МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Кафедра інформаційних систем

Реферат на тему

Одношаровий персептрон Розенблатта

Методи прогнозування часових рядів

з курсу "Розподілені інформаційні системи"

Виконав:
студент групи
ПМІ-44
Кізло Тарас Михайлович

Перевірила
доцент
Козій Ірина Ярославівна

Львів 2020

План

Одношаровий персептрон Розенблатта

Френк Розенблат — відомий американський учений у галузі психології, нейрофізіології та штучного інтелекту. За своє життя, він намагався знайти алгоритм який буде здатний розв’язувати складні задачі та при цьому навчатись. Як виявилось такий алгоритм вже існує — це біологічний мозок. Знання в галузях штучного інтелекту та нейрофізіології допомогли Розенблатту збагнути, що фундаментальні закони організації, загальні для всіх систем обробки інформації, як для машини, так і для людського розуму. Саме це надихнуло його на створення перцептрону.

Поява персептрона

Персептро́н — це математична або комп'ютерна модель сприйняття інформації мозком. Він став однією з перших моделей нейромереж. Попри свою простоту, перцептрон здатен навчатися і розв'язувати досить складні завдання.

Та сам перцептрон це трохи спрощена модель яка більш подібна до нейрона ніж до мозку. Так основні ідеї запозичені із біології можна висловити наступними тезами:

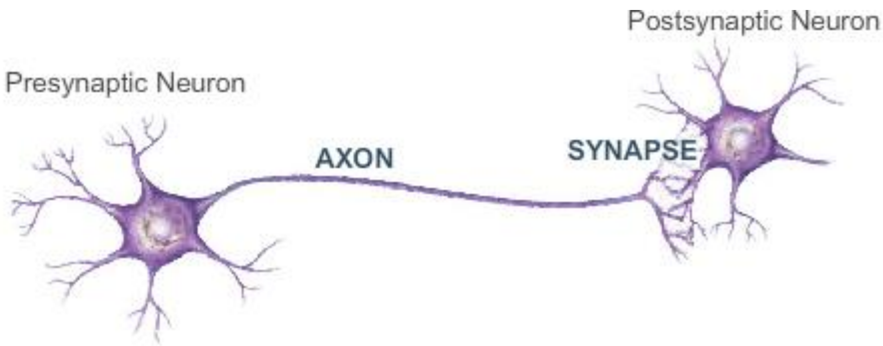


рис 1. Зображення нейрона

Нейрон — це клітина яка що **обробляє** та **передає** інформацію у вигляді електричного або хімічного **сигналу**

- Аксон — відросток нервової клітини, що **проводить** імпульси іншим нервовим клітинам
- Передача хімічних сигналів відбувається через **синапси** - контакти між нейронами та іншими клітинами
- Деякі зв’язки між нейронами **сильніші** за інші. Сильніше з'єднання передає **більше** сигналу
- Якщо нейрон отримує **достатньо** енергії він **передає** її дальше

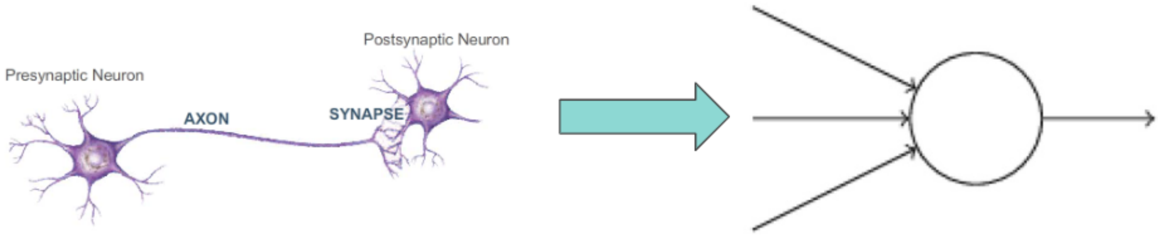


рис 2. Модель персептрона на основі нейрона

Будова персептрона

Перцептрон складається із таких основних типів елементів:

- Сенсори, **S-елемент** — є чутливими елемент, які від дії будь-якого з видів енергії (наприклад, світла, звуку, тиску, тепла тощо) виробляють сигнал. У фізичному втіленні вони відповідають, наприклад, світлочутливим клітинам сітківки ока або фоторезисторам матриці камери. Кожен рецептор може перебувати в одному з двох станів — спокою або збудження, і лише в останньому випадку він передає одиничний сигнал до наступного шару, асоціативним елементам.
- Асоціативні, **A-елемент** — асоціюються із декількома сенсорними елементами.
- Результуючий елемент, **R-елемент** — відповідає за результат персептрона
- **Ваги**, які з'єднують елементи між собою. Вони набувають значення -1 чи 1, та 0 що означає *відсутність* зв'язку.

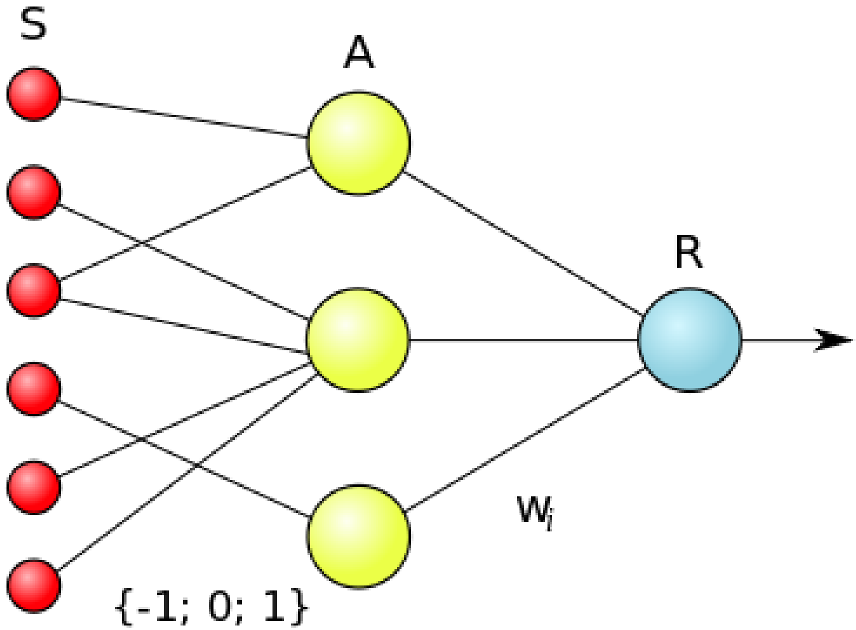


рис 3. Будова персептрона

Алгоритм персептрона

При взаємодії елементи передають сигнал по вагах зліва направо

Сенсорні S-елементи відповідають за *вхідні* параметри і в *активному* стані передають одиничний сигнал асоціативним елементам.

Асоціативні A-елемент передають сигнал у тому випадку, якщо сума сигналів перевищує заданий *поріг*. Сигнали від збуджених A-елементів, своєю чергою, передаються до суматора R, причому сигнал передається з коефіцієнтом ваги зв'язку між елементами.

Так само як і A-елементи, R-елемент підраховує суму значень вхідних сигналів, помножених на ваги. А до вихідного результату застосовує функцію активації. R-елемент, а разом з ним і елементарний перцептрон, видає сигнал “1”, “-1” та “0”. Результат перцептрона можна розуміти як “так”, “ні” і “не знаю” відповідно.

Весь алгоритм можна описати формулою — різниця суми сигналів помножених на ваги та порогу активації, до якого застосована функція активації.

$$f(x) = sign(\sum_{i=1}^{n} w_i x_i - \theta)$$

рис 4. Математичний опис алгоритму персептрона

Функція активації — це функція, яка задає поріг для активації нейрона. Для персептрона Розенблатта звично застосовувати *функцію знаку*. Вона повертає 1, якщо вхідний параметр додатній, та — -1 в іншому випадку.

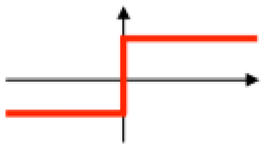


рис 5. Графік функції знаку.

Можна застосувати й іншу функцію активації.Головне щоб функція задовільняла наступним *вимогам:*

- існування похідної, необхідної для деяких алгоритмів навчання
- Y є [-1; 1] U [0; 1].

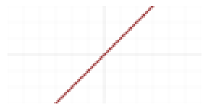
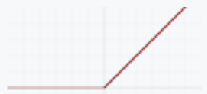
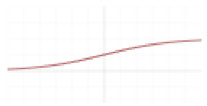
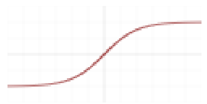
Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Relu		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Sigmoid		$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
Binary Step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Tanh		$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$	$f'(x) = 1 - f(x)^2$

рис 6. Приклади функцій активації.

Можливості та обмеження моделі

Можливості моделі

Основна задача яку ставлять перед персептроном Розенблата, це класифікації чи поділ на класи. Так персептрон можна навчити відрізняти коло від квадрата, тощо.Або навчити виконувати булівський оператор. Чи розділяти на класи.

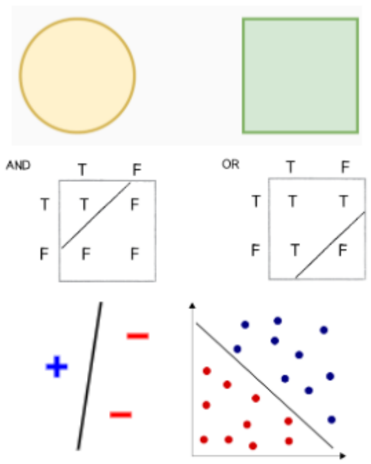


рис 7. Приклади задач класифікації.

Обмеження моделі

При цьому є обмеження, що задача повинна бути лінійно сепарабельною. **Лінійно сепарабельна задача** — це така задача в якій можна провести пряму лінію.

Можна побачити що навіть в тому самому класі не всі задачі є лінійно сепарабельні.

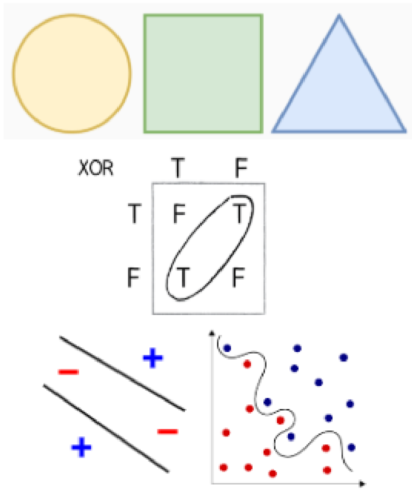


рис 7. Приклади не лінійно сепарабельних задач класифікації.

Правило навчання

Перцептрон не є універсальним алгоритмом і не буде розв’язувати задачі сходу. Його спочатку варто навчити за деяким правилом.

Правило навчання — це алгоритм, який змінює стан перцептрона так щоб вхідні аргументи давали необхідний результат
При навчанні відбуваються зміни:

- ваг
- функції активації

Метод корекції помилок

Один із методів навчання — це метод корекції помилок. Він складається із наступних кроків.

1. випадково вибираємо пороги активації
2. початкові ваги рівні нулю
3. готуємо вибірку з двох класів
4. показуємо об'єкт 1 класу. Ваги, А-елементів, що активуються *збільшуємо* на 1.
5. показуємо об’єкти 2 класу. Ваги, А-елементів, що активуються *зменшуємо* на 1.
6. переходимо до кроку 4 задану допоки не досягнемо умови припинення

Умову припинення можна представити у вигляді наступних предикатів:

- навчання триває N кількість кроків
- розв’язок задовільняє мінімальному критерію
- вичерпано дозволені ресурси (час, об’єм пам’яті тощо)

За теоремою збіжності, при любих початкових даних можна навчити персептрон за скінченний проміжок часу.

Приклад розв’язку задачі класифікації

Постановка задачі

Нехай є множина значень на декартовій площині. Необхідно класифікувати дві множини значень розділених прямою.

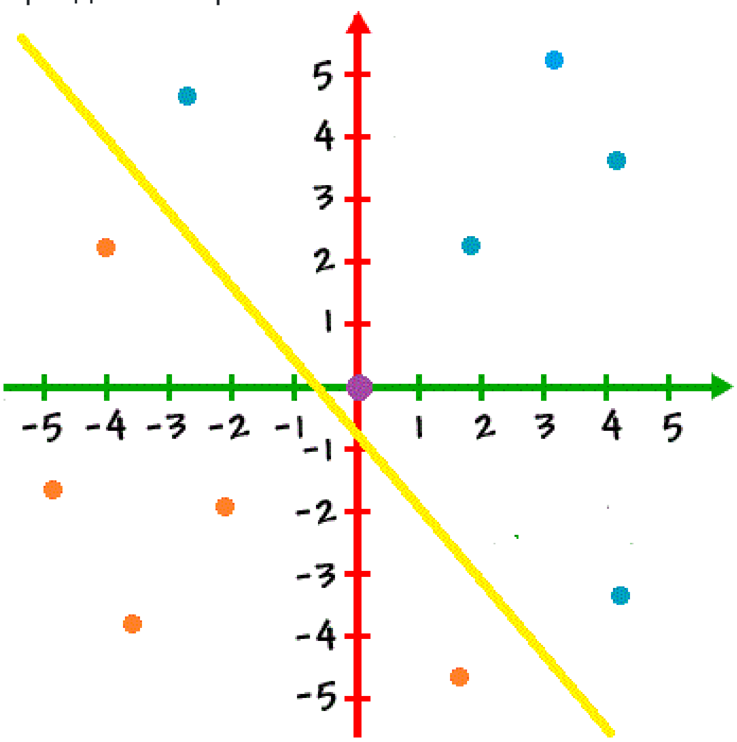


рис 8. Графічне представлення задачі.

Побудова моделі персептрона

Нехай перцептрон має два вхідних зв’язки.
На перший зв’язок будемо передавати значення координати X, а на другий — Y.
Результат перцептрона:

- +1 — якщо значення належить до першої множини, над прямою
- - 1 — якщо значення належить до другої множини, під прямою

Також варто враховувати значення (0, 0). Якщо її помножити на ваги і просумувати то результат буде 0. Хоча вона може бути як вище так і нижче лінії. У випадку коли вхідні дані вірні, але вони можуть вплинути на роботу перцептрона вводимо ще один зв’язок — *Bias*, який задає межу, наскільки важко активувати перцептрон.

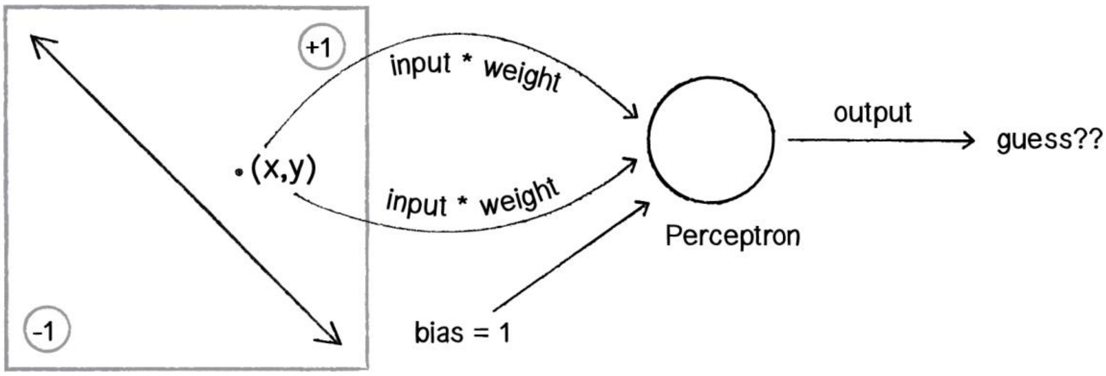


рис 9. Будова моделі

Розв’язок задачі

Можна побачити як персептрон намагається прокласифікувати значення.
Є два класи

- клас білих значень
- клас чорних значень

Жовта лінія, це де на думку персептрона знаходиться пряма.
Ті класи які прокласифіковані вірно позначені *зеленим* кольором, в іншому випадку — *червоним*.
Можна побачити, що після кількох ітерацій навчання, персептрону вдалось розв’язати задачу.

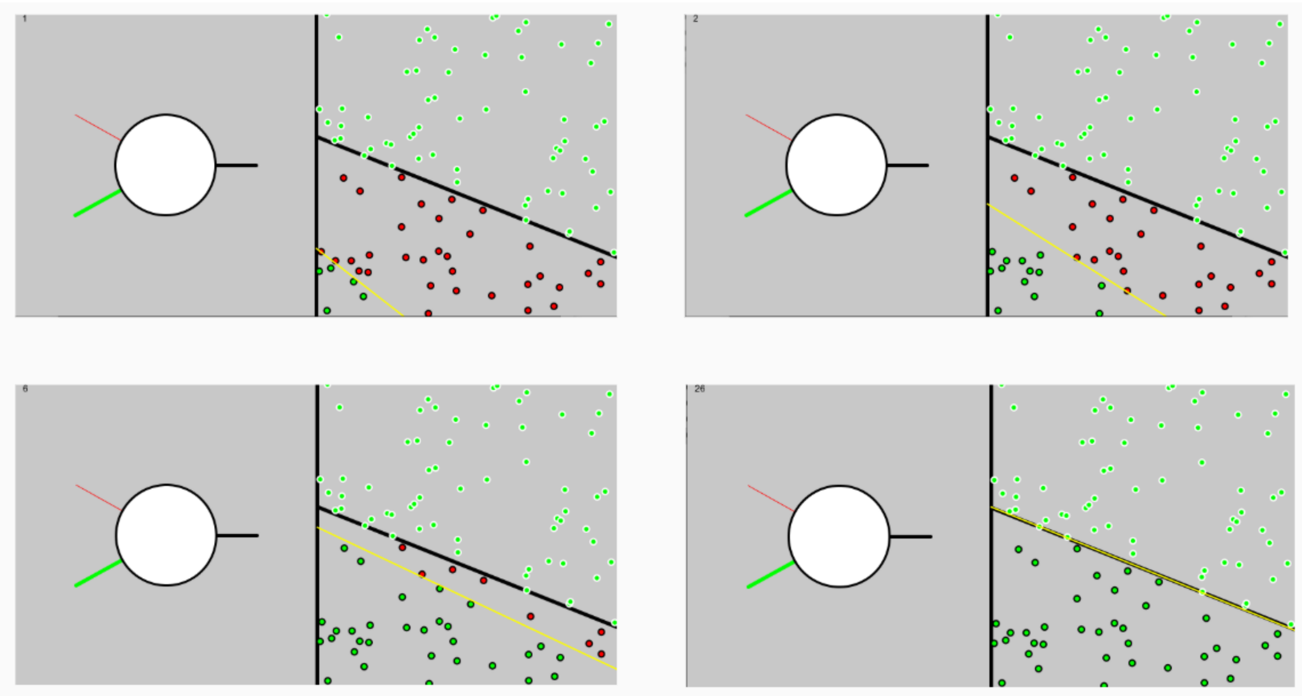


рис 10. Розв'язок задачі класифікації

Приклади архітектур

Щоб уникнути обмежень персептрона зазвичай використовують інші архітектури. Таким чином перцептрони об'єднують в шари, де вхідними сигналами наступного шару будуть вихідні попереднього. Інколи імітують пам’ять за допомогою зациклення сигналів тощо.

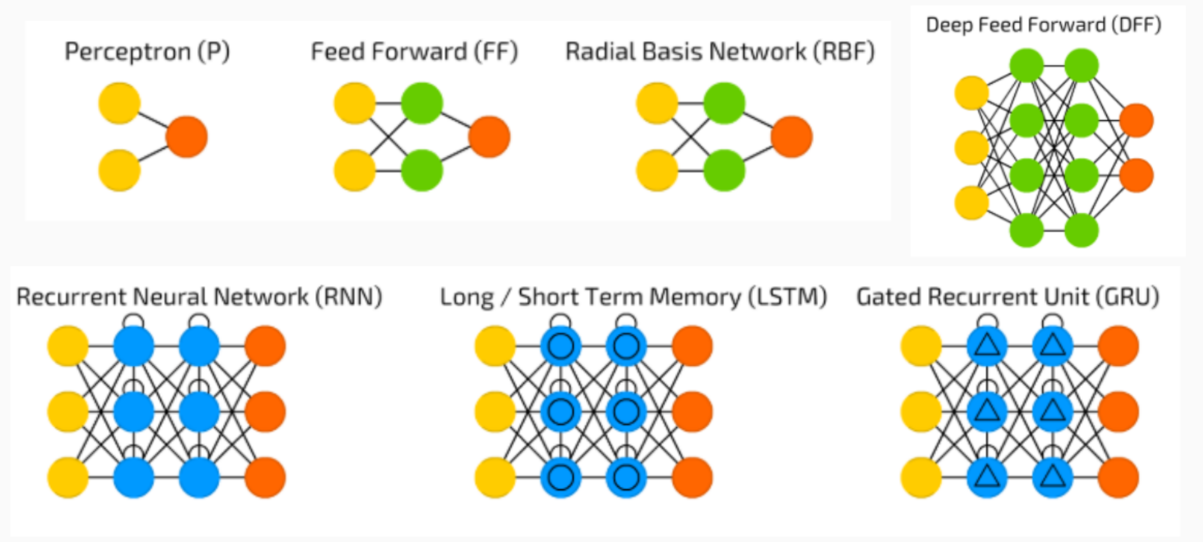


рис 11. Приклади архітектур нейронних мереж

Методи прогнозування часових рядів.

Основні терміни

Часовий ряд — це ряд даних, в хронологічному порядку.

Цей ряд можна представити у вигляді чисел, таблиць чи графіків.

1, 2, 3, 4, 5, ??

Номер	Об'єм перевезень (тис. т)	Поліноміальний тренд	Лінійний тренд	Логарифмічний тренд
1	8174,4	7572,9026	4136,846	4982,8
2	5078,33	6153,9296	4138,893	4733,087
3	4507,2	4472,595	4140,939	4587,014
4	2257,19	3017,0648	4142,985	4483,374
5	3400,69	2053,625	4145,032	4402,984
6	2968,71	1677,1536	4147,078	4337,301
7	2147,14	1856,6966	4149,124	4281,766
8	1325,56	2476,148	4151,17	4233,66

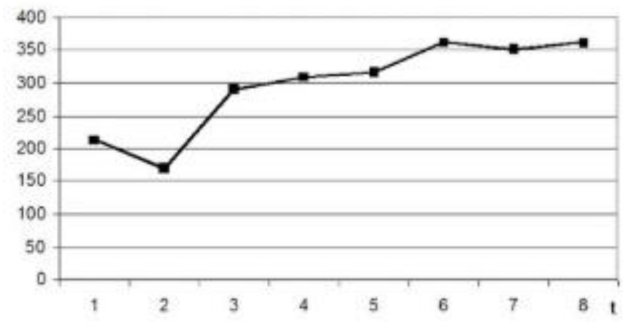


рис 12. Приклади представлення часового ряду

Часто графічне представлення дозволяє зрозуміти *характер* процесу, в той час як табличне ні.

Прикладами часових рядів може бути показник температури по днях, або ріст валюти протягом року, тощо.

Задача прогнозування полягає в тому, щоб спрогнозувати яке значення буде наступним базуючись на відомій вибірці.

Компоненти ряду

Тренд — плавна довгострокова зміна ряду.

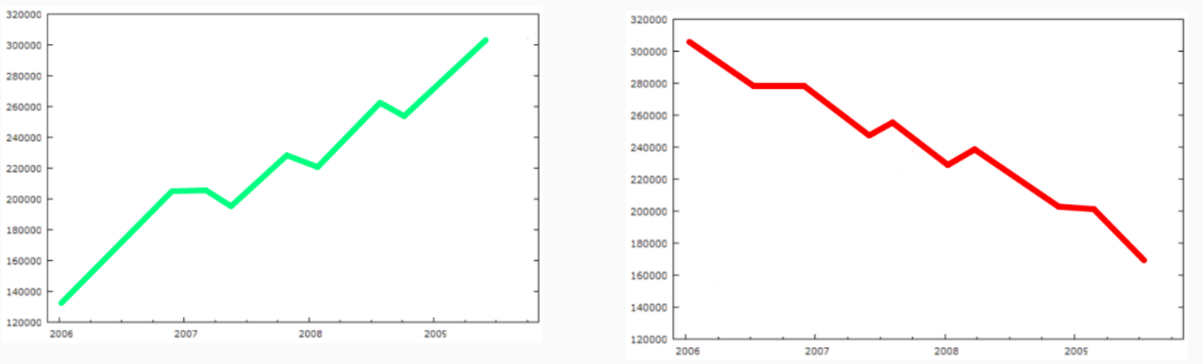


рис 13. Зростаючий та спадаючий тренд

Сезонність — циклічні зміни ряду із постійним періодом.

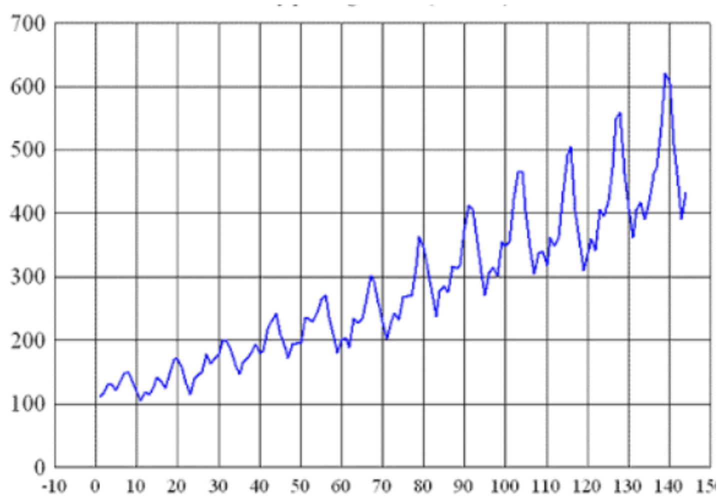


рис 15. Прояв сезонності ряду

Помилка (аномалія) — непрогнозована випадкова компонента ряду.

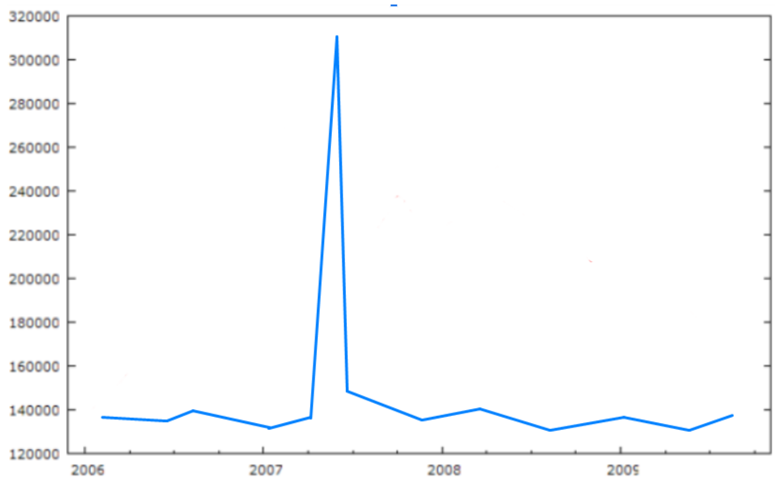


рис 16. Прояв аномалії в часовому ряді

Приклади методів прогнозування

Серед основних алгоритмів прогнозування часового ряду можна виділити наступні

- Авторегресійні моделі
- ARMA, ARIMA, Seasonal ARIMA ...
- Нейромережеві моделі
- Регресія
- Адаптивна селекція моделі
- Адаптивна композиція моделі
- тощо

Регресійний аналіз

Регресія — форма зв'язку між випадковими величинами, коли ми припускаємо, що величини залежать одна від одної.

Таким чином при **регресійному аналізі** ми використовуємо властивість *тренду* часових рядів. Наприклад, на графіку ми бачимо ярко виражений тренд і можемо зробити припущення, що графік і надалі зростатиме.

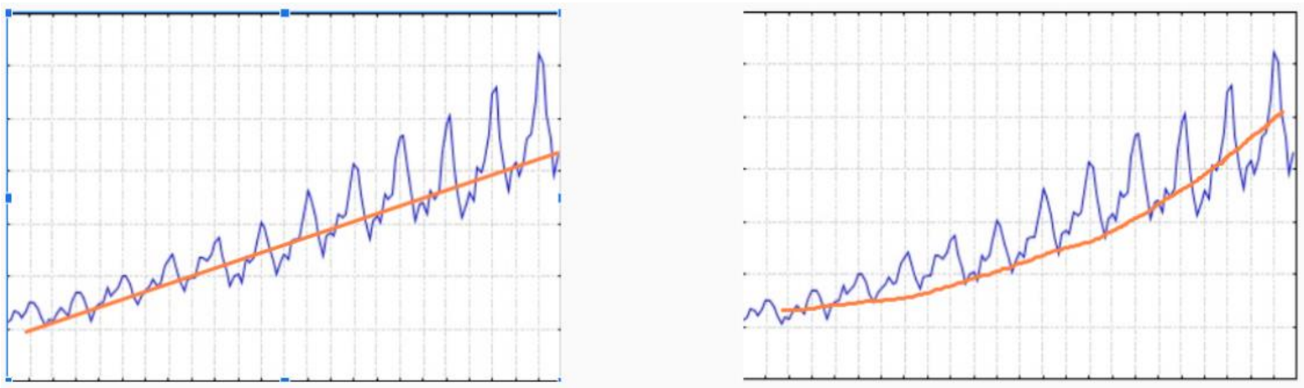


рис 17. Лінійна та поліноміальна регресія

Наївний сезонний аналіз

Метод **наївного сезонного аналізу** бере до уваги компоненту *сезонності* часового ряду. Ми прогнозуємо те що було в минулий період, або середнє за всі сезони.

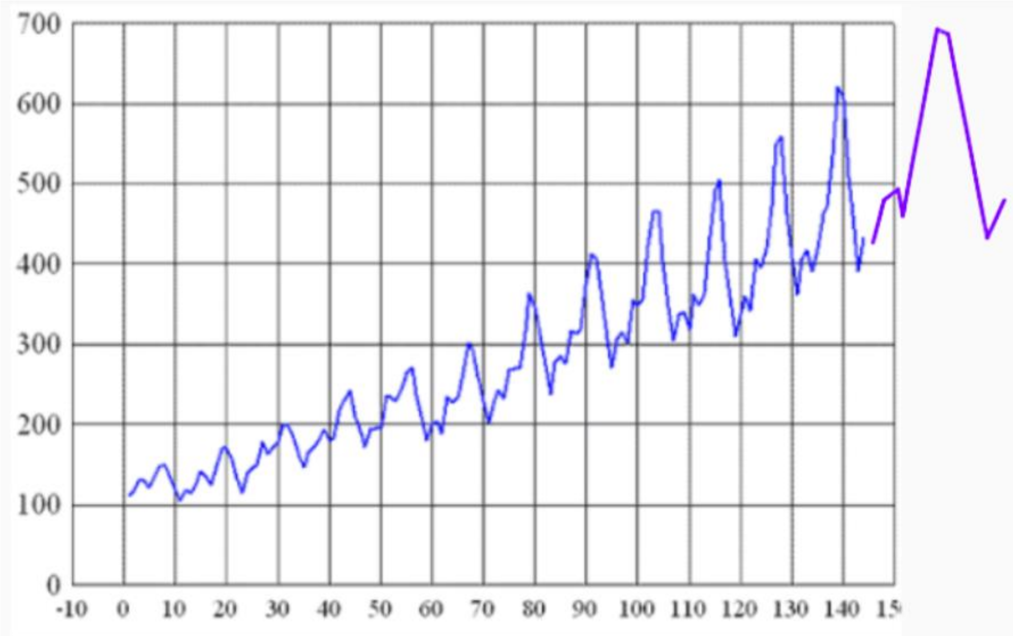


рис 19. Приклад наївного сезонного аналізу

Метод середньої оцінки інтервалу

Якщо на графіку немає ярко вираженого тренду чи сезонності, можна застосувати **метод середньої оцінки інтервалу.** Для цього враховуються дані за певний проміжок часу і рахується середнє значення. При обрахунку наступних значень враховуються спрогнозовані значення, тому такий метод не підходить при прогнозуванні на великий проміжок часу наперед.

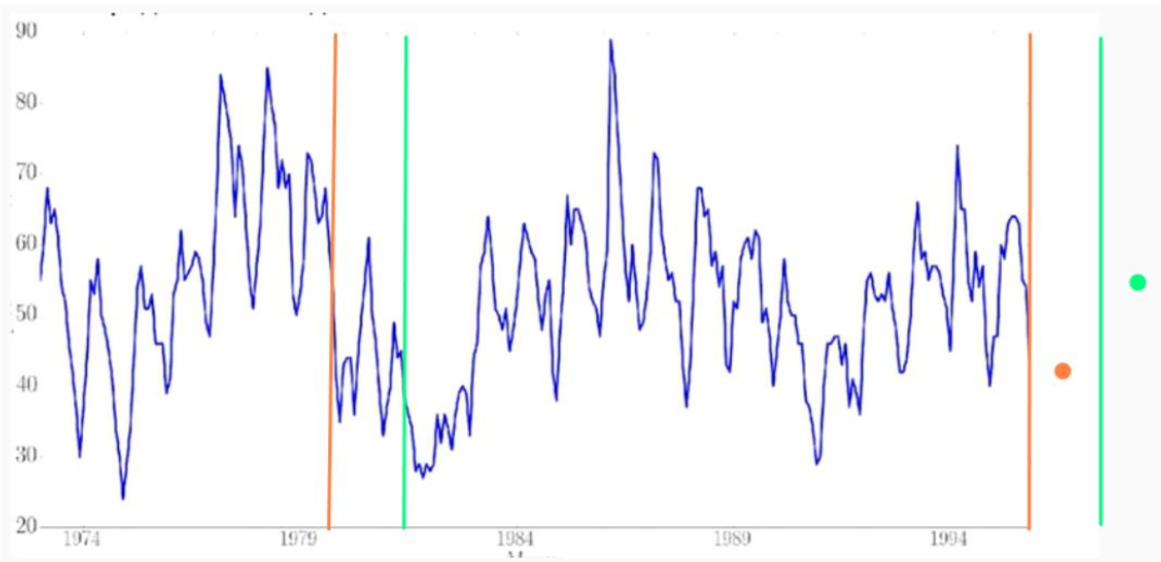


рис 20. Приклад методу середньої оцінки інтервалу

Метод згладження

При прогнозуванні **методом згладження** необхідно для всього ряду в околі знайти середнє значення. Таким чином можна отримати згладжений ряд на якому виділити сезонність чи тренд та застосувати один з методів, що базуються на цих компонентах.

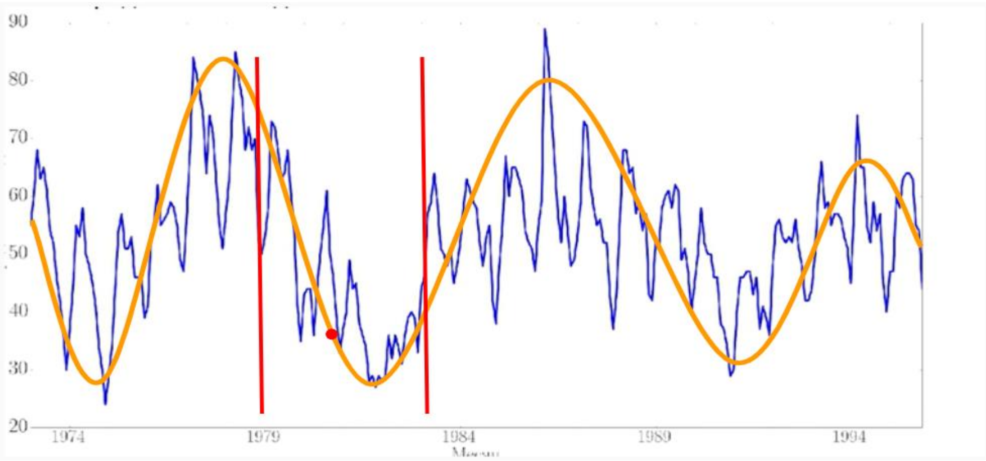


рис 21. Приклад методу згладження

Екстраполяція

Екстраполяція — пропонує прогнозування за допомогою математичного розв'язку.

$$\hat{y}_{T+h|T} = y_T + \frac{h}{T-1}\sum_{t=2}^{T}(y_t - y_{t-1}) = y_T + h\left(\frac{y_T - y_1}{T-1}\right)$$

рис 22. Формула екстраполяції

- Y — елементи часового ряду
- T — часовий інтервал
- h — крок часового інтервалу

У цьому випадку ми намагаємось продовжити ряд в майбутнє.

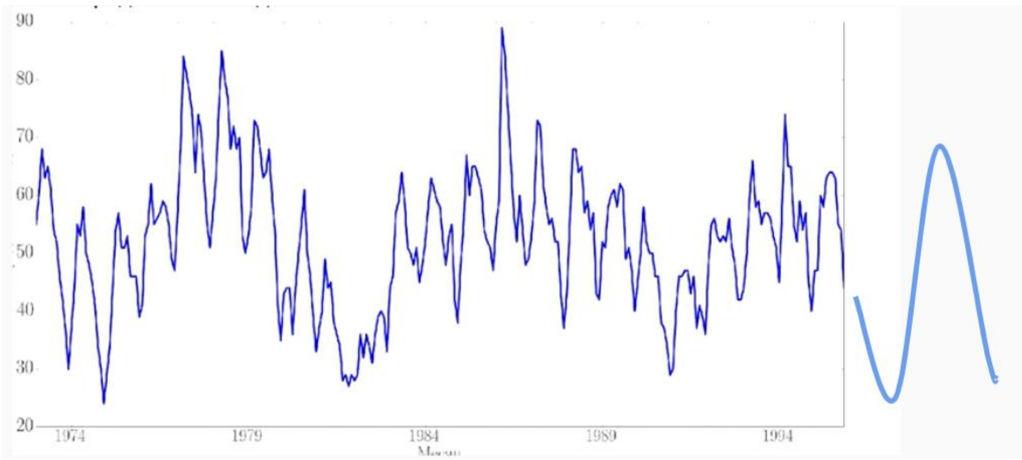


рис 23. Приклад розв'язку задачі прогнозування методом екстраполяції.

Метод помилки

При прогнозуванні **методом помилки**, необхідно прогнозувати дані заздалегідь на готовому ряді. Це робиться для того, щоб знайти помилку між спрогнозованим значенням і реальним, та враховуємо її при наступних обчисленнях з метою зменшення різниці між реальними даними та спрогнозованими.

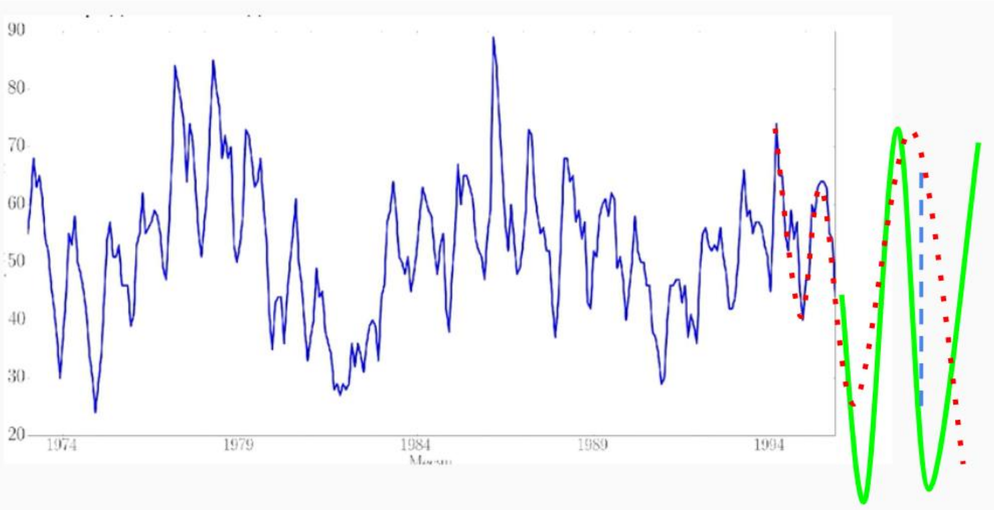


рис 24. Приклад методу помилки

Методи штучних мереж

Часто часові ряди прогнозують за допомогою штучних мереж. Наприклад, рекурентна нейронна мережа — це така нейронна мережа, яка володіє пам’яттю та при наступних обчисленнях використовує вхідні аргументи і результат минулих обчислень. Часто використовується при прогнозуванні погоди, де вхідними аргументами будуть вологість повітря, швидкість вітру та результат із минулих обчислень, через те, що вчорашня погода впливає на сьогоднішню також.

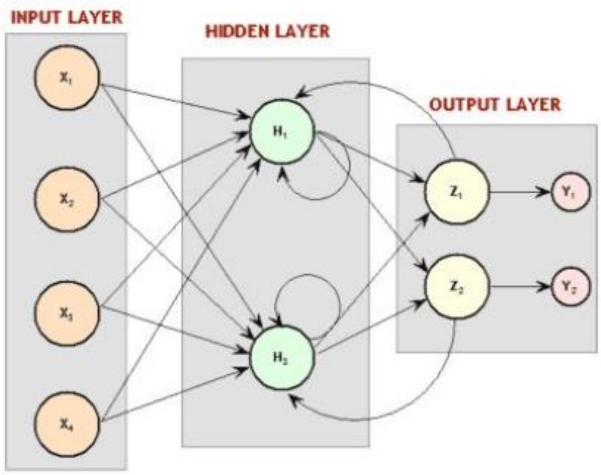


рис 25. Приклад рекурентної нейронної мережі

АРІМА

Метод **АРІМА** — це комбінація декількох підходів (ансамблі).

Складові АРІМА:

- AR— autoregressive (*авторегресія*)
- I — integrated (*диференціювання*)
- MA — moving average (*метод ковзного середнього*)

Диференціювання

Диференціювання — це метод, що дозволяє перетворити ряд в стаціонарний, за допомогою різниць сусідніх елементів денного ряду

$$y'_t = y_t - y_{t-1}$$

рис 26. Формула диференціювання часового ряду

Стаціонарний ряд — це ряд в якого протягом часу зберігаються його основні статистичні значення. Він не володіє характеристиками тренду чи сезонності. При прогнозуванні на такому ряді вдається досягнути кращих результатів.

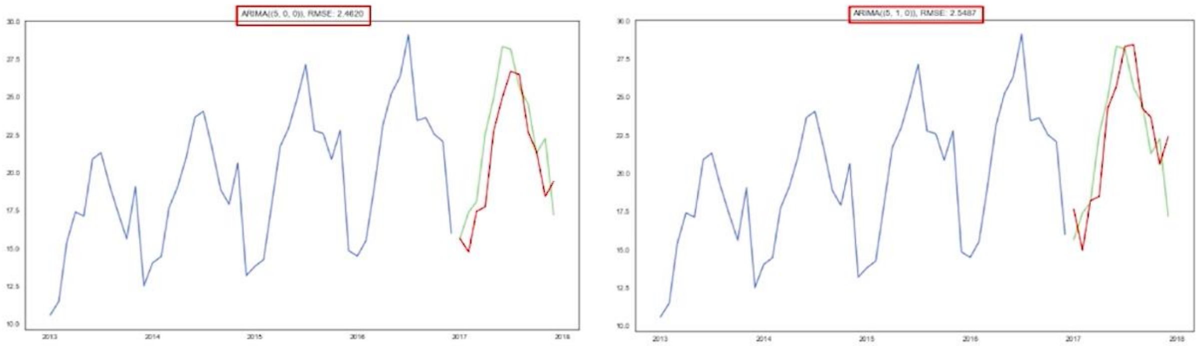


рис 27.Приклад прогнозування на часовому ряді без диференціювання (зліва) та на стаціонарному ряді (справа)

Авторегресія

Авторегресія — є сумою значень помножених на коефіцієнт регресії.. Цей метод являється модифікованою версію методу регресії, де результат коригується вагами.

$$x_t = \alpha_1 x_{t-1} + \ldots + \alpha_p x_{t-p} + \omega_t \ = \sum_{i=1}^{p} \alpha_i x_{t-i} + \omega_t$$

рис 28. Формула авторегресії

- a — коефіцієнт регресії
- x — елемент часового ряду
- w — ваги корегування

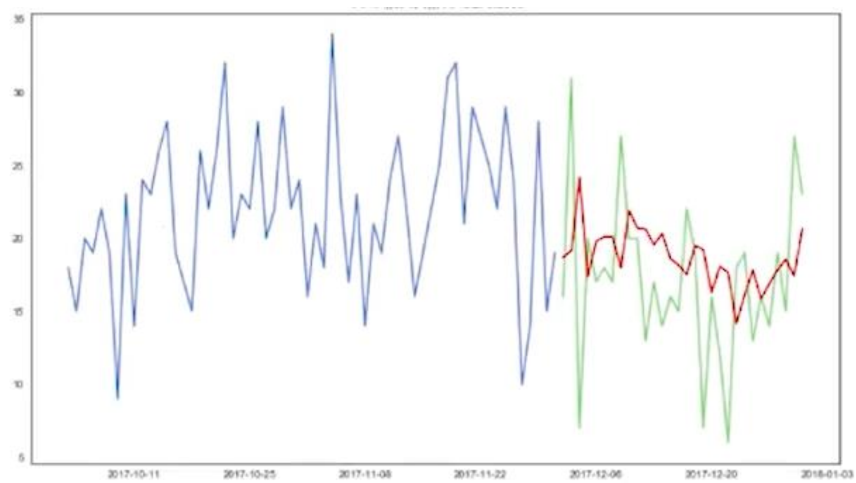


рис 29. Приклад авторегресії

Метод ковзного середнього

Метод ковзного середнього робить прогноз на основі помилки та теперішнього значення. Він являється комбінацією методу згладження та методу помилки.

$$x_t = \omega_t + \beta_1\omega_{t-1} + \ldots + \beta_p\omega_{t-p} = \omega_t + \sum_{i=1}^{p} \beta_i\omega_{t-i}$$

рис 30. Формула методу ковзного середнього.

- B — коефіцієнт згаджування
- w — різниця помилки між реальним та спрогнозованим значенням

рис 31. Приклад методу ковзного середнього

Після обрахунку значення диференційованого часового ряду метода авторегресії та методу ковзного середнього вираховується середнє значення, яке і буде прогнозованим.

Прогнозування на багатовимірних рядах

Багатовимірний ряд — це ряд який залежить не від одного параметра, а від декількох.

У таких випадках при прогнозуванні ряд розділяють на одновимірні ряди, і до кожного застосовують один із методів описаних вище. По завершенню аналізу, ряду знову об'єднують в один багатовимірний ряд.

рис 32. Приклад аналізу багатовимірного ряду

Джерела та література

- https://en.wikipedia.org/wiki/Perceptron
- https://habr.com/ru/company/ods/blog/327242/
- https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average
- http://ebib.pp.ua/vremennoy-ryad-vidyi-vremennyih-ryadov-7564.html
- https://en.wikipedia.org/wiki/Time_series
- *Бонгард М. М.[ru]. Проблема узнавания. — М. : Наука, 1967. — 320 с. (рос.)*
- *Брюхомицкий, Ю. А. Нейросетевые модели для систем информационной безопасности: Учебное пособие. — Таганрог : Изд-во ТРТУ, 2005. — 160 с. (рос.)*
- *Мак-Каллок, У. С., Питтс, В.[en]. Логическое исчисление идей, относящихся к нервной активности // Автоматы : сб.. — М., 1956. — С. 363—384. (рос.)*
- *Минский, М., Пейперт, С. Персептроны = Perceptrons. — М. : Мир, 1971. — 261 с. (рос.)*
- *Розенблатт, Ф. Принципы нейродинамики: Перцептроны и теория механизмов мозга = Principles of Neurodynamic: Perceptrons and the Theory of Brain Mechanisms. — М. : Мир, 1965. — 480 с. (рос.)*
- *Уоссермен, Ф. Нейрокомпьютерная техника: Теория и практика = Neural Computing. Theory and Practice. — М. : Мир, 1992. — 240 с. — ISBN 5-03-002115-9. (рос.)*
- *Яковлев С. С. Система распознавания движущихся объектов на базе искусственных нейронных сетей // ИТК НАНБ. — Минск, 2004. — С. 230—234. (рос.)*
- *Stormo G. D., Schneider T. D., Gold L., Ehrenfeucht A. Использование перцептрона для выделения сайтов инициации в E. coli // Nucleic Acids Research. — 1982. — С. P. 2997–3011. (англ.).*
- Mills, Terence C. (1990). Time Series Techniques for Economists. Cambridge University Press. ISBN 0521343399. (англ.)
- Francq, C.; Zakoïan, J.-M. (2005). Recent results for linear time series models with non independent innovations. У Duchesne, P.; Remillard, B. Statistical Modeling and Analysis for Complex Data Problems. Springer. с. 241–265..
- Магнус Я.Р., Катышев П.К., Пересецкий А.А. Эконометрика. Начальный курс. — М.: Дело, 2007. — 504 с. — ISBN 978-5-7749-0473-0.
- Эконометрика. Учебник / Под ред. Елисеевой И.И. — 2-е изд. — М.: Финансы и статистика, 2006. — 576 с. — ISBN 5-279-02786-3.
- Бокс Дж., Дженкинс Г. Анализ временных рядов, прогноз и управление: Пер. с англ. // Под ред. В. Ф. Писаренко. — М.: Мир, 1974, кн. 1. — 406 с.
- Андерсон Т. Статистический анализ временных рядов;
- Кендалл М., Стюарт А. Многомерный статистический анализ и временные ряды (том 3);
- Montgomery D.C., Jennings C.L., Kulahci M. Introduction to time series analysis and forecasting;
- Панкрац, А. (1983). Прогнозирование с одномерной модели Бокса-Дженкинса, концепций и дел. Нью-Йорк: John Wiley и сыновья ".