МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. М.В. Ломоносова

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И КИБЕРНЕТИКИ

КАФЕДРА МАТЕМАТИЧЕСКИХ МЕТОДОВ ПРОГНОЗИРОВАНИЯ

ДИПЛОМНАЯ РАБОТА:

«СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ АЛГЕБРАИЧЕСКОЙ КОРРЕКЦИИ ДЛЯ ОДНОГО КЛАССА АЛГОРИТМОВ ПРОГНОЗИРОВАНИЯ»

Выполнила студентка 517 группы:
Егорова Е.В.

Научные руководители:
чл.-корр. РАН, д.ф.-м.н. Рудаков К.В.
к.ф.-м.н. Воронцов К.В.

МОСКВА
2005 г.

Содержание

Аннотация

При прогнозировании зашумленных нестационарных временных рядов возникает проблема выбора адекватной модели временного ряда. Известные модели нестационарных процессов существенно опираются на априорные предположения о природе нестационарности (например, гипотезу о непостоянстве дисперсии) и потому являются в той же степени эвристическими, что и классические стационарные модели.

В данной работе рассматриваются методы построения динамически адаптируемых композиций алгоритмов прогнозирования. Их преимущество в том, что они позволяют автоматически выбирать наиболее адекватную модель временного ряда. Исследуется влияние регуляризации и дополнительного ограничения монотонности (неотрицательности коэффициентов) на обобщающую способность композиции. Наилучшее качество обучения при тестировании скользящим контролем показал алгоритм локальной адаптации весов с регуляризацией.

Предложенные алгоритмы реализованы в среде MATLAB и протестированы на реальных данных в прикладной задаче прогнозирования объемов продаж в сети супермаркетов.

1 Введение

В настоящей работе рассматривается задача прогнозирования временных рядов. Существует большое количество методов построения прогнозов, в том числе статистические регрессионные, нейросетевые и т.п. Все эти алгоритмы в большей или меньшей степени являются эвристическими, то есть основанными на соображениях здравого смысла, учитывающих специфические особенности конкретной задачи прогнозирования или класса похожих задач. При решении сложных задач прогнозирования часто оказывается, что ни один из имеющихся эвристических алгоритмов не даёт желаемого качества на обучении. В таких случаях имеет смысл строить композицию алгоритмов, в которой недостатки одних алгоритмов компенсировались бы достоинствами других. При определённых условиях качество композиции может оказаться заметно лучше, чем у отдельных составляющих её алгоритмов. Именно эта идея лежит в основе алгебраического подхода к построению корректных алгоритмов прогнозирования.

В данной работе рассматриваются различные методы построения корректирующих операций. Среди них есть классические методы, к которым относятся метод наименьших квадратов, метод наименьших квадратов с регуляризацией и новый метод коррекции прогнозов, основанный на локальной адаптации весов с регуляризацией. Также рассмотрены простейшие алгоритмы, такие как среднее арифметическое и выбор наилучшего из базовых алгоритмов. Они используются в качестве эталонных алгоритмов при экспериментальном сравнении качества работы различных методов коррекции прогнозов.

В первом разделе настоящей работы обсуждается общая постановка задачи прогнозирования временных рядов, алгебраический подход к синтезу корректных алгоритмов прогнозирования и рассматривается прикладная задача прогнозирования потребительского спроса. Формулируется постановка задачи дипломной работы.

Второй раздел содержит описание алгоритмов прогнозирования многомерных временных рядов, которые можно использовать в качестве базовых алгоритмов при построении композиций. Рассмотрены следующие алгоритмы: ARMA, ARIMA, многомерная линейная регрессия и нейронные сети.

В третьем разделе содержится подробное описание методов линейной коррекции, реализованных в рамках работы над дипломом.

Четвертый раздел содержит результаты вычислительных экспериментов и сравнительный анализ построенных корректоров.

Задачей настоящей работы является задача прогнозирования многомерных временных рядов. Она заключается в том, чтобы предсказать поведение измеряемых величин в будущие моменты времени на основе поведения этих величин в прошлом. Эта задача, безусловно, актуальна для современного бизнеса и промышленности, так как в настоящее время существует множество информационных систем, накапливающих информацию в виде временных рядов, и есть необходимость ее анализировать. Во многих случаях от качества решения поставленной задачи прогнозирования непосредственно зависит эффективность управления предприятием.

В качестве прикладной задачи была рассмотрена задача прогнозирования потребительского спроса. Разработанные алгоритмы были протестированы на реальных данных.

Работа содержит 41 страницу, 10 иллюстраций, 19 источников литературы.

1.1 Задачи прогнозирования временных рядов

Совокупность величин $\{x_1, x_2, \ldots, x_n\}$, представляющая собой значения какого-либо параметра, изменяющегося во времени, называется временным рядом, при этом каждое значение соответствует значению параметра в конкретное время $t_1, t_2, \ldots, t_n$. Задача прогнозирования заключается в определении значения измеряемой величины x в момент времени $t_{n+1}, t_{n+2}, \ldots$, то есть для выполнения прогнозирования необходимо выявить закономерность этого временного ряда.

1.2 Об алгебраическом подходе к синтезу корректных алгоритмов прогнозирования

Наиболее общее определение алгоритмической композиции даётся в алгебраическом подходе Журавлёва. Вводится множество R, называемое пространством оценок, и рассматриваются алгоритмы, имеющие вид суперпозиции $A(x) = C(B(x))$, где функция $B : X \to R$ называется алгоритмическим оператором, а функция $C : R \to Y$ – решающим правилом. Данное предположение о структуре алгоритмов не является хоть сколько-нибудь стесняющим ограничением. Отметим, что строение множества R пока никак не фиксируется. Во-вторых, всегда остаётся возможность отказаться от использования решающего правила, положив $R = Y$ и взяв тождественное отображение $C(B) \equiv B$. В задачах восстановления регрессии и прогнозирования обычно так и поступают.

Алгоритмической композицией, составленной из алгоритмических операторов $B_i : X \to R$, $t = 1, \ldots, T$, корректирующей операции $F : R^p \to R$ и решающего правила $C : R \to Y$ называется алгоритм $A : X \to Y$ вида

$$A(x) = C(F(B_1(x), \ldots, B_p(x))), \qquad x \in X. \qquad (1.1)$$

В англоязычной литературе в основном рассматривается случай $R = \Re$, и алгоритмические операторы называются вещественнозначными классификаторами (real-valued classifiers). В общем случае операторы B_i, составляющие алгоритмическую композицию, называют также базовыми алгоритмами (base algorithms).

Основные свойства алгоритмических композиций, отличающие их от обычных эвристических алгоритмов.

- Композиция объединяет базовые алгоритмы, способные самостоятельно решать ту же исходную задачу.
- Композиция не знает внутреннего устройства базовых алгоритмов. Для неё это «чёрные ящики», имеющие только две функции: обучения по заданной выборке и вычисления ответа для заданного объекта. Это свойство очень удобно с технологической точки зрения: для настройки базовых алгоритмов можно задействовать богатый арсенал стандартных методов обучения.
- Композиция позволяет получать высокое качество обучения, недостижимое для отдельных базовых алгоритмов.

Два основных принципа построения алгоритмических композиций.

- Специализация. Пространство объектов делится на области, в каждой из которых строится свой алгоритм, специализирующийся на объектах только этой области. Исходная задача разбивается на более простые подзадачи по принципу «разделяй и властвуй». К таким методам относятся комитеты старшинства, решающие деревья и смеси экспертов.

– Усреднение. В этом случае корректирующая операция не получает информации о том, в какой области пространства находится объект, и работает только с ответами, выданными базовыми алгоритмами. Если базовые алгоритмы достаточно различны, то в результате усреднения их погрешности компенсируют друг друга. Причём усреднение следует понимать в обобщённом смысле, это не обязательно среднее арифметическое, и даже не обязательно линейная операция. На идее усреднения основаны комитеты большинства, бустинг, баггинг, монотонная коррекция.

Основные стратегии построения алгоритмических композиций.

– Последовательная оптимизация. Базовые алгоритмы строятся по очереди, и каждый следующий старается компенсировать недостатки предыдущих. Это жадная стратегия. Она не гарантирует построения наилучшей композиции, но на практике оказывается наиболее удобной. К таким методам относятся комитеты, бустинг, монотонная коррекция.

– Глобальная оптимизация. Базовые алгоритмы перестраиваются по очереди с помощью итерационного процесса, называемого EM-алгоритмом. Фактически, это та же последовательная оптимизация, но повторяющаяся итерационно. «Настоящая» глобальная оптимизация всех базовых алгоритмов является тяжёлой многоэкстремальной задачей и требует знания их внутреннего устройства, что затрудняет применение стандартных методов обучения.

– Независимая оптимизация. Базовые алгоритмы настраиваются независимо друг от друга. Чтобы они не получались слишком похожими, настройка производится по различным частям обучающей выборки, либо по различным частям признакового описания, либо при различных начальных приближениях. Типичным представителем этого подхода является баггинг.

– Алгебраический подход, описанный в теоретических работах Журавлёва и его учеников, позволяет строить корректные алгоритмические композиции чисто алгебраическими методами, не прибегая к оптимизации. Этот подход крайне продуктивен при исследовании вопросов полноты моделей алгоритмов вида (1.1). Однако он плохо приспособлен для практического построения алгоритмов, так как не позволяет управлять сложностью композиции, и склонен к переобучению. Более практичные схемы, разработанные в рамках алгебраического подхода, используют упомянутые выше стратегии оптимизации.

1.3 Прикладная задача прогнозирования потребительского спроса

В настоящей работе рассматривается задача прогнозирования потребительского спроса. Она заключается в том, чтобы предсказать объем продаж товаров некоторой торговой компании и на основе прогноза правильно спланировать объемы и сроки поставок.

Решения о сроках и объемах закупок принимаются на основе:

– прогнозов потребительского спроса;

– календаря поставок;

– возможностей поставщика;

– маркетинговой политики.

Основным фактором в принятии решения является прогноз.

В качестве объекта исследования в настоящей задаче будем рассматривать временные ряды продаж товаров. Требуется на основе исторических данных о продажах построить прогноз об объемах продаж в будущие периоды.

Задача прогнозирования потребительского спроса обладает специфическими особенностями. Одной из основных особенностей является большое количество временных рядов, равное произведению количества магазинов на число товаров в них (для некоторых предприятий порядок может быть близок к 10^5). Следующей особенностью является высокое отношение шум/сигнал и существует гетероскедастичность (то есть непостоянство дисперсии), что затрудняет применение стандартных методов прогнозирования.

Одним из следствий этих особенностей является практическая невозможность моделировать временные ряды в рамках какой-либо одной модели алгоритмов, что и приводит к необходимости построения алгоритмической композиции. Также следствием является необходимость эффективной перенастройки как базовых алгоритмов, так и корректирующей операции в каждый момент времени.

В настоящей работе будем предполагать, что каждый алгоритм B_i и корректирующая операция F дают в качестве результата прогноз потребительского спроса за прогнозируемый период в будущем. Решающее правило C используется только в задачах классификации для преобразования оценки в решение. Как правило, в задачах регрессии и прогнозирования C не используется, и алгоритм A строят в виде: $A = F\left(B_1, \ldots, B_p\right)$.

1.4 Постановка задачи

1) Разработать эффективные алгоритмы настройки линейной корректирующей операции вида $F(t) = \dfrac{\sum w_t^i \cdot B_i(t)}{\sum w_t^i}$, где w_t^i – веса базовых алгоритмов.
2) Реализовать алгоритмы в среде MATLAB и провести вычислительные эксперименты на реальных данных потребительского спроса.
3) Провести сравнительный анализ алгоритмов на реальных данных по критериям точности и времени работы.

2 Стандартные методы прогнозирования временных рядов

Выбор метода построения прогноза по временному ряду зависит от стоящей задачи и, как правило, определятся тем, какой прогноз требуется получить - долговременный или кратковременный. Построение долговременного прогноза является (как это и можно ожидать) задачей значительно более сложной, не всегда имеющей решение (причем не только по техническим причинам - в ряде случаев существует так называемый «горизонт прогноза», то есть максимальное время, на которое принципиально возможно дать прогноз поведения) и обычно требующей привлечения дополнительной информации о системе.

2.1 Статистические методы

Статистические методы - важный класс методов, с помощью которых описываются явления, в которых присутствуют стохастические факторы, не позволяющие объяснить явление в чисто детерминистских терминах. Типичные примеры такого рода моделей представляют временные ряды в экономике и финансовой сфере, имеющие тренд-циклическую компоненту и случайную составляющую.

2.1.1 Прогнозирование многомерных временных рядов

Всякий эконометрический анализ основывается на исходных статистических данных. При этом если процесс регистрации исходных статистических данных происходит во времени t и само время фиксируется наряду со значениями анализируемых характеристик $x_i^j(t_k)$ (где переменная j пробегает значения $j = 1, \ldots, p$, номер объекта i пробегает значения $i = 1, \ldots, n$ и $k = 1, 2, \ldots, N$), то говорят о статистическом анализе так называемых панельных данных. Если зафиксировать номер переменной j и номер статистически обследуемого объекта i, то расположенную в хронологическом порядке последовательность значений

$$x_j^i(t_1),\ x_j^i(t_2), \ldots,\ x_j^i(t_k), \qquad (2.1)$$

называют одномерным временным рядом. Если же одновременно рассматривать p одномерных временных рядов вида (2.1), т.е. исследовать закономерности во взаимосвязанном поведении временных рядов (2.1) для $j = 1, 2, \ldots, p$, характеризующих динамику p переменных, измеренных на каком-то одном объекте, то тогда говорят о статистическом анализе многомерного временного ряда $X(t) = x_1(t_k), x_2(t_k), \ldots, x_p(t_k)$, $k = 1, 2, \ldots, N$. Проблема прогнозирования заключается в построении кратко-, средне- и долгосрочных прогнозов. Тем не менее, использование доступных к моменту времени $t = N$ наблюдений временного ряда для прогнозирования значения $x(t)$ на один или несколько временных тактов вперед (т.е. для прогнозной оценки значений $x(t_{N+l})$, $l = 1, 2, \ldots$) может явиться основой для:

- планирования в экономике, производстве, торговле;
- управления и оптимизации, протекающих в обществе социально-экономических процессов;
- частичного управления важными параметрами демографических процессов и экологической ниши общества;
- принятия оптимальных решений в бизнесе.

Принципиальные отличия временного ряда от случайной выборки состоят в том, что члены временного ряда не являются статистически независимыми и не являются одинаково распределенными, т.е. $P\{x(t_1) < x\} \neq P\{x(t_2) < x\}$ при $t_1 \neq t_2$.

Это значит, что нельзя распространять свойства и правила статистического анализа случайной выборки на временные ряды. С другой стороны, взаимозависимость членов временного ряда создает свою специфическую базу для построения прогнозных значений анализируемого показателя (т.е. для построения оценок $\tilde{x}(t_{N+k})$ для неизвестных значений $x(t_{N+k})$) по наблюденным значениям $x(t_1), x(t_2), \ldots, x(t_N)$.

Каждый временной ряд формируется под воздействием большого числа факторов, которые условно можно подразделить на три группы:

- факторы, формирующие тенденцию ряда, которая описывается с помощью той или иной неслучайной функции $f_{mp}(t)$, как правило, монотонной. Эту функцию называют функцией тренда или просто – трендом;
- факторы, формирующие сезонные колебания ряда (повторяющиеся с определенной периодичностью (год, неделя, сутки и т.п.) колебания анализируемого признака), результат действия сезонных факторов обозначается с помощью неслучайной функции $\varphi(t)$;
- факторы, формирующие циклические колебания ряда (изменения анализируемого признака, обусловленные действием долговременных циклов экономической, демографической или астрофизической природы). Результат действия циклических факторов будем обозначать с помощью неслучайной функции $\psi(t)$;
- случайные факторы.

Обычно в процессе участвуют не все факторы одновременно и никогда сезонные и циклические колебания вместе, так что можно их объединить. Участие случайных факторов должно учитываться всегда, так как они обуславливают стохастическую природу элементов ряда.

Каждый член ряда можно представить следующим образом:

$$x(t) = \chi(A) \cdot f_{mp}(t) + \chi(B) \cdot \varphi(t) + \chi(C) \cdot \psi(t) + \varepsilon(t)$$

где

$$\chi(D) = \begin{cases} 1, & \textit{если факторы типа } D \textit{ участвуют в формировании значений } x(t); \\ 0, & \textit{в противном случае}; \end{cases}$$

$D = A, B$ *или* C.

Выводы о том, участвуют или нет факторы данного типа в формировании значений $x(t)$, могут базироваться как на анализе содержательной сущности задачи (т.е. быть априорно-экспертными по своей природе), так и на специальном статистическом анализе исследуемого временного ряда.

Временные ряды можно разделить на стационарные и нестационарные. Их свойства существенно отличаются, и для моделирования рядов должны применяться различные методы.

Ряд y_t называется строго стационарным, если совместное распределение m наблюдений $y_{t_1}, y_{t_2}, \ldots, y_{t_m}$ не зависит от сдвига по времени, то есть совпадает с распределением $y_{t_1+t}, y_{t_2+t}, \ldots, y_{t_m+t}$ для любых $m, t, t_1, \ldots, t_m$.

Ряд y_t называется слабо стационарным, если его среднее, ковариация и дисперсия не зависят от момента времени t:

$$E(y_t) = \mu < \infty, \qquad V(y_t) = \gamma_0, \qquad Cov(y_t, y_{t-k}) = \gamma_k.$$

В дальнейшем под стационарностью понимается именно слабая стационарность.

2.1.1.1 Тренд

Тренд – это изменение, определяющее общее направление развития, основную тенденцию временных рядов.

Трендом называют конкретное, в форме определенной монотонной кривой описание тенденции развития.

Рассмотрим следующий временной ряд:

$$y_t = \alpha + \beta \cdot t + \varepsilon_t.$$

Он представляет собой сумму заданной составляющей $\alpha + \beta \cdot t$ (линейный тренд) и случайной составляющей ε_t, которая является стационарным временным рядом с нулевым средним. Также можно встретить квадратичный тренд: $y_t = \alpha + \beta \cdot t + \gamma t^2$, экспоненциальный: $y_t = \alpha \cdot e^{\beta \cdot t}$ и другие.

Для того чтобы выделить тренд в модели $y_t = \alpha + \beta \cdot t + \varepsilon_t$, можно применить обычную технику оценивания параметров регрессионных уравнений, считая t независимой переменной. После этого получается ряд остатков, для описания которого можно применить модели стационарных временных рядов.

2.1.1.2 Сезонность

Временные ряды могут учитывать сезонную компоненту. Например, в квартальных данных может наблюдаться сезонная компонента с периодом n:

$$y_t = S(t) + \varepsilon_t, \qquad S(t+n) \equiv S(t).$$

Здесь ряд y_t представлен в виде композиции периодической детерминированной составляющей $S(t)$ (сезонная компонента) и случайной составляющей ε_t, которая является стационарным временным рядом с нулевым средним. Сезонную компоненту $S(t)$ можно представить в виде $S(t) = \beta_1 d_{1t} + \beta_2 d_{2t} + \ldots + \beta_n d_{nt}$, где d_i – фиктивные (бинарные) переменные для кварталов. Для выделения сезонной компоненты мы можем применить методы оценивания параметров регрессий к уравнению:

$$y_t = \beta_1 d_{1t} + \beta_2 d_{2t} + \ldots + \beta_n d_{nt} + \varepsilon_t.$$

Как и в случае выделения тренда, методы моделирования стационарных временных рядов применяются далее к ряду остатков регрессии $y_t = \beta_1 d_{1t} + \beta_2 d_{2t} + \ldots + \beta_n d_{nt} + \varepsilon_t$.

2.1.1.3 Взятие последовательной разности

Рассмотрим процесс, называемый случайным блужданием (random walk):

$$y_t = y_{t-1} + \varepsilon_t, \qquad \varepsilon_t \approx iid\left(0, \sigma^2\right), \qquad t = 1, \ldots, n.$$

Он является нестационарным процессом, так как $V\left(y_t\right) \neq V\left(y_{t-1}\right)$. Покажем это: ошибка ε_t некоррелирована с y_{t-1}, значит, можем получить:

$$E\left(y_t\right) = E\left(y_{t-1}\right) + 0, \qquad V\left(y_t\right) = V\left(y_{t-1}\right) + \sigma^2$$

Для дальнейших рассуждений удобно применять оператор сдвига (lag operator) $Lx_t = x_{t-1}$ и полиномы от оператора сдвига $A\left(L\right) = 1 \pm \upsilon_1 L \pm \ldots \pm \upsilon_q L$.

Случайное блуждание можно привести к стационарному временному ряду при помощи операции взятия последовательной разности:

$$z_t = \Delta y_t = \left(1 - L\right) y_t = y_t - y_{t-1}, \qquad z_t = \varepsilon_t$$

Обобщим: для нестационарного процесса $A\left(L\right) y_t = \varepsilon_t$ это же преобразование $z_t = \Delta y_t$ приводит к стационарному процессу $B\left(L\right) z_t = \varepsilon_t$, причем $A\left(L\right)$ должен иметь один единичный корень, т.е. $A\left(L\right) = B\left(L\right)\left(1 - L\right)$, а все корни $B\left(L\right)$ должны лежать вне единичного круга.

Взятие разности также приводит к стационарному процессу ряд $y_t = \alpha + \beta \cdot t + \varepsilon_t$ с линейным трендом:

$$\Delta y_t = \beta + u_t, \qquad u_t = \Delta \varepsilon_t = \varepsilon_t - \varepsilon_{t-1}$$

Для получения стационарного процесса из временного ряда с квадратичным трендом $y_t = \alpha + \beta t + \gamma t^2$ нужно взять вторую разность $\Delta^2 y_t = \Delta\left(\Delta y_t\right) = \left(y_t - y_{t-1}\right) - \left(y_{t-1} - y_{t-2}\right)$, то

$$\Delta y_t = \beta + \gamma \cdot \left(2t - 1\right) + \Delta \varepsilon_t,$$

$$\Delta^2 y_t = 2\gamma + \Delta^2 \varepsilon_t.$$

Тогда получившийся временной ряд $\Delta^2 y_t$ уже является стационарным.

Если нестационарный временной ряд обладает сезонностью, то его также можно привести к стационарному ряду. Сезонная компонента в $y_t = S\left(t\right) + \varepsilon_t$, $S\left(t + n\right) \equiv S\left(t\right)$ удаляется при помощи оператора взятия сезонной последовательной разности $\Delta_n y_t = \left(1 - L^n\right) y_t = y_t - y_{t-n}$.

Но применение оператора последовательной разности не обязательно приводит нестационарный ряд к стационарному. Рассмотрим процесс

$$y_t = \beta \cdot y_{t-1} + \varepsilon_t, \qquad \beta > 1,$$

который не является стационарным. Дисперсия от обеих частей имеет вид $V\left(y_t\right) = \beta^2 \cdot V\left(y_{t-1}\right) + \sigma_\varepsilon^2$. Получается, что процесс может быть стационарным лишь при отрицательной дисперсии $V\left(y_t\right) = \dfrac{\sigma_\varepsilon^2}{1 - \beta^2} < 0$. Это означает, что процесс нестационарен Применим к этому ряду оператор разности и получим

$$\Delta y_t = \beta \cdot \Delta y_{t-1} + \Delta \varepsilon_t, \qquad \beta > 1,$$

то есть процесс по–прежнему остался нестационарным.

Таким образом, применяя выделение тренда, сезонности и/или оператор последовательной (и сезонной) разности, часто можно получить из исходного временного ряда стационарный.

2.1.1.4 Проверка на стационарность

Определить, является ли ряд стационарным, можно несколькими способами.

Самым простым способом является построение графика полученных наблюдений. В нем можно обнаружить тренд или периодичную компоненту (сезонность).

Также можно построить график выборочной автокорреляционной функцией ACF (autocorrelation function) или коррелограммы (correlogram):

$$\rho_k = \frac{Cov(y_t, y_{t-k})}{V(y_t)} = \frac{\gamma_k}{\gamma_0} \qquad \text{(ACF)}$$

$$r_k = \hat{\rho}_k = \frac{\sum_{t=k+1}^{n}(y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum_{t=1}^{n}(y_t - \bar{y})^2}, \quad k = 1, 2, \ldots,$$

где $\bar{y}$ – среднее, $\bar{y} = \sum_{i=1}^{n} y_i$. Коррелограмма стационарного временного ряда «быстро убывает» с ростом k после нескольких первых значений. Если же график убывает достаточно медленно, то есть основания предположить нестационарность ряда.

Кроме этого можно построить график частной автокорреляционной функцией PACF. Частная автокорреляционная функция PACF(k) (partial autocorrelation function) есть «чистая корреляция» между y_t и y_{t-k} при исключении влияния промежуточных значений $y_{t-1}, y_{t-2}, \ldots, y_{t-k+1}$. Она также должна быстро убывать для стационарного процесса.

Также можно использовать формальные тесты на наличие единичного корня (тест Дики–Фуллера DF, расширенный тест Дики–Фуллера ADF, тест МакКинли).

2.1.2 Модели авторегрессии и скользящего среднего ARMA

Рассмотрим класс моделей стационарных временных рядов:

$$y_t - \phi_1 y_{t-1} - \ldots - \phi_p y_{t-p} = \delta + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \ldots - \theta_p \varepsilon_{t-p}, \qquad \varepsilon_t \approx iid(0, \sigma^2) \qquad (2.2)$$

или в более короткой записи

$$\Phi(L) \cdot y_t = \delta + \Theta(L) \cdot \varepsilon_t, \qquad \varepsilon_t \approx iid(0, \sigma^2) \qquad (2.3)$$

где $\Phi(L) = 1 - \phi_1 L - \ldots - \phi_p L^p$ и $\Theta(L) = 1 - \theta_1 L - \ldots - \theta_p L^p$ – полиномы от оператора сдвига. Такая модель называется моделью авторегрессии и скользящего среднего (autoregressive moving average) или $ARMA(p, q)$.

Рассмотрим сначала простые примеры $ARMA$ моделей.

2.1.2.1 Процесс AR

Процесс ARMA (1,0)

$$y_t = \delta + \phi_1 \cdot y_{t-1} + \varepsilon_t \qquad \varepsilon_t \approx iid\left(0, \sigma^2\right)$$

является $AR(1)$ процессом. Рассмотрим его подробнее. Пусть $|\phi_1| < 1$. Перепишем его в виде:

$$\left(1 - \phi_1 L\right) \cdot y_t = \delta + \varepsilon_t$$

или

$$y_t = \left(1 - \phi_1 L\right)^{-1} \cdot \left(\delta + \varepsilon_t\right) = \left(1 + \phi_1 L + \phi_1^2 L^2 + \ldots\right) \cdot \left(\delta + \varepsilon_t\right) =$$
$$= \left(1 + \phi_1 + \phi_1^2 + \ldots\right) \cdot \delta + \varepsilon_t + \phi_1 \cdot \varepsilon_{t-1} + \phi_1^2 \cdot \varepsilon_{t-2} + \ldots .$$

Так как $|\phi_1| < 1$, то получаем, что

$$E\left(y_t\right) = \frac{\delta}{1 - \phi_1} \qquad V\left(y_t\right) = \gamma_0 = \frac{\sigma^2}{1 - \phi_1^2} \qquad \gamma_k = \phi_1^k \cdot \gamma_0 \qquad \rho_k = \frac{\gamma_k}{\gamma_0} = \phi_1^k \qquad (2.4)$$

Неравенство $|\phi_1| < 1$ является необходимым условием стационарного процесса y_t. Частная автокорреляционная функция процесса $AR(1)$ равна нулю для значений $k > 1$.

Возьмем в качестве примера авторегрессионного процесса высокого порядка процесс $AR(2)$ (для простоты положим свободный член равным нулю):

$$y_t = \phi_1 \cdot y_{t-1} + \phi_2 \cdot y_{t-2} + \varepsilon_t, \qquad \varepsilon_t \approx iid\left(0, \sigma^2\right). \qquad (2.5)$$

Для $k > 0$ вычислим ковариацию обеих частей (2.5) с y_{t-k}:

$$\gamma_k = Cov\left(y_t, y_{t-k}\right) = Cov\left(\phi_1 \cdot y_{t-1} + \phi_2 \cdot y_{t-2} + \varepsilon_t, y_{t-k}\right) = \phi_1 \cdot \gamma_{k-1} + \phi_2 \cdot \gamma_{k-2}, \qquad (2.6)$$

разделив на дисперсию γ_0, получим

$$\rho_k = \phi_1 \cdot \rho_{k-1} + \phi_2 \cdot \rho_{k-2}, \qquad k = 1, 2, \ldots. \qquad (2.7)$$

Взяв (2.7) при $k = 1{,}2$ и учитывая, что $\rho_0 = 1$, $\rho_{-1} = \rho_1$, получаем систему уравнений с неизвестными ρ_1 и ρ_2:

$$\begin{cases} \rho_1 = \phi_1 + \phi_2 \cdot \rho_1 \\ \rho_2 = \phi_1 \cdot \rho_1 + \phi_2 \end{cases}. \qquad (2.8)$$

Система (2.8) называется системой уравнений Юла–Уолкера (Yule–Walker) для $AR(2)$ процесса. Решая эту систему, найдем два первых значения автокорреляционной функции

$$\begin{cases} \rho_1 = \dfrac{\phi_1}{1 - \phi_2} \\ \rho_2 = \dfrac{\phi_1^2}{1 - \phi_2} + \phi_2 \end{cases}.$$

Следующие значения автокорреляционной функции вычисляются по формуле $\rho_k = \phi_1 \cdot \rho_{k-1} + \phi_2 \cdot \rho_{k-2}$ (2.7). Если умножить обе части (2.5) на y_t и взять математическое ожидание, получим следующее выражение для дисперсии y_t:

$$\gamma_0 = \phi_1 \cdot \gamma_1 + \phi_2 \cdot \gamma_2 + \sigma^2 .$$

Решая это уравнение совместно с двумя уравнениями (2.6) для $k = 1,2$, получаем:

$$\gamma_0 = \frac{(1-\phi_2)\cdot\sigma^2}{(1+\phi_2)\cdot(1-\phi_1-\phi_2)\cdot(1+\phi_1-\phi_2)}.$$

Отсюда, учитывая, что дисперсия должна быть положительна, получаем условия стационарности $AR(2)$ процесса:

$$|\phi_2| < 1,\ \phi_2 + \phi_1 < 1,\ \phi_2 - \phi_1 < 1.$$

В случае, когда корни характеристического полинома $\Phi(L) = 1 - \phi_1 L - \phi_2 L^2$ являются действительными, автокорреляционная функция процесса убывает экспоненциально, когда корни комплексные – изменяется по синусоиде с экспоненциально убывающей амплитудой.

Опишем схему вычисления частной автокорреляционной функции для процесса $AR(2)$. Запишем три уравнения Юла–Уолкера (типа (2.8)) для $AR(3)$ процесса. Коэффициент ϕ_3 равен коэффициенту частной корреляции между y_t и y_{t-3}. Для $AR(2)$ процесса из (2.7) получаем $\rho_3 = \phi_1 \cdot \rho_2 + \phi_2 \cdot \rho_1$. Подставляя это выражение в третье уравнение Юла–Уолкера, получаем $\phi_3 = 0$. Таким образом, PACF(k)=0 для $k > 2$.

Аналогично, можно показать, что для $AR(p)$ процесса частная автокорреляционная функция PACF(k) равна нулю, начиная с $k = p+1$. Следует иметь в виду, что этот результат верен для теоретической частной автокорреляционной функции и может не выполняться для выборочной частной автокорреляционной функции. Однако на практике следует ожидать резкое убывание PACF до значений, близких к нулю, за порогом, равным порядку авторегрессионного процесса.

2.1.2.2 Процессы скользящего среднего (MA)

Моделью скользящего среднего (moving average) порядка q называется модель $ARMA(0,q)$

$$y_t = \delta + \Theta(L)\varepsilon_t = \delta + \varepsilon_t - \theta_1 \cdot \varepsilon_{t-1} - \ldots - \theta_q \cdot \varepsilon_{t-q}, \qquad \varepsilon_t \approx iid(0,\sigma^2),$$

которая обозначается $MA(q)$. Очевидно, что процесс $MA(q)$ стационарен при любом q и любых θ_i.

Сформулируем условие обратимости процесса, т. е. возможности его представления в виде AR процесса.

Рассмотрим в качестве примера модель скользящего среднего первого порядка $MA(1)$

$$y_t = \delta + \Theta(L)\varepsilon_t = \delta + \varepsilon_t - \theta_1 \cdot \varepsilon_{t-1}, \qquad \varepsilon_t \approx iid(0,\sigma^2), \qquad (2.9)$$

Представим $MA(1)$ процесс в виде авторегрессионного процесса:

$$\Theta(L)^{-1} y_t = \Theta(L)^{-1}\delta + \varepsilon_t,$$

или

$$y_t = \frac{\delta}{1-\theta_1} - \theta_1 \cdot y_{t-1} - \theta_1^2 \cdot y_{t-2} - \ldots + \varepsilon_t, \qquad (2.10)$$

Ясно, что такое $AR(\infty)$ представление $MA(1)$ процесса (2.9) возможно только в случае обратимости оператора $\Theta(L)=1-\theta_1 L$, т.е. когда выполняется условие обратимости $|\theta_1|<1$.

Вычислим среднее, дисперсию и автокорреляционную функцию процесса $MA(1)$:

$$E(y_t)=\delta, \qquad V(y_t)=\sigma^2\cdot(1+\theta_1^2), \qquad \gamma_1=Cov(y_t,y_{t-1})=E((\varepsilon_t-\theta_1\varepsilon_{t-1})\cdot(\varepsilon_{t-1}-\theta_1\varepsilon_{t-2})).$$

Если раскрыть скобки, то только одно слагаемое из четырех будет отлично от нуля: $E(-\theta_1\cdot\varepsilon_{t-1}^2)=1-\theta_1\cdot\sigma^2$. Поэтому

$$\gamma_1=Cov(y_t,y_{t-1})=-\theta_1\cdot\sigma^2.$$

Аналогичные вычисления показывают, что $\gamma_k=0$ при $k>1$. Получаем:

$$\rho_1=\frac{\gamma_1}{\gamma_0}=-\frac{\theta_1}{1+\theta_1^2}, \qquad \rho_k=0, \qquad k>1.$$

Проделав аналогичные вычисления для $MA(q)$ процесса, получим, что его автокорреляционная функция ACF(k) равна 0 для $k>q$, т.е. ее вид аналогичен виду PACF для $AR(q)$ процесса.

Частная автокорреляционная функция PACF(k) для $MA(q)$ процесса, аналогично ACF(k) для $AR(q)$ процесса, экспоненциально убывает. Таким образом, имеет место некоторая симметрия: пара графиков (ACF, PACF) для $MA(q)$ процесса имеет такой же вид, как пара графиков (PACF, ACF) для $AR(q)$.

Отметим, что подобно $AR(\infty)$ представлению $y_t=\frac{\delta}{1-\theta_1}-\theta_1\cdot y_{t-1}-\theta_1^2\cdot y_{t-2}-\ldots+\varepsilon_t$ (2.10) для $MA(1)$ процесса $y_t=\delta+\Theta(L)\varepsilon_t=\delta+\varepsilon_t-\theta_1\cdot\varepsilon_{t-1}$ (2.9) существует $MA(\infty)$ представление для $AR(1)$ процесса $y_t=\delta+\phi_1\cdot y_{t-1}+\varepsilon_t$:

$$y_t=(1-\phi_1 L)^{-1}\cdot(\delta+\varepsilon_t)=\frac{\delta}{1-\phi_1}+\varepsilon_t+\phi_1\cdot\varepsilon_{t-1}+\phi_1^2\cdot\varepsilon_{t-2}+\ldots.$$

2.1.2.3 Смешанные процессы

Рассмотрим простейший смешанный $ARMA(1,1)$ процесс (2.3) с $\Phi(L)=1-\phi_1 L$ и $\Theta(L)=1-\theta_1 L$:

$$y_t-\phi_1\cdot y_{t-1}=\delta+\varepsilon_t-\theta_1\cdot\varepsilon_{t-1}, \qquad \varepsilon_t\approx iid(0,\sigma^2). \qquad (2.11)$$

Будем считать, что $|\phi_1|<1$ и $|\theta_1|<1$. Как и в случае $AR(1)$ и $MA(1)$ процессов, можно показать, что тогда процесс $ARMA(1,1)$ является стационарным и обратимым.

Применяя те же методы, что и ранее, получим следующие выражения для среднего, дисперсии и ковариации $ARMA(1,1)$ процесса:

$$E(y_t)=\frac{\delta}{1-\phi_1}, \qquad \gamma_0=V(y_t)=\sigma^2\frac{1+\theta_1^2-2\phi_1\theta_1}{1-\phi_1^2}, \qquad \gamma_1=Cov(y_t,y_{t-1})=\phi_1\gamma_0-\theta_1\sigma^2$$

(2.12 а,б,в)

Для автокорреляций порядка больше 1 получаем рекуррентное соотношение

$$\gamma_k = Cov(y_t, y_{t-1}) = \phi_1 \cdot \gamma_{k-1},\ k > 1.$$

Применяя рекуррентно это соотношение, получаем:

$$\rho_k = \phi_1^{k-1} \cdot \rho_1, \qquad k > 1$$

$$\rho_1 = \frac{(1 - \phi_1 \cdot \theta_1) \cdot (\phi_1 - \theta_1)}{1 + \theta_1^2 - 2 \cdot \phi_1 \cdot \theta_1}.$$

Из этих равенств видно, что ACF для $ARMA(1,1)$ процесса ведет себя так же, как ACF для $AR(1)$ процесса (ср.(2.4)). Хотя значение ρ_1 другое, но соотношение между ρ_1 и последующими значениями ACF точно такое же.

Этот вывод можно обобщить на случай $ARMA(p,q)$ процесса. Первые q значений ACF определяются взаимодействием AR и MA компонент, а дальнейшее поведение автокорреляционной функции такое же, как в $AR(p)$ процессе.

Аналогичный вывод справедлив для частичной автокорреляционной функции $ARMA(p,q)$ процесса. Она убывает подобно PACF для $MA(q)$ процесса.

2.1.3 Интегрированные модели авторегрессии и скользящего среднего ARIMA

Предположим, что временной ряд y_t после того, как к нему применили d раз оператор последовательной разности, стал стационарным рядом $\Delta^d y_t$ удовлетворяющим $ARMA(p,q)$ модели

$$y_t - \phi_1 \cdot y_{t-1} - \ldots - \phi_p \cdot y_{t-p} = \delta + \varepsilon_t - \theta_1 \cdot \varepsilon_{t-1} - \ldots - \theta_p \cdot \varepsilon_{t-p}, \qquad \varepsilon_t \approx iid(0, \sigma^2).$$

Тогда процесс y_t называется интегрированным процессом авторегрессии и скользящего среднего (integrated autoregression and moving average), $ARIMA(p,q,d)$.

2.1.3.1 Методология Бокса–Дженкинса

Методология Бокса–Дженкинса подбора $ARIMA$ модели для данного ряда наблюдений состоит из трех этапов.

I. Идентификация модели

I.1. Первый шаг – получение стационарного ряда. Мы тестируем ряд на стационарность, используя описанные выше методы: визуальный анализ графика, визуальный анализ ACF и PACF, тесты на единичные корни. Если получается стационарный ряд, то переходим к следующему пункту, если нет, то применяем оператор взятия последовательной разности и повторяем тестирование. На практике последовательная разность берется, как правило, не более двух раз.

I.2. После того как получен стационарный временной ряд, строятся его выборочные ACF и PACF, которые, как было показано выше, являются своеобразными «отпечатками пальцев» $ARMA(p,q)$ процесса и позволяют сформулировать несколько гипотез о возможных порядках авторегрессии (p) и скользящего среднего (q). Обычно рекомендуется использовать модели возможно более низкого порядка, как правило, с $p + q \le 3$ (если нет сезонной компоненты).

Выборочные ACF и PACF, конечно, не обязаны в точности следовать теоретическим аналогам, но должны быть «достаточно близки» к ним.

II. Оценивание модели и проверка адекватности модели

II.1. Для каждой из выбранных на первом этапе моделей оцениваются их параметры и вычисляются остатки.

II.2. Каждая из моделей проверяется, насколько она соответствует данным. Из моделей, адекватных данным, выбирается самая простая модель, т. е. модель с наименьшим количеством параметров.

III. Прогнозирование

После того как на втором этапе выбрана модель, можно строить прогноз на один или несколько шагов по времени и оценивать доверительные границы прогнозных значений.

Остановимся подробнее на втором и третьем этапах методики Бокса–Дженкинса.

2.1.3.2 Оценивание ARMA моделей

Рассмотрим пример $ARMA(1,1)$ модели (2.11). Запишем ее в виде:

$$\Theta(L)^{-1} y_t = \Theta(L)^{-1}(\delta + \phi_1 \cdot y_{t-1}) + \varepsilon_t, \qquad (2.13)$$

где $\Theta(L) = 1 - \theta_1 L$ и $\Theta(L)^{-1} = 1 + \theta_1 L + \theta_1 L^2 + \ldots$. В (2.13) надо каким–то образом интерпретировать переменную $y_t^* = \Theta(L)^{-1} y_t$, которая является бесконечной взвешенной суммой предыдущих значений y_t. Одним из возможных решений является следующее. Приравняем нулю все значения, предшествующие началу наблюдений: $y_0 = y_{-1} = \ldots = 0$. При этом получим:

$$y_1^* = y_1,\ y_2^* = y_2 + \theta_1 \cdot y_1,\ \ldots\ y_t^* = y_t + \theta_1 \cdot y_{t-1} + \ldots + \theta_1^{t-1} \cdot y_1.$$

В этих обозначениях уравнение (2.13) принимает вид

$$y_t^* = \delta^* + \phi_1 \cdot y_{t-1}^* + \varepsilon_t, \qquad \delta^* = \frac{\delta}{1-\theta_1}.$$

В том случае, если θ_1 известно, это уравнение является линейным по δ^*, ϕ_1, однако в общем случае оно нелинейно по параметрам.

Для оценивания уравнения (2.13) применим условный метод максимального правдоподобия (conditional ML), когда y_1 предполагается заданным, считая, что ошибки $\varepsilon_t \approx iidN(0, \sigma^2)$. Условная функция правдоподобия равна

$$L^* = p(y_2^*, y_2^*, \ldots, y_n^* \mid y_1^*) = \prod_{t=2}^{n} p(y_t^* \mid y_{t-1}^*) =$$
$$= \prod_{t=2}^{n} \frac{1}{\sqrt{2\pi} \cdot \sigma} \exp\left(-\frac{1}{2\sigma^2} (y_t^* - \delta^* - \phi_1 \cdot y_{t-1}^*)^2 \right).$$

Логарифм условной функции максимального правдоподобия равен

$$l^* = \ln L^* = const - \frac{n-1}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} \sum_{t=2}^{n} (y_t^* - \delta^* - \phi_1 \cdot y_{t-1}^*)^2.$$

Из вида функции l^* видно, что оценка коэффициентов δ, ϕ_1 по условному методу максимального правдоподобия совпадает с оценкой нелинейного метода наименьших квадратов. (Заметим, что сумма в правой части равенства является нелинейной функцией параметров δ, ϕ_1.)

Полный метод максимального правдоподобия (full ML) состоит в максимизации функции правдоподобия

$$L = p(y_1)\cdot L^*.$$

Известно, что при гипотезе нормальности ошибок $y_1^* \approx N\left(\frac{\delta^*}{1-\phi_1}, \frac{\sigma^2}{1-\phi_1^2}\right)$. Поэтому логарифм функции правдоподобия равен

$$l(\delta,\phi_1,\theta,\sigma^2) = \ln p(y_1) + l^* =$$

$$= const - \frac{n}{2}\ln\sigma^2 - \frac{1-\phi_1^2}{2\sigma^2}\left(y_1^* - \frac{\delta^*}{1-\phi_1}\right)^2 - \frac{1}{2\sigma^2}\sum_{t=2}^{n}\left(y_t^* - \delta^* - \phi_1\cdot y_{t-1}^*\right)^2$$

2.1.3.3 Проверка адекватности ARMA моделей

Есть несколько критериев оценки того, насколько $ARMA$ модель, которую мы оцениваем, соответствует нашим данным.

Во–первых, оценки коэффициентов модели должны статистически достоверно отличаться от нуля, т. е. соответствующие P–значения t–статистик должны быть меньше выбранного порогового значения.

Во–вторых, согласно модели, ошибки ε_t являются белым шумом. Соответственно их оценки, т.е. остатки регрессии e_t, должны быть также похожи на белый шум. Поэтому остатки должны иметь нулевую автокорреляцию.

В модели, включающей константу, среднее остатков равно 0. Поэтому выборочная автокорреляционная функция остатков вычисляется по формуле:

$$r_k = \frac{\sum_{t=k+1}^{n} e_t \cdot e_{t-k}}{\sum_{t=1}^{n} e_t^2}.$$

Если модель адекватна данным, ошибки являются белым шумом, и при больших значениях n и k величина r_k, имеет распределение, близкое к нормальному $N\left(0,\frac{1}{n}\right)$.

Другие тесты проверяют гипотезу равенства нулю сразу K первых значений автокорреляционной функции остатков.

Q–статистика Бокса–Пирса определяется как

$$Q = n\cdot\sum_{k=1}^{K} r_k^2 .$$

При нулевой гипотезе отсутствия автокорреляции Q имеет распределение $\chi^2(K-p-q)$, где p, q – параметры $ARMA$ модели. Нулевая гипотеза отвергается, если полученное значение Q больше соответствующего критического значения.

Тест Льюпга–Бокса является модификацией теста Бокса–Пирса. Соответствующая статистика

$$\widetilde{Q} = n\cdot(n+2)\cdot\sum_{k=1}^{K}\frac{r_k^2}{n-k}$$

имеет такое же асимптотическое распределение, как и Q, однако ее распределение ближе к χ^2 для конечных выборок.

Если тесты показывают наличие автокорреляции остатков, это означает, что рассматриваемая $ARMA$ модель не подходит, и ее надо модифицировать. Например, если в автокорреляционной функции отличны от нуля значения с номерами, кратными n, то стоит попробовать ввести сезонную авторегрессию n–го порядка. Если единственное отличающееся от нуля значение соответствует лагу, равному n, можно попробовать ввести сезонный MA–член порядка n.

Если возникает ситуация, когда несколько $ARMA$ моделей оказываются адекватными данным, то, руководствуясь принципом «экономии мышления», следует выбрать модель с наименьшим количеством параметров.

В компьютерных пакетах среди результатов оценивания приводится информационный критерий Акаике AIC (Akaike information criterion), определяемый формулой

$$AIC = 2\frac{p+q}{n} + \ln\left(\frac{\sum_{t=1}^{n} e_t^2}{n}\right).$$

Критерий Акаике является эвристической попыткой свести в один показатель два требования: уменьшение числа параметров модели и качество подгонки модели. Согласно этому критерию, из двух моделей следует выбрать модель с меньшим значением AIC.

Обычно также приводится значение критерия Шварца (Schwarz criterion)

$$\frac{p+q}{n} + \ln\left(\frac{\sum_{t=1}^{n} e_t^2}{n}\right),$$

отличие которого от AIC состоит в большем штрафе за количество параметров.

2.1.3.4 Прогнозирование с ARIMA моделями

Главная цель использования $ARIMA$ моделей – построение прогноза за пределы выборки. Есть два источника неточности прогноза: первый – игнорирование будущих ошибок ε_t, второй – отклонение оценок коэффициентов модели от их истинных значений. Рассмотрим только первый источник ошибок прогноза или, другими словами, прогнозирование в рамках теоретических моделей.

Рассмотрим проблему прогнозирования на примере $ARMA(1,1)$ и $ARIMA(1,1,0)$ моделей.

2.1.3.4.1 ARMA (1,1) модель. Прогнозирование

Из (2.11) получаем значение y в момент $n+1$:

$$y_{n+1} = \delta + \phi_1 \cdot y_n + \varepsilon_{n+1} - \theta_1 \cdot \varepsilon_n .$$

Используя обозначение $\mu = E(y_t) = \dfrac{\delta}{1-\phi_1}$, получаем:

$$y_{n+1} - \mu = \phi_1 \cdot (y_n - \mu) + \varepsilon_{n+1} - \theta_1 \varepsilon_n . \qquad (2.14)$$

Прогноз на один шаг, минимизирующий среднеквадратичное отклонение, равен $\hat{y}_{n+1} = E(y_{n+1} \mid I_n)$, где I_n – информация, доступная в момент n. Из (2.14) получаем:

$$\hat{y}_{n+1} - \mu = \phi_1 \cdot (y_n - \mu) - \theta_1 \cdot \varepsilon_n . \qquad (2.15)$$

Ошибка прогноза и ее дисперсия равны

$$e_{n+1} = y_{n+1} - \hat{y}_{n+1} = \varepsilon_{n+1}, \qquad V(e_{n+1}) = \sigma^2 .$$

Используя две итерации уравнения (2.14), получаем

$$y_{n+2} - \mu = \phi_1^2 \cdot (y_n - \mu) + \varepsilon_{n+2} + (\phi_1 - \theta_1) \cdot \varepsilon_{n+1} - \phi_1 \cdot \theta_1 \cdot \varepsilon_n .$$

Отсюда аналогично (2.15) вычисляется прогноз на два шага:

$$y_{n+2} - \mu = \phi_1^2 \cdot (y_n - \mu) - \phi_1 \cdot \theta_1 \cdot \varepsilon_n$$

$$V(\varepsilon_{n+2}) = \sigma^2 \cdot \left(1 + (\phi_1 - \theta_1)^2\right).$$

Продолжая итерации, можно получить

$$y_{n+s} - \mu = \phi_1^s \cdot (y_n - \mu) - \phi_1^{s-1} \cdot \theta_1 \cdot \varepsilon_n$$

откуда видно, что прогноз стремится к среднему μ, когда горизонт прогноза возрастает. Можно показать, что

$$\lim_{s \to \infty} V(\varepsilon_{n+2}) = \sigma^2 \cdot \left(\frac{1 - 2 \cdot \phi_1 \cdot \theta_1 + \theta_1^2}{1 - \phi_1^2} \right).$$

Заметим, что это выражение совпадает с дисперсией ряда y, полученной в (2.12 б).

2.1.3.4.2 ARIMA (1,1,0) модель. Прогнозирование

Прогноз нестационарного временного ряда несколько отличается от выше разобранного случая. Рассмотрим временной ряд y_t первые разности которого z_t являются $AR(1)$ процессом ($y_t = \delta + \phi_1 \cdot y_{t-1} + \varepsilon_t$):

$$z_t = y_t - y_{t-1}, \qquad z_t - \mu = \phi_1 \cdot (z_{t-1} - \mu) + \varepsilon_t \qquad (2.16)$$

Многократное применение (2.16) дает

$$y_{n+s} = y_n + z_{n+1} + z_{n+2} + \ldots + z_{n+s} = (y_n + s \cdot \mu) + (z_{n+1} - \mu) + \ldots + (z_{n+s} - \mu) \qquad (2.17)$$

Подставляя $z_t - \mu$ из (2.16) в (2.17), получаем:

$$y_{n+s} = y_n + s \cdot \mu + \frac{\phi_1 \cdot (1 - \phi_1^s)}{1 - \phi_1} \cdot (y_n - y_{n-1} - \mu) + e_{n+s} \qquad (2.18)$$

где

$$e_{n+s} = \varepsilon_{n+s} + (1+\phi_1)\cdot\varepsilon_{n+s-1} + \ldots + \left(1+\phi_1+\phi_1^2+\ldots+\phi_1^{s-1}\right)\cdot\varepsilon_{n+1} \qquad (2.19)$$

Очевидно, что прогноз, минимизирующий среднеквадратичное отклонение, равен сумме первых трех слагаемых в (2.18). Заметим, что второе и третье слагаемые растут с ростом s. Ошибка прогноза на s шагов равна e_{n+s}. В силу формулы (2.19) дисперсия ошибки равна

$$V(e_{n+s}) = \sigma^2\left(1+(1+\phi_1)^2+\ldots+\left(1+\phi_1+\phi_1^2+\ldots+\phi_1^{s-1}\right)^2\right)$$

Отсюда видно, что в случае нестационарного временного ряда диспепсия ошибки прогноза монотонно растет с ростом горизонта прогноза s.

2.2 Множественная линейная регрессия

2.2.1 Общие положения

В регрессионном анализе изучается связь и определяется количественная зависимость между зависимой переменной и одной или несколькими независимыми переменными. Множественная линейная регрессия предполагает, что результатом или искомой переменной является линейная функция, зависящая от входных и независимых переменных и ошибки.

$$Y_i = a_0 + a_1\cdot x_{i1} + a_2\cdot x_{i2} + \ldots + a_k\cdot x_{ik} + \varepsilon_i,$$

где Y_i – зависимая переменная, $a_0, a_1, \ldots, a_k$ – линейные коэффициенты, $x_{i1}, x_{i2}, \ldots, x_{ik}$ – независимые переменные, ε_i – ошибка.

Для проведения регрессионного анализа необходимо, чтобы случайные ошибки наблюдений имели нормальный закон распределения

$$\varepsilon_i \approx N(0,\sigma), \qquad M(\varepsilon_i) = 0, \qquad D(\varepsilon_i) = \sigma^2 = const,$$

также необходимым является взаимная некореллированность (отсутствие автокорреляции) между ошибками наблюдений, т.е. последовательные значения ε_i не зависят друг от друга

$$M(\varepsilon_i\varepsilon_j) = \begin{cases}\sigma^2 & при \quad i = j\\ 0 & при \quad i \neq j\end{cases}.$$

2.2.2 Метод наименьших квадратов

Для нахождения линейных коэффициентов модели используется метод наименьших квадратов (МНК). Пусть проведено n независимых наблюдений случайной величины Y_i при соответствующих значениях $x_{i0}, x_{i1}, \ldots, x_{ik}$, совместный закон распределения которых неизвестен. Необходимо оценить эмпирическую функцию регрессии

$$\tilde{Y}_i = \tilde{a}_0 + \tilde{a}_1\cdot x_{i1} + \ldots + \tilde{a}_k\cdot x_{ik}.$$

Согласно МНК, параметры подбираются таким образом, чтобы минимизировать сумму квадратов отклонений наблюдаемых значений от расчетных по модели значений

$$F = \sum_{i=1}^{n}\left(Y_i - \tilde{Y}_i\right)^2 = \sum_{i=1}^{n}\left(Y_i - \tilde{a}_0 - \tilde{a}_1\cdot x_{i1} - \ldots - \tilde{a}_k\cdot x_{ik}\right)^2 \to \min,$$

где Y_i – наблюдаемые значения выходной переменной; $\tilde{Y}_i$ – значения выходной переменной, рассчитанные по модели.

Введем матричные обозначения:

$y = \begin{pmatrix} y_1 & y_2 & \dots & y_k \end{pmatrix}^T$ – вектор наблюдений,

$a = \begin{pmatrix} a_0 & a_1 & \dots & a_k \end{pmatrix}^T$ – вектор параметров,

$A = \begin{pmatrix} 1 & x_{11} & x_{21} & \cdots & x_{k1} \\ 1 & x_{12} & x_{22} & & x_{k2} \\ \cdots & & & \ddots & \\ 1 & x_{1n} & x_{2n} & & x_{kn} \end{pmatrix}$ – регрессионная матрица размерности $(nk+1)$.

Получаем новую запись для F:

$$F = \|y - Aa\|^2 \to \min$$

Из необходимых условий минимума

$$\frac{\partial F}{\partial a} = 0,$$

или

$$A^T \cdot (y - Aa) = 0$$

Система нормальных уравнений имеет вид $(A^T \cdot A) \cdot a = A^T \cdot y$. При условии, что $A^T A$ – невырожденная матрица, решение системы можно записать в виде

$$a = (A^T \cdot A)^{-1} \cdot A^T \cdot y.$$

2.2.3 Проверка предпосылок регрессионного анализа

2.2.3.1 Оценка адекватности модели

Для оценки адекватности рассчитанной регрессионной модели вычисляется коэффициент детерминации

$$R^2 = \frac{\sum_{n=1}^{k} (\hat{Y}_n - \overline{Y})^2}{\sum_{n=1}^{k} (Y_n - \overline{Y})^2},$$

где $\hat{Y}_n$ – прогнозные значения, $\overline{Y}$ – среднее значение для Y, $\overline{Y} = \frac{1}{k} \cdot \sum_{n=1}^{k} Y_n$. Коэффициент детерминации показывает долю правильно спрогнозированных величин. Если $R^2 = 0.75$ это значит, что модель работает на 75%, а 25% приходится на ошибку или неучтенные в модели факторы

Далее вычисляется величина $R = \sqrt{R^2}$, которая является оценкой множественного коэффициента корреляции между результатами наблюдений и вычисленными значениями $\hat{Y}_n$.

2.2.3.2 Проверка значимости параметров модели

Для проверки значимости l-го параметра уравнения регрессии используют t-критерий:

$$t_l = \frac{a_l}{S(a_l)},$$

подчиняющуюся закону распределения Стьюдента с $(k-m-1)$ степенями свободы (где m – количество параметров x_{ik}). Знаменатель этого выражения – среднее квадратичное отклонение коэффициента уравнения регрессии a_m:

$$S(a_m) = \sqrt{(X^T \cdot X)^{-1}_{jm} \cdot S^2_{ост}},$$

где $S_{ост}$ – среднее квадратичное отклонение для остатков:

$$S^2_{ост} = \frac{1}{k-m-1} \cdot \sum_{n=1}^{k} \left(Y_n - \widetilde{Y}_n\right)^2.$$

Когда расчетное значение t-критерия превосходит его табличное значение при заданном уровне значимости, коэффициент регрессии считается значимым.

2.2.3.3 Проверка гипотезы о равенстве нулю всех коэффициентов

Для проверки гипотезы о равенстве нулю всех коэффициентов рассчитывается F-критерий:

$$F = \frac{\frac{1}{k-1} \cdot \sum_{n=1}^{k} \left(Y_n - \overline{Y}_n\right)^2}{S^2_{ост}},$$

где $\overline{Y}_n$ – среднее значение для Y, а F подчиняется распределению Фишера с $(k-1)$ и $(k-m-1)$ степенями свободы. Если расчетное значение F-критерия больше его табличного значения при заданном уровне значимости, то уравнение регрессии считается значимым.

2.2.3.4 Проверка на автокорреляцию случайных ошибок

Критерий Дарбина-Уотсона позволяет определить наличие автокорелляции остатков в модели:

$$DU = \frac{\sum_{n=1}^{k} (d_n - d_{n-1})^2}{\sum_{n=1}^{k} d_n{}^2},$$

где $d_n = Y_n - \widetilde{Y}_n$ – разность прогнозируемого и фактического значения (остаток) на интервале в k точек, для которых Y_n известно. Если значение DU близко к 2, то автокорреляция остатков отсутствует. Предельное отклонение от 2, при котором гипотеза о некоррелированности остатков принимается – табличная величина, зависящая от N.

2.2.4 Нелинейные модели регрессии и линеаризация

К сожалению, не все модели регрессии являются линейными, но некоторые из них можно преобразовать таким образом, что модифицированная модель будет линейной. Сам процесс приведения называется процедурой линеаризации. Рассмотрим пример: если функции $\varphi_0, \varphi_1, \ldots, \varphi_k$ являются функциями, определяющими переход к преобразованным переменным, то есть $\tilde{Y}_i = \varphi(Y_i)$, $\tilde{x}_{i1} = \varphi(x_{i1})$, …, $x_{ik} = \varphi(x_{ik})$, то линейная модель регрессии может быть представлена в виде: $\tilde{Y}_i = a_0 + a_1 \cdot \tilde{x}_{i1} + a_2 \cdot \tilde{x}_{i2} + \ldots + a_k \cdot \tilde{x}_{ik} + \varepsilon_i$.

В случае невозможности линеаризации модели приходится исследовать искомую регрессионную зависимость в виде $Y_i = f(X_i, a) + \varepsilon_i$ и вычислять МНК-оценки линейных коэффициентов по формуле $a_{МНК} = \arg\min_a \sum_{i=1}^{n} (Y_i - f(X_i, a))^2$.

Кроме того, нужно иметь в виду, что при вычислении параметров по методу МНК минимизируется сумма квадратов отклонений преобразованных, а не исходных данных.

2.3 Нейронные сети

2.3.1 Общие положения

Нейронные сети были разработаны на основе функционирования нервной клетки. Основой метода явилось то, что нейрон достаточно легко представить в машинном виде, а всю сложность нервной клетки и другие качества ее работы можно определить связями. Это обуславливает применимость нейронных сетей для различных видов задач обучения по прецедентам, в том числе распознавания, классификации, регрессии и прогнозирования.

Нейронная сеть состоит из простейших элементов – нейронов. Каждый нейрон обладает группой синапсов – однонаправленных входных связей, соединенных с выходами других нейронов, а также имеет аксон – выходную связь данного нейрона, с которой сигнал поступает на синапсы следующих нейронов. Каждый синапс характеризуется величиной синаптической связи или ее весом w_i.Общий вид нейрона приведен на рисунке 2.1.

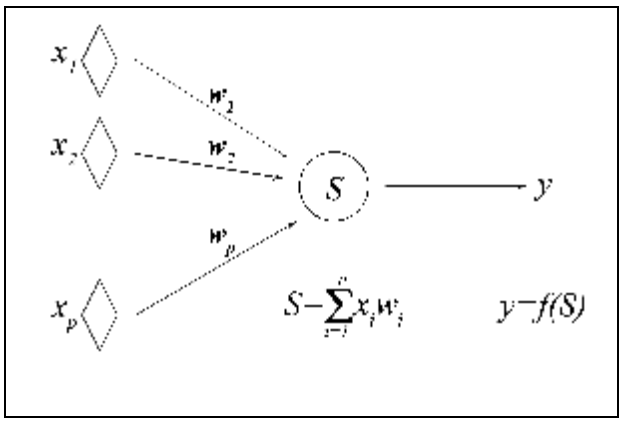


Рис. 2.1 Общий вид нейрона.

Текущее состояние нейрона определяется, как взвешенная сумма его входов:

$$s = \sum_{i=1}^{p} x_i \cdot w_i$$

Выход нейрона есть функция его состояния:

$$y = f(s)$$

Функция $f(s)$ называется функцией активации и может быть линейной, пороговой, сигмоидной и др. Примеры показаны на рисунке 2.2.

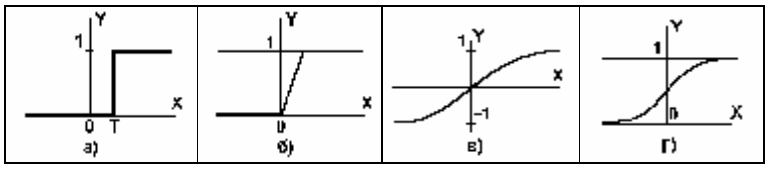


Рис. 2.2 а) пороговая функция единичного скачка; б) линейный порог (гистерезис); в) сигмоид – гиперболический тангенс; г) сигмоид – $f(x) = \dfrac{1}{1+e^{-\alpha x}}$

Одной из наиболее распространенных является нелинейная функция с насыщением (так называемая логистическая функция или сигмоид (т.е. функция S-образного вида)):

$$f(x) = \frac{1}{1+e^{-\alpha x}}$$

При уменьшении α сигмоид становится более пологим, в пределе при $\alpha = 0$ вырождаясь в горизонтальную линию на уровне 0.5, при увеличении α сигмоид приближается по внешнему виду к функции единичного скачка с порогом T в точке $x = 0$. Из выражения для сигмоида очевидно, что выходное значение нейрона лежит в диапазоне [0,1]. Одно из ценных свойств сигмоидной функции – простое выражение для ее производной:

$$f'(x) = \alpha \cdot f(x) \cdot (1 - f(x))$$

Следует отметить, что сигмоидная функция дифференцируема на всей оси абсцисс, что используется в некоторых алгоритмах обучения. Кроме того, она обладает свойством усиливать слабые сигналы лучше, чем большие, и предотвращает насыщение от больших сигналов.

Общей чертой, присущей всем нейронным сетям является принцип параллельной обработки сигналов, который достигается путем объединения большого числа нейронов в так называемые слои и соединения определенным образом нейронов различных слоев, а также, в некоторых конфигурациях, и нейронов одного слоя между собой, причем обработка взаимодействия всех нейронов ведется послойно.

Любая нейронная сеть состоит из входного и выходного слоя и может иметь один или более скрытых слоев. На входной слой подается вектор исходных данных $X = \{x_1, \ldots, x_p\}$, затем входные данные преобразуются нейронами сети и выдают результат на выходной слой, где производится сравнение результата с пороговой величиной и выдается результат – вектор $Y = \{y_1, \ldots, y_k\}$.

Помимо входного и выходного слоев в многослойной сети существуют так называемые скрытые слои. Они представляют собой нейроны, которые не имеют непосредственных входов исходных данных, а связаны только с выходами входного слоя и с входом выходного слоя. Таким образом, скрытые слои дополнительно преобразуют информацию и добавляют нелинейности в модели.

Теоретически число слоев и число нейронов в каждом слое может быть произвольным, однако фактически оно ограничено ресурсами компьютера или специализированной микросхемы, на которых обычно реализуется нейронные сети. Чем сложнее нейронная сеть тем масштабнее задачи, подвластные ей.

Общий вид нейронной сети показан на рисунке 2.3.

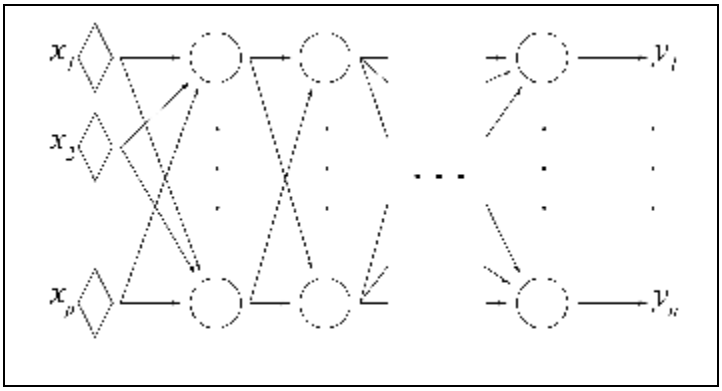


Рис. 2.3 Общий вид нейронной сети.

2.3.2 Классификация нейронных сетей

По архитектуре связей нейронные сети могут быть сгруппированы в два класса: сети прямого распространения, в которых связи не имеют петель, и сети рекуррентного типа, в которых возможны обратные связи.

Сети прямого распространения подразделяются на однослойные и многослойные персептроны. В них нейроны расположены в несколько слоев. Нейроны первого слоя получают входные сигналы, преобразуют их и через точки ветвления передают нейронам второго слоя. Далее срабатывает второй слой и т.д. до слоя k, который выдает выходные сигналы для интерпретатора и пользователя. Если противное не оговорено, то каждый выходной сигнал слоя i подается на вход всех нейронов слоя $i+1$. Число нейронов в каждом слое может быть любым и никак заранее не связано с количеством нейронов в других слоях. Стандартный способ подачи входных сигналов: все нейроны первого слоя получают каждый входной сигнал. Особенно широко распространены трехслойные персептроны, так как было доказано, что трех слоев достаточно для реализации практически любой функции p аргументов. В них каждый слой имеет свое наименование: первый – входной, второй – скрытый, третий – выходной.

Класс рекуррентных нейронных сетей гораздо обширнее, сами сети сложнее по своему устройству. Поведение рекуррентных сетей описывается дифференциальными или разностными уравнениями, как правило, первого порядка. Это существенно расширяет области применения нейронных сетей и способы их обучения. Сеть организована так, что каждый нейрон получает входную информацию от других нейронов и, возможно, нейронов того же слоя и от самого себя. Этот тип сетей имеет важное значение, так как с их помощью можно моделировать нелинейные динамические системы.

Среди рекуррентных сетей можно выделить сети Хопфилда и сети Кохонена. С помощью сетей Хопфилда можно обрабатывать неупорядоченные (рукописные буквы), упорядоченные во времени (временные ряды) или пространстве (графики) образцы.

Сеть Кохонена еще называют "самоорганизующейся картой признаков". Сеть такого типа рассчитана на самостоятельное обучение, во время обучения сообщать ей правильные ответы необязательно. В процессе обучения на вход сети подаются различные образцы. Сеть улавливает особенности их структуры и разделяет образцы на кластеры, а уже обученная сеть относит каждый вновь поступающий пример к одному из кластеров, руководствуясь некоторым критерием "близости".

В данной работе рассматриваются только сети прямого распространения.

2.3.3 Обучение нейронной сети

На этапе обучения происходит вычисление синаптических коэффициентов w_i в процессе решения нейронной сетью задач, в которых нужный ответ определяется не по правилам, а с помощью примеров, сгруппированных в обучающие множества. От того, насколько качественно будет выполнено обучение, зависит способность сети решать поставленные перед ней проблемы во время эксплуатации. На этапе обучения кроме параметра качества подбора весов важную роль играет время обучения. Как правило, эти два параметра связаны обратной зависимостью и их приходится выбирать на основе компромисса.

Одним из самых распространенных алгоритмов обучения нейронных сетей с учителем является алгоритм обратного распространения ошибки (Back Propagation Algorithm). Идея алгоритма заключается в том, что нейронная сеть за один проход «просматривается» дважды: на прямом проходе вычисляется ошибка нейронной сети на работе всех прецедентных векторов, на обратном проходе производится подсчет ошибок.

Согласно методу наименьших квадратов, минимизируемой целевой функцией ошибки нейронной сети является величина:

$$\Im(w) = \sum_{i=1}^{n} \sum_{j=1}^{k} \left(y_{ij}^{N} - \hat{y}_{ij} \right)^2 \to \min$$

где $y_{ij}^{(N)}$ – реальное выходное состояние нейрона j последнего (выходного) слоя N нейронной сети при подаче на ее входы i-го объекта; $\hat{y}_{ij}$ – идеальное (желаемое) выходное состояние этого нейрона.

Суммирование ведется по всем нейронам выходного слоя и по всем обрабатываемым сетью объектам. Минимизация ведется методом градиентного спуска, что означает подстройку весовых коэффициентов следующим образом:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \frac{\partial \Im}{\partial w_{ij}} \qquad \text{(НС 1)}$$

Здесь w_{ij} – весовой коэффициент синаптической связи, соединяющей i-ый нейрон слоя $n-1$ с j-ым нейроном слоя n, η – шаг градиентного спуска, $0 < \eta < 1$.

Далее

$$\frac{\partial \Im}{\partial w_{ij}} = \frac{\partial \Im}{\partial y_j} \cdot \frac{dy_j}{dv_j} \cdot \frac{\partial v_j}{\partial w_{ij}} \qquad \text{(НС 2)}$$

Здесь под y_j, как и раньше, подразумевается выход нейрона j, а под v_j – взвешенная сумма его входных сигналов, то есть аргумент активационной функции. Так как множитель $\frac{dy_j}{dv_j}$ является производной этой функции по ее аргументу, из этого следует, что производная активационной функция должна быть определена на всей оси абсцисс. В связи с этим функция единичного скачка и прочие активационные функции с неоднородностями не подходят для рассматриваемых нейронных сетей. В них применяются такие гладкие функции, как гиперболический тангенс или классический сигмоид с экспонентой.

Третий множитель $\frac{\partial v_j}{\partial w_{ij}}$, очевидно, равен выходу нейрона предыдущего слоя $y_i^{(n-1)}$.

Что касается первого множителя в (2), он легко раскладывается следующим образом:

$$\frac{\partial \Im}{\partial y_j} = \sum_k \frac{\partial \Im}{\partial y_k} \cdot \frac{dy_k}{dv_k} \cdot \frac{\partial v_k}{\partial y_j} = \sum_k \frac{\partial \Im}{\partial y_k} \cdot \frac{dy_k}{dv_k} \cdot w_{jk}^{(n+1)}$$

Здесь суммирование по k выполняется среди нейронов слоя $n+1$.

Введя новую переменную

$$\delta_j^{(n)} = \frac{\partial \Im}{\partial y_j} \cdot \frac{dy_j}{dv_j}$$

мы получим рекурсивную формулу для расчетов величин $\delta_j^{(n)}$ слоя n из величин $\delta_k^{(n+1)}$ более старшего слоя $n+1$.

$$\delta_j^{(n)} = \left[\sum_k \delta_k^{(n+1)} \cdot w_{jk}^{(n+1)} \right] \cdot \frac{dy_j}{ds_j}$$

Для выходного же слоя

$$\delta_l^{(N)} = (y_l^{(N)} - d_l) \cdot \frac{dy_l}{ds_l}$$

Теперь мы можем записать (2.1) в раскрытом виде:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \delta_j^{(n)} \cdot y_i^{(n-1)}$$

И скорректировать все веса в нейронной сети по формуле:

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t-1) + \Delta w_{ij}^{(n)}(t)$$

3 Методы линейной коррекции

Задача прогнозирования потребительского спроса обладает следующими специфическими особенностями:

- большое количество временных рядов, равное произведению количества магазинов на число товаров в них (для некоторых предприятий порядок может быть близок к 10^5);
- высокое отношение шума к сигналу;
- высокая изменчивость характеристик ряда, в том числе гетероскедастичность (непостоянство дисперсии).

Перечисленные особенности имеют ряд следствий:

- практическая невозможность моделировать временные ряды в рамках какой-либо одной модели алгоритмов, что приводит к необходимости построения алгоритмической композиции;
- необходимость использования эффективных методов настройки как базовых алгоритмов, так и корректирующей операции, в частности, желательно перенастраивать корректор в каждый момент времени только с учётом самых последних данных, не решая заново задачу построения алгоритма по всей предыстории.

В настоящей работе будем предполагать, что каждый базовый алгоритм B_i и корректирующая операция F дают в качестве результата прогноз потребительского спроса за прогнозируемый период в будущем. Таким образом, корректирующая операция A строится в виде суперпозиции: $A = F\left(B_1, \ldots, B_p\right)$.

Будем строить корректирующую операцию по формуле

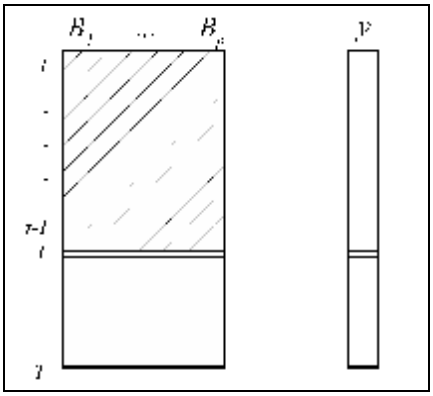
$$A_t = \sum_{i=1}^{p} w_t^i \cdot B_t^i \,,$$

где B_t^i – прогнозы базовых алгоритмов в момент времени t, w_t^i – весовые коэффициенты. На веса можно накладывать следующие ограничения:

$$w_t^i \geq 0 \qquad \text{и} \qquad \sum_{i=1}^{p} w_t^i = 1 \,.$$

Первое ограничение типа неравенства $w_t^i \geq 0$ отвечает за монотонность корректирующей операции, то есть при увеличении (уменьшении) ответа любого из базовых алгоритмов $B_1, \ldots, B_p$ ответ корректирующей операции A неубывает (невозрастает). Второе ограничение типа равенства $\sum_{i=1}^{p} w_t^i = 1$ определяет условие нормировки, его необходимость определяется по результатам экспериментов. Выполнение обоих условий означает выпуклость корректирующей операции, то есть ее ответ лежит в интервале между максимальным и минимальным прогнозами базовых алгоритмов.

В матричном варианте задача выглядит следующим образом:



Для каждого описанного ниже метода проводилось по два эксперимента: с ограничением неотрицательности весов и без ограничения.

3.1 Метод наименьших квадратов

Корректор построен таким образом, что зависит от всех алгоритмов из набора базовых алгоритмов. Каждому алгоритму присваивается вес w_t^i, который корректируется в процессе обучения таким образом, чтобы результат, выдаваемый корректором, был наилучший. В каждый момент времени t корректор строится по формуле $A_t = \sum_{i=1}^{p} w_t^i \cdot B_t^i$, где B_t – матрица из $(t-1)$-ой строки.

Проводилось два эксперимента: с наложением ограничения неотрицательности весов и без него. В первом случае решалась задача математического программирования, имеющая вид:

$$\begin{cases} A_t = \sum_{i=1}^{p} w_t^i \cdot B_t^i \\ \sum_{i=1}^{p} w_t^i = 1 \\ w_t^i \geq 0 \end{cases},$$

во втором – решаемая задача математического программирования имела вид

$$\begin{cases} A_t = \sum_{i=1}^{p} w_t^i \cdot B_t^i \\ \sum_{i=1}^{p} w_t^i = 1 \end{cases}.$$

Для нахождения весов модели используется метод наименьших квадратов (МНК).

$$\sum_{s=1}^{t-1} \left(\sum_{i=1}^{p} w_t^i \cdot B_s^i - y_s \right)^2 \to \min_{w_t^1, \ldots, w_t^p}$$

После перенастройки корректора по интервалу времени $[1, t-1]$ вычисляется прогноз алгоритмической композиции в момент времени t, ошибка прогноза $\varepsilon_t = y_t - A_t$ и ошибки усредняются по всем t. Полученная оценка называется оценкой скользящего контроля.

Для настройки МНК с одним ограничением равенства вида $\sum_{i=1}^{p} w_t^i = 1$ и одним ограничением неравенства вида $w_t^i \geq 0$ можно использовать стандартные методы математического программирования.

В системе MATLAB есть встроенные функции для решения задач вида

$$\frac{1}{2} w^T \cdot H \cdot w + f^T \cdot w \to \min_w$$

с ограничениями $A \cdot w \leq b$, $A_{eq} \cdot w = b_{eq}$, $lb \leq w \leq ub$, где H, A, A_{eq} являются матрицами, а f, b, b_{eq}, lb, ub и w – векторами.

Для того чтобы воспользоваться этой функцией, данные в матричной форме были приведены следующим образом:

$$\Phi = \left\| B \cdot w - y \right\|^2 \to \min_w ,$$

$$\Phi = w^T \cdot B \cdot B^T \cdot w - 2 \cdot y \cdot B^{\cdot T} \cdot w + y^2 \to \min_w ,$$

$$\frac{1}{2} \cdot w^T \cdot \left(B \cdot B^T \right) \cdot w + \left(- y \cdot B^T \right) \cdot w \to \min_w .$$

Таким образом, используя $H = B \cdot B^T$, $f = -y \cdot B^T$, $A_{eq} = \left[a_i = 1, \quad i = \overline{1, p} \right]$, $b_{eq} = 1$, $lb = 0$, вычисляем весовые коэффициенты корректирующей операции с помощью встроенной в MATLAB функции.

3.2 Метод наименьших квадратов с регуляризацией

Корректор строится таким же образом как в предыдущем пункте $A_t = \sum_{i=1}^{p} w_t^i \cdot B_t^i$, ограничения на веса остаются прежними $\sum_{i=1}^{p} w_t^i = 1$, $w_t^i \geq 0$. Однако при перенастройке корректора в каждый момент времени возникает проблема неустойчивости весовых коэффициентов. Потребуем, чтобы веса не изменялись сильно при переходе от момента времени t к моменту времени $t+1$. Для формализации этого естественного требования вводится штрафное слагаемое $\lambda \cdot \sum_{i=1}^{p} \left(w_t^i - w_{t-1}^i \right)^2$ в функционал качества:

$$\sum_{s=1}^{t-1} \left(\sum_{i=1}^{p} w_t^i \cdot B_s^i - y_s \right)^2 + \lambda \cdot \sum_{i=1}^{p} \left(w_t^i - w_{t-1}^i \right)^2 \to \min_{w_t^1, \ldots, w_t^p} .$$

Вид этого слагаемого соответствует обычной практике регуляризации задач, некорректно поставленных по Тихонову. Параметр λ позволяет регулировать стабильность весов по времени. Этот параметр находится следующим образом: вычисляется средняя ошибка прогнозов по скользящему контролю для каждого фиксированного λ и выбирается такое значения $\lambda = \lambda^*$, при котором ошибка минимальна.

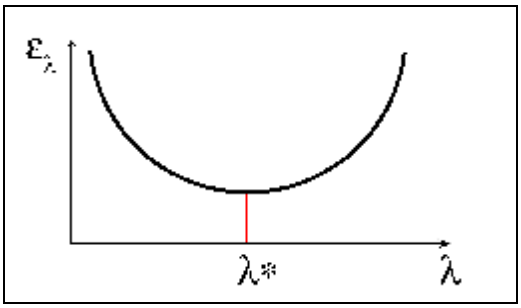


Рис. 3.1. Определение оптимального значения параметра регуляризации λ^*

Для инициализации процесса настройки весов в первый момент времени $t=1$ всем весам приписывается значение $w_t^i = \frac{1}{p}$, затем решается задача нахождения весов методом МНК. При переходе от момента времени t, $t=2,\ldots$ к моменту времени $t+1$ решается задача нахождения весов методом МНК с регуляризацией.

Эту задачу также можно свести к встроенной в MATLAB функции $\frac{1}{2} w^T \cdot H \cdot w + f^T \cdot w \to \min\limits_w$. При аналогичном приведении данных получаем $H = B \cdot B^T + \lambda \cdot I_p$, $f = -\left(y \cdot B^T + \lambda \cdot w_{t-1}^T\right)$, ограничения на A_{eq}, b_{eq} и lb остаются прежними. Далее задача решается встроенной в систему MATLAB функцией.

Приведение данных:

$$\Phi = \left\| B \cdot w - y \right\|^2 + \lambda \cdot \left(w - w_{t-1}\right)^T \cdot \left(w - w_{t-1}\right) \to \min_w ,$$

$$\Phi = w^T \cdot B \cdot B^T \cdot w - 2 \cdot y \cdot B^{\cdot T} \cdot w + y^2 + \lambda \cdot w^T \cdot w - 2 \cdot \lambda \cdot w^T \cdot w_{t-1} + \lambda \cdot w_{t-1}^T \cdot w_{t-1} \to \min_w ,$$

$$\frac{1}{2} \cdot w^T \cdot \left(B \cdot B^T + \lambda \cdot I_p\right) \cdot w + \left(- y \cdot B^T - \lambda \cdot w_{t-1}^T\right) \cdot w \to \min_w .$$

Удобно ввести вспомогательный коэффициент $\alpha(\tau)$ в одно из слагаемых функционала качества:

$$\Phi\left(w_t^1, \ldots, w_t^p\right) = \sum_{\tau=1}^{t} \alpha(\tau) \cdot \left(\sum_{i=1}^{p} w_\tau^i \cdot B_\tau^i - y_\tau \right)^2 \to \min_{w_t^i,\, i=1,\ldots,p} ,$$

который вычисляется следующим образом:

$$\alpha(\tau) = \begin{cases} K^{t-\tau}, & \tau = 1, \ldots, t-1 \\ 1, & \tau = t \end{cases} .$$

Коэффициент K является коэффициентом «забывания» предыстории.

3.3 Метод локальной адаптации весов с регуляризацией

Корректор построен таким же образом, что рассмотренные выше: $A_t = \sum\limits_{i=1}^{p} w_t^i \cdot B_t^i$. Кардинальное отличие состоит в том, что учитывается история продаж товара не во все моменты времени t, $t=1, 2, \ldots, t-1$, а только за предыдущий момент $t-1$ времени t. Веса корректора также вычисляются методом наименьших квадратов с регуляризацией, то есть в модель

введено штрафное слагаемое вида $\lambda \cdot \sum_{i=1}^{p} \left(w_t^i - w_{t-1}^i\right)^2$, которое позволяет регулировать стабильность весов по времени. Таким образом, веса находятся по следующей схеме:

$$\left(\sum_{i=1}^{p} w_t^i \cdot B_{t-1}^i - y_{t-1}\right)^2 + \lambda \cdot \sum_{i=1}^{p} \left(w_t^i - w_{t-1}^i\right)^2 \to \min,$$

в которой ограничения на веса остаются прежними: $\sum_{i=1}^{p} w_t^i = 1$, $w_t^i \geq 0$. Заметим, что этот случай соответствует рассмотренному в предыдущем разделе при $K = 0$.

Инициализация корректора повторяет шаги построения корректора методом наименьших квадратов с регуляризацией. Веса в начальный момент времени $t = 1$ приравниваются $w_t^i = \frac{1}{p}$, и в следующий момент времени t, $t = 2, \ldots$ находятся методом наименьших квадратов с регуляризацией.

Параметр λ находится следующим образом: вычисляется ошибка прогноза $\varepsilon_t(\lambda) = y_t - A_t$ для каждого фиксированного λ и выбирается такое значения $\lambda = \lambda^*$, при котором ошибка минимальна (рис. 3.1).

Предполагается, что в момент времени $t-1$ корректор настроен соответствующим образом, тогда для нахождения весов в момент времени t необходимо провести ту же процедуру.

В данной работе производится проверка эффективности корректора такого вида. Очевидно, что время работы такого алгоритма значительно меньше времени работы других корректоров, описанных в работе.

Задачу нахождения весовых коэффициентов методом локальной адаптации с регуляризацией также можно свести к встроенной в MATLAB функции $\frac{1}{2} w^T \cdot H \cdot w + f^T \cdot w \to \min_w$, при $H = B \cdot B^T + \lambda \cdot I_p$, (где B – вектор, а не матрица), $f = -\left(y \cdot B^T + \lambda \cdot w_{t-1}^T\right)$, ограничения на A_{eq}, b_{eq} и lb остаются прежними.

3.4 Простейшие (эталонные) алгоритмы коррекции

3.4.1 Выбор наилучшего из базовых алгоритмов (model selection)

Построение корректора производится на основе результатов базовых алгоритмов. Из всех базовых алгоритмов выбирается тот, результаты которого давали наименьшую ошибку на большем количестве алгоритмов на информации обучения, то есть $A_t = B_t^i$, для которого средняя ошибка прогнозов $\varepsilon_t = y_t - B_t^i$ на предыстории была бы наименьшей. Здесь также можно ввести коэффициент «забывания» предыстории K.

3.4.2 Простое усреднение

Корректор представляет собой среднее арифметическое всех базовых алгоритмов $A = \frac{1}{p} \cdot \sum_{i=1}^{p} B_i$. Далее вычисляется ошибка прогноза: $\varepsilon_t = y_t - A_t$.

4 Результаты вычислительных экспериментов

Далее используются следующие обозначения методов:

МНК-[η] — метод наименьших квадратов с «забыванием» предыстории η;

ЛАВР — локальная адаптация весов с регуляризацией, она же МНК-[0];

-М — монотонный корректор с ограничением неотрицательности весов;

-неМ — без ограничения неотрицательности весов.

4.1 Описание данных

Данные представлены в виде матрицы. Строки соответствуют периодам времени t, $t = 1,2,\ldots$, первый столбец соответствует реальным данным о продажах, остальные – прогнозам базовых алгоритмов $B_1, \ldots, B_p$.

4.2 Подбор параметров регуляризации

Параметр регуляризации λ позволяет регулировать стабильность весов по времени Его наилучшее значение находится по геометрической прогрессии следующим образом: на первом шаге параметру присваивается значение $\lambda = 1 \cdot 10^{-6}$, на каждом последующем шаге он изменяется в 2 раза $\lambda = 2 \cdot \lambda$.

4.3 Сравнение методов по точности контрольных прогнозов

Проводилось два эксперимента по подбору коэффициента регуляризации: с ограничением неотрицательности весов $w_t^i \geq 0$ и без ограничения. Заметим, что в случае линейной коррекции ограничение неотрицательности весов эквивалентно требованию монотонности корректирующей операции.

С неотрицательными весами точность прогнозов оказалась выше. Это подтверждает целесообразность предъявления требования монотонности к корректирующим операциям.

Неожиданные (на первый взгляд) результаты показал анализ зависимости точности прогнозов ε от параметра регуляризации λ. При отсутствии ограничения монотонности график $\varepsilon(\lambda)$ имеет классическую форму с чётко выраженным минимумом. При наличии ограничения график $\varepsilon(\lambda)$ становится почти монотонно возрастающей функцией, (рис. 4.1 и рис. 4.2). Чем меньше λ, тем лучше точность прогнозов. Она не ухудшается даже в пределе при $\lambda \to +0$, когда регуляризация фактически отключается.

Данный факт можно объяснить тем, что требование монотонности само по себе является хорошим регуляризатором. По всей видимости, обнуление весов «плохих» базовых алгоритмов как раз и обеспечивает устойчивость остальных весов. Это снова подтверждает целесообразность предъявления требования монотонности к корректирующим операциям.

Заметим также, что сильная регуляризация ($\lambda \to +\infty$) в обоих случаях эквивалентна простому усреднению и даёт худшую точность прогнозов.

В экспериментах вычислялся показатель неустойчивости весов как доля случаев, когда вес базового алгоритма w_t^i изменяется более чем на 10%. Оказалось, что неустойчивость весов у ЛАВР-М практически в 2 раза выше, чем у МНК-М, при этом средняя ошибка прогнозов на 11.7% выше. Менее устойчивый корректор обладает лучшими показателями точности на скользящем контроле (рис. 4.2).

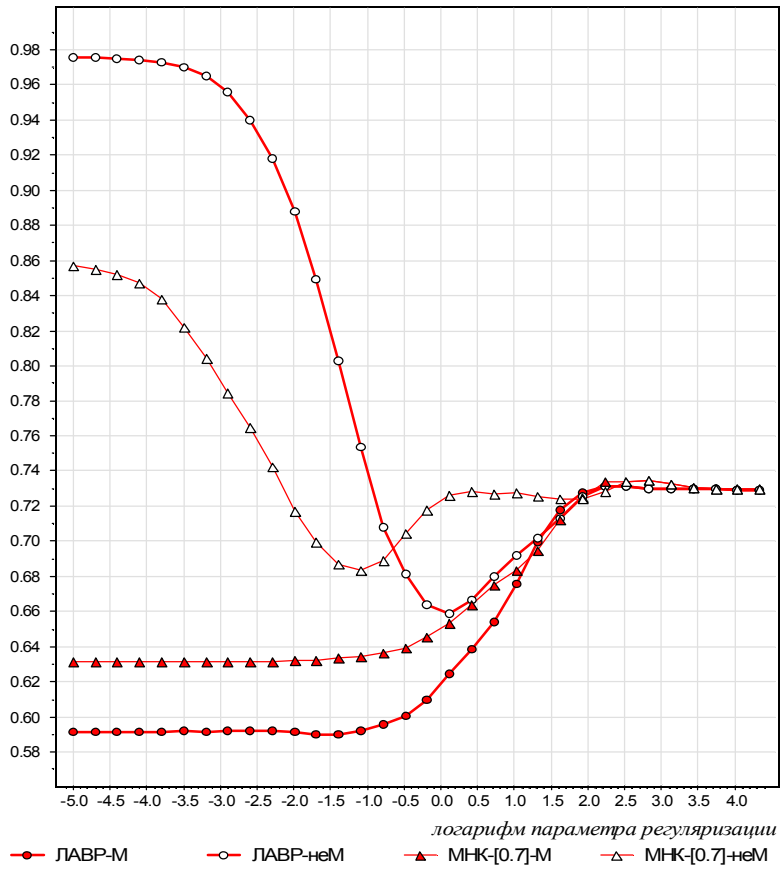


Рис. 4.1. Зависимость средней ошибки прогнозов ε от параметра регуляризации λ.

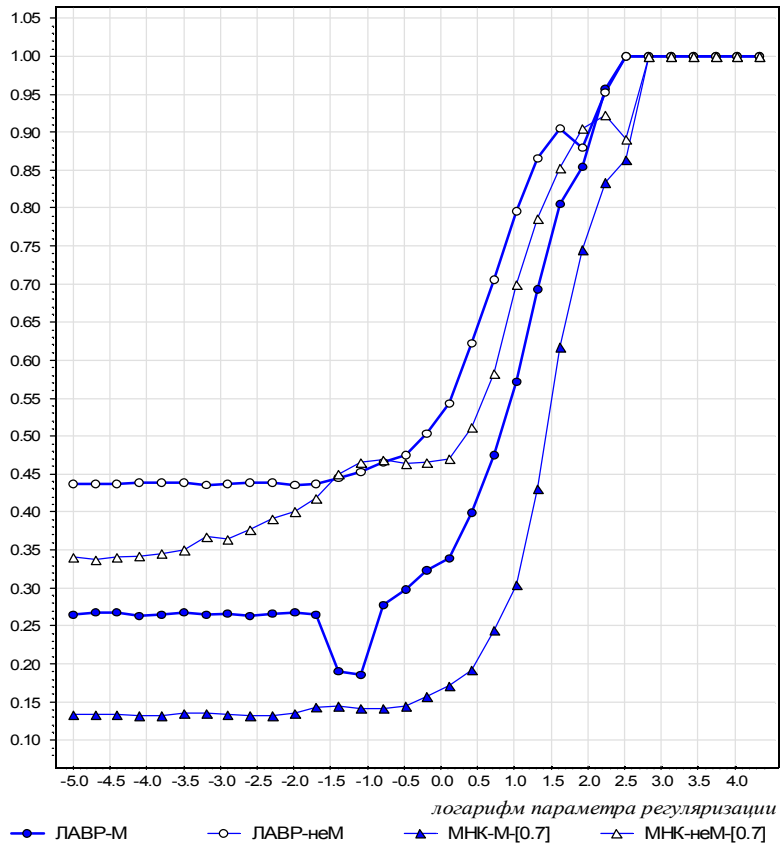


Рис. 4.2. Зависимость средней неустойчивости весов от параметра регуляризации λ.

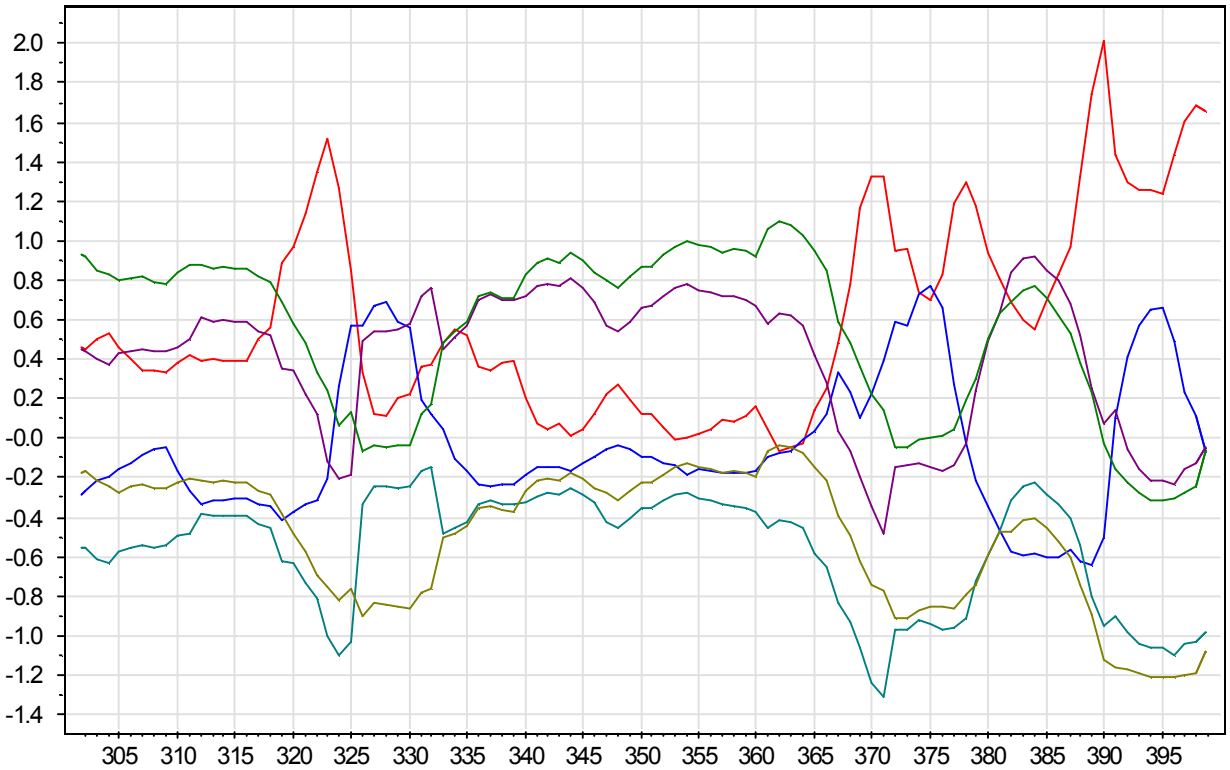


Рис. 4.3. Метод ЛАВР-неМ. Фрагмент динамики весов базовых алгоритмов при оптимальном λ.

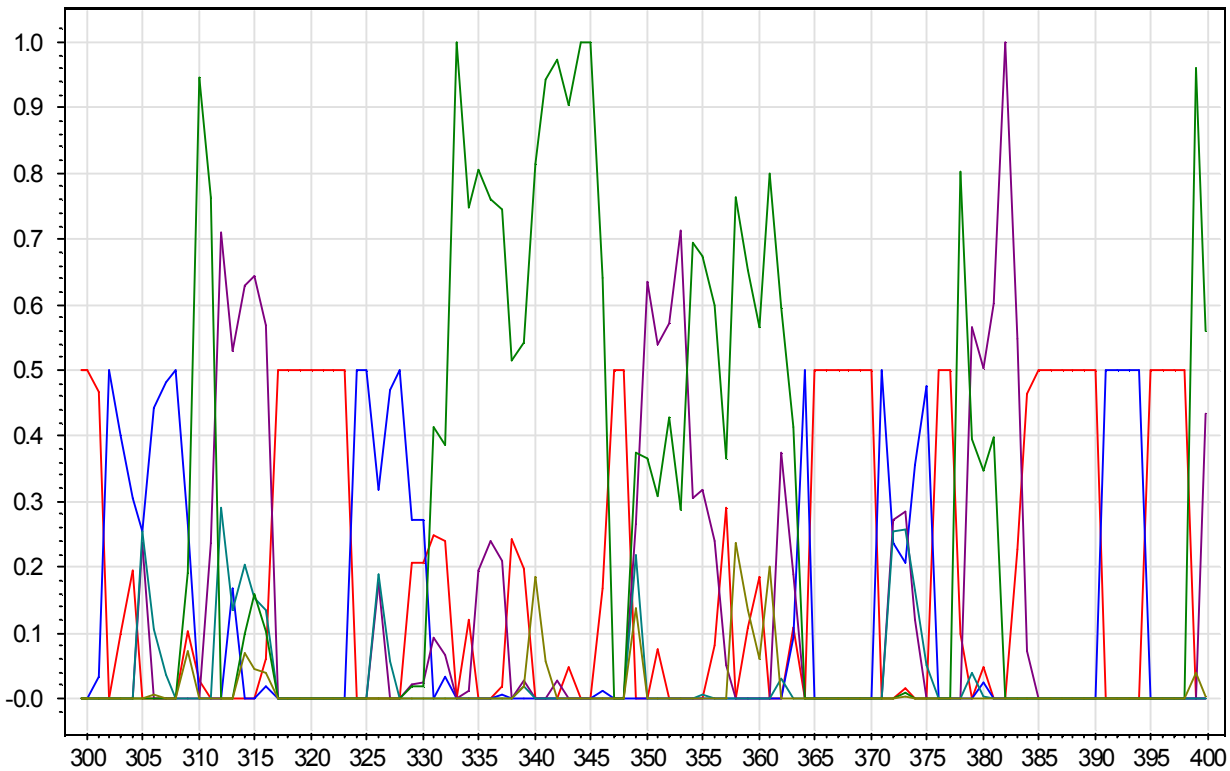


Рис. 4.4. Метод ЛАВР-М. Фрагмент динамики весов базовых алгоритмов при оптимальном λ.

Выбор наилучшего из базовых алгоритмов дает наилучший результат при использовании стратегии «забывания» предыстории (рис. 4.5). Оптимум достигается при $K = 0.2 \div 0.3$, что приблизительно эквивалентно выбору модели по 3–5 последним дням.

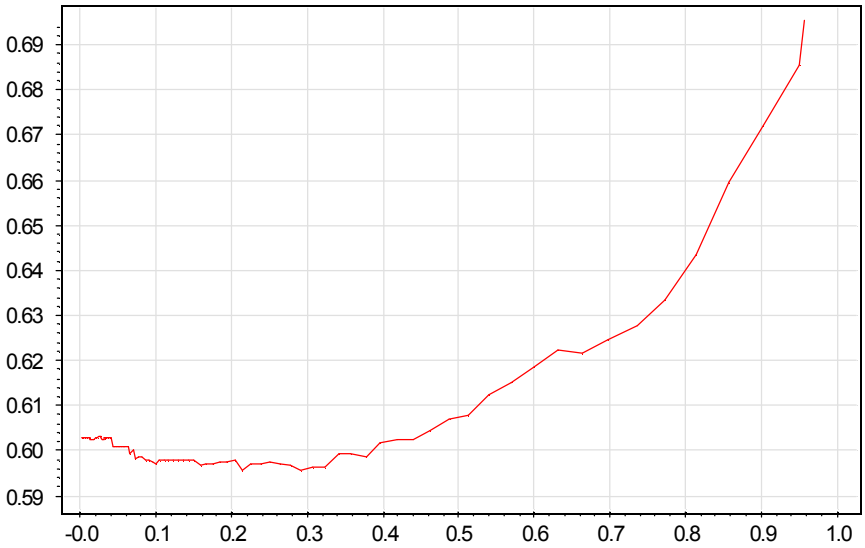


Рис. 4.5. Метод Model Selection.
Зависимость средней ошибки прогнозов ε от параметра сглаживания K.

На рис. 4.6. приведен график используемости каждого базового алгоритма в методе выбора наилучшего из базовых алгоритмов. Слева на рисунке отображен график изменения номера лучшей модели во времени. Справа – гистограмма используемости моделей, по горизонтальной оси — число включений модели, по вертикальной оси — номер модели.

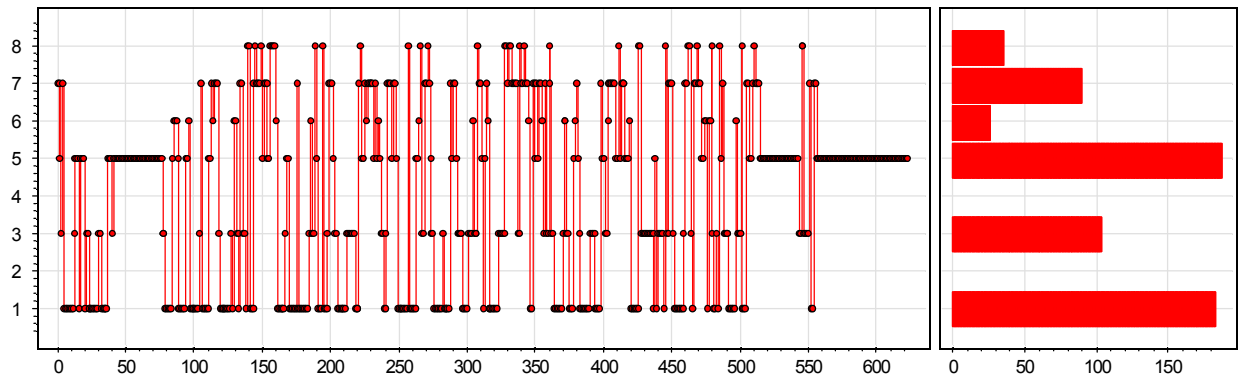


Рис. 4.6. Метод Model Selection.
Динамика переключения моделей при оптимальном значении параметра сглаживания K.

В качестве эталона для сравнения использовалось также взвешенное среднее базовых алгоритмов с весами, вычисленными по МНК на всём интервале времени. Заметим, что данный алгоритм невозможно использовать для прогнозирования, поскольку он «заглядывает в будущее».

Рассматривались два варианта: МНК-1 — метод наименьших квадратов с ограничением $\sum_{i=1}^{p} w_t^i = 1$; МНК-0 — метод наименьших квадратов без ограничений.

Оба варианта практически не улучшают точность прогнозирования по сравнению с базовыми алгоритмами.

Алгоритм		Средняя ошибка на скользящем контроле
Базовые алгоритмы		
B_1	базовый алгоритм	0.7624
B_2	базовый алгоритм	0.7624
B_3	базовый алгоритм	0.7793
B_4	базовый алгоритм	0.7793
B_5	базовый алгоритм	0.7294
B_6	базовый алгоритм	0.7664
B_7	базовый алгоритм (лучший)	0.7142
B_8	базовый алгоритм	0.7534
Эталонные (простейшие) алгоритмы		
Среднее	усреднение с равными весами	0.7294
Выбор	выбор лучшей модели при оптимальном значении параметра сглаживания	0.5956
Выбор	выбор лучшей модели без использования параметра сглаживания	0.9107
МНК-0	МНК по всем данным, без ограничения на сумму весов	0.6763
МНК-1	МНК по всем данным, с ограничением на сумму весов	0.7142
Алгоритмы МНК		
ЛАВР-М	локальная адаптация весов с регуляризацией, при ограничении монотонности	**0.5899**
ЛАВР-неМ	локальная адаптация весов с регуляризацией, без ограничения монотонности	0.6591
МНК-[0.7]-М	метод наименьших квадратов, параметр «забывания» предыстории = 0.7, при ограничении монотонности	0.6314
МНК-[0.7]-неМ	метод наименьших квадратов, параметр «забывания» предыстории = 0.7, без ограничения монотонности	0.6834

5 Заключение

В данной работе рассматривалась задача прогнозирования временных рядов. В связи с высокой изменчивостью характеристик ряда и высоким отношением шума к сигналу использовался алгебраический подход к синтезу корректных алгоритмов, предложенный академиком РАН Ю.И. Журавлевым.

Для набора базовых алгоритмов $B_1, \ldots, B_p$ строилась линейная корректирующая операция вида $A_t = \sum_{i=1}^{p} w_t^i \cdot B_t^i$, где w_t^i – веса базовых алгоритмов. Для настройки корректора были использованы следующие методы:

- метод наименьших квадратов,
- метод наименьших квадратов с регуляризацией,
- метод локальной адаптации весов с регуляризацией,
- простое усреднение,
- выбор наилучшего из базовых алгоритмов.

Для эффективной настройки использовались ограничения $\sum_{i=1}^{p} w_t^i = 1$, $w_t^i \geq 0$ и в некоторых из этих алгоритмов использовался коэффициент «забывания» предыстории.

Все эти алгоритмы реализованы в среде MATLAB и проведены вычислительные эксперименты на реальных данных потребительского спроса. На основании результатов экспериментов произведен сравнительный анализ алгоритмов по критерию точности.

Предложенный в данной работе метод прогнозирования временных рядов на основе локальной адаптации весов и регуляризации является новым. Он строит прогноз на основе информации только за предыдущий момент времени, а не всей истории, что значительно сокращает время его работы. Эксперименты показали, что этот метод является существенно более эффективным, чем другие методы, рассмотренные в работе, и хорошо подходит для решения прикладной задачи прогнозирования потребительского спроса.

По результатам экспериментов можно сказать, что с ограничением неотрицательности весов ($w_t^i \geq 0$) точность прогнозов оказалась выше. Это объясняется тем, что требование монотонности само по себе является хорошим регуляризатором.

Точность алгоритмов зависит от параметра регуляризации λ. Эксперименты показали, что точность прогнозов выше при малых значениях λ («слабой» регуляризации). Точность не ухудшается даже в пределе при $\lambda \to +0$, когда регуляризация фактически отключается. Заметим также, что «сильная» регуляризация ($\lambda \to +\infty$) эквивалентна простому усреднению и даёт худшую точность прогнозов.

Использование коэффициента «забывания» предыстории также обосновано. Экспериментально показано, что в данной прикладной задаче настройка корректирующей операции по короткой предыстории даёт лучшие результаты, чем настройка по длинной предыстории. Ярким примером является метод выбора наилучшего из базовых алгоритмов, качество прогноза которого улучшилось на 25% при переходе от использования всей предыстории к предыстории оптимальной длины (примерно в 5 дней) При использовании более 3–5 дней качество прогнозирования начинает ухудшаться. Таким образом, косвенно подтверждается гипотеза о существенной нестационарности временных рядов продаж.

Сделанные выводы справедливы, по всей видимости, только для временных рядов, характеризующихся существенной нестационарностью и высоким отношением шум/сигнал. Представляет интерес проведение аналогичных исследований и для временных рядов другой природы.

В работе также представлен общий обзор методов прогнозирования временных рядов (множественная линейная регрессия, ARMA, ARIMA, нейронные сети), изучены области применимости и сравнительные характеристики различных методов.

Список литературы

1. Воронцов К.В. О проблемно-ориентированной оптимизации базисов задач распознавания // ЖВМ и МФ. 1998 Т. 38, № 5. с. 870-880
2. Журавлев Ю.И. Корректные алгебры над множествами некорректных (эвристических) алгоритмов. I-III // Кибернетика. 1977. № 4. С. 5-17, 1977. № 6. С. 21-27, 1978. № 2. С. 35-43.
3. Журавлев Ю.И. Об алгебраическом подходе к решению задач распознавания и классификации // Проблемы кибернетики. Вып. 33 М.: Наука, 1978, С. 5-68.
4. Рудаков К.В. О применении универсальных ограничений при исследовании алгоритмов классификации //Кибернетика. 1988. № 1. с. 1-5.
5. Рудаков К.В. О симметрических и функциональных ограничениях для алгоритмов классификации // ДАН СССР. 1987. Т. 297, № 1. с. 43- 46.
6. Рудаков К.В. Об алгебраической теории универсальных и локальных ограничений для задач классификации // Распознавание, классификация, прогноз. М.: Наука, 1989. С. 176-201.
7. Рудаков К.В. Полнота и универсальные ограничения в проблеме коррекции эвристических алгоритмов классификации // Кибернетика. 1987. № 3. с. 106 - 109.
8. Рудаков К.В. Симметрические и функциональные ограничения для алгоритмов классификации // Кибернетика. 1987. № 4. с. 73 - 77.
9. Рудаков К.В. Универсальные и локальные ограничения в проблеме коррекции эвристических алгоритмов классификации // Кибернетика. 1987. № 2. с. 30- 35.
10. Jane E. Lappin A primer on consumer marketing research: procedures, methods, tools // 1994
11. Sven F. Crone Training artificial neural networks for time series prediction using asymmetric cost functions // Institute of Business Information Systems, University of Hamburg, Germany
12. Frank M. Thicsing, Oliver Vornberger Forecasting sales using neural networks
13. Ralph Snider Forecasting sales of slow and fast moving Inventories // 1999
14. T. Tchaban, J. P. Griffin A comparison between single and combined back-propagation neural networks in the prediction of turnover
15. Sanjay Goil, Alok Choudhary An infrastructure for scalable parallel multidimension analysis
16. Jingtao Yao, Nicholas Teng, Hean-Lee Poh, Chew Lim Tan Forecasting and analysis of marketing data using neural networks //1997
17. G. Antoniol, G Casazza, M. Di Penta, E. Merlo Modeling clones evolution through time series
18. Taras Tchaban, Joe P. Griffin, Malcolm J. Taylor The application of intelligent decision support systems in profiling of retail outlets
19. Магнус Я.Р., Катышев П.К., Пересецкий А.А. Эконометрика //Москва, Дело 2004