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

МЕТОДИ ПРОГНОЗУВАННЯ

Навчальний посібник

ЧЕРКАСИ, 2012

ББК 22.17

УДК 519.2

Ганчук А. А., Соловйов В. М., Чабаненко Д. М.
Методи прогнозування. Навч. посібник. – Черкаси: Брама-Україна, 2012. – 140 с.

Рецензенти: д.ф-м.н., проф. Стеблянко П. О.,
Дніпродзержинський державний технічний університет
д.п.н., проф. Триус Ю. В.,
Черкаський державний технологічний університет

У посібнику розглянуті як класичні методи прогнозування часових рядів, так і сучасні, основані на концепціях нелінійної динаміки, нейронних мереж, генетичних алгоритмів, методу групового урахування аргументів. Висвітлені авторські методи прогнозування рядів фінансово-економічної динаміки, такі як методика складних ланцюгів Маркова та дискретне Фур'є-продовження. Детальні інструкції у лабораторних роботах дозволять студентам освоїти усі наведені методи, реалізувати розглянуті моделі засобами табличного процесору Microsoft Excel або системи Matlab та використовувати у практичній діяльності.

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

Рекомендовано до друку рішенням Вченої ради Черкаського національного університету імені Богдана Хмельницького. Протокол № 7 від 17 квітня 2012 р.

Зміст

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 у її розв'язанні. На сьогодні відомо понад 200 різноманітних методів прогнозування [1–3], також з'являються нові підходи, що означає існування проблем, не розв'язаних на даний момент.

Читачу доступна значна кількість підручників [4–7] та монографій [8–11], присвячених прогнозуванню. Більшість підручників широко розглядають класичні методи та Microsoft Excel як популярне середовище побудови прогнозних моделей, але рідко містять сучасні методи прогнозування та інструкції щодо їх застосування. Наукові монографії розкривають передові досягнення сучасних методів, але далеко не завжди описані методи зручно і легко повторити самостійно навіть підготовленому старшокурснику або аспіранту. Тому ціллю даного посібника є огляд методів прогнозування, а також комп'ютерних та програмних засобів побудови і використання прогнозних моделей.

Подяка

Автори даної роботи висловлюють щиру вдячність Бахрушину Володимиру Євгеновичу, Бідюку Петру Івановичу, Грицюку Петру Михайловичу, Зайченку Юрію Петровичу, Кириченко Людмилі Олеговні, Савченко Євгенії Анатоліївні, Сапціну Володимиру Михайловичу, Снитюку Віталію Євгеновичу, Степашку Володимиру Семеновичу, Триусу Юрію Васильовичу за конструктивну критику наших досліджень та цікаві бесіди, результатом яких з'явилась дана робота.

Розділ 1

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 певних 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 фундаментального значення, можна стверджувати, що акції або недооцінені, або переоцінені, а ринкова ціна коливається навколо фундаментального значення.

Існують деякі недоліки, притаманні фундаментальному аналізу: потреба значного часу для збору даних та розрахунків, що зумовлює відставання прогнозу від потоку 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 її майбутнього значення. У цьому 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 випливає

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 розвивається технічний аналіз – метод вивчення рухів цін, головним інструментом якого служать графіки [14–17]. Своє коріння сучасний технічний аналіз бере з теорії Чарльза Доу, випливаючи з неї прямо або побічно. Технічний аналіз увібрав у себе такі принципи і поняття, як напрям руху цін, ("ціни враховують усю відому інформацію"), збіжність і розбіжність ковзних середніх, динаміка об'ємів як дзеркало цінових змін, а також лінії підтримки та опору.

Розглянемо деякі підходи до класифікації сучасних методів прогнозування.

Підручники з економічного прогнозування [5, 19] 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 систем [5, 19]. Інтуїтивні методи 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].

У посібнику [3] розглядається структурний та формалізований підхід до побудови прогнозних моделей. Під структурним підходом розуміють 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 чорної скриньки).

Виділимо, зокрема, недоліки загальноприйнятих класифікацій:

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.

Нами пропонується удосконалена класифікація методів прогнозування, яка наведена на рисунку 1.1.

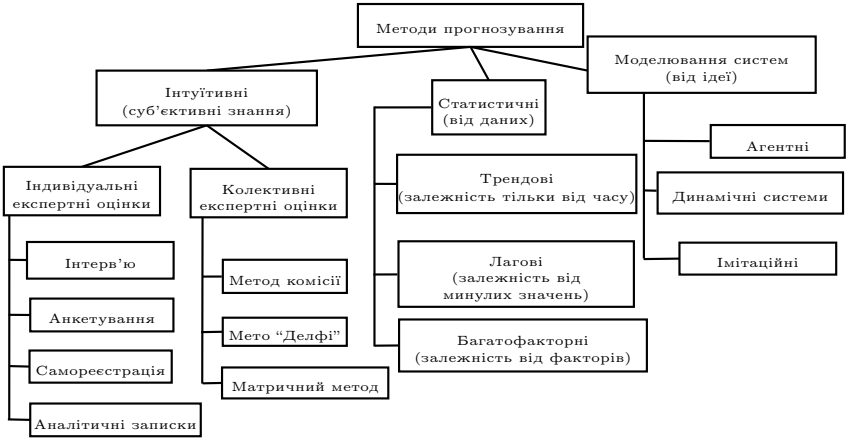


Рис. 1.1: Класифікація методів прогнозування

За основу класифікації взято об'єкт, вихідні дані чи модель якого дозволяє побудувати прогноз. У першій, експертній групі методів таким об'єктом є експерт, який опитується чи анкетується. Основна ідея даних методів полягає у проведенні опитування та статистичного опрацювання суб'єктивних прогнозів експертів. Оскільки формально джерелами знань є експерти, які володіють власними, здебільшого інтуїтивними правилами прийняття рішень, а експертні 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 вбачаємо у побудові системи видобування знань із джерел новин мережі Інтернет, аналізу правдоподібності будь-якої інформації, включаючи гіпотези та чутки, саме засобами інформаційних технологій.

Одним із нових напрямів дослідження є теорія несилової взаємодії, запропонована у монографії [18]. Автором пропонується методи прогнозування, які базуються на цьому принципі

існування природи та суспільства. У рамках даної роботи вказаний напрям розглядатись не буде через його складність.

Основна увага в даній роботі приділяється формалізованим методам прогнозування. Згідно з запропонованою нами класифікацією (рис. 1.1) дані методи поділяються на статистичні методи та методи моделювання систем. Методи з групи "Моделювання систем" направлені на побудову моделей, які описують безпосередньо функціонування досліджуваного об'єкту чи складної системи. Статистичні методи тут треба розуміти як методи, направлені на дослідження тільки похідних показників досліджуваного об'єкту, представлених у вигляді часових рядів.

Методи прогнозування з групи "Моделювання систем" базуються на моделях об'єкту дослідження. До них можна віднести імітаційне моделювання, під яким традиційно розуміється 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 у досліджуваній системі. Динамічні системи – це такі моделі, в яких закономірності розвитку об'єкта моделюються на основі диференціальних чи різницевих рівнянь. Якщо розглядати моделі з класу "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 [20–24]. 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 випливають 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: недостатня складність у поведінці агентів, неможливість 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. Термін "статистичні" 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 як нелінійні аналітичні моделі, так і 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 1.1. *Часовим рядом* називається послідовність рівнів ряду

$$\{y_i\} = y_1, y_2, ...y_n \tag{1.1}$$

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

Усі методи прогнозування можуть бути подані у вигляді наступної математичної моделі:

$$y_{t+1} = f(t, y_t, ..., y_{t-m}, x_t^{(1)}, ..., x_{t-m_1}^{(1)}, x_t^{(2)}, ..., x_{t-m_2}^{(2)}, x_t^{(k)}, ..., x_{t-m_k}^{(k)}), \tag{1.2}$$

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; $m, m_1, ...m_k$ – довжини "пам'яті" ряду, які використовуються при прогнозуванні. Відмінності різних методів будуть стосуватися структури вхідних даних x та виду функції $f(...)$. Таким чином, можна виділити такі види залежностей, які використовуються у сучасних методах прогнозування:

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

В наступних розділах даної роботи будуть розглядатись вищезазначені залежності, що моделюються рядом методів прогнозування.

1.1 Перспективи використання трендових моделей при побудові сучасних методів прогнозування

Технічний аналіз [14–17] полягає у вивченні динаміки руху цін у минулому з метою визначення ймовірного напряму їх розвитку в майбутньому. Поточна динаміка цін (тобто поточні очікування) порівнюється із зіставною динамікою цін у минулому, за допомогою чого досягається більш менш реалістичний прогноз.

Незалежно від того, якому типу аналізу віддає перевагу трейдер [16] – 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 [16]. Графічний аналіз ліній тренду втратив минулу суб'єктивність і перетворився на суто 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 1.2 і 1.3 представлено 2 лінії тренду: пропозиція спадаючою лінією, попит – зростаючою.

Можна звернути увагу, що на рисунку 1.2 поступове зниження цін відображене низхідною лінією "пропозиції": цінові списки і западини також послідовно знижуються.

На рисунку 1.3 поступове підвищення цін відображене висхідною лінією "попиту": цінові списки і западини також послідовно підвищуються. Складність полягає у виборі особливих точок, через які проходять ці прямі лінії (див. рис. 1.3). Як правило, аналітик привносить неабияку частку суб'єктивізму при побудові 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

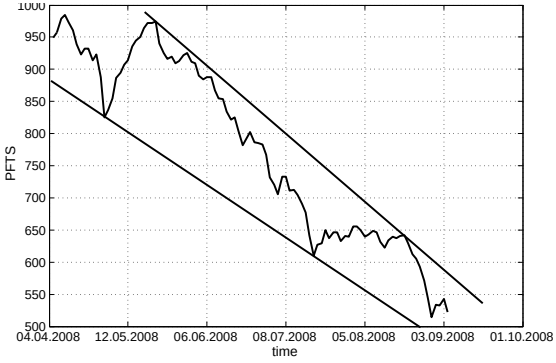


Рис. 1.2: Графік лінії тренду (низхідна лінія)

Рис. 1.3: Графік лінії тренду (висхідна лінія)

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, пов'язані із застосуванням економетричних моделей, кожну 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 динаміка середнього значення ряду описується аналітичною залежністю тільки від часу, називаються трендовими моделями [4, 5, 25]. У більшості робіт [4, 5, 25] під трендовими моделями мають на увазі лінійні моделі тренду або ті, що до них зводяться.

Означення 1.2. *Трендовою моделлю* будемо називати аналітичну функцію виду

$$y = f(t, \vec{a}), \tag{1.3}$$

де t – час, заданий в певних одиницях (для даного дослідження – у днях); $\vec{a}$ – вектор параметрів, при яких модель найточніше описує залежність досліджуваного ряду $\{y_i\}$ від часу t та забезпечує мінімальні відхилення фактичних та змодельованих рівнів:

$$F(\vec{a}) = \|y_t - f(t, \vec{a})\| \tag{1.4}$$

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 [26–28], складною нейронною мережею тощо.

Для побудови даних моделей здебільшого використовується 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 нев'язки) для усіх параметрів разом, крім того декількох гармонік одночасно, дозволяють дещо 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 роль функції для 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. Строге обгрунтування 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:

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

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

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:

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

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

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

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

В підручниках [4, 5, 19] даний метод використовується для побудови багатофакторних та авторегресійних прогнозних моделей.

Передумови використання даного методу базуються на твердженнях

- незміщенності 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. Трендові моделі, які допускають зведення до лінійного виду, наведено у таблиці 1.1.

Варто зауважити, що оцінювання коефіцієнту детермінації та статистичної значущості результатів здійснюється не для вихідних змінних, а для лінеаризованих, що обов'язково необхідно враховувати.

Розглянемо обчислення довірчих інтервалів для прогнозу згідно трендової моделі. Для нормального розподілу $N(\mu, \sigma)$ з невідомими параметрами μ та σ процес відповідає наступній допоміжній статистиці [29] :

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

Табл. 1.1: Нелінійні трендові моделі

Вид трендової моделі	Формула залежності	Лінеаризація
Лінійна	$y(t) = a_0 + a_1 t$	–
Поліном n-го степеня	$y(t) = a_0 + a_1 t + +a_2 t^2 + ... + a_n t^n$	$y(t) = a_0 + a_1 z_1 + ... + a_n z_n,$ $z_i = t^i$
Експонента	$y(t) = a_o e^{a_1 t}$	$ln(y(t)) = ln(a_0) + a_1 t$
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	$y(t) = a_o t^{a_1}$	$Y(t) = ln(y(t)), t^* = ln(t)$
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	$y(t) = a + bc^t$	$c = \frac{(n-1)\sum_{t=0}^{n-1} y_t y_{t+1} - \sum_{t=1}^{n-1} y_t \sum_{t=1}^{n-1} y_{t+1}}{(n-1)\sum_{t=1}^{n-1} y_t^2 - \left(\sum_{t=1}^{n-1} y_t\right)^2},$ $b = \frac{n\sum c^t y_t - \sum c^t \sum y_t}{n\sum c^{2t} - (\sum c^t)^2},$ $a = \frac{\sum y_t - b\sum c^t}{n}$
крива Гомперця	$y(t) = a_0 a_1^{a_2^t}$	$ln(y(t)) = ln(a_0) + a_2{}^t ln(a_1)$
Логістична	$y(t) = \frac{1}{a_0 + a_1 a_2^t}$	$Y(t) = \frac{1}{y(t)}, t^* = \frac{1}{t},$ $Y = a_0 + a_1 a_2^t$

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

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

Ймовірність того, що значення X_{n+1} буде знаходитись у заданому інтервалі, наступна:

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

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

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

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

На нашу думку, до трендових моделей необхідно віднести також лінійно-гармонічну модель, наведену у монографії [26]. Розглянемо її.

$$y(t) = c_0 + c_1 t + \sum_{i=1}^{n}\left(a_i cos\frac{2\pi}{T_i}t + b_i sin\frac{2\pi}{T_i}t\right), \tag{1.5}$$

де $t = 1, 2, \cdots, n$ – 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 багатофакторної 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

$$Z(c_0, c_1, a_i, b_i) = \sum_{t=T_{min}}^{T_{max}}\left[Y(t) - c_0 - c_1 t - a_i cos\frac{2\pi}{T_i}t + b_i sin\frac{2\pi}{T_i}t\right]^2 \to min. \tag{1.6}$$

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

- повний перебір частот не є оптимальним, такий підхід не дає можливості точно оцінити період коливань;
- неможливість оптимізації моделі за параметром T_i;

- можливі проблеми з визначенням параметрів за МНК (мультиколінеарність, гетероскедастичність тощо).

Удосконалення даного методу розглядається в розділі 5 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) [30–32]. 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

$$y_t^* = y_t - \bar{y_t},$$

або

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

де $\bar{y_t}$ - середнє значення ряду, σ - середньоквадратичне відхилення.
2. Побудова траєкторної матриці

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

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 $v^{(1)}, v^{(2)}, \ldots, v^{(m)}$ – власні вектори матриці C, а $\lambda_1, \lambda_2, \ldots, \lambda_m$ – її власні значення.

5. Вибір мінімального базису власних векторів. З множини власних векторів пропонується вибрати підмножину векторів, які описують тільки суттєву варіацію часового ряду. Для цього упорядкуємо пари $\left(\lambda_i, v^{(i)}\right)$ у порядку спадання власних значень. Нехай отримані номери власних векторів є наступними:

$k_1, k_2, \ldots, k_m$. Способи вибору підмножини власних векторів для аналізу часових рядів детально обговорюються в [33]. Розглянемо найпростіший метод. Визначимо τ як кількість найбільших власних векторів, питома вага власних значень яких більша, наприклад, за 75% ваги усих власних значень.

Запишемо матрицю V_x у вигляді отриманих власних векторів-стовбців:

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

6. Відображення матриці X в новому базисі власних векторів:

$$U_x = V_x^T X = (U_1, \ldots, U_\tau).$$

7. Відновлення вихідної матриці за першими r головними компонентами:

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

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:

$$f_s = \begin{cases} \frac{1}{s} \sum\limits_{i=1}^{s} \tilde{x}_{i,s-i+1}, 1 \le s \le \tau, \\ \frac{1}{\tau} \sum\limits_{i=1}^{\tau} \tilde{x}_{i,s-i+1}, \tau \le s \le n, \\ \frac{1}{N-s+1} \sum\limits_{i=1}^{N-s+1} \tilde{x}_{i+s-n,n-i+1}, \tau \le s \le N. \end{cases}$$

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 [33] 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, 34–36]. Головна ідея економетричних методів полягає у наближенні часового ряду до лінійного виду за допомогою лінеаризації. Далі будується лінійна регресійна модель та використовується 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.

Лабораторна робота № 1. Трендові моделі

За означенням, трендовими називаються моделі, у яких використовується залежність показника, що прогнозується, від часу.

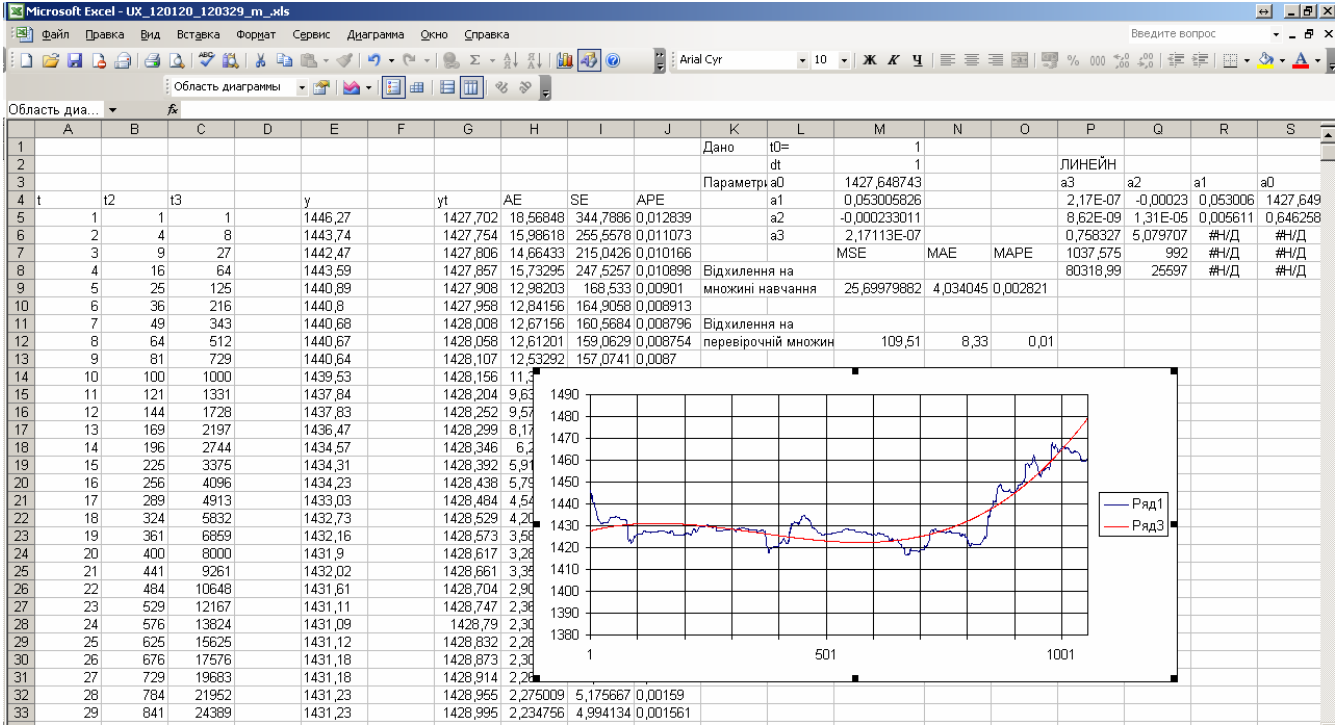
$$y = f(t),$$

де t – параметр часу. Розглянемо процес побудови трендової моделі на прикладі поліноміальної моделі, 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 багатофакторної, здійснивши заміни $x_3(t) = t^3, x_2(t) = t^2, x_1(t) = t$. Таким чином, доцільно 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 секції "Дано", "Таблиці даних" та "Графіки" (див. рис. 1.4).

Секцію "Дано" зручно використовувати для введення та змін параметрів моделі та параметрів розрахунку. До таких параметрів необхідно віднести початкове значення t_0 та крок часу Δt. Також секцію "Дано" знизу зручно доповнити розрахунковими результатами прогнозної моделі. Наприклад, до них можна віднести:

1. параметри моделі a_3, a_2, a_1, a_0;
2. середнє лінійне відхидення (MAE) на навчальній вибірці;
3. середньоквадратичне відхилення (MSE) на навчальній вибірці;
4. дисперсію процесу σ;
5. коефіцієнт детермінації R^2;
6. середнє лінійне відхилення (MAE) на перевірочній вибірці;
7. середньоквадратичне відхилення (MSE) на перевірочній вибірці.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1											Дано	t0=	1						
2												dt	1			ЛИНЕЙН			
3											Параметри	a0	1427,648743			a3	a2	a1	a0
4	t	t2	t3		y		yt	AE	SE	APE		a1	0,053005826			2,17E-07	-0,00023	0,053006	1427,649
5	1	1	1		1446,27		1427,702	18,56848	344,7886	0,012839		a2	-0,000233011			8,62E-09	1,31E-05	0,005611	0,646258
6	2	4	8		1443,74		1427,754	15,98618	255,5578	0,011073		a3	2,17113E-07			0,758327	5,079707	#Н/Д	#Н/Д
7	3	9	27		1442,47		1427,806	14,66433	215,0426	0,010166			MSE	MAE	MAPE	1037,575	992	#Н/Д	#Н/Д
8	4	16	64		1443,59		1427,857	15,73295	247,5257	0,010898	Відхилення на					80318,99	25597	#Н/Д	#Н/Д
9	5	25	125		1440,89		1427,908	12,98203	168,533	0,00901	множині навчання		25,69979882	4,034045	0,002821				
10	6	36	216		1440,8		1427,958	12,84156	164,9058	0,008913									
11	7	49	343		1440,68		1428,008	12,67156	160,5684	0,008796	Відхилення на								
12	8	64	512		1440,67		1428,058	12,61201	159,0629	0,008754	перевірочній множин		109,51	8,33	0,01				
13	9	81	729		1440,64		1428,107	12,53292	157,0741	0,0087									
14	10	100	1000		1439,53		1428,156	11,3											
15	11	121	1331		1437,84		1428,204	9,63											
16	12	144	1728		1437,83		1428,252	9,57											
17	13	169	2197		1436,47		1428,299	8,17											
18	14	196	2744		1434,57		1428,346	6,2											
19	15	225	3375		1434,31		1428,392	5,91											
20	16	256	4096		1434,23		1428,438	5,79											
21	17	289	4913		1433,03		1428,484	4,54											
22	18	324	5832		1432,73		1428,529	4,20											
23	19	361	6859		1432,16		1428,573	3,58											
24	20	400	8000		1431,9		1428,617	3,28											
25	21	441	9261		1432,02		1428,661	3,35											
26	22	484	10648		1431,61		1428,704	2,90											
27	23	529	12167		1431,11		1428,747	2,36											
28	24	576	13824		1431,09		1428,79	2,30											
29	25	625	15625		1431,12		1428,832	2,28											
30	26	676	17576		1431,18		1428,873	2,30											
31	27	729	19683		1431,18		1428,914	2,26											
32	28	784	21952		1431,23		1428,955	2,275009	5,175667	0,00159									
33	29	841	24389		1431,23		1428,995	2,234756	4,994134	0,001561									

Рис. 1.4: Побудова трендової моделі в Excel

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),
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 (У);

константа – задає, яким чином обчислювати значення параметру $_0$. Якщо обчислювати звичайним чином, то значення параметру має бути “ИСТИНА” або 1; якщо $_0 = 0$, то значення параметру має бути 0 або “ЛОЖЬ”;

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. 1.4, 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) в усі наступні комірки.

3. Додаткові комірки (значення квадратів і кубів для t у якості факторів багатофакторної моделі) можут бути обчислені простим піднесенням до 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.1.4). Радимо залишати декілька значень для післяпрогнозної 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 1.1, яка наведена у теоретичній частині даного розділу, частіше такими 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 виділити кольором або за допомогою зміни формату діапазон комірок, у які будуть повернені параметри (Розмірність масиву має бути $(m + 1)$ рядків та 5 стовбців, де m – кількість факторів). Підпишемо 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, виберіть 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 посилань 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 перевірочної 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.

8. Обчислити абсолютні та квадратичні похибки для навчальної та прогнозної вибірки. Для цього необхідно знайти модулі (наприклад, комірка H5 містить формулу =ABS(G5-E5) та 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 перевірочної 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.

Лабораторна робота № 2. Методи оцінювання параметрів нелінійних трендових моделей

Згідно алгоритму, розглянутого у попередній лабораторній роботі, можна побудувати будь-яку трендову модель з таблиці 1.1, яка дозволяє зведення до лінійного виду. Далеко не усі трендові моделі допускають таке зведення. Нами пропонується метод оцінювання параметрів будь-якої трендової моделі, який базується на нелінійній оптимізації.

Нехай нелінійна трендова модель задана у вигляді функції $y = f(t, a_0, a_1, \ldots, a_m)$, де $a_0, a_1, \ldots, a_m$ – параметри моделі, t – час. Нехай задано навчальну вибірку, яка складається з точок $\{(t_i, y_i)\}$. Тоді функціонал нев'язки може бути записаний наступним чином:

$$L(a_0, a_1, \ldots, a_m) = \sum_{i=1}^{n} |y_i - f(t_i, a_0, a_1, \ldots, a_m)|^k, \qquad (1.7)$$

де k – параметр чутливості до відхилень. При $k = 1$ мають місце абсолютні похибки, при $k = 2$ – квадратичні похибки, при $k \to \infty$ міра L відповідає максимальному відхиленню у навчальній вибіці. Необхідно знайти значення вектору параметрів $a_0, a_1, \ldots, a_m$, при яких вибрана функція L набуває свого мінімального значення.

Використовуючи табличний процесор Excel та надстройку "Поиск решения", можна реалізувати пошук параметрів, при яких функціонал нев'язки 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, вибрати опцію "Переместить/скопировать", поставивши галочку "Создать копию").

Хід роботи:

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. 1.4, формула EXCEL для комірки початкового значення A5 матиме вигляд =M1), а кожне наступне більше

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.1.4). Радимо залишати декілька значень для післяпрогнозної 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 у попередній роботі за допомогою лінеаризації. У випадку коли лінеаризація неможлива, можна лінеаризувати спрощений варіант моделі, зафіксувавши певні початкові значення параметрів, які не вдалося звести до лінійних залежностей. Зауважимо, що для методів локальної нелінійної оптимізації початкове значення відіграє значну роль. Тому необхідно продумати не одне, а певну множину для перебору початкових 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: =$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 перевірочної 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 прогнозної вибірки. Для цього необхідно знайти модулі (наприклад, комірка H5 містить формулу =ABS(G5-E5) та містить модуль відхилення фактичного та теоретичного значень 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 перевірочної вибірки (продовження часового ряду, дані якого не використовувались при побудові моделі). Середнє значення похибки для вибірки навчання буде грати роль цільової функції, яку будемо мінімізовувати.

7. Застосуємо функцію "Поиск решения", яка доступна з меню "Сервис-Поиск решения" (див. рис. 1.5). У випадку, коли такої функції нема у меню, необхідно її встановити, скориставшись опцією меню "Сервис-Надстройки" і поставивши 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 екстремуму, 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 роботах, можна завантажити за веб-адресою

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

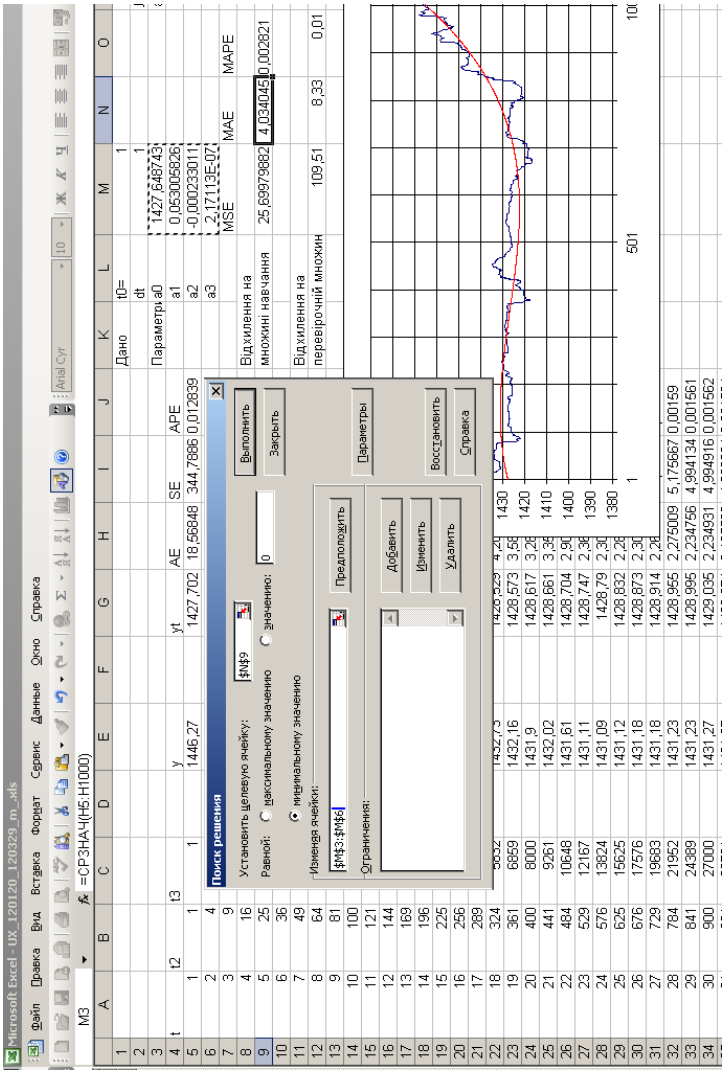


Рис. 1.5: Побудова нелінійної трендової моделі в Excel

1.2 Авторегресійні методи прогнозування

Означення 1.3. *Процесом з авторегресією* $AR(p)$ називається послідовність рівнів ряду $\{y_i\}$, яка задовольняє рівняння:

$$y_{n+1} = a_0 + a_1 y_n + a_2 y_{n-1} + \cdots + a_{k-1} y_{k-2} + a_k y_{k-1} + e, \quad (1.8)$$

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

У монографіях [3, 37–39] пропонується метод знаходження аналітичного розв'язку авторегресійного рівняння (1.8) відносно параметру часу t, що дозволяє звести авторегресійну модель часового ряду до трендової. Дане перетворення базується на пошуку частинних 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 дійсних 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 частинний розв'язок для однорідних збурень:

$$y^p = 2,1 + 0,8y^p - 0,15y^p => y^p = 6.$$

Об'єднуємо однорідний розв'язок з частинними:

$$y(k) = 6.0 + A_1 \cdot (0,5)^k + A_2 \cdot (0,3)^k,$$

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

$$\{1.0 = 6.0 + A_1 + A_2 2.0 = 6.0 + 0.5 \cdot A_1 + 0.3 \cdot A_2\}$$

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

$$y(k) = 6,0 - 12,5 \cdot (0,5)^k + 7,5 \cdot (0,4)^k.$$

Для рівнянь вищого порядку можна виконати аналогічні перетворення, але виникають труднощі, пов'язані з розв'язанням алгебраїчного рівняння високого степеня. На нашу думку, даний метод не покращує прогнозні 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, 40] 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, 40] називаються авторегресійні моделі порядку 1.

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

Означення 1.4. *Процесом ковзного середнього* $MA(q)$ називається послідовність рівнів ряду $\{y_i\}$, яка задовольняє рівняння:

$$y_{n+1} = b_0 + b_1 e_n + b_2 e_{n-1} + \cdots + b_{k-1} e_{k-2} + b_k e_{k-1}, \qquad (1.9)$$

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

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

Означення 1.5. *Процесом авторегресії – ковзного середнього* $ARMA(p, q)$ називається послідовність рівнів ряду $\{y_i\}$, яка задовольняє рівняння:

$$\begin{aligned} y_{n+1} = a_0 + a_1 y_n + a_2 y_{n-1} + \cdots + a_{k-1} y_{k-2} + a_k y_{k-1} + \\ + b_0 + b_1 e_n + b_2 e_{n-1} + \cdots + b_{k-1} e_{k-2} + b_k e_{k-1}, \end{aligned} \qquad (1.10)$$

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 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.

Спрогнозуємо невідоме значення, x_{t+l}, $l > 1$ 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

$$\begin{aligned}&(1 - \alpha_1 L - ... - \alpha_p L^p) \cdot \sum_{j=0}^{k} (-1)^k C_k^j x_{\tau-1} = \\ &= \delta_\tau - \theta_1 \delta_{\tau-1} - ... - \theta_q \delta_{\tau-q},\end{aligned} \tag{1.11}$$

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

Зі співвідношення (1.11) можна виразити x_τ для будь-якого $\tau = t - q, \ldots, t - 1, t + 1$. Отримуємо:

$$\begin{aligned}x_\tau = &\left(\sum_{j=1}^{p} \alpha_j L^j\right) x_\tau + \left(1 - \sum_{j=1}^{p} \alpha_j L^j\right) \cdot \left(\sum_{i=0}^{k} (-1)^j C_k^i x_{\tau-i}\right) + \\ &+ \left(1 - \sum_{j=1}^{q} \theta_j L^j\right) \delta_\tau.\end{aligned} \tag{1.12}$$

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 (1.12) 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 момент часу t з випередженням l є умовне математичне сподівання випадкової величини x_{t+1}, обчислене за умови, що всі значення x_τ до моменту часу t. Цей результат є окремим випадком загальної теорії прогнозування [41–43].

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

1) за формулою (1.12) обчислюються ретроспективні прогнози $x^l_{t-q-l}...x^l_{t-1}$ на базі попередніх значень часового ряду;

2) використовуючи формули (1.12) для $\tau > t$, підраховуються умовні математичні сподівання для обчислення прогнозних значень.

Метод експоненціального згладжування

Досить ефективним і надійним методом прогнозування є метод експоненціального згладжування [5, 43]. 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 найчастіше даний метод застосовується 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, 41].

Нехай початковий динамічний ряд описується функцією

$$y_t = a_0 + a_1 t + \frac{a_2}{2} t^2 + ... + \frac{a_p}{p!} t^p + \varepsilon_t. \qquad (1.13)$$

Метод експоненціального згладжування, що є узагальненням методу ковзного середнього, дозволяє побудувати такий опис процесу (1.13), 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 середньою користуються рекурентною формулою [44]

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

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 обгрунтованого вибору α можна використовувати процедуру узагальненого згладжування, яка дозволяє отримати співвідношення, що пов'язують дисперсію прогнозу і параметр згладжування [43]. Істотним для практичного використання є питання про вибір порядку прогнозуючого полінома, що багато в чому визначає якість прогнозу. У роботі [43] 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 [43].

Огляд літератури дозволив виявити ряд недоліків загальновідомих методів прогнозування і завдань, які необхідно вирішити в даній області.

Головний недолік цього методу полягає в тому, що часовий ряд розглядається ізольовано від інших явищ. Крім того, точність прогнозу помітно падає при довгостроковому прогнозуванні [43].

Слабким 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.

Недоліками методу найменших квадратів є використання

процедури оцінки, що передбачає обов'язкове виконання цілого ряду передумов, порушення яких може привести до значних помилок:

1) випадкові похибки повинні мати нульову середню та 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 мають бути 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 та ін. Класичний метод найменших квадратів передбачає рівноцінність початкової інформації в моделі. У реальній практиці майбутня поведінка процесу значно більшою мірою визначається більш новими спостереженнями, ніж минулими. Ця властивість вказує на зменшення цінності ранньої (більш віддаленої у часі, старої) інформації.

Важливим моментом отримання прогнозу за допомогою 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 тільки 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.

Проблеми з методом Бокса-Дженкінса (моделі авторегресії – ковзного середнього) пов'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 та 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 [45]. 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 [44, 46–49].

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 особливостями авторегресійних моделей, більш глибоко усвідомити їх суть і можливості при побудові прогнозів.

Пропонуємо розробити лист Excel наступним чином (див. рис.1.6). 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 прогнозної 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 $a_0, a_1, \ldots, a_m$;

2. середнє лінійне відхилення (MAE) на навчальній вибірці;

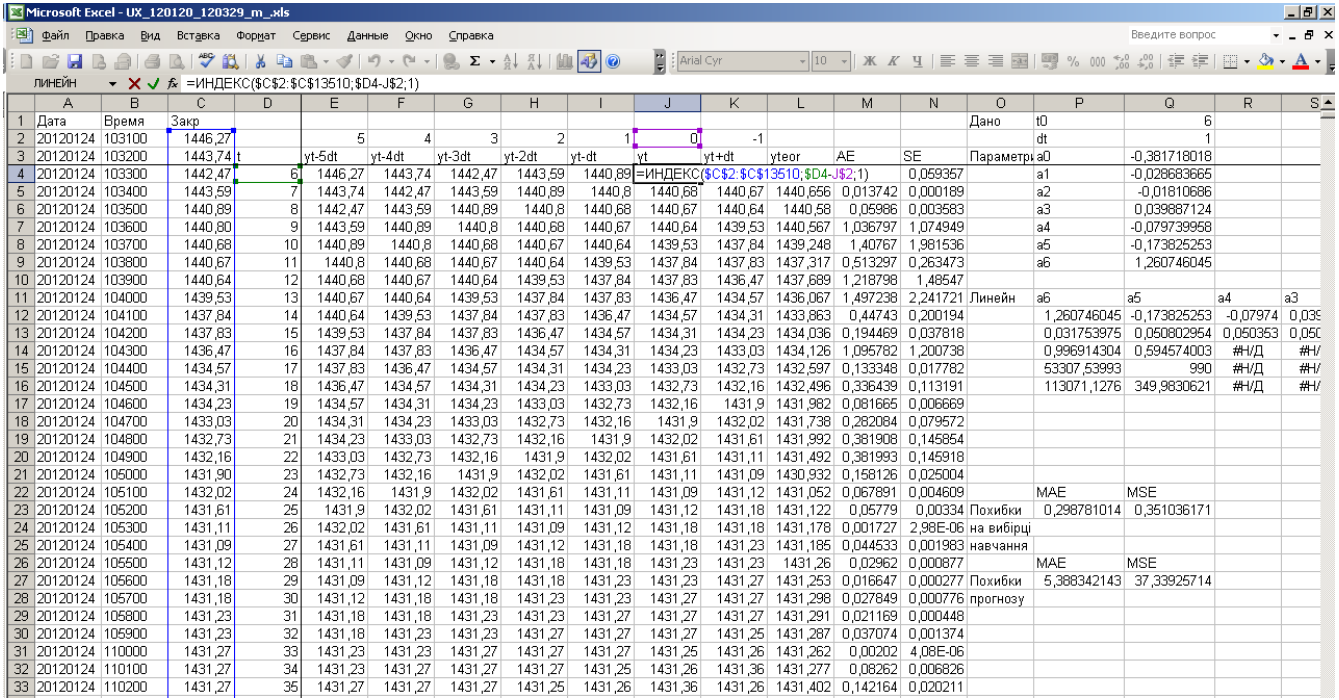
3. середньоквадратичне відхилення (MSE) на навчальній вибірці;

4. дисперсія процесу;

5. коефіцієнт детермінації;

6. середнє лінійне відхилення (MAE) на перевірочній вибірці;

7. середньоквадратичне відхилення (MSE) на перевірочній вибірці.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Дата	Время	Закр												Дано	t0	6		
2	20120124	103100	1446,27		5	4	3	2	1	0	-1					dt	1		
3	20120124	103200	1443,74	t	yt-5dt	yt-4dt	yt-3dt	yt-2dt	yt-dt	yt	yt+dt	yteor	AE	SE	Параметри	a0	-0,381718018		
4	20120124	103300	1442,47	6	1446,27	1443,74	1442,47	1443,59	1440,89	=ИНДЕКС($C$2:$C$13510;$D4-J$2;1)				0,059357		a1	-0,028683665		
5	20120124	103400	1443,59	7	1443,74	1442,47	1443,59	1440,89	1440,8	1440,68	1440,67	1440,656	0,013742	0,000189		a2	-0,01810686		
6	20120124	103500	1440,89	8	1442,47	1443,59	1440,89	1440,8	1440,68	1440,67	1440,64	1440,58	0,05986	0,003583		a3	0,039887124		
7	20120124	103600	1440,80	9	1443,59	1440,89	1440,8	1440,68	1440,67	1440,64	1439,53	1440,567	1,036797	1,074949		a4	-0,079739958		
8	20120124	103700	1440,68	10	1440,89	1440,8	1440,68	1440,67	1440,64	1439,53	1437,84	1439,248	1,40767	1,981536		a5	-0,173825253		
9	20120124	103800	1440,67	11	1440,8	1440,68	1440,67	1440,64	1439,53	1437,84	1437,83	1437,317	0,513297	0,263473		a6	1,260746045		
10	20120124	103900	1440,64	12	1440,68	1440,67	1440,64	1439,53	1437,84	1437,83	1436,47	1437,689	1,218798	1,48547					
11	20120124	104000	1439,53	13	1440,67	1440,64	1439,53	1437,84	1437,83	1436,47	1434,57	1436,067	1,497238	2,241721	Линейн	a6	a5	a4	a3
12	20120124	104100	1437,84	14	1440,64	1439,53	1437,84	1437,83	1436,47	1434,57	1434,31	1433,863	0,44743	0,200194		1,260746045	-0,173825253	-0,07974	0,039
13	20120124	104200	1437,83	15	1439,53	1437,84	1437,83	1436,47	1434,57	1434,31	1434,23	1434,036	0,194469	0,037818		0,031753975	0,050802954	0,050353	0,050
14	20120124	104300	1436,47	16	1437,84	1437,83	1436,47	1434,57	1434,31	1434,23	1433,03	1434,126	1,095782	1,200738		0,996914304	0,594574003	#Н/Д	#Н/
15	20120124	104400	1434,57	17	1437,83	1436,47	1434,57	1434,31	1434,23	1433,03	1432,73	1432,597	0,133348	0,017782		53307,53993	990	#Н/Д	#Н/
16	20120124	104500	1434,31	18	1436,47	1434,57	1434,31	1434,23	1433,03	1432,73	1432,16	1432,496	0,336439	0,113191		113071,1276	349,9830621	#Н/Д	#Н/
17	20120124	104600	1434,23	19	1434,57	1434,31	1434,23	1433,03	1432,73	1432,16	1431,9	1431,982	0,081665	0,006669					
18	20120124	104700	1433,03	20	1434,31	1434,23	1433,03	1432,73	1432,16	1431,9	1432,02	1431,738	0,282084	0,079572					
19	20120124	104800	1432,73	21	1434,23	1433,03	1432,73	1432,16	1431,9	1432,02	1431,61	1431,992	0,381908	0,145854					
20	20120124	104900	1432,16	22	1433,03	1432,73	1432,16	1431,9	1432,02	1431,61	1431,11	1431,492	0,381993	0,145918					
21	20120124	105000	1431,90	23	1432,73	1432,16	1431,9	1432,02	1431,61	1431,11	1431,09	1430,932	0,158126	0,025004					
22	20120124	105100	1432,02	24	1432,16	1431,9	1432,02	1431,61	1431,11	1431,09	1431,12	1431,052	0,067891	0,004609		MAE	MSE		
23	20120124	105200	1431,61	25	1431,9	1432,02	1431,61	1431,11	1431,09	1431,12	1431,18	1431,122	0,05779	0,00334	Похибки	0,298781014	0,351036171		
24	20120124	105300	1431,11	26	1432,02	1431,61	1431,11	1431,09	1431,12	1431,18	1431,18	1431,178	0,001727	2,98E-06	на вибірці				
25	20120124	105400	1431,09	27	1431,61	1431,11	1431,09	1431,12	1431,18	1431,18	1431,23	1431,185	0,044533	0,001983	навчання				
26	20120124	105500	1431,12	28	1431,11	1431,09	1431,12	1431,18	1431,18	1431,23	1431,23	1431,26	0,02962	0,000877		MAE	MSE		
27	20120124	105600	1431,18	29	1431,09	1431,12	1431,18	1431,18	1431,23	1431,23	1431,27	1431,253	0,016647	0,000277	Похибки	5,388342143	37,33925714		
28	20120124	105700	1431,18	30	1431,12	1431,18	1431,18	1431,23	1431,23	1431,27	1431,27	1431,298	0,027849	0,000776	прогнозу				
29	20120124	105800	1431,23	31	1431,18	1431,18	1431,23	1431,23	1431,27	1431,27	1431,27	1431,291	0,021169	0,000448					
30	20120124	105900	1431,23	32	1431,18	1431,23	1431,23	1431,27	1431,27	1431,27	1431,25	1431,287	0,037074	0,001374					
31	20120124	110000	1431,27	33	1431,23	1431,23	1431,27	1431,27	1431,27	1431,25	1431,26	1431,262	0,00202	4,08E-06					
32	20120124	110100	1431,27	34	1431,23	1431,27	1431,27	1431,27	1431,25	1431,26	1431,36	1431,277	0,08262	0,006826					
33	20120124	110200	1431,27	35	1431,27	1431,27	1431,27	1431,25	1431,26	1431,36	1431,26	1431,402	0,142164	0,020211					

Рис. 1.6: Побудова авторегресійної моделі в Excel

Секцію "Таблиця даних" пропонується розділити на 2 частини. Перша буде містити вихідні дані (дати та значення 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 $x_t = (y_{t-(m-1)\Delta t}, \ldots, _{t-\Delta t})$, 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.

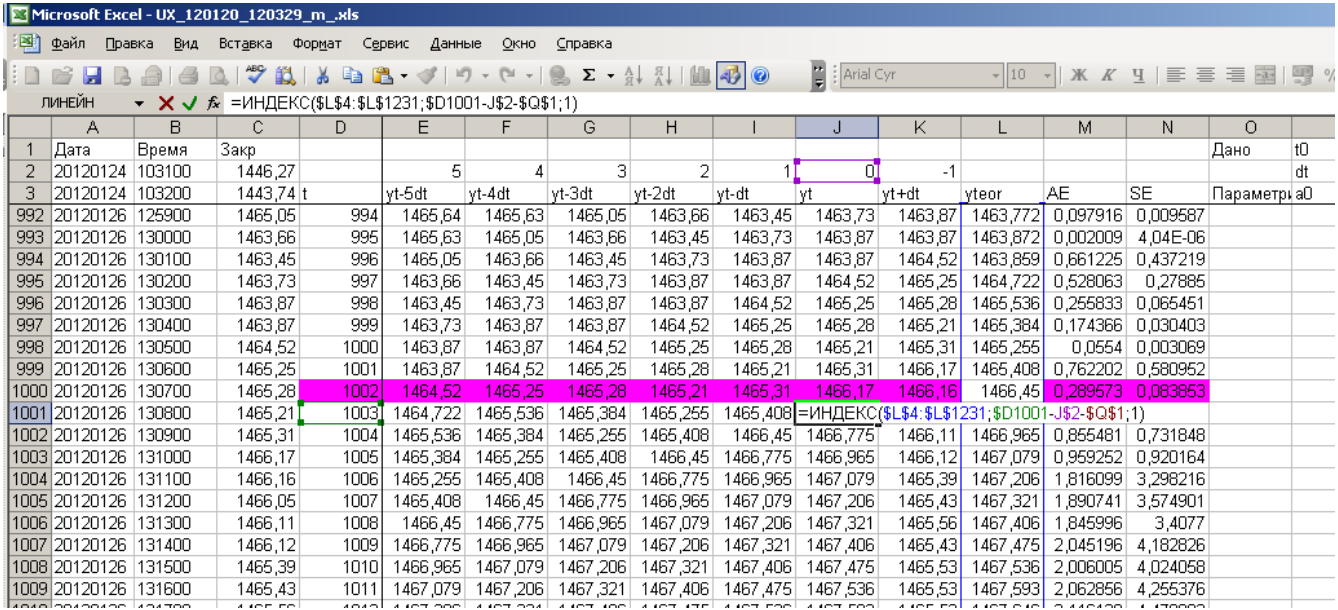
Розглянемо приклад листа Excel для побудови авторегресійної моделі. Стовбець часу t задає послідовність індексів, що відповідають часу поточного лагового вектору. Тому значення t_0 в секції "Дано" повинне бути задане з відступом, щоб даних вистачило для першого вектору, враховуючи заданий крок Δt. Перше 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 =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 ИНДЕКС, 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+$Q$2, при якій минула величина лагу збільшується на Δt. Усі формули зліва від розглянутих отримуються копіюваннням формули в комірці I2 у комірки E2... H2. Для значення відгуку системи (комірка K2) формула матиме вигляд =J2-$Q$2. 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 листі. 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 =ИНДЕКС($C$2:$C$13510;$D4-J$2;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 K4) також може мати аналогічний вигляд, так як абсолютні та відносні посилання 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 попередньою роботою, можна запропонувати два методи оцінювання параметрів: метод найменших квадратів, реалізований 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.



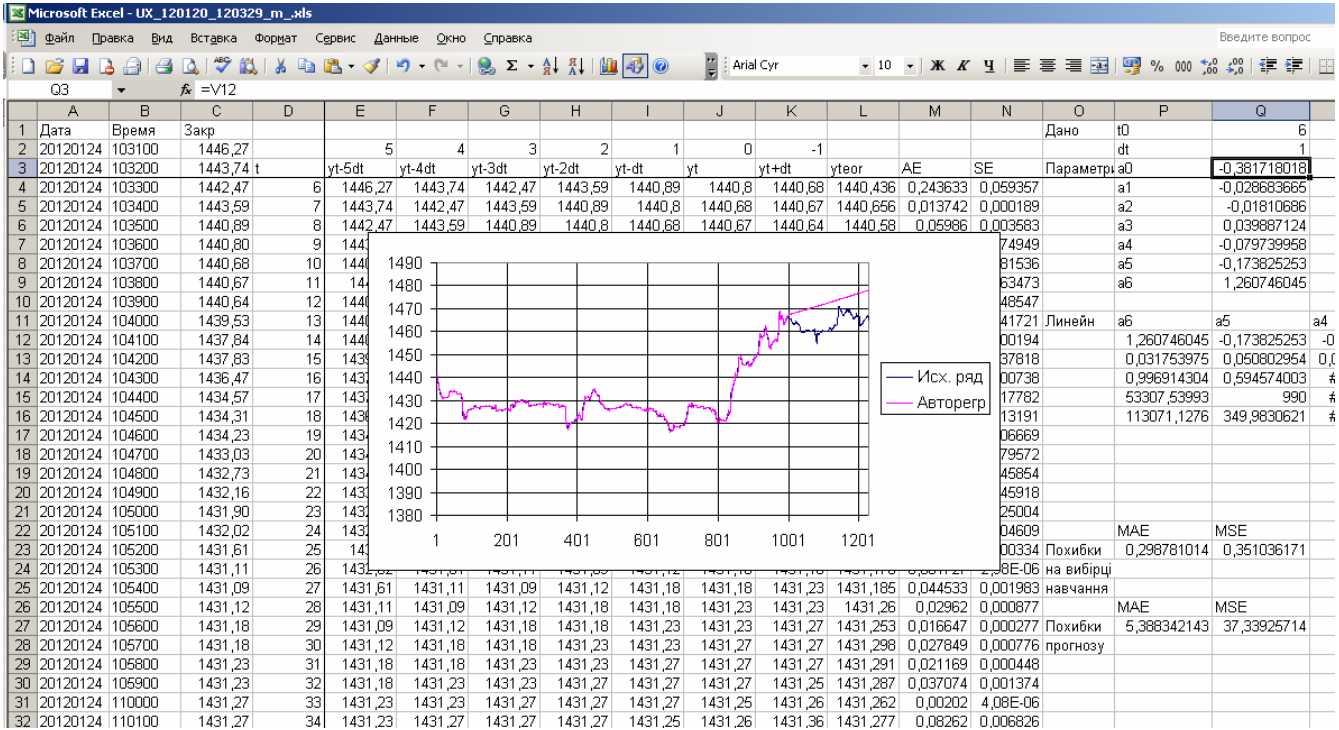
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1	Дата	Время	Закр												Дано	t0
2	20120124	103100	1446,27		5	4	3	2	1	0	-1					dt
3	20120124	103200	1443,74	t	yt-5dt	yt-4dt	yt-3dt	yt-2dt	yt-dt	yt	yt+dt	yteor	AE	SE	Параметри	a0
992	20120126	125900	1465,05	994	1465,64	1465,63	1465,05	1463,66	1463,45	1463,73	1463,87	1463,772	0,097916	0,009587		
993	20120126	130000	1463,66	995	1465,63	1465,05	1463,66	1463,45	1463,73	1463,87	1463,87	1463,872	0,002009	4,04E-06		
994	20120126	130100	1463,45	996	1465,05	1463,66	1463,45	1463,73	1463,87	1463,87	1464,52	1463,859	0,661225	0,437219		
995	20120126	130200	1463,73	997	1463,66	1463,45	1463,73	1463,87	1463,87	1464,52	1465,25	1464,722	0,528063	0,27885		
996	20120126	130300	1463,87	998	1463,45	1463,73	1463,87	1463,87	1464,52	1465,25	1465,28	1465,536	0,255833	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	1465,255	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	0,580952		
1000	20120126	130700	1465,28	1002	1464,52	1465,25	1465,28	1465,21	1465,31	1466,17	1466,16	1466,45	0,289573	0,083853		
1001	20120126	130800	1465,21	1003	1464,722	1465,536	1465,384	1465,255	1465,408	=ИНДЕКС($L$4:$L$1231;$D1001-J$2-$Q$1;1)						
1002	20120126	130900	1465,31	1004	1465,536	1465,384	1465,255	1465,408	1466,45	1466,775	1466,11	1466,965	0,855481	0,731848		
1003	20120126	131000	1466,17	1005	1465,384	1465,255	1465,408	1466,45	1466,775	1466,965	1466,12	1467,079	0,959252	0,920164		
1004	20120126	131100	1466,16	1006	1465,255	1465,408	1466,45	1466,775	1466,965	1467,079	1465,39	1467,206	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	1466,775	1466,965	1467,079	1467,206	1467,321	1467,406	1465,43	1467,475	2,045196	4,182826		
1008	20120126	131500	1465,39	1010	1466,965	1467,079	1467,206	1467,321	1467,406	1467,475	1465,53	1467,536	2,006005	4,024058		
1009	20120126	131600	1465,43	1011	1467,079	1467,206	1467,321	1467,406	1467,475	1467,536	1465,53	1467,593	2,062856	4,255376		

Рис. 1.7: Побудова прогнозу за авторегресійною моделлю в Excel

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.1.7). Таким чином, можна реалізувати багатокроковий прогноз. Формула для лагових векторів має брати дані не з колонки реальних значень (колонка C в прикладі), а з колонки прогнозів (колонка K). Зауважимо, що значення колонки K зміщені відносно C 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 векторів у часі можна отримати, скопіювавши вміст комірки J1001. Формули для обчислення теоретичних (прогнозних) значень та похибок залишиться без зміни.

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

Після виконання усіх вищезазначених розрахунків, можна зробити копію отриманого листа та розрахувати прогнози для інших інтервалів дискретизаці 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).



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Дата	Время	Закр												Дано	t0	6
2	20120124	103100	1446,27		5	4	3	2	1	0	-1					dt	1
3	20120124	103200	1443,74	t	yt-5dt	yt-4dt	yt-3dt	yt-2dt	yt-dt	yt	yt+dt	yteor	AE	SE	Параметри	a0	-0,381718018
4	20120124	103300	1442,47	6	1446,27	1443,74	1442,47	1443,59	1440,89	1440,8	1440,68	1440,436	0,243633	0,059357		a1	-0,028683665
5	20120124	103400	1443,59	7	1443,74	1442,47	1443,59	1440,89	1440,8	1440,68	1440,67	1440,656	0,013742	0,000189		a2	-0,01810686
6	20120124	103500	1440,89	8	1442,47	1443,59	1440,89	1440,8	1440,68	1440,67	1440,64	1440,58	0,05986	0,003583		a3	0,039887124
7	20120124	103600	1440,80	9	144									74949		a4	-0,079739958
8	20120124	103700	1440,68	10	144									81536		a5	-0,173825253
9	20120124	103800	1440,67	11	14									63473		a6	1,260746045
10	20120124	103900	1440,64	12	144									48547			
11	20120124	104000	1439,53	13	144									41721	Линейн	a6	a5	a4
12	20120124	104100	1437,84	14	144									00194		1,260746045	-0,173825253	-0
13	20120124	104200	1437,83	15	143									37818		0,031753975	0,050802954	0,0
14	20120124	104300	1436,47	16	143									00738		0,996914304	0,594574003	#
15	20120124	104400	1434,57	17	143									17782		53307,53993	990	#
16	20120124	104500	1434,31	18	143									13191		113071,1276	349,9830621	#
17	20120124	104600	1434,23	19	143									06669				
18	20120124	104700	1433,03	20	143									79572				
19	20120124	104800	1432,73	21	143									45854				
20	20120124	104900	1432,16	22	143									45918				
21	20120124	105000	1431,90	23	143									25004				
22	20120124	105100	1432,02	24	143									04609		MAE	MSE	
23	20120124	105200	1431,61	25	141									00334	Похибки	0,298781014	0,351036171	
24	20120124	105300	1431,11	26	1431,82									8E-06	на вибірці			
25	20120124	105400	1431,09	27	1431,61	1431,11	1431,09	1431,12	1431,18	1431,18	1431,23	1431,185	0,044533	0,001983	навчання			
26	20120124	105500	1431,12	28	1431,11	1431,09	1431,12	1431,18	1431,18	1431,23	1431,23	1431,26	0,02962	0,000877		MAE	MSE	
27	20120124	105600	1431,18	29	1431,09	1431,12	1431,18	1431,18	1431,23	1431,23	1431,27	1431,253	0,016647	0,000277	Похибки	5,388342143	37,33925714	
28	20120124	105700	1431,18	30	1431,12	1431,18	1431,18	1431,23	1431,23	1431,27	1431,27	1431,298	0,027849	0,000776	прогнозу			
29	20120124	105800	1431,23	31	1431,18	1431,18	1431,23	1431,23	1431,27	1431,27	1431,27	1431,291	0,021169	0,000448				
30	20120124	105900	1431,23	32	1431,18	1431,23	1431,23	1431,27	1431,27	1431,27	1431,25	1431,287	0,037074	0,001374				
31	20120124	110000	1431,27	33	1431,23	1431,23	1431,27	1431,27	1431,27	1431,25	1431,26	1431,262	0,00202	4,08E-06				
32	20120124	110100	1431,27	34	1431,23	1431,27	1431,27	1431,27	1431,25	1431,26	1431,36	1431,277	0,08262	0,006826				

Рис. 1.8: Результат прогнозування за авторегресійною моделлю в Excel

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.1.9). 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 стовбці після стовбця даних: стовбець індексів розріджених даних, стовбець розріджених даних та стовбець прогнозованих вихідних даних (для процедури відновлення прогнозу різниць, див. рис.1.10). 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

=ИНДЕКС($C$2:$C$13510;D3;1)-ИНДЕКС($C$2:$C$13510;D2;1).

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

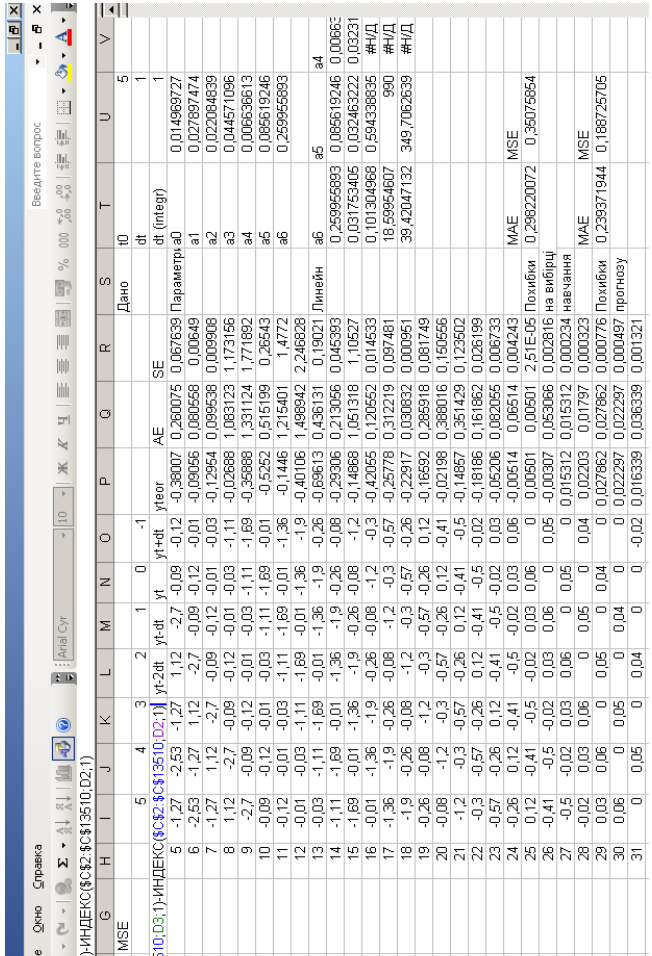


Рис. 1.9: Прогнозування за авторегресійною моделлю в Excel з 1-м порядком інтегрованості

Після цього необхідно замінити формули для формування лагових векторів. Формула для комірки I4 має вигляд: =ИНДЕКС($E$3:$E$13510;$H4-I$2;1). Усі комірки діапазону I5:01000 можна замінити, скопіювавши формулу з комірки I4 у ці комірки. Формули на прогнозній 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.

Оцінювання параметрів ARIMA-моделі та прогнозування часового ряду можна виконати за допомогою Garch Toolbox, яка включена в Econometrics Toolbox системи Matlab. Розглянемо та прокоментуємо просту matlab-функцію для прогнозування за допомогою Garch Toolbox:

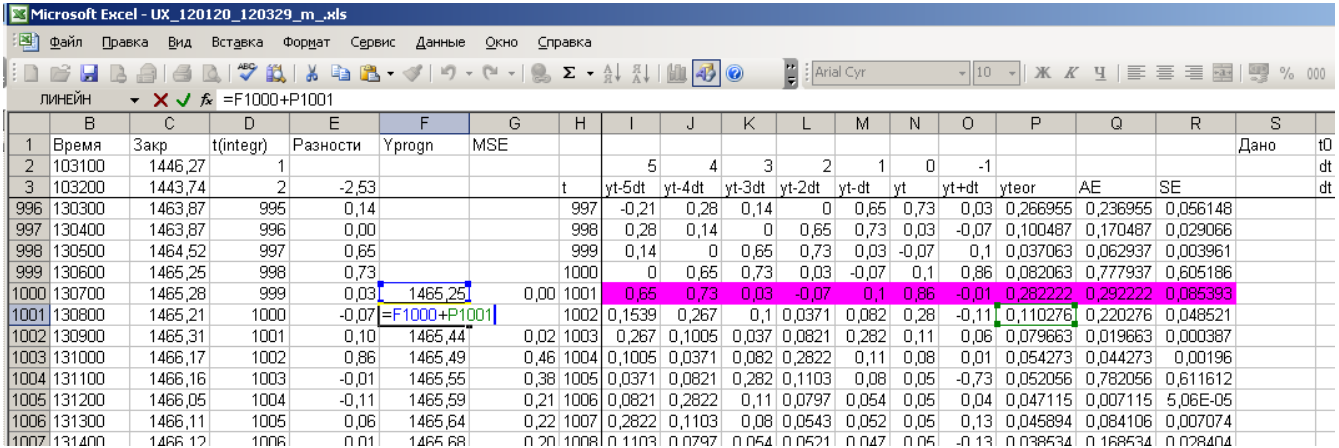
```
function garchpred_try2(infile, outfile);

%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
[sigmay,meany]=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;meany];
% зберігаємо ряд верхньої межі прогнозу
outfile_max=strrep(outfile,'.txt','_max.txt');
dlmwrite(outfile_max,[y;meany+sigmay];
% зберігаємо ряд нижньої межі прогнозу
outfile_min=strrep(outfile,'.txt','_min.txt');
dlmwrite(outfile_min,[y;meany-sigmay];
% будуємо 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 можливості 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`.



	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	Время	Закр	t(integr)	Разности	Yprogn	MSE												Дано	t0
2	103100	1446,27	1					5	4	3	2	1	0	-1					dt
3	103200	1443,74	2	-2,53			t	yt-5dt	yt-4dt	yt-3dt	yt-2dt	yt-dt	yt	yt+dt	yteor	AE	SE		dt
996	130300	1463,87	995	0,14			997	-0,21	0,28	0,14	0	0,65	0,73	0,03	0,266955	0,236955	0,056148		
997	130400	1463,87	996	0,00			998	0,28	0,14	0	0,65	0,73	0,03	-0,07	0,100487	0,170487	0,029066		
998	130500	1464,52	997	0,65			999	0,14	0	0,65	0,73	0,03	-0,07	0,1	0,037063	0,062937	0,003961		
999	130600	1465,25	998	0,73			1000	0	0,65	0,73	0,03	-0,07	0,1	0,86	0,082063	0,777937	0,605186		
1000	130700	1465,28	999	0,03	1465,25	0,00	1001	0,65	0,73	0,03	-0,07	0,1	0,86	-0,01	0,282222	0,292222	0,085393		
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	0,220276	0,048521		
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	0,019663	0,000387		
1003	131000	1466,17	1002	0,86	1465,49	0,46	1004	0,1005	0,0371	0,082	0,2822	0,11	0,08	0,01	0,054273	0,044273	0,00196		
1004	131100	1466,16	1003	-0,01	1465,55	0,38	1005	0,0371	0,0821	0,282	0,1103	0,08	0,05	-0,73	0,052056	0,782056	0,611612		
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	0,007115	5,06E-05		
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	0,045894	0,084106	0,007074		
1007	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	0,038534	0,168534	0,028404		

Рис. 1.10: Результат прогнозування за авторегресійною моделлю з 1-м порядком інтегрованості

Розділ 2

Використання нелінійних функцій, нейронних мереж при побудові прогнозних 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 квазіперіодичну, або змішану (стохастика + хаос-динаміка+детермінізм) основу [48–50]. У даній ситуації адекватним апаратом для вирішення завдань аналізу і прогнозування ринку цінних паперів можуть служити штучні нейронні мережі, що реалізовуються на базі концепцій штучного інтелекту. Програмно-математичною основою цих методів є нейронні мережі, що самоорганізуються [51–56].

Нейронна мережа – це сукупність нейронних елементів і зв'язків між ними (рис. 2.2). Основний елемент нейронної мережі – це формальний нейрон (рис. 2.1), що здійснює операцію нелінійного перетворення суми добутків вхідних сигналів на вагові коефіцієнти

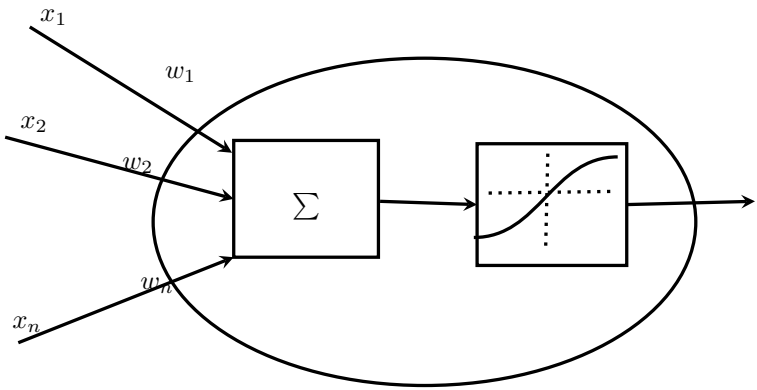


Рис. 2.1: Математична модель нейрона

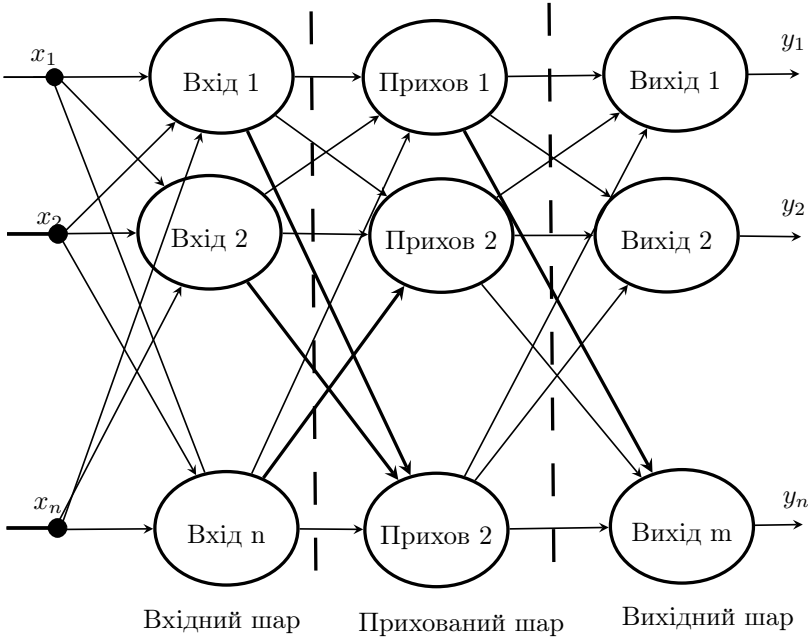


Рис. 2.2: Приклад нейронної мережі – багатошаровий персептрон

$$Y = F\left(\sum_{i=1}^{n} w_i \cdot x_i\right) \cdot F\left(WX\right), \tag{2.1}$$

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

В даний час методи штучних нейронних мереж вже довели свою високу ефективність при дослідженні фінансово-економічних систем. Штучні нейронні мережі є апаратом з області нейрокомп'ютингу (neural computing), області обчислювальних технологій, що останнім часом швидко розвиваються. Обчислювальні операції в таких мережах виконуються великим числом порівняно простих процесорних елементів (processing element). Структура мережі (network) тотожна математично визначеній структурі обчислювальної системи (графу мережі), в якій усі 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 [45]. 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 виступає набір параметрів, який дослідник розміщує 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 [45].

Алгоритм роботи мережі, що навчається без учителя, грунтується на навчанні змаганнями. Забезпечується така поведінка 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 символьним 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. Ця властивість дозволяє 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 оптимальну структуру нейронної мережі [6, 57, 58].

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 [54]. У 1949 р. Хебб запропонував правило навчання, яке стало математичною основою для навчання ряду нейронних мереж [53]. У 1957-1962 рр. Ф. Розенблатт запропонував і дослідив модель нейронної мережі, яку він назвав персептроном [55]. У 1959 р. В. Відроу и М. Хофф запропонували процедуру навчання для лінійного адаптивного елементу – AD ALINE. Процедура навчання отримала назву "Дельта правило" [54]. 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 дослідив нейронні

мережі, що самоорганізуються. Ряд авторів запропонували алгоритм зворотного поширення помилки, який став потужним засобом для навчання багатошарових нейронних мереж [53–55]. В даний час розроблено велике число нейросистем, які використовуються у різних областях: прогнозуванні фінансових показників, управлінні, діагностиці в медицині та техніці, розпізнаванні образів тощо [59, 60].

Для навчання мережі використовуються різноманітні алгоритми навчання і їх модифікації [61–65].

Методам аналізу і підвищення якості навчальних вибірок для нейромережевих прогнозних моделей присвячена робота [66]. 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 [67, 68] розроблений алгоритм навчання мережі, 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 багатошарових нейронних мереж авторів [67, 68] і алгоритмів їх навчання дозволив виявити ряд недоліків і проблем, що можуть виникнути:

- невизначеність у виборі числа шарів і кількості нейронних елементів у 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 нейронної мережі на пошук мінімуму функції средньоквадратичної похибки.

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 вихідних параметрів має бути визначена до початку навчання. У таких випадках метод АНМ доцільно доповнити генетичними алгоритмами.

Прогнозування з використанням інструментарію генетичних алгоритмів вперше було запропоновано в 70-х роках XX століття [52, 69–71].У другій половині 1980-х рр. 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 [72].

Однією з особливостей генетичних алгоритмів є складність інтерпретації розв'язку та програмна реалізація. Проте, перевагою є ефективність у пошуку глобальних мінімумів багатоекстремальних цільових функцій, оскільки при навчанні нейронних мереж досліджуються великі області допустимих значень вектору параметрів. Генетичні алгоритми дають можливість оперувати дискретними значеннями параметрів нейронних мереж, що може привести до скорочення загального часу навчання [6, 57, 58].

Методи індуктивного моделювання у задачах прогнозування.

Особливо відзначимо, що в 70-і роки увагу фахівців в області прогнозування привертали методи індуктивного моделювання. [73–75]. 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 [76], полягає у перекладенні на ЕОМ задачі відшукання моделі, яка найкраще описує процес, що досліджується. Пошук організовано, базуючись на еволюційному принципі, починаючи з найпростіших моделей, які основуються на опорній функції, що найчастіше має 2 аргументи, наприклад:

$$\begin{aligned} y &= a_0 + a_1 x_i x_j, \\ y &= a_0 + a_1 x_i + a_2 x_j, \\ y &= a_0 + a_1 x_i + a_2 x_j + a_3 x_i x_j, \\ y &= a_0 + a_1 x_i + a_2 x_j + a_3 x_i^2 + a_4 x_j^2 + a_5 x_i x_j. \end{aligned} \tag{2.2}$$

У загальному випадку, використовується поліном Колмогорова-Габора:

$$y = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n} a_{ij} x_i x_j + \sum_{i<j<k} a_{ijk} x_i x_j x_k. \tag{2.3}$$

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

$$y(t) = F(x_1(t), x_2(t), ..., x_m(t)). \tag{2.4}$$

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 $y(t + \Delta t) = f\left(x_i(t), x_j(t)\right)$, де Δt – період упередження. У якості функції f вибирається одна з опорних функцій, наприклад, задані рівняннями (2.2) або (2.3). Якщо вибрати внутрішнім критерієм якості моделей критерій мінімальності середньоквадратичного відхилення прогнозів $f\left(x_i(t_k), x_j(t_k)\right)$ 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 наступного 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 моделі, після чого точність почне падати.

У монографії [8] запропоновано нечіткий МГУА. У роботах [77, 78]розглянута лінійна інтервальна модель регресії

$$Y = A_0 x_0 + A_1 z_x + \ldots + A_n x_n, \tag{2.5}$$

де $A_i = (a_i, c_i)$ – нечіткі числа трикутного виду, де a_i – центр інтервалу, c_i – його ширина. $c_i \geq 0$, $x_1, x_2, \ldots, x_n$ – вхідні змінні ($x_i \in \mathbb{R}$). Тоді відгук Y моделі (2.5) буде теж нечітким числом з центром інтервалу

$$a_y = \sum a_i z_i = a^T \cdot z, \tag{2.6}$$

та шириною інтервалу

$$c_y = \sum c_i \mid z_i \mid = c^T \cdot \mid z \mid . \tag{2.7}$$

Щоб дана модель коректно описувала навчальну вибірку $\{(y_i, x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)})\}$, $i = 1..n$, необхідно знайти такі значення нечітких параметрів $A_i = (a_i, c_i)$ моделі регресії (2.5), щоб межі нечітких відгуків моделі $Y(x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)})$ 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:

$$min(c_0 \cdot n + c_1 \sum_{k=1}^{n} \mid x_i^{(k)} \mid + c_2 \sum_{k=1}^{n} \mid x_j^{(k)} \mid + c_3 \sum_{k=1}^{n} \mid x_j^{(k)} \cdot x_j^{(k)} \mid + \\ + c_4 \sum_{k=1}^{n} \mid x_i^{(k)} \mid^2 + c_5 \sum_{k=1}^{n} \mid x_j^{(k)} \mid^2), \tag{2.8}$$

при обмеженнях:

$$\begin{array}{l} a_0 + a_1 x_i^{(k)} + a_2 x_j^{(k)} + a_3 x_i^{(k)} \cdot x_j^{(k)} + a_4 \left(x_j^{(k)}\right)^2 + \left(a_5 x_{kj}\right)^2 - \\ - \left(c_0 + c_1 \mid x_i^{(k)} \mid + c_2 \mid x_j^{(k)} \mid + c_3 \mid x_i^{(k)} \cdot x_{kj} \mid + a_4 \mid \left(x_j^{(k)}\right)^2 \mid + a_5 \mid \left(x_j^{(k)}\right)^2 \mid \right) \le y_k \\ - \left(c_0 + c_1 \mid x_i^{(k)} \mid + c_2 \mid x_j^{(k)} \mid + c_3 \mid x_i^{(k)} \cdot x_{kj} \mid + a_4 \mid \left(x_j^{(k)}\right)^2 \mid + a_5 \mid \left(x_j^{(k)}\right)^2 \mid \right) \le y_k \\ - \left(c_0 + c_1 \mid x_i^{(k)} \mid + c_2 \mid x_j^{(k)} \mid + c_3 \mid x_i^{(k)} \cdot x_{kj} \mid + a_4 \mid \left(x_j^{(k)}\right)^2 \mid + a_5 \mid \left(x_j^{(k)}\right)^2 \mid \right) \le y_k \\ a_0 + a_1 x_i^{(k)} + a_2 x_j^{(k)} + a_3 x_i^{(k)} \cdot x_{kj} + a_4 \left(x_j^{(k)}\right)^2 + a_5 \left(x_j^{(k)}\right)^2 + \\ + \left(c_0 + c_1 \mid x_i^{(k)} \mid + c_2 \mid x_j^{(k)} \mid + c_3 \mid x_i^{(k)} \cdot x_j^{(k)} \mid + a_4 \mid \left(x_j^{(k)}\right)^2 \mid + a_5 \mid \left(x_j^{(k)}\right)^2 \mid \right) \ge y_k \end{array}, \tag{2.9}$$

де k – номер точки вхідних даних $k = 1, 2, \ldots, n$. 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.

У монографії [8] розглянуто методи нечіткої регресії типу (2.5) для нечітких чисел гаусового та дзвоноподібного виду, а також для регресії у вигляді ортогональних поліномів Чебишева та Лагерра та для тригонометричних поліномів. Зведення регресії до нечіткого виду дозволяє подолати відомі проблеми регресійного аналізу, як мультиколінеарність та 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.

Лабораторна робота № 4. Нейромережеві 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 виявляти залежності у даних. Щодо механізмів або методик формування наборів даних для побудови нейромережевих моделей прогнозування та здійснення прогнозів, то такі процедури часто описуються поверхнево у вітчизняних наукових публікаціях і нерідко є відсутніми навіть у сучасних реалізаціях нейромережевого 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 $y_{t+1} = f(y_t, y_{t-1}, y_{t-2}, \ldots, y_{t-m})$. Тому можна скористатись табличним процесором Excel, сформувати 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 обчислення прогнозного ряду.

```
function neural_prepare(infilename,nprognoz,dt,k)
y=dlmread(infilename);% 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
    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; % персептрон
% 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
    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(infilename,'.txt',['_dt_' num2str(dt) ...
                '_k_' num2str(k) '.txt']);
dlmwrite(outfilename,y_progn_vidn,'\n');
outfilename=strrep(outfilename,'.txt','.mat');
save(outfilename);

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.2.3).

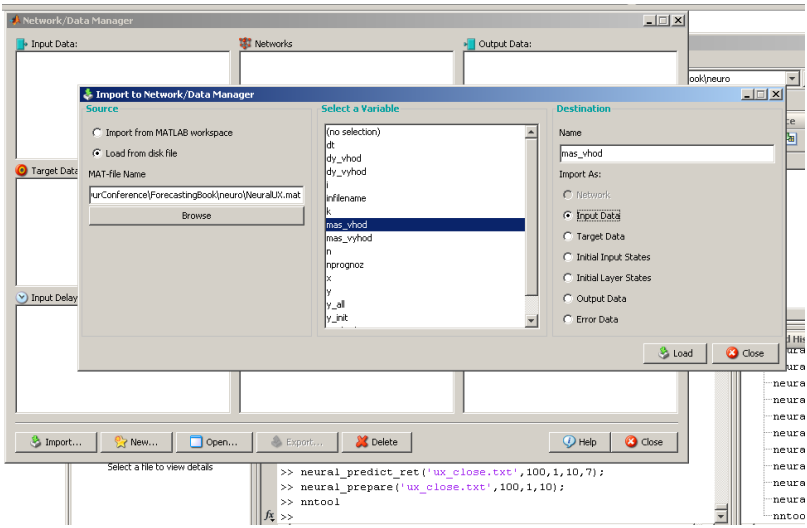


Рис. 2.3: Інтерфейс створення нейронної мережі

Необхідно імпортувати дані в систему. Для цього необхідно натиснути кнопку "Import". З'явиться вікно імпорту (показано на рис. 2.3).

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 вкажіть, що це вхідні дані (Input data). Для імпорту натисніть кнопку Load справа внизу. Після імпорту, з цього ж вікна можна імпортувати вихідний масив. Виберіть 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.2.4).

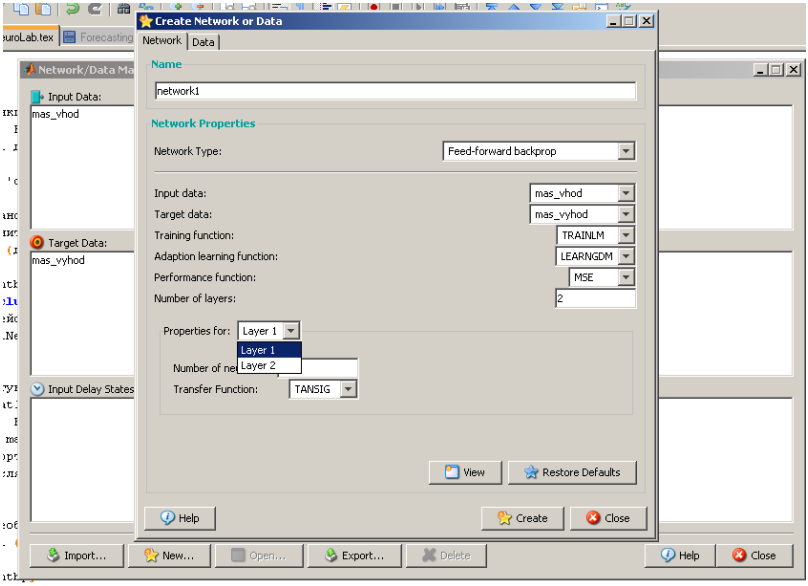


Рис. 2.4: Інтерфейс створення нейронної мережі

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), 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 Enter). 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 задання 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.2.5). На першій закладці зображена архітектура нейронної мережі. Нам необхідно перейти до закладки навчання (Train, див. рис. 2.6)

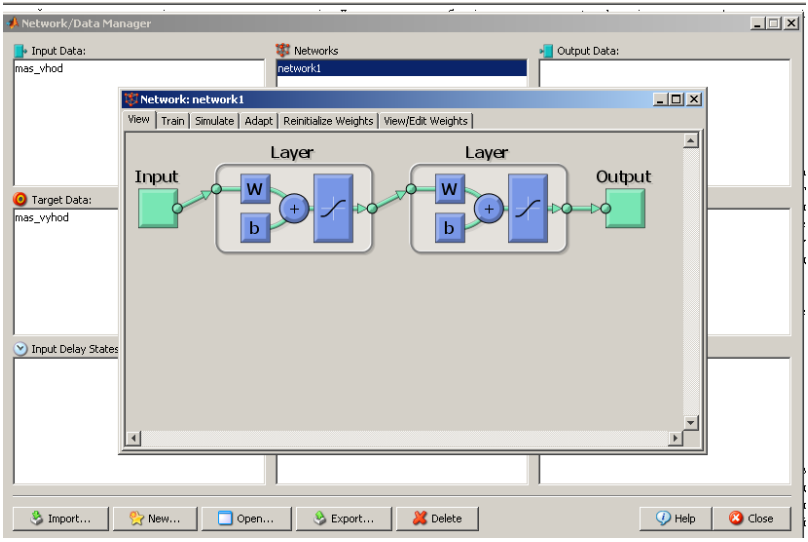


Рис. 2.5: Налаштування нейронної мережі

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.2.7). Процес навчання часто займає значний час. Тому під час навчання 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 переривання 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".

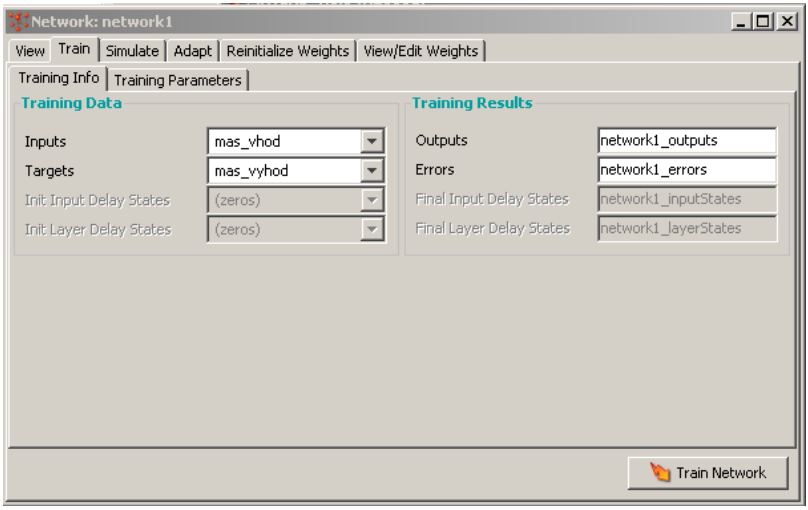


Рис. 2.6: Налаштування навчання нейронної мережі

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

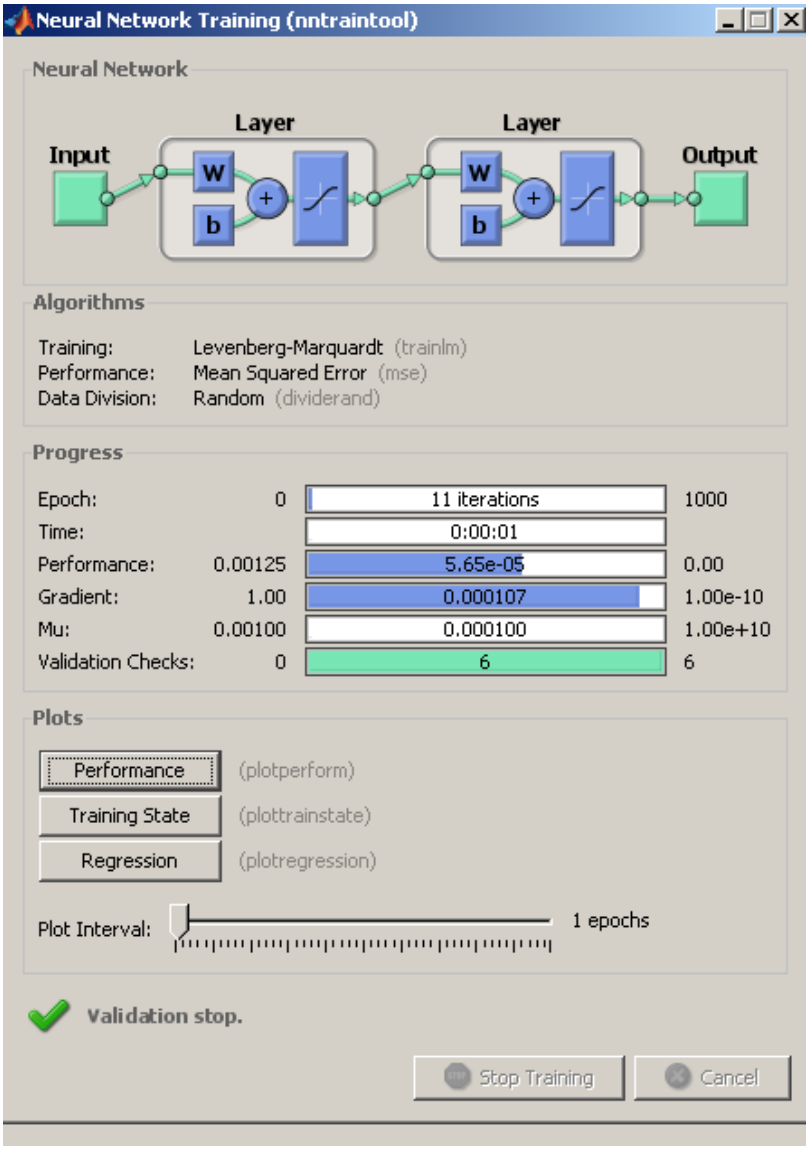


Рис. 2.7: Процес навчання нейронної мережі

```
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(infilename,'.txt', ...
    ['_dt_' num2str(dt) '_k_' num2str(k) '.txt']);
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 навченої мережі (збережена програмою nntool після навчання). Після завантаження файлів буде обчислено остаточний прогнозний 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(infilename,nprognoz,dt,k,network_type)

infilename - ім'я вхідного файла, рядок символів,

nprognoz - 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).

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

`http://prognoz.ck.ua/neuro.rar`

Розділ 3

Випадкові процеси та методи побудови марківських моделей при прогнозуванні.

У даному розділі розглядаються моделі для прогнозування фінансових рядів, які базуються на ланцюгах Маркова. На відміну від аналогічних досліджень з фінансового прогнозування [79], не будуються моделі регресійних рівнянь, а пропонується модель машинного навчання (приховані марківські моделі, ПММ) для аналізу фінансових часових рядів. Також проводиться аналіз попередніх робіт, розкриваються недоліки розроблених раніше моделей та пропонуються шляхи удосконалення, зокрема, використання неперервних сумішей функцій розподілу Гауса та відповідна модифікація класичного алгоритму максимізації правдоподібності. В роботі [80] 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.

Розглянемо три основні проблеми задач прогнозування

часових рядів, які пропонується вирішити за допомогою моделей ПММ. 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 короткочасними 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 [80] націлена на подолання вищезазначених проблем за допомогою системи передбачення, яка базується на прихованих марківських моделях (Hidden Markov Model, HMM, далі - ПММ) [81–84]. 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 нестаціонарності результуючого часового ряду.

Новий експоненційно зважений алгоритм МП дає можливість подолати другу та третю з вищезазначених проблем. Даний алгоритм надає більшого значення новим записам у навчальній вибірці у порівнянні зі старими, налаштовуючи вагові коефіцієнти на зміну волатильності. Баланс між довгими та короткими рядами знайдено в процесі комбінованої торгівлі та передбачення.

Базуючись на експоненціально зваженому алгоритму МП, пропонується подвійно зважений алгоритм МП. Даний алгоритм 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 [80].

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.

Приховані марківські моделі

Почнемо з визначення властивостей марківського процесу, після чого розглянемо характеристики прихованих марківських моделей. В роботах Рабінера [81] 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 $S = \{1, 2, \cdots, s\}$. 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.

У 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: 1) обмежений горизонт

$$P(X_{t+1} = s_k | X_1, \ldots, X_t) = P(X_{t+1} = s_k | X_t) \tag{3.1}$$

2) часова інваріантність

$$= P(X_2 = s_k | X_1) \tag{3.2}$$

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

$$a_{ij} = P(X_{t+1} = s_j | X_t = s_i) \tag{3.3}$$

a_{ij} є ймовірностями, отже: $a_{ij} \geq 0$;

$$\forall i, j \sum_{j=1}^{N} a_{ij} = 1 \tag{3.4}$$

Також необхідно зафіксувати ймовірність кожного стану на початку певного часового інтервалу та початковий розподіл ймовірностей:

$$\pi_i = P(X_1 = s_i) \tag{3.5}$$

при

$$\sum_{i=1}^{N} \pi_i = 1. \tag{3.6}$$

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 вище, є обмеженою, тому ми розширюємо її для покращення можливостей в приховану марківську модель (далі ПММ). В ПММ процес, який генерує послідовність, що розглядається, 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 ймовірнісною функцією. В момент часу t, послідовність станів o_t генерується ймовірнісною функцією $b_j(o_t)$, яка зв'язана зі станом j, та має такий вигляд:

$$b_j(o_t) = P(o_t|X_t = j). \tag{3.7}$$

Розглянемо приклад прихованої марківської моделі, яка описує діяльність інвестора з передбачення змін ринкової ціни, основану 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 експерта. 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 ймовірностей 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 П.

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. 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 здатний приймати вхідні послідовності різної довжини, без необхідності зводити навчання до "шаблону". Удосконалений авторами роботи [80] алгоритм МП дає можливість збільшити "вагу" більш свіжих даних при навчанні, а також відкривається можливість налаштовувати баланс між вимогами до чутливості та точності.

ПММ застосовуються під час розпізнання мови, у 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 $O = (o_1, ..., o_T)$, 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 $(1, \cdots, N)$, 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, так як 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. $S = \{1, \ldots, N\}$ – множина станів. Стан системи в момент часу t позначимо s_t.

2. $K = \{k_1, \ldots, k_M\}$ – 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

$$\pi_i = P(s_1 = i). \tag{3.8}$$

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

$$a_{ij} = P(s_{t+1}|s_t); 1 \leq \quad i; j \quad \leq \quad N \tag{3.9}$$

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

$$b_j(o_t) = P(o_t|s_t = j) \tag{3.10}$$

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 базуються на відомих даних та отримуємться 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 T:

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:

1. Ініціалізація: $\alpha_1(i) = \pi_i b_i(d_1)$.
2. Крок індукції: $\alpha_{t+1}(j) = \left[\sum_{i=1}^{n} \alpha_t(i) a_{ij}\right] b_j(d_{t+1})$.
3. Після T ітерацій підрахуємо приховану ймовірність:

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

Алгоритм знаходження найбільш ймовірних послідовностей станів:

1. Ініціалізація: $\delta_1(i) = \pi_i b_i(d_1),\ \psi_1(i) = \{\}$.
2. Крок індукції:

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

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

3. Після T ітерації визначити:

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

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 [83]. 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 "викидами" мається на увазі 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, основаного 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 за показником волатильності, то такий поділ буде суб'єктивним, а ряди з проміжними значеннями волатильності не обов'язково демонструватимуть однакові властивості. В роботі [80] наводяться моделі з сумішами розподілів, в яких налаштовуються окремі параметри кожного компоненту суміші.

У 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.

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

$$b_j(o_t) = \sum_{k=1}^{M} w_{jk} b_{jk}(o_t), j = 1, ..., N, \tag{3.11}$$

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

$$\sum_{k=1}^{M} w_{jk} = 1, j = 1, \ldots, N; w_{jk} \geq 0; j = 1, \ldots, N; k = 1, \ldots, M. \tag{3.12}$$

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

$$b_{jk}(o_t) = N(o_t, \mu_{jk}, \Sigma_{jk}). \tag{3.13}$$

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 IBM, є сенс включити у вхідні дані ціни акцій Microsoft та інших високотехнологічних компаній.

Звичайна функція розподілу Гауса має наступний вигляд:

$$b_{jk}(o_t) = N(o_t, \mu_{jk}, \Sigma_{jk}) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \tag{3.14}$$

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

$$\begin{aligned} &b_{jk}(o_t) = N(o_t, \mu_{jk}, \Sigma_{jk}) = \\ &= \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma_{jk}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(o_t - \mu_{jk})^T \Sigma_{jk}^{-1}(o_t - \mu_{jk})\right), \end{aligned} \tag{3.15}$$

де o_t – вектор явних значень прогнозованого ряду. Пропонується модифікація алгоритму Баума-Велча [80] для випадку багатовимірних часових рядів з дійсними значеннями. Вводяться наступні допоміжні змінні:

$$\delta_i(t) = \sum_{m=1}^{M} \delta_{im}(t) = \sum_{m=1}^{M} \frac{1}{P} a_i(t-1) a_{ji} w_{im} b_{im}(o_i) \beta_i(t). \tag{3.16}$$

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:

$$\nu_{j,k}(t) = \frac{\alpha_j(t)\beta_j(t)}{\sum_{j=1}^{N}\alpha_j(t)\beta_j(t)} \cdot \frac{w_{jk}N(o_i,\mu_{jk},\sigma_{jk})}{\sum_{k=1}^{M}w_{jk}N(o_i,\mu_{jk},\sigma_{jk})} \tag{3.17}$$

З новими допоміжними змінними запишемо модифіковані формули уточнення параметрів:

$$\mu'_{im} = \frac{\sum_{t=1}^{T}\delta_{im}(t)\,o_t}{\sum_{t=1}^{T}\delta_{im}(t)} \tag{3.18}$$

$$\sigma'_{im} = \frac{\sum_{t=1}^{T}\delta_{im}(t)\,(o_t-\mu'_{im})\,(o_t-\mu'_{im})'}{\sum_{t=1}^{T}\delta_{im}(t),} \tag{3.19}$$

$$w'_{im} = \frac{\sum_{t=1}^{T}\delta_{im}(t)\,o_t}{\sum_{t=1}^{T}\delta_{i}(t)}, \tag{3.20}$$

$$a'_{ij} = \frac{\sum_{t=1}^{T}a_i(t)\,\alpha_i(t)\,a_{ij}b_j(o_{t+1})\,\beta_i(t+1)}{\sum_{t=1}^{T}\alpha_i(t)\,\beta_i(t)}. \tag{3.21}$$

В даному розділі розглянуто ПММ, адаптовану 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 сумішшю 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, показують високі результати в обробці та розпізнаванні 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 нові дані, залишаючи у вибірці навчання і старі, віддалені у часі фрагменти, але використовуючи їх зі зменшеною довірою. Усвідомлюючи, що зміна важливості даних з часом та мотивована цим концепція експоненційного ковзного середнього (далі ЕКС) фінансових часових рядів, в роботі [80] 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, що містять проміжні стани. Для знаходження кращого балансу між короткочасними та довготривалими трендами, авторами [80] пропонується подвійно зважений алгоритм МП, який має властивості налаштовувати чутливість за допомогою включення інформації про волатильність.

Доведено збіжність експоненціально зваженого алгоритму МП за допомогою зваженої версії теореми про максимізацію правдоподібності.

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

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

$$P(X|\Omega) = \prod_{i=1}^{N} p(x_i|\Omega) = L(\Omega|X) \tag{3.22}$$

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:

$$\Omega* = \arg\max_{\Omega} L(\Omega|X). \tag{3.23}$$

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:

$$\begin{aligned} &Q\left(\Theta, \Theta^{(i-1)}\right) = E\left[\log p\left(X, Y|\Theta\right)|X, \Theta^{(i-1)}\right] = \\ &= \int\limits_{y\in\gamma} \log p\left(X, y|\Theta\right) f\left(y|X, \Theta^{(i-1)}\right) dy. \end{aligned} \tag{3.24}$$

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. Нагадуємо, 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 часово-залежні вагові коефіцієнти η, очікуючи відобразити інформацію про часовий параметр.

Нова форма функції правдоподібності Q^* матиме наступний вигляд:

$$Q^*\left(\Theta, \Theta^{(i-1)}\right) = E\left[\log \eta p\left(X, Y|\Theta\right)|X, \Theta^{(i-1)}\right] = \\ = \int\limits_{y\in\gamma} \log \eta p\left(X, y|\Theta\right) f\left(y|X, \Theta^{(i-1)}\right) dy. \tag{3.25}$$

Аналогічно формулам (3.18)-(3.21) виводяться ітераційні формули уточнення параметрів моделі при зваженому врахуванні фрагментів даних.

Розділ 4

Методика прогнозування на основі складних ланцюгів Маркова

4.1 Технологія прогнозування на основі складних ланцюгів Маркова

Розглянемо поняття ланцюгів Маркова. Припустимо існує послідовність дискретних станів певної системи. З цієї послідовності 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) [85, 86]. Кількість станів в передісторії називається порядком ланцюга Маркова.

Теорія простих ланцюгів Маркова широко викладена в літературі, наприклад в [85, 86], що стосується ланцюгів Маркова вищих порядків, то в літературі даний термін використовується рідко. Наприклад, в статті [87] автор використовує термін "ланцюги Маркова" для процесів з пам'яттю об'ємом декілька попередніх значень. У роботах [88, 89] пропонується використання класу періодичних ланцігів Маркова для побудови прогнозів часових рядів технічних систем.

Ланцюг Маркова порядку вище першого можна звести до простого ланцюга Маркова за допомогою узагальнення поняття стану: "наявний стан", як ряд послідовних станів [40]. 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 сучасних методів прогнозування максимально використати інформацію з відомого відрізку 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 s 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 випадку всі можливі прирости вихідного ряду необхідно класифікувати на s класів [90]. 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 послідуюче відновлення можна здійснити для різних приростів часу Δt. Для ефективного використання інформації, представленій в наявному часовому ряді, прогнозування здійснюється для різних приростів часу Δt =1,2,4,8..., або більш складної ієрархії приростів, яка розглянута в даній роботі в підрозділі 4.3, 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 [40], або комбінації лінійного тренду та гармонійних коливань [91]. Дана процедура розглядається в розділі 5 даної роботи.

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 (межі крайніх лівого та правого стану 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. Проблему 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 прецендентів, а по-друге, усереднення відхилень 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.

Із рис. 4.1-4.4 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.

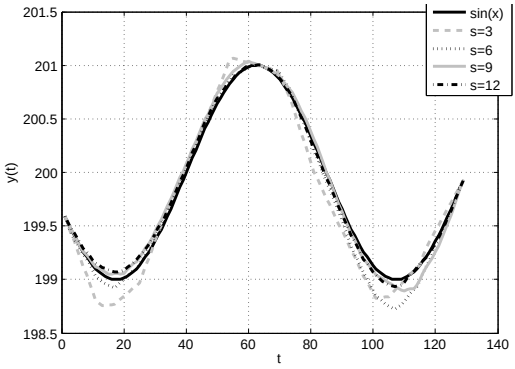


Рис. 4.1: Дискретизація та відновлення $y = sin(x)$. Проста ієрархія

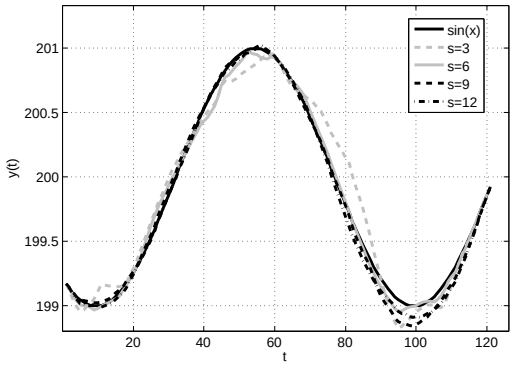


Рис. 4.2: Дискретизація та відновлення $y = sin(x)$. Складна ієрархія

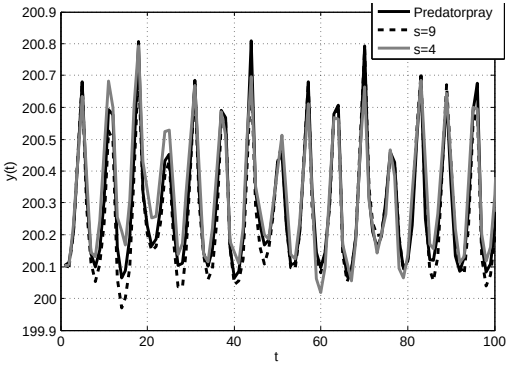


Рис. 4.3: Дискретизація та відновлення для моделі Хижак-жертва. Проста ієрархія

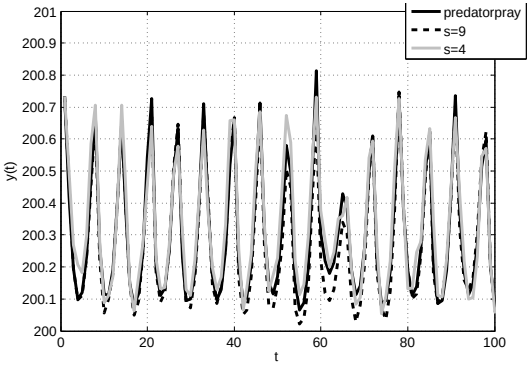


Рис. 4.4: Дискретизація та відновлення для моделі Хижак-жертва. Складна ієрархія

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 4.3.

3) Величина порогу δ, та мінімальна кількість представників кожного стану СЛМ N_{min}.

Отже, алгоритм побудови прогнозу зводиться до наступних кроків:

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. При цьому треба:

   (a) обчислити прирости з дискретизацією Δt;

   (b) перевести ряд приростів у ряд номерів станів $\{1, \ldots, s\}$;

   (c) 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 тренду, побудованого по всім попередньо відомим точкам.

Ряд, склеєний з лінійним трендом, є результатом прогнозування.

4.2 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, зв'язавши рядки матриці зі станами, з яких відбувається перехід, а стовбці (елементи цих рядків) – зі станами, у які визначається ймовірності переходу.

Наприклад, для трьох станів можна записати матрицю ймовірностей переходів A:

$$A = \begin{pmatrix} 0,0967 & 0,0462 & 0,0366 & 0,0648\,0,113 \\ 0,1628 & 0,1354 & 0,1317 & 0,18650,1477 \\ 0,4765 & 0,5621 & 0,6658 & 0,56390,4521 \\ 0,1612 & 0,1937 & 0,133 & 0,13860,1799 \\ 0,1048 & 0,0627 & 0,033 & 0,04620,1073 \end{pmatrix},$$

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 Байеса, 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 розмірності (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 з кількістю 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 99, 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. Із усіх можливих наступних станів вибирається стан з максимальною ймовірністю. У випадку декількох станів з максимальною ймовірністю процес прийняття рішення буде описаний далі.

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

Далі по одержаному ряду станів та відомому початковому значенню y_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 прогнозу

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 обчислююватись 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.

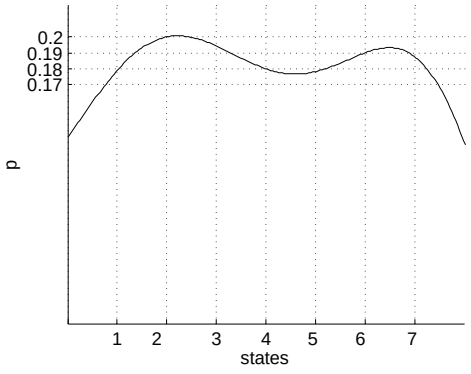


Рис. 4.5: Розподіл ймовірностей на певному кроці

Як видно з рис.4.5, для того, щоб не допустити втрати станів, ймовірності яких обчислені з похибкою (наприклад, стан 6 на рис.4.5), 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.4.5), 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 вибрати 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 (багатомодальний розподіл), то необхідно розглянути 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. 4.5) 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 центру розподілу - стан 4).

Якщо прийняти ($\delta = 0,01$), кластерами найбільш ймовірних станів будуть множини $\{2,3\}$ та $\{6,7\}$. Проміжок між станами 4 та 5 необхідно взяти як центр розподілу (стан, що відповідає нульовому відхиленню, мав би знаходитись між станами 4 та 5), таким чином 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 сценаріїв, 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.

4.3 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 ряді 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.

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: $y^k_{\Delta t_k}, y^k_{2\Delta t_k}, ..., y^k_N; k = 1, 2, ..., K_{max}$ для кроків дискретизації $\Delta t_1 = 1; \Delta t_2, ...,\Delta t_{K_{max}}$ відповідно.

Нехай процедура склеювання проведена для сукупності кроків $\Delta t_1 = 1; \Delta t_2; ...; \Delta t_{K_{max}}$ та отриманий ряд значень $X^k_0, X^k_1, ...X^k_N$ з інтервалом дискретизації Δt_k. Тоді продовження процедури склеювання та перехід до ряду значень $X^{k-1}_0, X^{k-1}_1, ...X^{k-1}_N$ відбувається за алгоритмом:

початкове значення:

$$x^{k+1}_0 = x^k_0 = x_0;$$

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

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

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

$$x_{\Delta t_{k+1}+1}^{k+1} = x_{\Delta t_{k+1}}^{k+1} + \left(x_{\Delta t_{k+1}+1}^k - x_{\Delta t_{k+1}}^k\right) +$$
$$+1 \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2\cdot\Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k\right)}{\Delta t_{k+1}};$$
$$x_{\Delta t_{k+1}+2}^{k+1} = x_{\Delta t_{k+1}}^{k+1} + \left(x_{\Delta t_{k+1}+2}^k - x_{\Delta t_{k+1}}^k\right) +$$
$$+2 \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2\cdot\Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k\right)}{\Delta t_{k+1}};$$
$$\ldots$$
$$x_{2\Delta t_{k+1}-1}^{k+1} = x_{\Delta t_{k+1}}^{k+1} + \left(x_{2\Delta t_{k+1}-1}^k - x_{\Delta t_{k+1}}^k\right) + \quad \cdots$$
$$+ (\Delta t_{k+1} - 1) \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2\cdot\Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k\right)}{\Delta t_{k+1}};$$
$$x_{2\Delta t_{k+1}}^{k+1} = x_{\Delta t_{k+1}}^{k+1} + \left(x_{2\Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k\right) +$$
$$+\Delta t_{k+1} \cdot \frac{x_{\Delta t_{k+1}}^{k+1} y_{2\Delta t_{k+1}}^{k+1} - \left(x_{2\cdot\Delta t_{k+1}}^k - x_{\Delta t_{k+1}}^k\right)}{\Delta t_{k+1}} \equiv$$
$$\equiv x_{\Delta t_{k+1}}^{k+1} \left(1 + y_{2\Delta t_{k+1}}^{k+1}\right);$$

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

$$
\begin{aligned}
&x^{k+1}_{N-\Delta t_{k+1}+1} = x^{k+1}_{N-\Delta t_{k+1}} + \left(x^{k}_{N-\Delta t_{k+1}+1} - x^{k}_{N-\Delta t_{k+1}}\right) + \\
&+1 \cdot \frac{x^{k+1}_{N-\Delta t_{k+1}} y^{k+1}_{N} - \left(x^{k}_{N} - x^{k}_{N-\Delta t_{k+1}}\right)}{\Delta t_{k+1}}; \\
&x^{k+1}_{N-\Delta t_{k+1}+2} = x^{k+1}_{N-\Delta t_{k+1}} + \left(x^{k}_{N-\Delta t_{k+1}+2} - x^{k}_{N-\Delta t_{k+1}}\right) + \\
&+2 \cdot \frac{x^{k+1}_{N-\Delta t_{k+1}} y^{k+1}_{N} - \left(x^{k}_{N} - x^{k}_{N-\Delta t_{k+1}}\right)}{\Delta t_{k+1}}; \\
&\ldots \\
&x^{k+1}_{N-1} = x^{k+1}_{N-\Delta t_{k+1}} + \left(x^{k}_{N-1} - x^{k}_{N-\Delta t_{k+1}}\right) + \\
&+ (\Delta t_{k+1} - 1) \cdot \frac{x^{k+1}_{N-\Delta t_{k+1}} y^{k+1}_{N} - \left(x^{k}_{N} - x^{k}_{N-\Delta t_{k+1}}\right)}{\Delta t_{k+1}}; \\
&x^{k+1}_{N} = x^{k+1}_{N-\Delta t_{k+1}} + \left(x^{k}_{N} - x^{k}_{N-\Delta t_{k+1}}\right) + \\
&+\Delta t_{k+1} \cdot \frac{x^{k+1}_{N-\Delta t_{k+1}} y^{k+1}_{N} - \left(x^{k}_{N} - x^{k}_{N-\Delta t_{k+1}}\right)}{\Delta t_{k+1}} \equiv \\
&\equiv x^{k+1}_{N-\Delta t_{k+1}} \left(1 + y^{k+1}_{N}\right).
\end{aligned}
$$

Цю процедуру проводимо для $k = 1, 2, 3, \ldots, K_{\max}$, починаючи з $k = 1$. Початковий ряд (при $k = 1$ для $\Delta t_1 = 1$)$x^1_0, x^1_1, \ldots, x^1_N$ знаходимо за алгоритмом:

$$
\begin{aligned}
&x^1_0 = x_0; \\
&x^1_1 = x^1_0 \left(1 + y^1_1\right); \\
&x^1_2 = x^1_1 \left(1 + y^1_2\right); \\
&\ldots \\
&x^1_N = x^1_{N-1} \left(1 + y^1_N\right).
\end{aligned}
$$

Розглянемо процедуру склеювання часового ряду на прикладі фрагменту індексу PFTS:

На рис.4.6 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 Δt=1,2,4 (рис.1, пунктирна лінія) "підтягуються" до точок нового ряду з дискретизацією Δt=8 (рис. 1, штрихпунктирна лінія). В результаті отримується ряд, склеєний з $\Delta t 1, 2, 4, 8$ (рис. 4.6, суцільна лінія). Як видно з рисунку, характерні риси прогнозу при $\Delta t = 4$ зберігаються після склеювання з прогнозним рядом при $\Delta t = 8$.

Ієрархія приростів дискретизацій, як степені числа 2 – не єдиний спосіб побудови ієрархії дискретизацій. Для покращення прогнозу можна використовувати більш часту

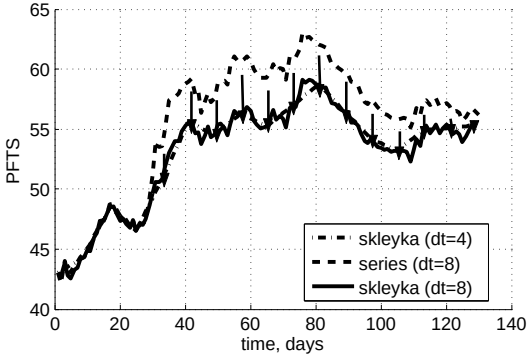


Рис. 4.6: Процес склеювання двох ієрархій часового ряду

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 приростів Δt_i, таким чином використавши більше інформації, наявної у часовому ряді.

Припустимо, що довжина ряду, який необхідно розкласти на ієрархії – N. Взявши послідовність перших простих чисел p_1, p_2, ..., 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 повинна бути максимізована.

Задача полягає в тому, щоб знайти послідовність простих чисел та відповідні їх степені, для яких, по-перше, добуток $p_1^{m1}\cdot p_2^{m2}\cdot\dots\cdot p_n^{mn} = N_1$ 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:

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

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

Вибираємо $k = 4$:

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

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.

Вибір оптимальних параметрів алгоритму прогнозування

Описаний вище алгоритм має параметри, які необхідно підбирати експериментально. Для цього проводиться процедура оптимізації параметрів, проводячи тестові прогнози для відомої частини ряду та оцінюючи результат.

Вихідний ряд розбивається на 2 частини, першу приймаємо як навчальну вибірку, а іншу як тестову. Будемо проводити тестові прогнози при різних значеннях параметрів та оцінювати 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:
   (a) Для кожної пари параметрів (r, s), при яких будуть перевірятись прогнози, та для декількох початкових точок виконуємо наступні дії:
      i. Обчислюємо прогноз на даному частотному рівні 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.

4.4 Прогнозування часових рядів з точним періодом

При експериментах на рядах з точним періодом, наприклад, залежності $y(t)$=sin$(at+b)$, оптимальний порядок ланцюга Маркова r прямує до максимально можливого значення, оскільки ряд станів містить абсолютно періодичні фрагменти. Але цей параметр обмежений тим, що для визначення значення елементів матриці ймовірностей переходів кожен перехід між станами повинен бути представлений достатньою кількістю повторень у ряді станів. Для цього необхідно, щоб кількість елементів матриці була б однакового порядку з розміром вибірки станів. Якщо припустити, що в середньому кількість переходів між станами на кожну комірку матриці повинні бути k а 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, як видно з рис. 4.8, обумовлені похибкою квантування на Δt, більших періоду синусоїди. На рис. 4.7 у вибірці навчання представлена достатня кількість періодів для відновлення на великих ієрархіях, а на рис. 4.8 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.

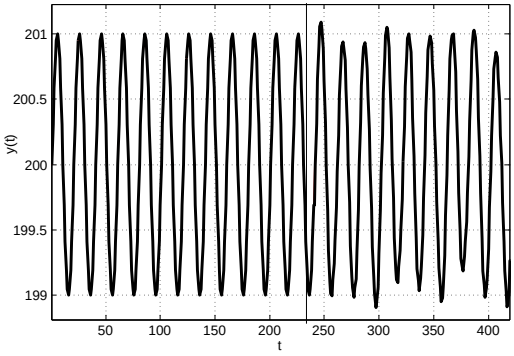


Рис. 4.7: Прогнозування y=$\sin\left(\frac{2\pi}{20}t\right)$ при $k = 5$, $s = 9$

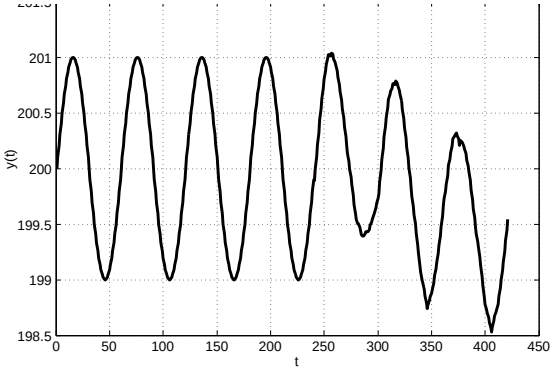


Рис. 4.8: Прогнозування $y = sin\left(\frac{2\pi}{60}t\right)$ при $k = 2$, $s = 9$

Прогнозування рядів динамічного хаосу

Для налаштування методики, за допомогою моделі детермінованого хаосу типу "Хижак-жертва" з параметрами, близькими до критичного режиму, були згенеровані тестові ряди. Дані ряди мають коротку пам'ять, так як генеруються рекурентними співвідношеннями. На рис.4.9 рис.4.10 зображено прогноз таких рядів.

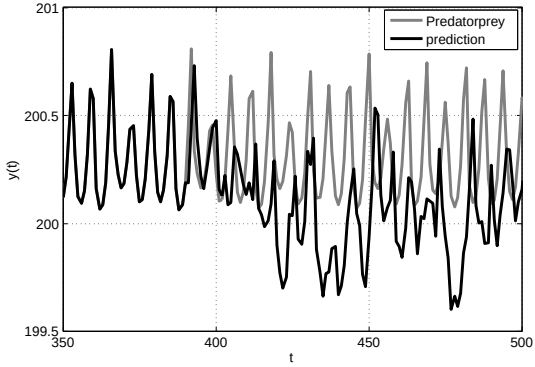


Рис. 4.9: Прогнозування ряду моделі "Хижак-жертва" s=5

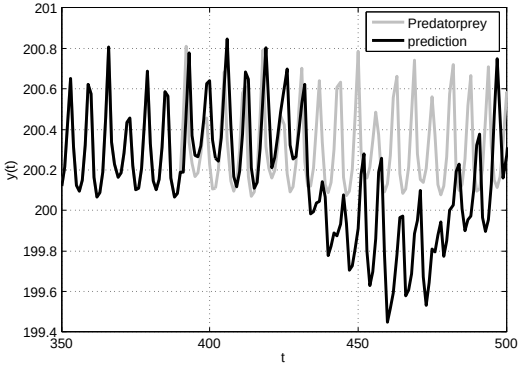


Рис. 4.10: Прогнозування ряду моделі "Хижак-жертва" $s = \sqrt{\Delta t}$

На рис.4.10 зображено прогноз при динамічному зростанні s пропорційно зростанню величини 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 прогнозних значень.

Експериментальна робота з рядами динамічного хаосу показала, що запропонована методика дозволяє виявляти закономірності у модельних часових рядів, навіть якщо такі закономірності приховані від статистичних методів (прогнозування рядів детермінованого хаосу). Виведене оптимальне співвідношення порядку ланцюгу Маркова r та кількості станів s, зроблено попередній підбір оптимальних параметрів.

4.5 Багатосценарний підхід до прогнозування рядів фінансово-економічної динаміки

У даному розділі представлені результати прогнозування світових фондових індексів, які доступні на сайті `http:\\ finance.yahoo.com`. Експерименти показали, що при різній довжині навчальної вибірки, прогнозні криві дещо відрізняються (див. рис.4.11, 4.12, 4.13 та 4.14). Ряди прогнозів для індексу Dow Jones Industrial Average (далі DJIA) зображено на рис.4.11 є більш корельовані між собою, ніж прогнози індексу FTSE 100, зображеного на рис. 4.13. На рисунках 4.12 та 4.14 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 таких прогнозів на одному графіку пропонується процедура нормалізації. Нормалізовані значення ряду будуть знаходитись у інтервалі [0...1] та обчислюються за наступною формулою:

$$y_n(t) = \frac{y(t) - min\,(y(t))}{max\,(y(t)) - min\,(y(t))}. \tag{4.1}$$

Нормалізовані прогнозні ряди показані на рис. 4.15 (Американські ринки), рис. 4.16 (Європа, зліва ринки розвинутих країн, справа: ринки країн PIIGS), рис. 4.17 (азіацькі країни). Усі рисунки містять ряд усередненого прогнозу, які зображені

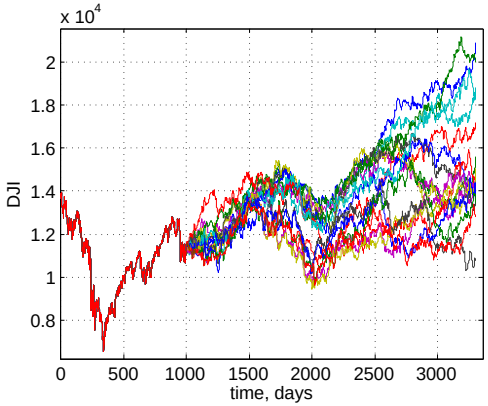


Рис. 4.11: Dow Jones Industrial Average - DJIA (США). Ряд прогнозів, обчислених з різною довжиною навчальної вибірки

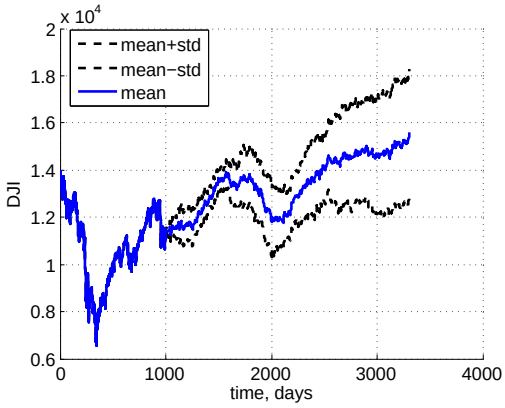


Рис. 4.12: Dow Jones Industrial Average - DJIA (США). Середнє значення та стандартне відхилення прогнозних кривих.

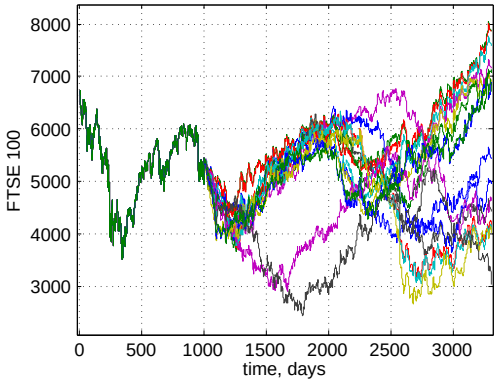


Рис. 4.13: Прогноз індексу FTSE 100 (Великобританія). Ряд прогнозів, обчислених з різною довжиною навчальної вибірки

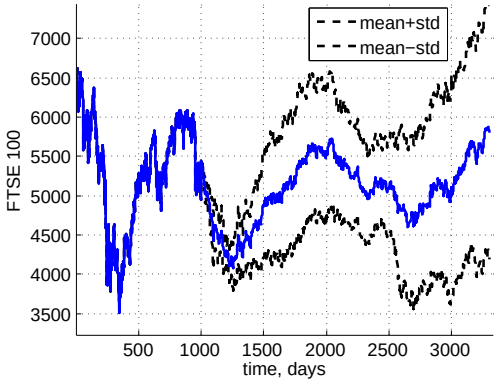


Рис. 4.14: Прогноз індексу FTSE 100 (Великобританія). Середнє значення та стандартне відхилення прогнозних кривих

чорною лінією та є середньозваженими з вагами ВВП відповідних країн.

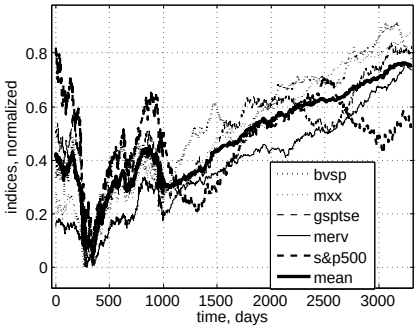


Рис. 4.15: Нормалізовані усереднені прогнози американських фондових індексів. Бразилія (BVSP), Мексика (MXX), Канада (GSPTSE), Аргентина (MERV), США (S&P 500)

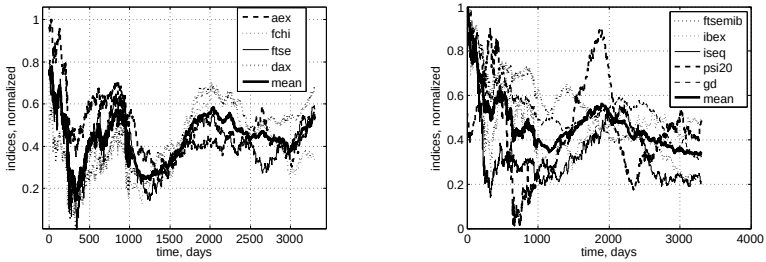


Рис. 4.16: Нормалізовані усереднені прогнози європейських фондових індексів. а) Розвинуті країни: FTSE (Великобританія), DAX (Німеччина) FCHI (Франція), AEX (Нідерланди); б) Португалія (PSI20), Італія (FTSEMIB), Ірландія (ISEQ), Греція (GD) та Іспанія (IBEX)

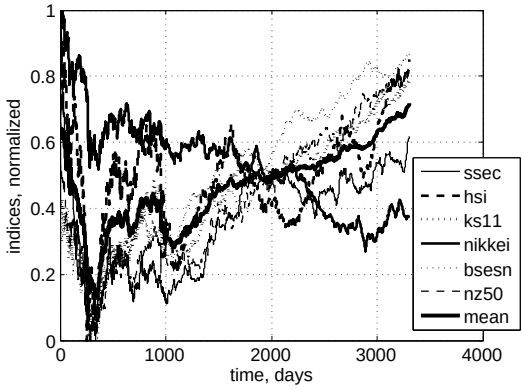


Рис. 4.17: Нормалізовані усереднені прогнози фондових індексів Азії. Китай (SSEC, HSI), Корея (KS11), Японія (NIKKEI), Індія (BSESN), Нова Зеландія (NZ50)

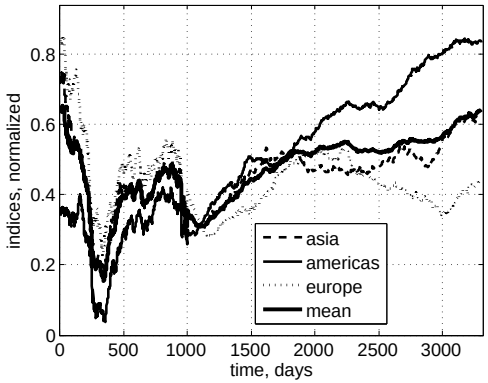


Рис. 4.18: Усереднені та нормалізовані прогнози індексів світових фондових ринків

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. 4.19)

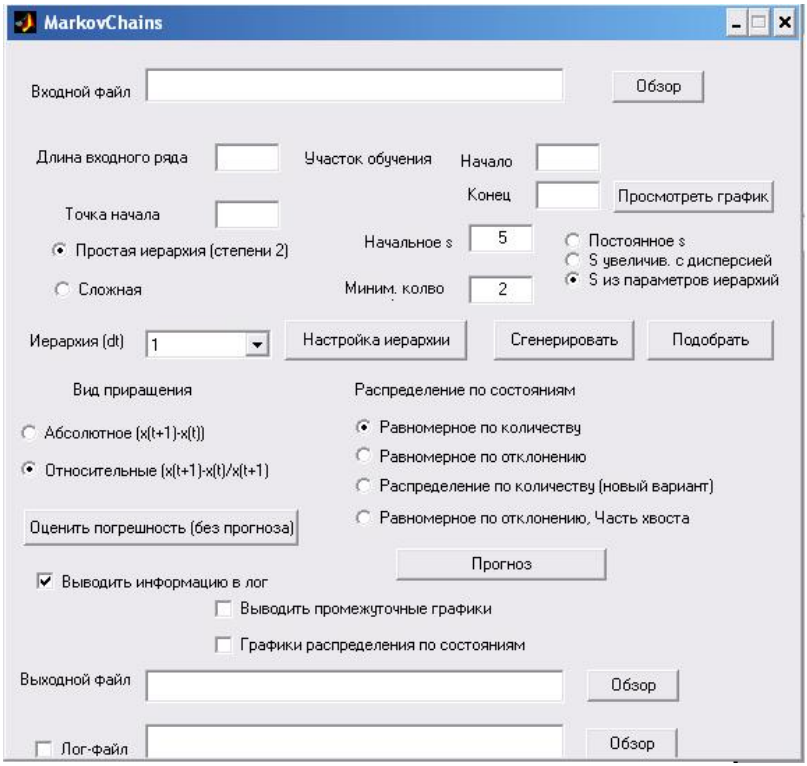


Рис. 4.19: Робоче вікно програми MarkovChains

Програма приймає в якості вхідних даних текстові файли, які містять ряд чисел, розділених пробілом або символами кінця рядка. 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 відрізку, на якому буде вестись навчання. Переглянути графік проміжку навчання можна, натиснувши кнопку “Просмотреть график” (рис. 4.20).

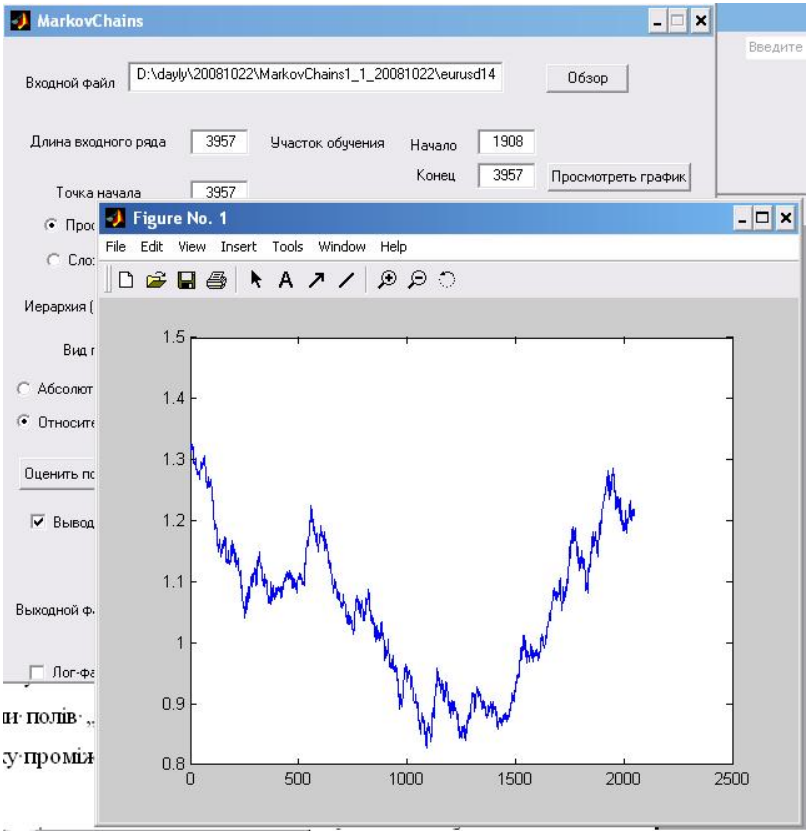


Рис. 4.20: Робоче вікно програми MarkovChains

Значення поля “Точка начала” задає точку, починаючи з якої здійснюється прогнозування. Ця точка задається відносно відрізку навчання, заданого значеннями полів “Начало” та

“Конец”. Це значення краще всього визначити по графіку проміжку навчання.

Мережу приростів часу можна згенерувати двома 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 дт повинні бути кратними найбільшому приросту Δt. 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 “Конец” здійснюється генерація ієрархій та вирізання проміжку навчання, який стає рівним максимальному Δt. При цьому значення поля “Конец” не змінюється, а значення “Начало” збільшується, задаючи довжину проміжку навчання $N1 = \Delta t_{max}$. Пам’ятайте, що значення кінця проміжку незмінне, а значення початку проміжку може змінюватись, це необхідно враховувати при заданні проміжку навчання.

Під час автоматичної генерації ієрархії приростів часу кожному приросту Δt задаються налаштування: кількість станів s, порядок ланцюга Маркова r та мінімальна кількість повторень k. Ці параметри генеруються автоматично, але є можливість підкоректувати їх для кожного приросту часу (кнопка “Настройка иерархии”) або згенерувати з іншими значеннями (кнопка “Сгенерировать”). Процес налаштування параметрів кожного приросту буде описаний нижче.

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” та “Относительные приращения”.

Також реалізовано 4 способи розподілення приростів на стани: рівномірний за кількістю представників станів (стани визначаються так, щоб кількість представників кожного стану були приблизно рівними), рівномірним за відхиленнями (ділиться проміжок відхилень від максимального додатного до мінімального відємного на рівних s частин), “Равномерное по количеству (новый вариант)” – попереджує можливу ситуацію, коли в ряді приростів присутні 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".

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. 4.21).

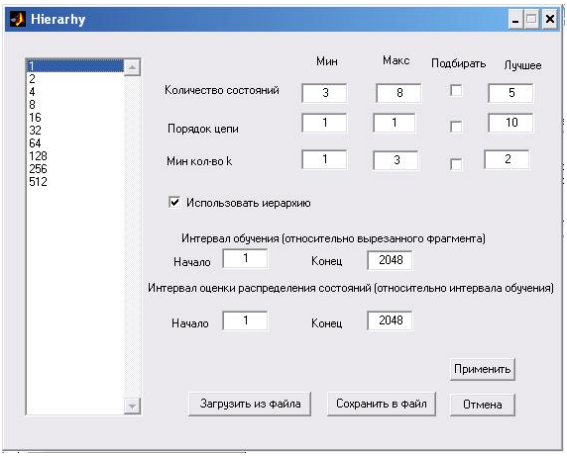


Рис. 4.21: Налаштування параметрів ієрархії приростів часу

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 повторень k. Оптимальні значення цих параметрів можна підібрати, для цього необхідно відмітити галочку "Подобрать" навпроти відповідного параметру. Значення в полях введення "Макс" та "Мин" 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 приростах часу може відрізнятись, керуючи довжиною проміжку навчання для кожного приросту

можна уникнути так званих “крайових ефектів”. Також за допомогою галочки “Использовать иерархию” можна вилучити 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. Налаштування 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 значеня параметрів ієрархії беруться із полів “Начальное s” та “Миним. кол-во k”. Якщо Ви хочете згенерувати параметри, які відміні від заданих по замовчуванню, то можна скористатись кнопкою “Сгенерировать”. Після натискання Ви побачите вікно параметрів генерування, яке зображене на рис. 4.22.

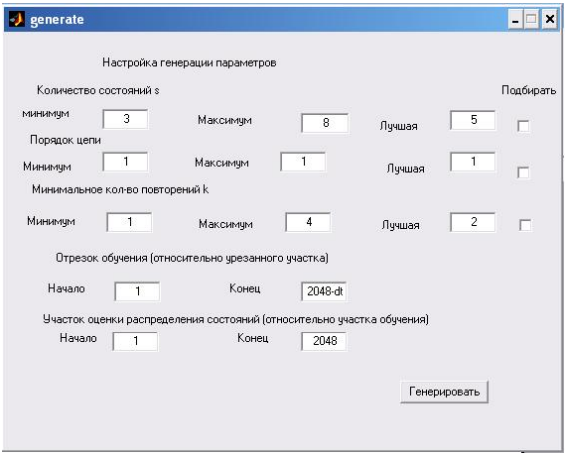


Рис. 4.22: Генерація параметрів ієрархій приростів часу

В кожне поле введення необхідно задати значення параметру, яке буде застосоване для всіх приростів ієрархії. Можна використовувати вирази з параметром Δt, 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 (див. рис. 4.22).

Для перевірки адекватності представлення ряду у вигляді послідовності станів при різних значеннях приростів часу необхідно проробити наступні дії: закодувати ряд на усіх рівнях ієрархії та відновити їх значення по реальним послідовностям станів (без прогнозування) і проробити процедуру склеювання. 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. 4.23. зображено результат цієї процедури.

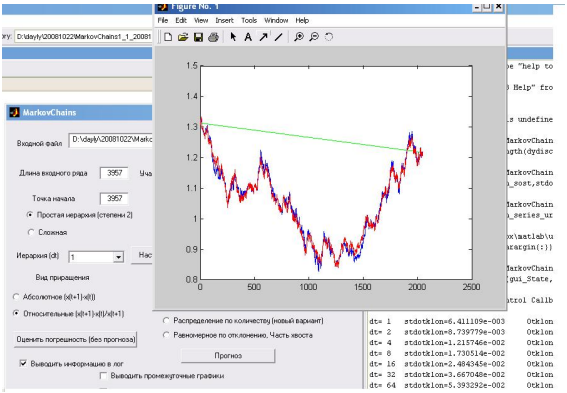


Рис. 4.23: Процедура перевірки представлення (без прогнозування)

Результати виконання процедури виводяться текстовими повідомленнями на консоль Matlab, який можна звичайним способом виділити і скопіювати в буфер обміну. На рис. 4.23. 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 представлення.

Пакет має можливість виводити текстовий лог не на консоль, а в будь-який файл. Для цього слугують нижнє поле введення та кнопка "Обозор”, 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.

Існує можливість проілюструвати процес прогнозування та вищерозглянутого тестування за допомогою виводу проміжних графіків при відповідних Δt у процесі виконання склеювання. Виведення цих графіків вмикається вибором опції "Выводить промежуточные графики", яка знаходиться в нижній частині головного вікна.

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 вімкнута опція виводу проміжних графіків, то в кінці прогнозу кожної наступної ієрархії виводиться графік прогнозу даної ієрархії і результат склеювання з попередніми ієрархіями.

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 MarkovChains доступний за допомогою команди MarkovChainsCmd. Набір вхідних параметрів даної команди наступний:

MarkovChainsCmd(infilename, predictlength, outfilename);

infilename – 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. Буде округлене до найближчого степеня числа 2.

outfilename – ім'я вихідного файлу, має бути текстовим файлом (*.txt). Після виконання команди даний файл буде містити значення навчальної вибірки та продовження (прогноз) як один ряд.

Також, окрім текстового вихідного файла, 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.ck.ua/MarkovChains1_2_20111201_I_usuallly_use.rar`.

Усі інші параметри, які можуть бути змінені у методі, задані значеннями, які дозволяють отримати найкращі прогнози. Для зміни певних параметрів, можна 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.ck.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 створіть окрему папку, в яку запишіть

три 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 та 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.m буде доповнений новими командами. Тому необхідно очищувати (видаляти) файл commands_markov_predict.m після обчислення прогнозів. Цією особливістю можна скористатись, 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 з "if(1)" на "if(0)". Після усереднення для генерації наступних завдань зміни необхідно повернути. Після редагування для кожного вхідного файлу запустіть команду MarkovChainsUsred, вкажіть ім'я того ж файлу та ті самі параметри (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. Зауважимо, якщо вказати неіснуюче ім'я файлу, побудова графіку похибок просто буде пропущена.

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.ck.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)).

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 паралельних 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) просмотр, -> 2) добавить несколько заданий.

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.

   (c) 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 комп'ютери порахують усі завдання даного пакета, на кожному комп'ютері папка завдань запаковується (для зручності в імені вказується назва завдання та назва комп'ютера) 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 із 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).

10. 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, але вимагає значної кількості неавтоматизованих дій. В подальшому ми працюємо над більш автоматизованою версією зі зручним графічним інтерфейсом. Версію планується розмістити на тому ж сайті `http://prognoz.ck.ua`.

Розділ 5

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

Основна ідея даного розділу – 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 [92], де запропоновано використання дискретного перетворення Фур'є. Перетворення Фур'є дає можливість дослідити амплітудно-частотну характеристику сигналу, що досліджується, але має певні недоліки, пов'язані з задачами прогнозування. Також спектр частот, кратних двум, 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 [93], дає можливість подолати першу з вищезазначених проблем, але залишається проблема неефективності, збитковості кодування частотних складових. Нами запропоновано 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 методикою складних ланцюгів Маркова показує достатньо високу точність 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 [28, 91, 94–96].

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:

$$y_{abs}\,(t) = a + bt + \sum_{i=1}^{m} c_i \sin\,(d_i t + e_i), \tag{5.1}$$

або для відносного масштабу:

$$y_{rel}\,(t) = ae^{bt} \prod_{i=1}^{m} c_i \sin\,(d_i t + e_i). \tag{5.2}$$

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

$$F\left(a,b,c_1,...c_m,d_1,...d_m,e_1,...,e_m\right)=\sum_{i=1}^{n}\left(y_i-y_{lin}(t_0+i\Delta t)\right)^2, \tag{5.3}$$

або для відносного варіанту:

$$F\left(a,b,c_1,...c_m,d_1,...d_m,e_1,...,e_m\right)=\sum_{i=1}^{n}\left(1-\frac{y_i}{y_{lin}(t_0+i\Delta t)}\right)^2. \tag{5.4}$$

При розв'язуванні задачі 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 проілюстровано на рис.5.1 та 5.2. Тому необхідною має бути умова відповідності не меньш 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, пов'язаною з наявністю великої кількості хибних локальних мінімумів, а по-друге, можливою некоректністю такого наближення на високих частотах. Емпірично було вибране обмеження 10 коливань за час навчальної вибірки.

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 залишків для одночастинного наближення:

$$r_i\left(t\right)=r_{i-1}\left(t\right)-\left(a+bt+c_i\sin\left(d_it+e_i\right)\right), \tag{5.5}$$

або для m-частинного наближення:

$$r_i\left(t\right)=r_{i-1}\left(t\right)-\left(a+bt+\sum_{j=1}^{m}c_{ij}\sin\left(d_{ij}t+e_{ij}\right)\right). \tag{5.6}$$

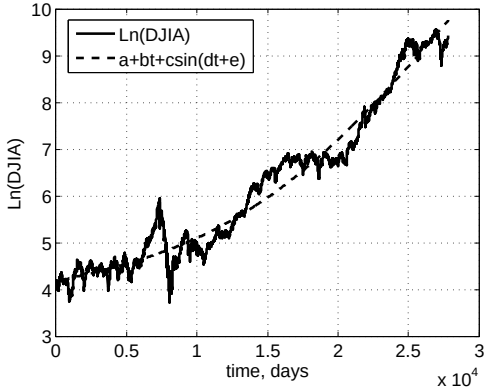


Рис. 5.1: Низькочастотна апроксимація індексу Dow Jonew Indusrtial

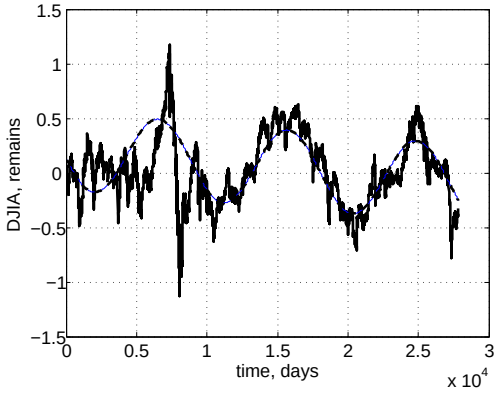


Рис. 5.2: Залишки апроксимації індексу DJIA та їх апроксимація

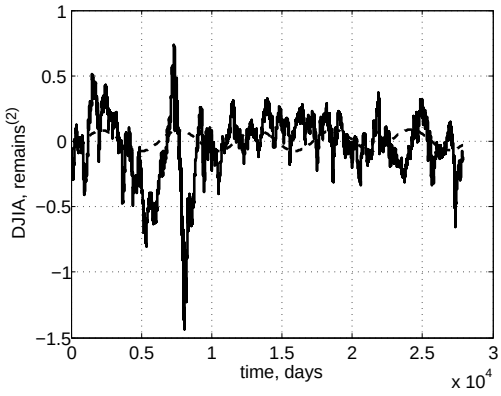


Рис. 5.3: Залишки другого порядку індексу DJIA та їх апроксимація

Мінімізація нелінійного функціоналу нев'язки (5.3) або (5.4) 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 параметрів тренду вибираються коефіцієнти лінійного тренду, визначені за допомогою МНК.

Вибирається два 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) $nf = 5$.

Лабораторна робота № 7. 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, якйи 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(infilename, outfilename, niter, logmode, sinmode,

maxt,mint, nfreq);

infilename - ім'я вхідного файлу, рядок символів.

outfilename - Імя 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 замовчуванні включене.

sinmode - Необов'язковий параметр, перемикач режиму 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 запитує 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 цьому 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 вімкнутий, 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), максимальний 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 nfreq, 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.

Висновки та перспективи досліджень

У даному посібнику розглянуто ряд методів прогнозування, серед яких: трендові (нелінійні, ті що зводяться до лінійних та лінійна гармонічна), авторегресійні (лінійні та нелінійні, реалізовані у вигляді нейронної мережі) та марківські моделі (приховані марківські моделі та складні ланцюги Маркова). Ряд лабораторних робіт, наведених у тексті, дозволяє познайомитись з особливостями кожного з методів.

Даний посібник не претендує на повноту викладу усіх методів прогнозування, які застосовуються сьогодні, а охоплює тільки методи, які активно використовуються авторами у своїх дослідженнях. За новими розробками методів прогнозування можна слідкувати на наших персональних сайтах.

Автори будуть раді відповісти на будь-які запитання, а також зауваженням та пропозиціям.

З повагою,

авторський колектив:
Андрій Анатолійович Ганчук
ganchuk2782@rambler.ru
Володимир Миколайович Соловйов
vnsoloviev@rambler.ru
Дмитро Миколайович Чабаненко
chdn6026@mail.ru
`http:\\kafek.at.ua`
`http:\\prognoz.ck.ua`

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. — 364 с.

[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. — 344 с.

[9] Марьясов Д. А. Анализ и прогнозирование финансового рынка на основе модели детерминированного хаоса: Дис. канд. техн. наук: 05.13.01 / 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. — 416 с.

[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. Effcient capital markets: A review of theory and empirical work / E. F. Fama // Journal of Finance. — 1970. — Vol. 25. — P. 383–417.

[14] Швагер Д. Технический анализ. Полный курс / Д. Швагер. — Москва: Альпина Бизнес Букс, 2005. — 806 с.

[15] Мерфи Д. Межрыночный технический анализ / Д. Мерфи. — Москва: Диаграмма, 2002. — 317 с.

[16] Колби Р. Энциклопедия технических индикаторов / Р. Колби. — Москва: Альпина Бизнес Букс, 2004. — 837 с.

[17] Удовенко В. Forex. Практика спекуляций на курсах валют / В. Удовенко. — Москва: ООО “И.Д. Вильямс”, 2007. — 384 с.

[18] Тесля Ю. Н. Введение в информатику природы. Монография. / Ю. Н. Тесля. — Киев: Маклаут, 2010. — 255 с.

[19] Грабовецкий Б. Є. Основи економічного прогнозування, Навчальний посібник. / Б. Є. Грабовецкий. — Вінниця: ВФ ТАНГ, 2004. — 162 с.

[20] Рассел С. Искусственный интеллект: современный подход / С. Рассел, П. Норвиг; Под ред. . е изд..: Пер. с англ. — Москва: Издательский дом “Вильямс”, 2006. — 1408 с.

[21] 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.

[22] Синергетичні та еконофізичні методи дослідження динамічних та структурних характеристик економічних систем. Монографія / В. Дербенцев, О. Сердюк, В. Соловйов, О. Шарапов. — Черкаси: Брама-Україна, 2010. — 287 с.

[23] Тобілевич Ю. Агентно-орієнтоване моделювання / Ю. Тобілевич // Вісник Черкаського університету. — 2010. — Т. 173. — С. 79–89.

[24] Krause A. Evaluating the performance of adapting trading strategies with different memory lengths / A. Krause // ArXiv e-prints. — 2009.

[25] Лукашин Ю. П. Адаптивные методы прогнозирования временных рядов: учеб. пособ. / Ю. П. Лукашин. — Москва: Финансы и статистика, 2003. — 416 с.

[26] Грицюк П. М. Аналіз, моделювання та прогнозування динаміки врожайності озимої пшениці в розрізі областей України : монографія / П. М. Грицюк. — Рівне: НУВГП, 2010. — 350 с.

[27] 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.

[28] Soloviev V. Discrete Fourier forecasting of economic dynamic's time series / V. Soloviev, V. Saptsin, D. Chabanenko // 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.

[29] Geisser S. Predictive inference: an introduction / S. Geisser. Monographs on statistics and applied probability. — Chapman & Hall, 1993.

[30] Голяндина Н. Анализ и прогноз временных рядов: программная реализация метода “гусеница”. — Електронний ресурс. - Режим доступу: http://www.gistatgroup.com/gus/programs.html.

[31] Голяндина Н. Метод “Гусеница”–SSA: прогноз временных рядов / Н. Голяндина. — СПб.: ВВМ, 2004. — 52 с.

[32] 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.

[33] Данилов Д. Л. Главные компоненты временных рядов: метод «Гусеница» / Д. Л. Данилов, А. А. Жиглявский. — СПБ: Издательство Санкт-Петербургского университета, 1997. — 150 с. `http://www.gistatgroup.com/gus/book1/manual.html`.

[34] Черняк О. І. Динамічна економетрика [Текст] : навч. посіб. / О. І. Черняк, А. В. Ставицький. — Київ: КВІЦ, 2000. — 120 с.

[35] 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. — 26 с.

[36] Цыплаков А. Эконометрический ликбез: прогнозирование временных рядов. / А. Цыплаков // Квантиль. — 2006. — Т. 1. — С. 1–60.

[37] Бідюк П. І. Моделювання та прогнозування нелінійних динамічних процесів / П. . Бідюк; — Київ: ЕКМО, 2004. — 120 с.

[38] Бідюк П. І. Часові ряди: моделювання та прогнозування / П. І. Бідюк. — Київ: ЕКМО, 2003. — 144 с.

[39] Бідюк П. І. Методи прогнозування / П. І. Бідюк, О. С. Меняйленко, О. В. Половцев. — Луганськ: Луганський національний ун-т ім. Тараса Шевченка, 2008. — Т. 1. — 305 с.

[40] 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.

[41] 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.

[42] Присняков В. Ф. Нестационарная макроэкономика: – Учебное пособие. / В. Ф. Присняков. — Донецк: Дон-НУ, 2000. — 209 с.

[43] Сигел Э. Практическая бизнес-статистика / Э. Сигел. — Москва: Издательский дом «Вильямс», 2002. — 1056 с.

[44] Растригин Л. А. Экстраполяционные методы проектирования и управления / Л. А. Растригин, Ю. П. Пономарев. — Москва: Машиностроение, 1986. — 120 с.

[45] Сизов Ю. С. Анализ портфелей ценных бумаг и управление ими в современной России. — Електронний ресурс. - Режим доступу:. http://mirkin.eufn.ru/articles_1.htm\#sizov4.

[46] Алтунин А. Е. Модели и алгоритмы принятия решений в нечетких условиях / А. Е. Алтунин, М. В. Семухин. — Тюмень: Изд-во ТюмГУ, 2000. — 352 с.

[47] Асаи К. Прикладные нечеткие системы: Пер. с япон. / К. Асаи, Д. Ватада, С. Иваи. — Москва: Мир, 1993. — 368 с.

[48] 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.

[49] Сергеева Л. Н. Моделирование поведения экономических систем методами нелинейной динамики (теории хаоса) / Л. Н. Сергеева. — Запорожье: ЗГУ, 2002. — 227 с.

[50] 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.

[51] 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.

[52] Holland J. The dynamics of searches directed by Genetic Algorithms / J. Holland. — Singapore: Word Scientific, 1988.

[53] Галушкин А. И. Теория нейронных сетей. Кн. 1: Учеб. Пособие для вузов / А. И. Галушкин. — Москва: ИПРЖР, 2001. — 385 с.

[54] Головко В. А. Нейронные сети: обучение, организация и применение. Кн.4:Учеб.пособие для вузов/Общая ред. А.И. Галушкина. / В. А. Головко. — Москва: ИПРЖР, 2001. — 256 с.

[55] Розенблат Ф. Принципы нейродинамики: Персептрон и теория механизмов мозга. Пер. с англ. / Ф. Розенблат. — Москва: Мир, 1965.

[56] Сигеру О. Нейроуправление и его приложения. Пер. с англ. под ред. А.И. Галушкина / О. Сигеру. — Москва: ИПРЖР, 2001. — 321 с.

[57] Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, М. Пилиньский, Л. Рутковский; Ed. by П. с польск. И. Д. Рудинского. — Москва: Горячая линия -Телеком, 2006. — 452 с. `http://www.sernam.ru/book_gen.php?id=21`.

[58] Тимчій М. Побудова архІтектури нейронних мереж зворотного розповсюдженнЯ за допомогою генетиЧних алгоритмІв / М. Тимчій // Вісник Черкаського університету. — 2010. — Т. 172. — С. 94–103.

[59] Poggio T. Theory of networks for approximation and learning / T. Poggio, F. A. Girosi // A.I. Memo. — 1994. — Т. 1140, № 1140. — 63 p.

[60] Вороновский Г. К. Генетические алгоритмы, нейронные сети и проблемы виртуальной реальности / Г. К. Вороновский. — Харьков: Основа, 1997. — 112 с.

[61] 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.

[62] 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.

[63] 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.

[64] Гусак А. Н. Подход к послойному 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.

[65] Шибхузов З. М. Конструктивный TOWER алгоритм для обучения нейронных сетей из ТП – нейронов / З. М. Шибхузов // Труды VIII Всероссийской конференции «Нейрокомпьютеры и их применение»: Сб. докл. — Москва: Век книги, 2002. — С. 1066–1072.

[66] 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. — 19 с.

[67] Тихонов Э. Е. Совершенствование методов прогнозирования с использованием нейронных сетей и системы остаточных классов: Дисс. к.т.н. / СГУ. — Ставрополь, 2004.

[68] Широков Р. В. Нейросетевые модели систем автоматического регулирования промышленных объектов: Дисс. к.т.н. / СГУ. — Ставрополь, 2003.

[69] Генетические алгоритмы. — Електронний ресурс. `http://www.iki.rssi.ru/ehips/Genetic.htm`.

[70] Исаев С. А. Генетические алгоритмы и машинное обучение. — Електронний ресурс. Режим доступу:. `http://www.blind.alfint.ru/modules.php?name=News\&new\_topic=3`.

[71] Лекции по нейронным сетям и генетическим алгоритмам. — Електронний ресурс. `http://infoart.baku.az/inews/30000007.htm`.

[72] Батищев Д. И. Генетические алгоритмы решения экстремальных задач / Д. И. Батищев. — Воронеж: ВГУ, 1994. — 135 с.

[73] Івахненко О. Г. Метод групового урахування аргументів - конкурент методу стохастичної апроксимації / О. Г. Івахненко // Автоматика. — 1968. — Т. 3, № 3. — С. 58–72.

[74] Ивахненко А. Г. Помехоустойчивость моделирования / А. Г. Ивахненко, В. С. Степашко. — Киев: «Наук. думка», 1985. — 216 с.

[75] Ивахненко А. Г. Моделирование сложных систем по экспериментальным данным / А. Г. Ивахненко, Ю. П. Юрачковский. — Москва: «Радио и связь», 1987. — 120 с.

[76] Ивахненко А. Долгосрочное прогнозирование и управление сложными системами / А. Ивахненко. — Москва: Техника, 1975. — 312 с.

[77] Зайченко Ю. П. Нечеткий метод индуктивного моделирования в задачах прогнозирования макроэкономических показателей / Ю. П. Зайченко // Системні дослідження та інформаційнітехнології. — 2003. — Т. 3. — С. 25–45.

[78] Зайченко Ю. П. Синтез і адаптація нечітких прогнозуючих моделей на основі методу самоорганізації / Ю. П. Зайченко, І. О. Заєць // Наукові вісті НТУУ “КПІ”. — 2001. — Т. 3. — С. 34–41.

[79] Granger C. W. J. Forecasting Economic Time Series / C. W. J. Granger, P.Newbold. — San Diego: Academic Press, 1986.

[80] 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.

[81] 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.

[82] Rabiner L. R. An introduction to hidden Markov models / L. R. Rabiner, B. H. Juang // IEEE ASSP Mag. — 1986. — Vol. June. — P. 4–16.

[83] 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.

[84] 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 конференції. / відповід. ред. к.т.н., доц.. Берідзе Т.М. — Кривий Ріг: КФ ДВНЗ ЗНУ, 2009. — С. 84–85.

[85] Романовский В. И. Дискретные цепи Маркова / В. И. Романовский. — Москва: Гостехиздат, 1949. — 436 с.

[86] Дынкин Е. Марковские процессы / Е. Дынкин. — Москва: Физматгиз, 1963. — 860 с.

[87] Гупал Н. Применение байесовской процедуры распознавания для прогнозирования динамики курсов акций / Н. Гупал, Ю. Матусов // Компьютерная математика. — 2009. — Т. 1. — С. 105–110.

[88] Приймак М. В. Оцінювання матриць переходів періодичних ланцюгів Маркова / М. В. Приймак, С. Ю. Прошин // Електроніка та системи управління. — 2009. — Т. 3 (21). — С. 26–33.

[89] Приймак М. Похибка оцінок матриць переходів періодичного ланцюга маркова / М. Приймак, О. Віцентій, С. Прошин // Вісник ТНТУ. — 2010. — Т. 15, № 3. — С. 150–159.

[90] 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.

[91] 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.

[92] 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. — Т. 1. — С. 240–243.

[93] 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.

[94] Chabanenko D. Discrete Fourier forecasting of economic dynamic's time series / D. Chabanenko, O. Nechinennaya // Information Technologies, Management and Society. Theses of the International Conference. — Riga: 2010. — P. 43–44.

[95] 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.

[96] 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.