обчислення прогнозів (фактично система запускає файл `commands_markov_predict.m` на виконання) запуститься усереднення отриманих прогнозів. Отримані зображення (графік з множиною отриманих прогнозів та графік усереднення) необхідно зберегти.

Зауважимо, що при кожному наступному запуску `MarkovChainsUsred`, будуть створені нові файли-фрагменти, а файл `commands_markov_predict` буде доповнений новими командами. Тому необхідно очищувати (видаляти) файл `commands_markov_predict.m` після обчислення прогнозів. Цією особливістю можна скористуватись, згенерувавши файли-фрагменти та завдання для усіх вхідних часових рядів, перериваючи запуск прогнозування (після запрошення почати обчислення прогнозу натиснути `CTRL+C`) та запуск генерації наступного завдання для іншого вхідного файлу. Важливо запам'ятати або записати максимальну та мінімальну довжини та крок для кожного ряду, який був заданий для генерації.

Після генерації усіх завдань, можна запустити команду `commands_markov_predict`, при цьому буде виконане прогнозування усіх завдань. Після цього необхідно додатково запустити процедуру усереднення. Це робиться тією ж програмою `MarkovChainsUsred`, але необхідно пропустити генерацію завдань. Для цього необхідно підредагувати файл `MarkovChainsUsred.m` (для початку редагування введіть в командному рядку `edit MarkovChainsUsred`), змінивши рядок № 51

"if(1)" на if(0). Після усереднення для генерації наступних завдань зміни необхідно повернути. Після редагування для кожного вхідного файлу запустить команду MarkovChainsUsred, вкажіть той же файл та ті ж максимальну, мінімальну довжини та крок, після чого з'являться графіки сценаріїв та усереднення, які необхідно зберегти.

Також в програмі MarkovChainsUsred реалізована можливість порівняти прогнози з реальним продовженням та побудувати графік залежності похибок від довжини навчальної вибірки. Для цього необхідно підготувати файл продовження часового ряду, що прогнозувався, записати його в робочу папку даних прогнозів та вказати його ім'я та значення початку прогнозу у файлі MarkovChainsUsred. Для цього редагуємо файл MarkovChainsUsred.m (для початку редагування введіть в командному рядку edit MarkovChainsUsred), в рядку 101 `realprodovzh=dlmread('file.txt');` замість 'file.txt' необхідно вказати ім'я підготовленого файлу продовження. У наступному рядку 102 `begpr=2470;` необхідно присвоїти у змінну `begpr` значення рядку (дня) початку прогнозу. Також у наступному рядку 103 може бути змінена довжина останнього фрагменту вибірки навчання `fragment2plot`, яка буде виведена на остаточні графіки. Часто цей параметр необхідно зменшувати, коли прогнозується ряд коротший ніж 1000. Після зміни файлу MarkovChainsUsred.m та запуску усереднення, з'явиться додатковий графік залежності похибок від довжини навчальної вибірки. Зауважемо, якщо вказати неіснуюче ім'я файлу, побудова графіку похибок просто буде пропущена.

8.11.1 Паралельно-розподілене обчислення прогнозів методами ланцюгів Маркова

При великій кількості рядів для прогнозування є необхідність організації паралельного та розподіленого розрахунку прогнозів. Для цього розроблено сервер паралельних розрахунків, доступний за адресою <http://prognoz.sk.ua>. Сервер реалізований на основі технологій CGI (php) та баз даних MySQL та дозволяє організувати розподіл завдань та паралельне обчислення прогнозів. Система має наступні функції:

1. Додавання пакету вхідних даних на сервер (файлообмінник для архівів вхідних рядів та результатів прогнозування)
2. Перегляд/скачування завантажених даних.
3. Менеджер завдань
4. Форма завантаження команд для розрахунку (кожна окрема команда закінчується символом нового рядка (Enter, n))

Крім сервера, необхідний розрахунковий клієнт, який працює під керівництвом Matlab та реалізує комунікацію з сервером та фактичне обчислення прогнозів.

Розглянемо приклад використання даної системи для паралельного обчислення усередненого прогнозу методами складних ланцюгів Маркова.

1. Згенерувати завдання. Запуск команди `MarkovChainsUsredn` з папки, у якій міститься вхідний файл. Після появи надпису: "Сейчас начнется прогнозирование" перервати виконання програми (комбінація клавіш `Ctrl+C`). Після генерації завдань, програма `MarkovChainsUsredn` згенерує набір вхідних файлів для прогнозування та послідовність команд `matlab` для запуску прогнозів (вміст файлу `Commands_markov_prognoz.m`).
2. Скачати клієнт парал. розрахунків. Його вміст (файли `parallel_workamashine.m` та `wwwread.m`) розархівувати у папку із згенерованими завданнями.
3. Всі завдання + файли `parallel_workamashine.m` та `wwwread.m` запакувати архіватором і відправити архів на сервер.
4. Відкрити вміст файлу `commands_markov_predict.m`, скопіювати команди, які у ньому записані, та вставити у форму додавання завдання. Дану форму також можна знайти з титульної сторінки проекту: 1) Сервер для ... -> 1) прогляд, -> 2) додати декілька завдань.
5. На усіх доступних для розрахунків комп'ютерах виконати наступні дії:
 - (a) Скачати тулбокс для прогнозування методами складних ланцюгів Маркова і прописати до нього шляхи;
 - (b) У списку завантажених файлів завдань скачати файл потрібного завдання.

- (с) Вибрати поточною папкою Matlab папку, куди розпаковано файли даних завдань.
6. Запустити команду `parallel_workamashine`. Її m-файл мав бути запакований у архіві завдань. Спостерігати за процесом обчислення можна з наступної сторінки
 7. Коли усі компютери порахують усі завдання даного пакета, на кожному компютері папка завдань запаковується (для зручності в імені вказується назва завдання та назва компютера) і відсилається на сайт.
 8. Користувач скачує усі архіви з порахованими прогнозами, розпаковує їх в одну папку, пропускаючи файли з однаковими іменами. 9. У папку, в якій знаходяться усі обчисленні завдання, копіюється файл програми `MarkovChainsUsredn.m` (його доречно запакувати разом зі завданнями), в ньому рядок 51 змінюється `if(1)` на `if(0)`, після змін, файл запускається на виконання (необхідно буде вказати ті ж параметри, що і при генерації завдань).
 9. На екран будуть виведені графіки усіх прогнозів з різною довжиною навчальної вибірки, а також графік середнього значення прогнозу та верхньої та нижньої границі. Отримані графіки необхідно зберегти, при необхідності відправити на сервер.

Використання вищеописаного серверу дозволяє значно пришвидшити розрахунки прогнозів, хоча і вимагає значної кількості неавтоматизованих дій. В подальшому ми працюємо

для більш автоматизованою версією зі зручним графічним інтерфейсом. Версію планується розмістити на тому ж сайті <http://prognoz.sk.ua>.

Розділ 9

Дискретне Фур'є-продовження низькочастотної складової рядів економічної динаміки.

Основна ідея даної глави - дослідження частотних характеристик коливань, наявних у динамічному ряді, побудова моделі, яка апроксимує найбільш виражені коливання та екстраполяція побудованої моделі у майбутнє.

Задача прогнозування часових рядів на основі частотних складових розглядалась у роботі [85], де запропоновано використання дискретного перетворення Фур'є. Перетворення Фур'є дає можливість дослідити амплітудно-частотну характеристику сигналу, що досліджується, але має певні недоліки, зв'язані з задачами прогнозування. Також спектр частот, кратних 2 не дає можливості порівняно невеликою кількістю доданків ряду Фур'є представити сигнал з частотою, не кратною 2 до кроку часової дискретизації ряду. Підхід вейвлет-аналізу, запропонований у роботі [86], дає можливість подолати першу з вищезазначених проблем, але залишається проблема неефективності, збитковості кодування частотних складових. В даній роботі пропонується підхід, який дозволяє подолати зазначені проблеми, описуючи сигнал достатньо малою кількістю гармонік (до 10) зі змінною фазою та здійснити прогноз ряду за допомогою екстраполяції аналітичного представлення його частотних характеристик.

Апробація запропонованої у попередніх розділах методики показала, що прогнозування динамічних рядів за цією методикою дає непогані результати для рядів із високою регулярністю, повторюваністю значень для короткострокових прогнозів. При довгостроковому прогнозуванні також слід враховувати низькочастотні коливання у вихідному ряді. Для подолання цієї проблеми, у якості процедури попередньої підготовки ряду, пропонується апроксимація вихідного ряду сумою декількох низькочастотних гармонічних функцій. Далі проводиться детрендування ряду, тобто обчислення залишків апроксимації та застосування методу складних ланцюгів Маркова до залишків апроксимації. Прогноз

залишків піддається оберненій процедурі відновлення ряду з використанням побудованого прогнозу залишків та екстраполяції гармонічної моделі тренду.

Метод побудови гармонічної моделі тренду може бути використаний як окремий метод прогнозування, його суть та результати застосування опубліковані у роботах [87, 27, 84, 88, 89].

Нехай ряд заданий послідовністю дискретних рівнів зі сталим кроком дискретизації часу Δt . Необхідно побудувати функцію $y(t)$, яка апроксимує задані емпіричні значення, оцінити параметри даної функції та експериментально перевірити ефективність прогнозів часового ряду на основі екстраполяції побудованої функції.

Пропонується низькочастотна апроксимуюча функція виду

$$y_{abs}(t) = a + bt + \sum_{i=1}^m c_i \sin(d_i t + e_i), \quad (9.1)$$

або для відносного масштабу:

$$y_{rel}(t) = ae^{bt} \prod_{i=1}^m c_i \sin(d_i t + e_i). \quad (9.2)$$

Параметри моделі $a, b, c_1, c_2, \dots, c_m, d_1, \dots, d_m, e_1, \dots, e_m$ мають мінімізувати наступний критерій оптимальності:

$$F(a, b, c_1, \dots, c_m, d_1, \dots, d_m, e_1, \dots, e_m) = \sum_{i=1}^n (y_i - y_{lin}(t_0 + i\Delta t))^2, \quad (9.3)$$

або для відносного варіанту:

$$F(a, b, c_1, \dots, c_m, d_1, \dots, d_m, e_1, \dots, e_m) = \sum_{i=1}^n \left(1 - \frac{y_i}{y_{lin}(t_0 + i\Delta t)}\right)^2. \quad (9.4)$$

При розв'язуванні задач оптимізації необхідно задати початкові оцінки параметрів, які оптимізуються, а також накласти обмеження на їх значення. Суттєвим обмеженням необхідно піддати частоту d_i . У випадку короткої навчальної вибірки можлива ситуація, коли найкраща апроксимація відповідає невеликій частині повного гармонічного коливання. При цьому продовження даної аналітичної кривої може бути не характерним для ряду, що досліджується. Це може бути проілюстровано рисунками 9.1 та 9.2. Тому необхідною має бути умова відповідності не менш ніж половині повного коливання протягом навчальної вибірки, тобто частота коливання не може бути нижчою величини $\frac{\pi}{N}$, де N – довжина вибірки навчання. Також обмеження має бути накладене на високі частоти, що пояснюється, по-перше, складністю оптимізаційної задачі для високих частот, зв'язаною з наявністю великої кількості хибних локальних мінімумів, а по-друге, можливою некоректністю такого наближення на високих частотах. Емпірично було вибрано обмеження 10 коливань за

час навчальної вибірки. Експерименти з вибором обмежень, які дозволяють отримати кращі прогнози, наведені у розділі ??.

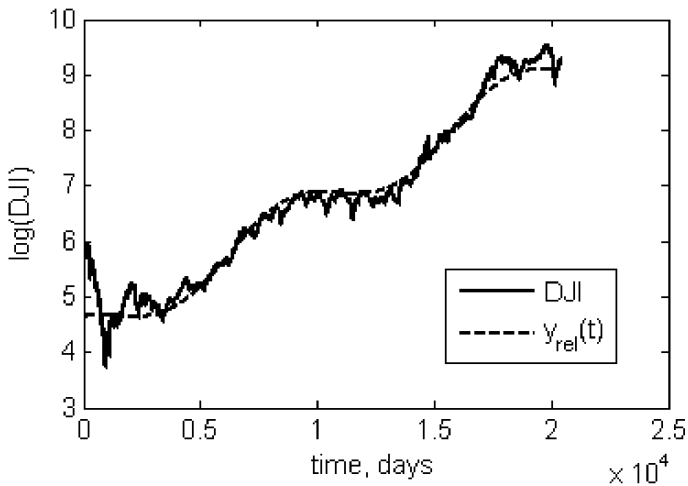


Рис. 9.1: Низькочастотна апроксимація індексу Dow Jones Industrial

Оскільки число параметрів функції F зростає зі збільшенням числа гармонік m , пропонується ітераційна апроксимація по одній (або двох) гармоніках, обчислення залишків та застосування наступної ітерації до більш високочастотних

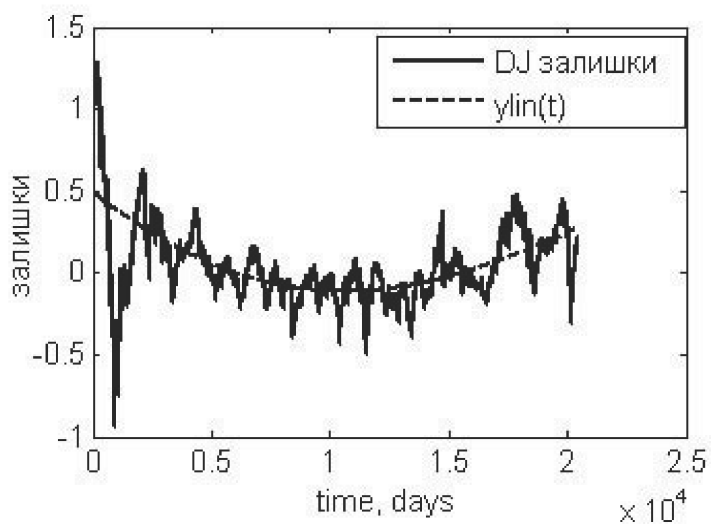


Рис. 9.2: Залишки апроксимації індексу Dow Jones Industrial та їх повторна апроксимація

залишків для одночастинного наближення:

$$r_i(t) = r_{i-1}(t) - (a + bt + c_i \sin(d_i t + e_i)), \quad (9.5)$$

або для m -частинного наближення:

$$r_i(t) = r_{i-1}(t) - \left(a + bt + \sum_{j=1}^m c_{ij} \sin(d_{ij} t + e_{ij}) \right). \quad (9.6)$$

Мінімізація нелінійного функціоналу нев'язки (9.3) або (9.4) пов'язана з труднощами, спричиненими існуванням декількох локальних мінімумів функції F у просторі значень параметрів. Для подолання цієї проблеми пропонується здійснювати оптимізацію з використанням декількох початкових оцінок значень параметрів. У якості початкових наближень для коефіцієнтів тренду вибираються коефіцієнти лінійного тренду, визначені за допомогою МНК.

Вибирається 2 початкових значення для фази: 0 та π радіан. Це дає можливість підкоректувати фазу апроксимуючої функції як в сторону збільшення, так і в сторону зменшення. Пропонується декілька початкових значень для частоти, вибраних із рівномірною сіткою в інтервалі між мінімальною та максимальною частотою. Емпірично вибрано кількість початкових значень частоти для низькочастотних коливань (перша ітерація, апроксимація безпосередньо вхідного ряду) $nf = 3$, для середньочастотних (друга та подальші ітерації) $nf = 5$.

9.1 Лабораторна робота № 6. Дискретне Фур'є-продовження.

Для побудови моделі дискретного Фур'є-продовження можна скористатись методикою, розглянутою у лабораторній роботі №2 “Методи оцінювання параметрів нелінійних трендових моделей”. Але у цьому випадку необхідно здійснювати перебір початкових наближень, який можна автоматизувати за допомогою програмування в системі Matlab. Нами розроблено тулбокс `Sintrend` для побудови даної моделі (вільнодоступний з <http://prognoz.ck.ua/SintrendAbsNew20120331.rar>. Розглянемо етапи її використання.

Інтерфейс програми є текстовим та реалізується через команду `sintrend_prognoz`. Параметри наступні:

`sintrend_prognoz(infile, outfile, niter, logmode, sinmode,`

`maxt, mint, nfreq);`

`infile` - ім'я вхідного файлу, рядок символів.

`outfile` - Ім'я вихідного файлу, рядок символів.

`niter` - Кількість ітерацій, ціле число

`logmode` - Необов'язковий параметр, перемикач режиму логарифмування перед апроксимацією. При значенні 1 перед апроксимацією вхідні значення замінюються на їх логарифми (відбувається перехід до логарифмічної шкали). При значенні 0 даного параметру, апроксимація застосовується безпосередньо для вхідного ряду. Якщо даний параметр не вказаний, логарифмування по замовчуванню вклучене.

`sinmode` - Необов'язковий параметр, перемикач режиму

9.1. ЛАБ.6. ДИСКРЕТНЕ ФУР'Є-ПРОДОВЖЕННЯ 187

одночастинного (значення 1) та двочастинного наближення (значення 2). Якщо даний параметр не задано, використовується одночастинне наближення.

`maxt` - Необов'язковий параметр, максимальний період коливань, дозволений при апроксимації. Використовується як обмеження та як відлік при переборі частот коливань. По замовчуванню задається як подвоєна довжина ряду.

`mint` - Необов'язковий параметр, мінімальний період коливань, дозволений при апроксимації. Використовується як обмеження та як відлік при переборі частот коливань. По замовчуванню задається як десята частина довжини ряду.

`nfreq` - кількість частот (густина сітки) при переборі початкових умов. Інтервал частот, який відповідає діапазону від вибраних мінімального до максимального періоду, ділиться на рівні `nfreq` частот. При ігноруванні даного параметру задається рівним 10.

Розглянемо приклади запуску функції прогнозування на основі дискретного Фур'є-продовження.

```
sintrend_prognoz;
```

Після запуску функції без жодного параметру, програма запитає у користувача вхідний файл (з'явиться діалогове вікно відкриття файлу, за допомогою якого можна вказати файл для відкриття. Можна для зручності скопіювати вхідний файл у робочу папку Matlab або змінити поточну папку Matlab, перейшовши у папку з вхідним файлом. Аналогічно програма запитає імя вихідного файлу. Якщо закрити вікно введення вихідного файлу, то вихідний файл буде згенерований, використовуючи імя вхідного файлу та дописавши `_SintrendPrognoz.txt` до нього. Кількість ітерацій при цьо-

му буде взята по замовчуванні - 5. Інші параметри - теж по замовчуванні.

```
sintrend_prognoz('dj.txt');
```

В даному прикладі задано ім'я вхідного файлу, а ім'я вихідного файлу буде запитане за допомогою діалогового вікна збереження файлу. Якщо закрити діалогове вікно, не вказавши вихідний файл, то ім'я вихідного файлу буде згенероване автоматично: до імені вхідного файлу буде дописано `_SintrendPrognoz.txt`. У даному прикладі це буде файл `dj_SintrendPrognoz.txt`. Кількість ітерацій - по замовчуванні 5, інші параметри - теж по замовчуванні.

```
sintrend_prognoz('dj.txt','dj_sin.txt');
```

У даному прикладі задано вхідний ('dj.txt') і вихідний файл ('dj_sin.txt'). Усі інші параметри буде взято по замовчуванню. Нагадую, що кількість ітерацій по замовчуванню 5, режим логарифмування вимкнтий, одночастинне наближення (одна гармоніка при підборі).

```
sintrend_prognoz('dj.txt','dj_sin.txt', 1);
```

У цьому випадку, крім імен вхідного та вихідного файлів, задана тільки 1 ітерація. Усі інші параметри - по замовчуванні.

```
sintrend_prognoz('dj.txt','dj_sin.txt', 1, 0, 1);
```

У цьому випадку задано 1 ітерація (3-й параметр =1), режим логарифмування вимкнений (4-й параметр = 0) , режим одночастинного наближення (5-й параметр = 1).

```
sintrend_prognoz('dj.txt','dj_sin.txt', 2, 1, 2, 15000,1000,4);
```

У даному прикладі задано кількість ітерацій 2 (3-й параметр =2), режим логарифмування ввімкнений (4-й параметр = 1), 2-частинне наближення (5-й параметр =2), максималь-

9.1. ЛАБ.6. ДИСКРЕТНЕ ФУР'Є-ПРОДОВЖЕННЯ 189

ний період коливань 15000, мінімальний період - 1000, кількість коливань при підборі - 4.

Після запуску команди, відбувається пошук оптимальних параметрів, викликаючи методи локальної оптимізації з різних початкових умов, заданих сіткою частот. Це може займати значний час, особливо при заданні великого значення pfreq , а особливо при включенні двочастинного наближення. Тому при виборі двочастинного наближення рекомендуємо конкретизувати діапазон періодів коливань для пошуку та зменшити кількість початкових частот для пошуку. Код програми оптимізований для обчислення у кластері (використовується паралельна версія циклу `for: parfor`). Тому при наявності налаштованого обчислювального кластера `matlab`, перебір початкових умов може бути розпаралелено. При виклику на локальній машині та ж програма працює в один потік без змін.

Після виконання оптимізації, на екрані з'являються графік залишків останньої ітерації, графіки кожної з гармонік, починаючи з низькочастотної та графік відновленої кривої, якщо використовувався режим логарифмування. Дані графіки можуть бути збережені користувачем, а можуть бути повторно побудовані з вихідних текстових файлів, куди були збережені обчислені прогнози.

Бібліографія

- [1] Бидюк, П. И. Вероятностное прогнозирование процессов ценообразования на фондовых рынках / П. И. Бидюк, А. В. Федоров // Системні дослідження та інформаційні технології. — 2009. — Т. 1, № 1. — С. 65–73.
- [2] Бідюк, П. І. Застосування сучасних методів прогнозування. — Електронний ресурс. — 2010. <http://sait.kpi.ua/eproc/2010/0/0304.pdf/view>.
- [3] Бідюк, П. І. Аналіз часових рядів (навчальний посібник) / П. І. Бідюк, В. Романенко, О. Тимощук. — Київ: Політехніка, 2010. — 317 с.
- [4] Єріна, А. М. Статистичне моделювання та прогнозування: навч. посібник / А. М. Єріна. — Київ: КНЕУ, 2001. — 170 с.

- [5] Присенко, Г. В. Прогнозування соціально-економічних процесів / Г. В. Присенко, Є. І. Равікович. — Київ: КНЕУ, 2005. — 378 с.
- [6] Снитюк, В. Є. Прогнозування. Моделі. Методи. Алгоритми: Навчальний посібник / В. Є. Снитюк. — Київ: Маклаут, 2008. — С. 364.
- [7] Осовский, С. Нейронные сети для обработки информации / Пер. с польск. Н.Д. Рудинского / С. Осовский. — Москва: Финансы и статистика, 2004. — 344 с.
- [8] Зайченко, Ю. П. Нечеткие модели и методы в интеллектуальных системах. Монография / Ю. П. Зайченко. — Киев: Изд Дом «Слово», 2008. — С. 344.
- [9] Марьясов, Д. А. АНАЛИЗ И ПРОГНОЗИРОВАНИЕ ФИНАНСОВОГО РЫНКА НА ОСНОВЕ МОДЕЛИ ДЕТЕРМИНИРОВАННОГО ХАОСА: Ph.D. thesis / Томский политехнический университет. — 2007.
- [10] Максишко, Н. Моделювання економіки методами дискретної нелінійної динаміки: Монографія / Наук. ред. проф. В.О. Перепелиця. / Н. Максишко. — Запоріжжя: Поліграф, 2009. — С. 416.
- [11] Алехин, Е. ОСНОВЫ АНАЛИЗА ВРЕМЕННЫХ РЯДОВ. Методические рекомендации студентам факультета экономики и управления ОГУ / Е. Алехин. — Орел, 2005.

- [12] Fama, E. F. The behaviour of stock market prices / E. F. Fama // Journal of Business. — 1965. — Vol. 38. — P. 34–105.
- [13] Fama, E. F. Efficient capital markets: A review of theory and empirical work / E. F. Fama // Journal of Finance. — 1970. — Vol. 25. — P. 383–417.
- [14] Грабовецкий, Б. Є. Основи економічного прогнозування, Навчальний посібник. / Б. Є. Грабовецкий. — Вінниця: ВФ ТАНГ, 2004. — 162 с.
- [15] Рассел, С. Искусственный интеллект: современный подход / С. Рассел, П. Норвиг; Под ред. . е изд.: Пер. с англ. — Москва: Издательский дом “Вильямс”, 2006. — С. 1408.
- [16] Ежов, А. А. НАУЧНАЯ СЕССИЯ МИФИ-2007. VIII ВСЕРОССИЙСКАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ «НЕЙРОИНФОРМАТИКА-2007»: ЛЕКЦИИ ПО НЕЙРОИНФОРМАТИКЕ. Часть 1 / А. А. Ежов. — МИФИ, 2007. — С. 11–51.
- [17] Синергетичні та еконофізичні методи дослідження динамічних та структурних характеристик економічних систем. Монографія / В. Дербенцев, О. Сердюк, В. Соловйов, О. Шарапов. — Черкаси: Брама-Україна, 2010. — С. 287.
- [18] Тобілевич, Ю. Агентно-орієнтоване моделювання / Ю. Тобілевич // Вісник Черкаського університету. — 2010. — Т. 173. — С. 79–89.

- [19] Krause, A. Evaluating the performance of adapting trading strategies with different memory lengths / A. Krause // ArXiv e-prints. — 2009.
- [20] Мерфи, Д. Межрыночный технический анализ / Д. Мерфи. — Москва: Диаграмма, 2002. — 317 с.
- [21] Швагер, Д. Технический анализ. Полный курс / Д. Швагер. — Москва: Альпина Бизнес Букс, 2005. — 806 с.
- [22] Удовенко, В. Форекс. Практика спекуляций на курсах валют / В. Удовенко. — Москва: ООО “И.Д. Вильямс”, 2007. — С. 384.
- [23] Колби, Р. Энциклопедия технических индикаторов / Р. Колби. — Москва: Альпина Бизнес Букс, 2004. — 837 с.
- [24] Лукашин, Ю. П. Адаптивные методы прогнозирования временных рядов: учеб. пособ. / Ю. П. Лукашин. — Москва: Финансы и статистика, 2003. — 416 с.
- [25] Грицюк, П. М. Аналіз, моделювання та прогнозування динаміки врожайності озимої пшениці в розрізі областей України : монографія / П. М. Грицюк. — Рівне: НУВГП, 2010. — 350 с.
- [26] Грицюк, П. М. До питання про циклічність урожайності зернових / П. М. Грицюк // Моделювання та інформаційні системи в економіці : зб. наук. праць / відп. ред. В.К. Галіцин. — 2008. — Т. 77. — С. 299–314.

- [27] Soloviev, V. Discrete Fourier forecasting of economic dynamic's time series / V. Soloviev, V. Saptsin, D. Chabankenko // Information Technologies, Management and Society. Theses of the International Conference. Information Technologies and Management, 2009 April 16-17. — Riga: Information System Institute, 2009. — P. 43–44.
- [28] Geisser, S. Predictive inference: an introduction / S. Geisser. Monographs on statistics and applied probability. — Chapman & Hall, 1993.
- [29] Голяндина, Н. Анализ и прогноз временных рядов: программная реализация метода “гусеница”. — Электронный ресурс. - Режим доступа: <http://www.gistatgroup.com/gus/programs.html>.
<http://www.gistatgroup.com/gus/programs.html>.
- [30] Голяндина, Н. Метод “Гусеница”–SSA: прогноз временных рядов / Н. Голяндина. — СПб.: ВВМ, 2004. — С. 52.
- [31] Александров, Ф. Автоматизация выделения трендовых и периодических составляющих временного ряда в рамках метода “гусеница”–SSA / Ф. Александров, Н. Голяндина // Exponenta Pro. — 2004. — Т. 3,4. — С. 54–61.
- [32] Данилов, Д. Л. Главные компоненты временных рядов: метод «Гусеница» / Д. Л. Данилов, А. А. Жиглявский. — СПб: Издательство Санкт-Петербургского университета, 1997. — С. 150. <http://www.gistatgroup.com/gus/book1/manual.html>.

- [33] Черняк, О. І. Динамічна економетрика [Текст] : навч. посіб. / О. І. Черняк, А. В. Ставицький. — Київ: КВІЦ, 2000. — С. 120.
- [34] Черняк, О. І. Навчально-методичний комплекс з дисципліни “Часові ряди” для студентів спеціальності “Економічна кібернетика” та “Прикладна економіка” / О. І. Черняк, А. В. Ставицький. — Київ: Видавничо-поліграфічний центр “Київський університет”, 2004. — С. 26.
- [35] Цыплаков, А. Эконометрический ликбез: прогнозирование временных рядов. / А. Цыплаков // Квантиль. — 2006. — Т. 1. — С. 1–60.
- [36] Бідюк, П. . Моделювання та прогнозування нелінійних динамічних процесів / П. . Бідюк; Ed. by ред. П. І. Бідюк. — Київ: ЕКМО, 2004. — Р. 120.
- [37] Бідюк, П. . Часові ряди: моделювання та прогнозування / П. . Бідюк. — Київ: ЕКМО, 2003. — С. 144.
- [38] Бідюк, П. . Методи прогнозування / П. . Бідюк, О. С. Меньяйленко, О. В. Половцев. — Луганськ: Луганський національний ун-т ім. Тараса Шевченка, 2008. — Т. 1. — С. 305.
- [39] Сапцин, В. М. Опыт применения генетически сложных цепей Маркова для нейросетевой технологии прогнозирования / В. М. Сапцин // Вісник Криворізького економічного інституту КНЕУ. — 2009. — Т. 2 (18). — С. 56–66.

- [40] Морозова, Т. Г. Прогнозирование и планирование в условиях рынка. Учебное пособие для вузов. Под ред. Т.Г. Морозовой, А.В. Пикулькина. / Т. Г. Морозова, А. В. Пикулькин, В. Ф. Тихонов. — Москва: ЮНИТИ-ДАНА, 1999. — 318 с.
- [41] Присняков, В. Ф. Нестационарная макроэкономика: — Учебное пособие. / В. Ф. Присняков. — Донецк: ДонНУ, 2000. — 209 с.
- [42] Сигел, Э. Практическая бизнес-статистика / Э. Сигел. — Москва: Издательский дом «Вильямс», 2002. — 1056 с.
- [43] Растринин, Л. А. Экстраполяционные методы проектирования и управления / Л. А. Растринин, Ю. П. Пономарев. — Москва: Машиностроение, 1986. — 120 с.
- [44] Сизов, Ю. С. Анализ портфелей ценных бумаг и управление ими в современной России. — Электронный ресурс. - Режим доступа: http://mirkin.eufn.ru/articles_1.htm\#sizov4.
- [45] Алтунин, А. Е. Модели и алгоритмы принятия решений в нечетких условиях / А. Е. Алтунин, М. В. Семухин. — Тюмень: Изд-во ТюмГУ, 2000. — 352 с.
- [46] Асаи, К. Прикладные нечеткие системы: Пер. с япон. / К. Асаи, Д. Ватада, С. Иваи. — Москва: Мир, 1993. — 368 с.

- [47] Сергеева, Л. Н. Клеточные сети с опосредованным взаимодействием в микроэкономическом моделировании / Л. Н. Сергеева // Искусственный интеллект. — 1999. — Т. 2 (специальный выпуск), № 2 (специальный выпуск). — С. 398–406.
- [48] Сергеева, Л. Н. Моделирование поведения экономических систем методами нелинейной динамики (теории хаоса) / Л. Н. Сергеева. — Запорожье: ЗГУ, 2002. — 227 с.
- [49] Clements, M. P. Forecasting economic and financial time-series with non-linear models / M. P. Clements, P. H. Franses, N. R. Swanson // Int. journal of forecasting. — 2004. — Vol. 20. — P. 169–183.
- [50] Billings, S. A. Dual - orthogonal radial function networks for nonlinear time series prediction / S. A. Billings, X. Hong // Neural Networks. — 1998. — Vol. 11, № 11. — P. 479–493.
- [51] Holland, J. The dynamics of searches directed by Genetic Algorithms / J. Holland. — Singapore: Word Scientific, 1988.
- [52] Галушкин, А. И. Теория нейронных сетей. Кн. 1: Учеб. Пособие для вузов / А. И. Галушкин. — Москва: ИПР-ЖР, 2001. — 385 с.
- [53] Головки, В. А. Нейронные сети: обучение, организация и применение. Кн.4: Учеб.пособие для вузов/Общая ред.

- А.И. Галушкина. / В. А. Головки. — Москва: ИПРЖР, 2001. — 256 с.
- [54] Розенблат, Ф. Принципы нейродинамики: Перцептрон и теория механизмов мозга. Пер. с англ. / Ф. Розенблат. — Москва: Мир, 1965.
- [55] Сигеру, О. Нейроуправление и его приложения. Пер. с англ. под ред. А.И. Галушкина / О. Сигеру. — Москва: ИПРЖР, 2001. — 321 с.
- [56] Poggio, T. Theory of networks for approximation and learning / T. Poggio, F. A. Girosi // A.I. Memo. — 1994. — Т. 1140, № 1140. — С. 63 р.
- [57] Вороновский, Г. К. Генетические алгоритмы, нейронные сети и проблемы виртуальной реальности / Г. К. Вороновский. — Харьков: Основа, 1997. — 112 с.
- [58] Барский, А. Б. Обучение нейросети методом трассировки / А. Б. Барский // Труды VIII Всероссийской конференции «Нейрокомпьютеры и их применение»: Сб. докл. — Москва: Век книги, 2002. — С. 862–898.
- [59] Белим, С. В. Математическое моделирование квантового нейрона / С. В. Белим // Труды VIII Всероссийской конференции «Нейрокомпьютеры и их применение»: Сб. докл. — Москва: Век книги, 2002. — С. 899–903.
- [60] Бутенко, А. А. Обучение нейронной сети при помощи алгоритма фильтра Калмана / А. А. Бутенко // Труды

- VIII Всероссийской конференции «Нейрокомпьютеры и их применение»: Сб. докл. — Москва: Век книги, 2002. — С. 1120–1125.
- [61] Гусак, А. Н. Подход к послыному обучению нейронной сети прямого распространения / А. Н. Гусак // Труды VIII Всероссийской конференции «Нейрокомпьютеры и их применение»: Сб. докл. — Москва: Век книги, 2002. — С. 931–933.
- [62] Шибхузov, З. М. Конструктивный tower алгоритм для обучения нейронных сетей из тп – нейронов / З. М. Шибхузov // Труды VIII Всероссийской конференции «Нейрокомпьютеры и их применение»: Сб. докл. — Москва: Век книги, 2002. — С. 1066–1072.
- [63] Тарасенко, Р. О. Метод аналізу і підвищення якості навчальних вибірок нейронних мереж для прогнозування часових рядів: Автореф. дис. канд. техн. наук : 05.13.06 / Одес. нац. політехн. ун-т. — Одеса, 2002. — С. 19.
- [64] Тихонов, Э. Е. Совершенствование методов прогнозирования с использованием нейронных сетей и системы остаточных классов: Дисс. к.т.н. / СГУ. — Ставрополь, 2004.
- [65] Широков, Р. В. Нейросетевые модели систем автоматического регулирования промышленных объектов: Дисс. к.т.н. / СГУ. — Ставрополь, 2003.

- [66] (ehips) ики ран. генетические алгоритмы. — Электронный ресурс. <http://www.iki.rssi.ru/ehips/Genetic.htm>.
- [67] Исаев, С. А. Генетические алгоритмы и машинное обучение. — Электронный ресурс. Режим доступу: http://www.blind.alfint.ru/modules.php?name=News\&new_topic=3.
- [68] Лекции по нейронным сетям и генетическим алгоритмам. — Электронный ресурс. <http://infoart.baku.az/inews/30000007.htm>.
- [69] Батищев, Д. И. Генетические алгоритмы решения экстремальных задач / Д. И. Батищев. — Воронеж: ВГУ, 1994. — 135 с.
- [70] Ивахненко, О. Г. Метод группового учета аргументів - конкурент методу стохастичної апроксимації / О. Г. Ивахненко // Автоматика. — 1968. — Т. 3, № 3. — С. 58–72.
- [71] Ивахненко, А. Г. Помехоустойчивость моделирования / А. Г. Ивахненко, В. С. Степашко. — Киев: «Наук. думка», 1985. — 216 с.
- [72] Ивахненко, А. Г. Моделирование сложных систем по экспериментальным данным / А. Г. Ивахненко, Ю. П. Юрачковский. — Москва: «Радио и связь», 1987. — 120 с.

- [73] Ивахненко, А. Долгосрочное прогнозирование и управление сложными системами / А. Ивахненко. — Москва: Техника, 1975. — С. 312.
- [74] Granger, C. W. J. Forecasting Economic Time Series / C. W. J. Granger, P. Newbold. — San Diego: Academic Press, 1986.
- [75] Zhang, Y. Prediction of Financial Time Series with Hidden Markov Models: Master of applied science: Simon Fraser University / School of Computer Science. — 2005. — P. 102.
- [76] Rabiner, L. R. A tutorial on hidden Markov models and selected applications in speech recognition / L. R. Rabiner // In Proceedings of IEEE. — 1989. — Vol. 77. — P. 257–286.
- [77] Rabiner, L. R. An introduction to hidden Markov models / L. R. Rabiner, B. H. Juang // IEEE ASSP Mag. — 1986. — Vol. June. — P. 4–16.
- [78] Shi, S. Taking time seriously: Hidden Markov experts applied to financial engineering / S. Shi, A. S. Weigend // In Proceedings of the IEEE/IAFE 1997 Conference on Computational Intelligence for Financial Engineering. — 1997. — P. 244–252.
- [79] Чабаненко, Д. М. Застосування прихованих марківських моделей для дослідження економічних систем / Д. М. Чабаненко, Н. С. Коровяцька // «ПРОБЛЕМИ ТРАНСФОРМАЦІЙНОЇ ЕКОНОМІКИ» Збірник наукових тез II Всеукраїнської науково-практичної конфе-

- ренції. / відповід. ред. к.т.н., доц. Берідзе Т.М. — Кривий Ріг: КФ ДВНЗ ЗНУ, 2009. — С. 84–85.
- [80] Романовский, В. И. Дискретные цепи Маркова / В. И. Романовский. — Москва: Гостехиздат, 1949. — С. 436.
- [81] Дынкин, Е. Марковские процессы / Е. Дынкин. — Москва: Физматгиз, 1963. — Р. 860.
- [82] Гупал, Н. Применение байесовской процедуры распознавания для прогнозирования динамики курсов акций / Н. Гупал, Ю. Матусов // Компьютерная математика. — 2009. — Т. 1. — С. 105–110.
- [83] Чабаненко, Д. М. Алгоритм прогнозування фінансових часових рядів на основі складних ланцюгів маркова / Д. М. Чабаненко // Вісник Черкаського університету. — 2009. — Т. 173, № 173. — С. 90–102.
- [84] Чабаненко, Д. М. Дискретне Фур'є-продовження низькочастотних складових рядів економічної динаміки / Д. М. Чабаненко // Системний аналіз та інформаційні технології: матеріали 12-ї Міжнародної науково-технічної конференції SAIT 2010. — Київ: ННК «ІПСА» НТУУ «КПІ», 2010. — С. 336.
- [85] Капранова, Л. Г. Прогноз експорту української металопродукції методом аналітичного вирівнювання за гармоніками ряду Фур'є / Л. Г. Капранова // Проблеммы развития внешнеэкономических связей и

привлечение иностранных инвестиций: региональный аспект. — 2010. — Vol. 1. — P. 240–243.

- [86] Анушина, Е. С. Система краткосрочного прогнозирования электрической нагрузки: Ph.D. thesis / Работа выполнена в Санкт-Петербургском государственном электротехническом университете “ЛЭТИ” им. В.И. Ульянова (Ленина). — Санкт-Петербург, 2009.
- [87] Chabanenko, D. Discrete Fourier forecasting of economic dynamic's time series / D. Chabanenko, O. Nechinenaуа // Information Technologies, Management and Society. Theses of the International Conference. — Riga: 2010. — P. 43–44.
- [88] Чабаненко, Д. М. Алгоритми дослідження та прогнозування низькочастотної складової динамічного ряду / Д. М. Чабаненко // Інформаційні технології в освіті, науці і техніці / Матеріали VIII Всеукраїнської конференції молодих науковців ІТОНТ-2010. — Черкаси: ЧДТУ, 2010.
- [89] Чабаненко, Д. М. Дискретне Фур'є-продовження часових рядів / Д. М. Чабаненко // Системні технології. Регіональний міжвузівський збірник наукових праць. — 2010. — Т. 1 (66), № 1 (66). — С. 114–121.