ИСПОЛЬЗОВАНИЕ ГРАДИЕНТНОГО БУСТИНГА НАД РЕШАЮЩИМИ ДЕРЕВЬЯМИ ДЛЯ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ

Хмельницкий В. В.
Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектороники
Минск, Республика Беларусь
E-mail: vovchyt@gmail.com

В данной статье рассматривается применение градиентного бустинга для решения задачи прогнозирования временных рядов. Приводится краткое описание модели машинного обучения, а также сравнение со статистическими моделями, решающими аналогичную задачу.

Введение

Временные ряды являются важной частью современной аналитики данных и имеют большое практическое применение в различных областях, начиная от финансов и заканчивая метеорологией.

Существует множество способов прогнозирования временных рядов, однако многие из них имеют свои ограничения, среди которых большие затраты ресурсов и времени, а также необходиомсть в дополнительной обработке входных данных. Применение градиентного бустинга может помочь обойти часть данных ограничений.

I. Основные проблемы прогнозирования временных рядов

На данный момент существует большое количество моделей, прогнозирующих временные ряды, но они делятся на две основные категории: статистические и машинное обучение. Особой популярностью среди статистических моделей пользуется ARIMA (Autoregressive Integrated Moving Average) и её модификации. Данная модель является комбинацией двух моделей: авторегрессии и скользящего среднего.

Основные проблемы при прогнозировании статистическими моделями:

- требование к рядам данных: для построения адекватной модели ARIMA требуется не менее 40 наблюдений, а для рядов, обладающих сезонностью – порядка 6–10 сезонов, что на практике не всегда возможно;
- неадаптивность моделей авторегрессии: при получении новых данных модель нужно периодически переоценивать;
- большие затраты ресурсов и времени на подбор параметров модели и предварительную обработку ряда [1].

Прогнозирование временных рядов моделями машинного обучения основывается на наличии набора данных для обучения и валидации (что уже является недостатком по отношению к статистическим моделям). Однако при этом данные модели являются более адаптивными (нет необходимости переобучаться при прогнозировании нового ряда), а также многие из них не требуют сложной предварительной обработки входных данных. Градиентный бустинг обладает всеми этими преимуществами.

II. Применение градиентного бустинга

Градиентный бустинг – это техника машинного обучения для задач классификации и регрессии, которая строит модель предсказания в форме ансамбля слабых предсказывающих моделей. При этом обучение ансамбля проводится последовательно.

Бустинг, использующий деревья решений в качестве базовых алгоритмов, называется градиентным бустингом над решающими деревьями (Gradient Boosting on Decision Trees, GBDT). Он способен эффективно находить нелинейные зависимости в данных различной природы. Этим свойством обладают все алгоритмы, использующие деревья решений, однако именно GBDT обычно выигрывает в подавляющем большинстве задач. Благодаря этому он широко применяется во многих прикладных задачах (поисковом ранжировании, рекомендательных системах, таргетировании рекламы, прогнозировании погоды и многих других).

Рассмотрим решение задачи регрессии с помощью градиентного бустинга. В качестве оптимизируемой функции возьмём квадратичную функцию потерь:

$$L(x, y) = \frac{1}{2}\sum_{i=1}^{N}(y_i - a(x_i))^2.$$

Для решения будем строить композицию из K базовых алгоритмов:

$$a(x) = a_K(x) = b_1(x) + b_2(x) + ... + b_K(x).$$

В качестве базовых алгоритмов выберем, как и условились, семейство решающих (регрессионных) деревьев некоторой фиксированной глубины.

Используя известные методы построения решающих деревьев, обучим алгоритм $b_1(x)$, который наилучшим образом приближает целевую

переменную:

$$b_1(x) = argmin, L(y, b(x)).$$

Построенный алгоритм $b_1(x)$, скорее всего, работает не идеально. Вычислим, насколько сильно отличаются прогнозы этого дерева от истинных значений:

$$s_i^1 = y_i - b_1(x_i).$$

Теперь мы хотим скорректировать $b_1(x)$ с помощью $b_2(x)$. Для этого второе решающее дерево будет обучаться прогнозировать разности s_i^1:

$$b_2(x) = argmin, L(s^1, b(x)).$$

Ожидается, что композиция из двух таких моделей $a_2(x) = b_1(x) + b_2(x)$ станет более качественно предсказывать целевую переменную y.

Далее все действия повторяются до построения всей композиции. На k-ом шаге вычисляется разность между истинным ответом и текущим прогнозом композиции, затем k-й алгоритм обучается предсказывать эту разность, а затем обновляется вся композиция [2]:

$$s_i^k = y_i - a_k - 1(x_i),$$

$$b_k(x) = argmin, L(s^{k-1}, b(x)),$$

$$a_k(x) = a_{k-1}(x) + b_k(x).$$

Для обучения описанной модели и её применения в прогнозировании временных рядов необходимо иметь набор данных (т.е. некоторое множество рядов). При этом данных должно быть как можно больше, они должны содержать исчерпывающее количество видов зависимостей, а также в них должен отсутствовать выраженный дисбаланс.

Временной ряд можно представить как вектор признаков, где каждая его компонента - значение исследуемой величины в определенный промежуток времени. Однако для увеличения скорости обучения и работы алгоритма можно взять лишь несколько самых важных признаков (учитывающих недельную, месячную и годовую сезонность, а также текущий тренд). Таким образом, уменьшится размерность входных данных и, следовательно, их объём.

В качестве целевой переменной выступает следующее (неизвестное) значение временного ряда. Для прогнозирования первого неизвестного значения в качестве входного параметра выступает вектор, полученный из первоначального ряда:

$$X^0 = (x_1, x_2, ..., x_n).$$

Затем из данного вектора выбираются самые важные признаки (по порядковым номерам, определенным заранее):

$$x^0 = (x_i, x_j, ..., x_k).$$

После получения первого неизвестного значения $y^0 = a(x^0)$, оно добавляется в конец исходного временного ряда:

$$X^1 = (x_1, x_2, ..., x_n, y^0)$$

Затем уже из нового вектора X^1 выбираются самые важные признаки:

$$x^1 = (x_{i+1}, x_{j+1}, ..., x_{k+1}).$$

Далее все действия повторяются до прогнозирования последнего p-го значения:

$$X^p = (x_1, x_2, ..., x_n, y^0, y^1, ..., y^{p-1})$$

$$x^p = (x_{i+p}, x_{j+p}, ..., x_{k+p}).$$

$$y^p = a(x^p)$$

Таким образом, полученная модель на основе градиентного бустинга может принимать на вход любой временной ряд и прогнозировать следующие значения, основываясь на данных о других зависимостях из обучающей выборки (в отличие от статистических моделей, которые для прогноза используют лишь данные об исследуемом ряде).

III. Заключение

Описанный метод прогнозирования временных рядов обладает своими преимуществами и недостатками по отношению к рассмотренным статистическим методам. Выбор должен осуществляться исходя из имеющегося времени и ресурсов, а также из требований, предъявляемых к моделям и прогнозам.

Список литературы

1. Бизнес-прогнозирование / Д. Э. Ханк [и др.]. – пер. с англ. 7-е изд. М.: Вильямс, 2003. 506 с.
2. Градиентный бустинг [Электронный ресурс] / Школа Анализа Данных. – Режим доступа: `https://academy.yandex.ru/handbook/ml/article-gradientnyj-busting`. – Дата доступа: 17.10.2023.