Временные ряды потребления контента и усиление их прогнозирующих свойств построением сплит-моделей

В. Г. Мосин

Аннотация

В На основании имеющегося временного ряда данных о потреблении контента на канале одного из ведущих хостингов построена регрессионная модель, аппроксимирующая его поведение кубическим полиномом. После разбиения данных на несколько непересекающихся классов построена серия подобных регрессионных моделей. Показано, что объединение частичных моделей приводит к существенному повышению точности аппроксимации.

Содержание

1 Введение

Временные ряды — это наборы данных, в которых значения измеряются в последовательные моменты времени. Они используются для анализа и прогнозирования переменных, которые изменяются во времени, таких как цены акций, количество продаж и другие. В анализе временных рядов применяются различные методы и модели для выявления скрытых закономерностей, трендов, цикличности, сезонности и шума в данных. Результаты таких исследований используются для предсказания будущих значений на основе предыдущих наблюдений.

1.1 Теоретическая часть

Если временной ряд обладает высокой волатильностью, то его анализ представляет определенные сложности из-за своих особенностей. Вот некоторые из них:

1. Шум и непостоянство. Высокая волатильность приводит к шуму и непостоянству в данных. Это затрудняет определение трендов и паттернов во временных рядах.
2. Нестационарность. Временные ряды с высокой волатильностью часто бывают нестационарными, то есть их статистические свойства (среднее и дисперсия) меняются со временем. Нестационарность усложняет применение классических методов анализа, которые предполагают стационарность.

3. Сложности прогнозирования. Волатильность создает большую неопределенность в прогнозировании будущих значений временных рядов. Это связано с тем, что даже небольшие изменения в данных или внешние факторы могут привести к значительным изменениям в прогнозах.
4. Необходимость использования продвинутых методов. С учетом всех вышеперечисленных сложностей, анализ временных рядов с высокой волатильностью требует применения более продвинутых методов и моделей для достижения точных результатов.

Анализ временных рядов с высокой волатильностью может представлять определенные вызовы, но с правильным подходом можно извлечь полезную информацию. Один из таких подходов состоит в построении ансамбля моделей (см. [5], [6], [7]). Ансамблирование моделей в машинном обучении — это метод, который объединяет предсказания нескольких моделей, чтобы получить более точный и стабильный прогноз. Он основан на идее, что объединение нескольких слабых моделей может привести к созданию более сильной и устойчивой модели. Процесс ансамблирования включает в себя два основных шага:

1. Создание множества моделей. Вместо использования единственной модели для предсказания, ансамбль создает несколько моделей, которые могут быть обучены с использованием различных методов или различных наборов данных. Например, можно использовать разные алгоритмы машинного обучения, разные фрагменты данных и т. д.
2. Комбинирование предсказаний. После того как модели обучены, их предсказания объединяются для получения окончательного прогноза.

1.2 Постановки задачи

1.2.1 Предмет исследования

Мы исследуем данные о просмотрах образовательного контента на канале одного из ведущих хостингов. Данные представлены в виде временного ряда с дневной периодичностью с 2023-03-28 по 2023-06-28.

1.2.2 Методика исследования

Для исследования мы применяем методы линейной и полиномиальной регрессии в два этапа. На первом этапе мы анализируем полные данные по всей выборке. На втором этапе мы разбиваем выборку на несколько фрагментов и анализируем их по отдельности, после чего агрегируем результаты.

1.2.3 Цель исследования

Наша цель состоит в повышении прогнозирующей способности регрессионной модели временного ряда данных. Для этого мы сравниваем результаты моделирования, которые получаются на первом и втором этапах нашего исследования.

1.3 Библиотеки

Для выполнения вычислений и анализа данных мы пользуемся средой `Jupyter Notebook`, которая предоставляет удобные средства для работы с языком программирования Python и его главными библиотеками: `NumPy`, `Pandas`, `sklearn` и `matplotlib`. Благодаря этим инструментам, мы можем эффективно работать с данными, выполнять исследования и визуализировать результаты (см. [1], [2]).

2 Описание данных

Данные содержат сведения о количестве просмотров образовательного контента на канале одного из ведущих хостингов, представленные в виде временного ряда. Период временного ряда — с 2023-03-28 по 2023-06-28, частота наблюдений — ежедневно.

3 Алгоритм

3.1 Чтение данных

Методом `read_csv` библиотеки `pandas` загружаем в среду исполнения набор данных и формируем дата-фрейм.

	Дата	Просмотры
0	2023-03-28	1557
1	2023-03-29	1313
2	2023-03-30	1248
...	...	...
91	2023-06-27	980
92	2023-06-28	902

Признак 'Дата' относится к строковому типу, признак 'Просмотры' — к типу целых чисел, пропущенных данных нет.

3.2 Визуализация временного ряда

Пользуясь методом `plot` библиотеки `matlpotlib`, визуализируем временной ряд: по горизонтали отложены номера дат, по вертикали — количество просмотров (см. рис. 1).

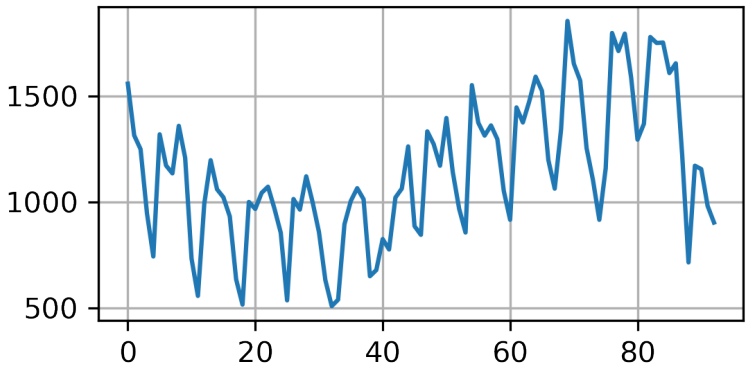

Рис. 1: Временной ряд просмотров

3.3 Построение и обучение моделей

Мы проведем исследование и сравним несколько регрессионных моделей: сначала построим линейную модель, а затем две полиномиальные — квадратичную и кубическую.

3.3.1 Предварительная подготовка данных

Данные признака 'Дата' являются строковыми, и мы не можем использовать этот признак в качестве независимой переменной регрессионной задачи, поскольку библиотека `sklearn` поддерживает только числовые данные. Мы добавляем в дата-фрейм столбец, совпадающий с индексом, и объявляем его признаком 'Номер':

	Дата	Номер	Просмотры
0	2023-03-28	1	1557
1	2023-03-29	2	1313
2	2023-03-30	3	1248
...	...	...	...
91	2023-06-27	92	980

	Дата	Номер	Просмотры
92	2023-06-28	93	902

Визуально ничего не изменилось. Если посмотреть на рис. 1, то на нем и раньше по горизонтали откладывались не сами даты, а их номера. Зато теперь данные являются числовыми, и мы можем строить регрессионные модели.

3.3.2 Линейная модель

Сначала мы формируем одномерную регрессионную задачу с линейной моделью. Пользуясь методом `drop` библиотеки `pandas`, удаляем из датафрейма признаки 'Просмотры' и 'Дата', пользуясь методом `to_numpy`, переводим оставшийся фрагмент в массив `numpy` и объявляем его левой частью регрессионной задачи (обозначение — `X`). В качестве правой части — наоборот, используем признак 'Просмотры' (обозначение — `y`). Затем, пользуясь методом `LinearRegression` модуля `sklearn.linear_model` из библиотеки `sklearn`, мы формируем объект `model` и, применяя к нему метод `fit`, обучаем модель на `X` и `y`. При помощи метода `score` мы вычисляем коэффициент детерминации: $R^2 = 0.22$, и наконец, средствами библиотеки `matplotlib` визуализируем модель (см. рис. 2(a)).

3.3.3 Квадратичная модель

Полученный выше коэффициент детерминации кране низок. Чтобы усилить нашу модель, мы вводим еще один предиктор, равный квадрату предиктора 'Номер':

	Дата	Номер	Квадрат номера	Просмотры
0	2023-03-28	1	1	1557
1	2023-03-29	2	4	1313
2	2023-03-30	3	9	1248
...	...	...	...	...
91	2023-06-27	92	8281	980
92	2023-06-28	93	8464	902

Затем, повторяя действия, оцениваем (получается $R^2 = 0.25$, это лучше чем было, но по-прежнему очень мало) и визуализируем квадратичную модель (см. рис. 2(b)).

3.3.4 Кубическая модель

Следующая попытка усилить модель состоит в том, что мы вводим еще один предиктор, равный кубу предиктора 'Номер':

	Дата	Номер	Квадрат номера	Куб номера	Просмотры
0	2023-03-28	1	1	1	1557
1	2023-03-29	2	4	8	1313
2	2023-03-30	3	9	27	1248
...	...	...	...	...	...
91	2023-06-27	92	8281	753571	980
92	2023-06-28	93	8464	778688	902

После чего, повторяя действия, обучаем, оцениваем и визуализируем кубическую модель (см. рис. 2(c)).

На этот раз поучаем $R^2 = 0.48$. Это неудовлетворительно низкий результат. Причем, характер поведения кривой временного ряда таков, что кубическая парабола хорошо воспроизводит ее основной тренд, но не справляется с дневными колебаниями, и никакая алгебраическая кривая большей степени с ними тоже не справится, то есть, путь полиномиального усиления регрессионной модели следует признать тупиковым.

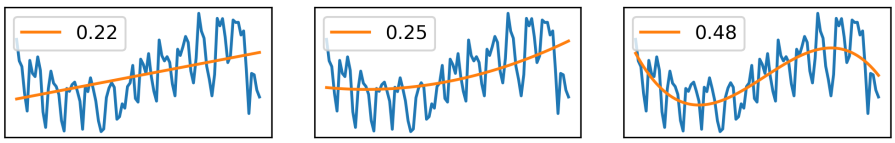

Рис. 2: Модели и их оценки

Это значит, что нам нужны какие-то другие идеи для решения.

3.4 Повторное чтение данных

Мы прочитали данные из файла *.csv, а этот формат не содержит сведений о типе данных. Поэтому признак 'Дата' был прочитан как строковый, в тот момент как на самом деле он относится к типу дата-время. Чтобы изменить тип этого признака, мы методом `read_csv` библиотеки `pandas` снова читаем данные из файла *.csv, но на этот раз используем атрибут `parse_dates`, для того чтобы интерпретировать признак 'Дата' в нужном формате. Теперь признак 'Дата' имеет тип дата-время, а это значит, что он приобретает дополнительную информативность.

3.5 Построение дополнительного признака

Формат дата-время позволяет апеллировать к календарям и вычислять по любой дате день недели, который приходился (или еще придется) на эту дату. Для этого в библиотеке `pandas` есть метод `dt.dayofweek`, применяя который мы заносим в наш дата-фрейм еще один предиктор 'День недели':

	Дата	День недели	Просмотры
0	2023-03-28	1	1557
1	2023-03-29	2	1313
2	2023-03-30	3	1248
...	...	...	...
91	2023-06-27	1	980
92	2023-06-28	2	902

Дни недели закодированы, начиная с понедельника, таким образом: понедельник — 0, воскресенье — 6, и все промежуточные значения по порядку.

3.6 Визуализация дополнительного признака

На этом шаге мы применяем методы библиотеки `matplolib`, для того чтобы отметить чередование дней недели на общей кривой временного ряда (см. рис. 3).

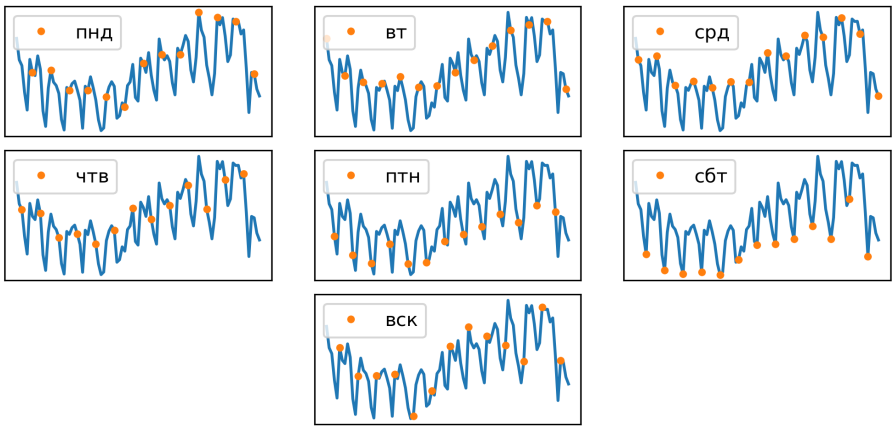

Рис. 3: Просмотры, локализованные по дням недели

Бросается в глаза разительное отличие, например, суббот от вторников,

хотя само по себе это наблюдение, конечно, не является неожиданным. Действительно, мы изучаем потребление образовательного контента, а образовательный процесс обладает естественной недельной цикличностью: по выходным студенты, как правило, отдыхают, а не учатся — отсюда глубокие пятничные и субботние провалы в потреблении. В начале недели студенты проявляют наибольшую активность, а воскресенье, хотя и выходной, но уже заставляет студентов задуматься о том, что завтра понедельник.

3.7 Частичные модели

Наша основная идея состоит в том, чтобы моделировать потребление контента не на всем массиве данных, а на отдельных его фрагментах: построить и обучить отдельную модель для понедельников, отдельную — для вторников и так далее, по всем дням недели.

3.7.1 Предварительная подготовка данных

Здесь мы, так же как и выше применяем сквозную нумерацию объектов, добавляя в дата-фрейм дополнительный признак 'Номер'.

	Дата	Номер	День недели	Просмотры
0	2023-03-28	1	1	1557
1	2023-03-29	2	2	1313
2	2023-03-30	3	3	1248
...	...	...	...	...
91	2023-06-27	92	1	980
92	2023-06-28	93	2	902

3.7.2 Модель на понедельниках

Прежде всего, пользуясь методом `loc` библиотеки `pandas`, мы выполняем локализацию дата-фрейма по условию 'День недели' = 0 и получаем локальный дата-фрейм:

	Дата	Номер	День недели	Просмотры
6	2023-04-03	0	6	1171
13	2023-04-10	0	13	1196

	Дата	Номер	День недели	Просмотры
20	2023-04-17	0	20	966
...	...	...	...	...
83	2023-06-19	0	83	1749
90	2023-06-26	0	90	1154

В нем, в отличие от полных данных не 93, а всего 13 записей, причем, номера и даты следуют с шагом 7. Затем строим и обучаем кубическую модель, вычисляем ее метрику эффективности $R^2 = 0.79$ и визуализируем в виде кубической параболы (см. рис. 4(а)).

3.7.3 Модели на остальных днях недели

Повторяем в цикле по дням недели. Получаем еще 6 моделей (каждую на своей локализации), обучаем их, вычисляем метрики эффективности и визуализируем (см. рис. 4).

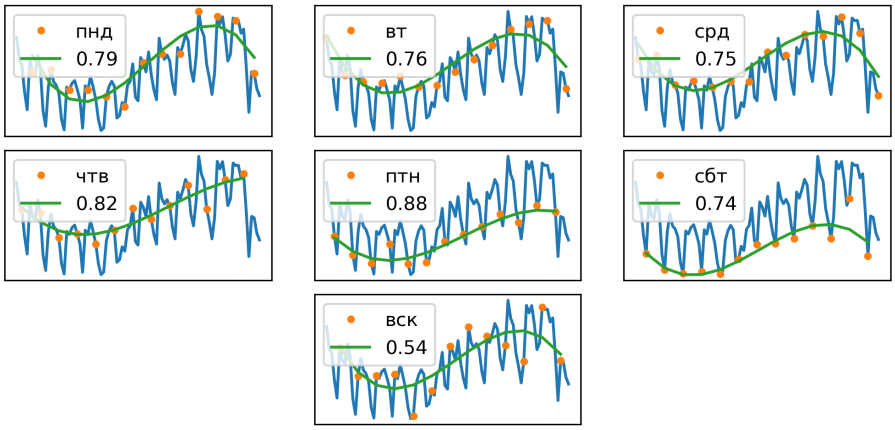

Рис. 4: Частичные модели и их оценки

4 Результаты

Как линейная, так и полиномиальная модели потребления контента на полных данных оказались неудовлетворительными. Поэтому мы разбили данные на классы по дням недели, после чего построили и обучили частичные регрессионные модели третьей степени. Каждая из них по сравнению с начальной полиномиальной моделью обладает большим коэффициентом детерминации, однако, окончательная прогнозирующая

способность всей совокупности частичных моделей оказывается еще выше (см. далее).

4.1 Композиция моделей

Итак, у нас есть семь моделей потребления контента в зависимости от дня недели. Пользуясь методом `coef_` библиотеки `sklearn`, которая выводит наборы коэффициентов, а также методом `intercept_`, который возвращает свободный член неоднородной регрессионной модели, мы можем описать потребление в виде явной функции от времени:

Функция	**Условие**
$1789.4219 - 85.5687n + 2.2736n^2 - 0.0153n^3$	если понедельник
$1575.2983 - 61.1187n + 1.7041n^2 - 0.0118n^3$	если вторник
$1536.0606 - 57.3905n + 1.6779n^2 - 0.0119n^3$	если среда
$1331.5792 - 32.1736n + 0.8119n^2 - 0.0046n^3$	если четверг
$1018.4471 - 33.0131n + 0.9056n^2 - 0.0057n^3$	если пятница
$0889.4622 - 41.2487n + 1.2501n^2 - 0.0089n^3$	если суббота
$1621.7125 - 68.7657n + 1.8182n^2 - 0.0122n^3$	если воскресенье

Здесь n означает порядковый номер дня в ряду от 0 до 92. Эта функция не имеет аналитического описания, но это не делает ее менее привлекательной с точки зрения эффективности ее прогнозирующих свойств. Напротив, она эффективна именно потому, что не претендует на универсальность, а разнесена на семь записей в виде семи полиномов.

4.2 Метрика композиции моделей

Чтобы получить общую метрику эффективности ансамбля моделей, нужно вспомнить определение коэффициента детерминации (см. [3], [4]):

$$R^2 = 1 - S^*/S'.$$

Здесь S^* — накопленный квадрат отклонения предсказанных значений от истинных значений целевой функции, а S' — накопленный квадрат отклонения среднего значения целевой функции от ее истинных значений. Применяя метод `predict` библиотеки `sklearn`, получаем предсказанные значения `y_star` для каждой из частичных моделей, вычитаем их из истинных значений `y`, возводим в квадрат и суммируем методом `sum`. После этого в цикле по дням недели складываем все частичные суммы, получаем `S_star` для всего ансамбля моделей. Для вычисления знаменателя

`S_line` применяем метод `mean`, который возвращает среднее значение массива, к правой части общей регрессионной задачи, отнимаем среднее значение от общего массива истинных значений, возводим результат в квадрат и суммируем элементы массива методом `sum`.

В результате имеем $R^2 = 0.82$. Если сравнить это значение с коэффициентом детерминации, полученным выше, на шаге, когда мы строили кубическую модель на полных данных (тогда получалось $R^2 = 0.48$), то окажется, что мы повысили метрику эффективности на 4 единицы в первом знаке после запятой! Это колоссальный рост коэффициента детерминации (обычно при повышении прогнозирующей способности регрессионной модели борьба ведется за хотя бы какой-нибудь сдвиг во втором знаке).

5 Выводы

Регрессионный анализ на частичных фрагментах данных с последующим построением ансамбля моделей обладает более высокой прогнозирующей способностью по сравнению с регрессионной моделью, построенной на полных данных по всей выборке. Это происходит, прежде всего, из-за учета локальной структуры данных: алгоритмы ансамблирования моделей, построенные на частичных фрагментах данных, точнее учитывают локальную структуру данных и нелинейные зависимости в выборке. Модель, построенная на полных данных, может упустить некоторые особенности или несоответствия в данных, а это приводит к менее точным прогнозам.

5.1 Обобщения

В нашем исследовании нам удалось существенно повысить точность регрессионной модели после того, как мы включили в данные дополнительный такой показатель как день недели, что связано с недельным образовательным циклом. Однако ярко выраженная недельная цикличность наблюдается не только в образовании и потреблении образовательного контента, но и во многих других общественных, экономических и производственных процессах.

1. Например, в большинстве стран установлен 5-дневный рабочий график с выходными днями в субботу и воскресенье. Недельная цикличность имеет прямое отношение к этому графику.
2. Потребление продуктов и услуг также имеет недельную цикличность: в выходные дни у потребителей больше времени для похода в рестораны, совершения покупок или посещения культурно-развлекательных мероприятий.

3. Отдельного внимания заслуживает изучение транспортного потока: интенсивность транспортного движения существенно меняется в зависимости от дня недели. В будние дни транспортный поток обычно увеличивается, особенно в часы пик, в выходные дни и вечером движение обычно снижается.
4. В туристической отрасли можно наблюдать выраженную недельную цикличность. Многие люди предпочитают ездить в путешествия на выходных, поэтому спрос на отели, рестораны и развлекательные мероприятия оказывается выше в эти дни.

И это лишь некоторые примеры процессов, в которых можно наблюдать ярко выраженную недельную цикличность.

5.2 Рекомендации

С учетом сказанного выше, вне зависимости от предметной области, если процесс обладает ярко выраженной недельной цикличностью, то целесообразно провести регрессионное моделирование, учитывая этот фактор.

Прежде всего, абсолютно необходимо добавление фактора дня недели, так как включение этого фактора в регрессионную модель позволит учесть недельную цикличность. Для этого можно создать фиктивные переменные, соответствующие каждому дню недели, и включить их в модель в качестве предикторов. Это поможет учесть различия в поведении процесса по дням недели.

Кроме того, рекомендуется ансамблирование моделей, разделенных по дням недели. Это означает построение отдельных модели для каждого дня недели и использование их для прогнозирования в соответствующие дни. Когда такие модели построены и обучены, их можно объединить в виде ансамбля для получения общего прогноза.

Вместе с тем нужно отметить, что ансамблирование моделей может быть полезным, только если недельная цикличность действительно важна, и различия в поведении процесса действительно ярко выражены по дням недели. Следует помнить, что использование ансамблей моделей требует дополнительных ресурсов и анализа результатов, поэтому решение об организации ансамблирования нужно принимать на основе анализа данных, контекста и конкретных требований задачи прогнозирования.

6 Литература

1. Хейдт М. Изучаем Pandas / М. Хейдт; — Москва: ДМК Пресс, 2018. — 438 с.

2. Бурков А. Машинное обучение без лишних слов / А. Бурков; — СПб: Питер, 2020. — 192 с.
3. Вьюгин, В. В. Математические основы теории машинного обучения и прогнозирования / В. В. Вьюгин; — М.: МЦИМО. — 2013. — 387 с.
4. Бринк Х. Машинное обучение / Х. Бринк, Дж. Ричардс, М. Феверолф — СПб.: Питер, 2017. — 336 с.
5. Газизов Д. И. Обзор методов статистического анализа временных рядов и проблемы, возникающие при анализе нестационарных временных рядов // Научный журнал. 2016. № 3 (4). С. 9–14.
6. Фирулина М. М., Корунова Н. В. Разработка автоматизированных методов прогнозирования временных рядов // В сборнике: Прикладные информационные системы. Вторая Всероссийская НПК: сборник научных трудов. 2015. С. 401–405.
7. Безверхий О. В., Курейчик В. М. Применение интеллектуальных информационных технологий при прогнозировании временных рядов // В сборнике: Искусственный интеллект в автоматизированных системах управления и обработки данных. Сборник статей Всероссийской научной конференции. В 2-х томах. Москва, 2022. С. 160–165.