

Реалізація методу градієнтного бустингу для прогнозування фінансових показників

Вінницький національний технічний університет

Анотація

Розглянуто особливості реалізації методу градієнтного бустингу для прогнозування фінансових показників в інтерактивній програмній системі моніторингу та аналізу фінансових інструментів.

Ключові слова: фінансові інструменти, градієнтний бустинг, методи прогнозування.

Abstract

The features of implementing the gradient boosting method for forecasting financial indicators in an interactive software system for monitoring and analyzing financial instruments are considered.

Keywords: financial instruments, gradient boosting, forecasting methods.

Градієнтний бустинг є потужним методом для задач прогнозування, особливо в умовах великих обсягів даних і складних залежностей. Розглянемо детальніше принципи роботи цього методу та теоретичне обґрунтування кожного етапу [1].

Градієнтний бустинг належить до класу методів ансамблевого навчання. В цьому методі для побудови моделі використовуються кілька "слабких" моделей (зазвичай дерев рішень), кожна з яких коригує помилки попередньої. У градієнтному бустингу ми не об'єднуємо прогнози дерев рішень через їх усереднення чи голосування, як це робиться в інших методах ансамблю, а сумуємо їх, поступово мінімізуючи залишкові помилки.

Модель поступово "вчиться" на помилках, і на кожному кроці додається нова модель, яка компенсує залишкову похибку поточного ансамблю.

Градієнтний бустинг працює з будь-якою диференційованою функцією втрат, що дає йому високу гнучкість. Найчастіше використовуються такі функції втрат:

- Середньоквадратична помилка (MSE):

$$L(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (1)$$

де y_i – реальні значення;

\hat{y}_i – прогнозовані значення.

Ця функція втрат зручна для задач регресії, оскільки сильно штрафує великі помилки.

- Функція логарифмічної втрати (Log Loss):

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)) \quad (2)$$

де y_i – реальні бінарні мітки (0 або 1);

\hat{y}_i – імовірності.

Функція використовується для бінарної класифікації.

Градiєнтний бустинг реалізує **градiєнтний спуск** для оптимізації функції втрат шляхом побудови нових дерев, які мінімізують залишки. Залишки, або *градiєнти*, обчислюються як похідні функції втрат по прогнозованих значеннях, що дозволяє знайти напрямок, у якому модель повинна скоригувати свої передбачення, щоб зменшити помилку.

Процес градiєнтного спуску для бустингу:

1. На кожній ітерації m обчислюється залишок $r_i^{(m)}$ для кожного прикладу:

$$r_i^{(m)} = -\frac{\partial L(y_i, f_{m-1}(x_i))}{\partial f_{m-1}(x_i)} \quad (3)$$

Цей залишок відображає, на скільки потрібно скоригувати прогноз, щоб зменшити значення функції втрат.

2. Далі будується нове дерево $h_m(x)$, яке намагається передбачити значення залишків $r_i^{(m)}$. Дерево навчається на залишках, щоб компенсувати помилки попереднього ансамблю.

3. Прогноз оновлюється з використанням коефіцієнта навчання α , що контролює вплив нового дерева:

$$f_m(x) = f_{m-1}(x) + \alpha h_m(x) \quad (4)$$

Малий коефіцієнт α допомагає уникнути переобучення і забезпечує поступове навчання моделі.

В процесі побудови ансамблю дерев на кожній ітерації додається нове дерево, яке враховує попередні помилки. Кількість таких дерев, або "глибина" ансамблю, обмежена, оскільки надмірне додавання дерев може призвести до переобучення. Зазвичай для градiєнтного бустингу обираються неглибокі дерева, кожне з яких є слабким учнем.

Розглянемо приклад ітерацій побудови ансамблю. Припустимо, що в нас є дані про зміну ціни акцій за 5 днів:

$$y = [100, 102, 104, 103, 105] \quad (5)$$

1. **Початкова оцінка:** обчислюємо середнє значення як початковий прогноз:

$$f_0(x) = \frac{100 + 102 + 104 + 103 + 105}{5} = 102.8 \quad (6)$$

2. **Обчислення залишків для першого дерева:**

$$\begin{aligned} r_1 &= 100 - 102.8 = -2.8 \\ r_2 &= 102 - 102.8 = -0.8 \\ r_3 &= 104 - 102.8 = 1.2 \\ r_4 &= 103 - 102.8 = 0.2 \\ r_5 &= 105 - 102.8 = 2.2 \end{aligned} \quad (7)$$

3. **Навчання першого дерева $h_1(x)$ на залишках**:** дерево будується так, щоб передбачати ці значення залишків. Наприклад, перше дерево може виявити, що залишки є систематично заниженими або завищеними.

4. **Оновлення прогнозу:** Припустимо, що ми використовуємо коефіцієнт навчання $\alpha = 0.1$. Тоді новий прогноз для кожного значення буде оновлено на основі прогнозів першого дерева:

$$f_1(x) = f_0(x) + 0.1 * h_1(x) \quad (8)$$

Процес повторюється кілька разів, з кожним новим деревом, поки залишкова помилка не стане досить малою.

Переваги градієнтного бустингу.

1. **Гнучкість:** градієнтний бустинг може використовувати різні функції втрат, що робить його придатним для регресії, класифікації, ранжування та інших задач.

2. **Точність:** цей метод забезпечує високу точність завдяки послідовній корекції помилок. Кожне дерево враховує помилки попередніх дерев, що робить модель дуже потужною.

3. **Можливість обробляти складні залежності:** градієнтний бустинг здатен виявляти нелінійні зв'язки та взаємодії між змінними, які важко виявити за допомогою простіших моделей.

Недоліки градієнтного бустингу.

1. **Високі обчислювальні витрати:** навчання великої кількості дерев вимагає значних обчислювальних ресурсів.

2. **Чутливість до параметрів:** градієнтний бустинг має багато гіперпараметрів (кількість дерев, глибина дерев, коефіцієнт навчання), які потребують ретельного налаштування.

3. **Схильність до переобучення:** при надмірній кількості дерев або завеликих значеннях параметрів модель може почати відображати шум у навчальних даних.

Градієнтний бустинг є одним із найефективніших методів машинного навчання для прогнозування фінансових показників. Завдяки гнучкості та здатності обробляти складні взаємозалежності він чудово підходить для прогнозування цін акцій, прибутковості та інших фінансових інструментів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ораельлі Реджан, Шлегель Марк. Gradient Boosting with XGBoost and Scikit-learn: Learn Advanced Machine Learning Techniques with Python, 2022. - Packt Publishing – 420 с.

Карпенко Максим Анатолійович, студент групи 2ПІ-23м, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Україна

Maksym Karpenko, student of group 2PI-23m, Faculty for Information Technologies and Computer Engineering, Vinnytsia National Technical University, Ukraine, diskmax008@gmail.com.