

Смеси моделей векторной авторегрессии в задаче прогнозирования временных рядов

Федоряка Д.С.¹, Нейчев Р.Г.¹, Стрижов В.В.²

fedimser@phystech.edu

¹МФТИ, ²ВЦ РАН

В данной статье исследуется задача краткосрочного прогнозирования временных рядов. Рассматриваются временные ряды разного масштаба, связанные между собой и обладающие свойством периодичности. Задача прогнозирования сводится к задаче регрессии, которая решается с помощью линейной модели. Для повышения её точности предлагается применить композицию моделей. Композиции строятся с помощью бэггинга, метода случайных подпространств и алгоритма бустинга AdaBoost. Также предлагается эвристический итерационный алгоритм композиции моделей, основанный на идее алгоритма кластеризации K-means. С помощью предлагаемых методов производится построение прогноза потребления электроэнергии в Турции и Польше, а также цен на электроэнергию в Германии с учетом информации о погоде.

Ключевые слова: *временные ряды, прогнозирование, авторегрессия, линейная регрессия, бустинг, AdaBoost, бэггинг, смесь экспертов.*

1 Введение

В данной статье рассматривается задача краткосрочного прогнозирования временных рядов. Рассматриваются наборы связанных между собой временных рядов, обладающих свойством сезонности. Данная задача сводится к задаче регрессии, которая решается композицией линейных моделей.

Это важная практическая задача, т.к. на основании точных прогнозов можно принимать более оптимальные решения, чем на основании только знаний об уже произошедших событиях. Данная задача имеет приложения в экономике, торговле, транспортных перевозках, обслуживании систем и т.д.

Ввиду прикладной важности данной задачи, она исследовалась с середины прошлого века, и к настоящему времени существует множество различных методов её решения. В книге [1] рассмотрены линейные модели. В этих моделях прогноз линейно зависит от значений ряда в предыдущие моменты времени. Существует три основных класса линейных моделей: авторегрессионные модели, интегральные модели и модели скользящего среднего. На их основе построены модели авторегрессионного скользящего среднего (Autoregressive Moving Average, ARMA), авторегрессионного интегрированного скользящего среднего (Autoregressive Integrated Moving Average, ARIMA) и другие.

В книге [2] рассматриваются адаптивные методы — простые модели, в которых значение в будущий момент времени вычисляется на основе значения и предсказания для текущего момента времени.

Существует множество других подходов: анализ сингулярного спектра [3], нейросетевые модели [4, 5], квантильная регрессия [6], вероятностное моделирование [7].

Далее рассмотрим новейшие исследования в области. В работе [8] решается задача определения периода периодического временного ряда с помощью применения метода

главных компонент к матрице Ганкеля для данного ряда. В работах [9] и [10] рассматривается отбор признаков для решения задачи прогнозирования временных рядов методом авторегрессии. В работе [11] решается задача предсказания дискретных временных рядов (а именно – музыкальных произведений) с помощью изучения вероятностного распределения N-грамм в обучающей выборке. В работе [12] рассматривается прогнозирование набора временных рядов с учётом иерархической структуры порождающих их объектов с применением теории игр. В работе [13] решается задача прогнозирования объёмов железнодорожных грузоперевозок. Для построения прогноза используется алгоритм, основанный на свертке эмпирической плотности распределения значений временного ряда с функцией потерь.

В данной работе предлагается свести задачу прогнозирования временного ряда к задаче векторной регрессии. Действительно, если ряд обладает периодичностью, то в пределах одного периода можно выявить некоторую зависимость поздних значений от предшествующих им значений того же ряда и от одновременных с ними значений связанных рядов, которая повторяется от периода к периоду. Затем на основании этой зависимости можно предсказать неизвестные значения ряда.

Полученную задачу предлагается решать методом линейной регрессии. Для повышения точности предлагается строить композиции линейных моделей. Рассматривается построение композиций с помощью бэггинга и метода случайных подпространств [14] и с помощью бустинга [14, 15].

Кроме того, предлагается эвристический итерационный алгоритм для композиции моделей, основанный на идее алгоритма кластеризации K-means. Он строит систему из нескольких моделей линейной регрессии и разбиение объектов обучающей выборки на кластеры так, чтобы каждый кластер описывался одной моделью и каждый объект принадлежал кластеру с той моделью, которая его лучше всего описывает. Предсказание строится как смесь экспертов [14], в которой экспертами являются полученные линейные модели.

Для практической проверки предлагаемых методов решается задача предсказания потребления электроэнергии и цен на неё. Сравняется точность предсказания разными методами.

2 Постановка задачи

Дан набор вещественнозначных временных рядов: $\mathfrak{D} = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_p\}$. Отсчёты каждого ряда сделаны через равные промежутки времени, но могут иметь разный масштаб (см. рис. 1):

$$\mathbf{s}_i = [s_i(\Delta t_i), s_i(2\Delta t_i), \dots, s_i(L_i\Delta t_i)],$$

где L_i — длина i -го ряда, Δt_i - его период дискретизации.

При этом ряды описывают один и тот же промежуток времени, и их можно уложить в общую сетку, т.е. существуют $T \in \mathbb{R}$, $l_1, \dots, l_p \in \mathbb{N}$, $m, A \in \mathbb{N}$, такие что

$$l_1\Delta t_1 = l_2\Delta t_2 = \dots = l_p\Delta t_p = T,$$

$$L_1\Delta t_1 = L_2\Delta t_2 = \dots = L_p\Delta t_p = m \cdot A \cdot T.$$

При этом известно, что временные ряды коррелируют между собой, обладают свойством периодичности с периодом T , а также с периодом $A \cdot T$. Под периодичностью понимается повторение вида ряда через период: $s(t) \approx s(t - T)$.

Значения рядов на последний период неизвестны, и требуется их предсказать для одного ряда \mathbf{s}_1 , т.е. в моменты времени

$$(L_1 - l_1 + 1)\Delta t_1, (L_1 - l_1 + 2)\Delta t_1, \dots, (L_1)\Delta t_1.$$

Пусть $s_1(t)$ — действительные значения временного ряда в момент времени t , а $\hat{s}_1(t)$ — предсказанные некоторой моделью f на основании данного набора \mathfrak{D} . Введём функцию потерь как сумму квадратов разностей предсказанных и истинных значений:

$$S(f) = \sum_{i=L_1-l_1+1}^{L_1} [s_1(i\Delta t_1) - \hat{s}_1(i\Delta t_1)]^2.$$

Тогда формально задача прогнозирования ставится следующим образом:

$$f^* = \arg \min_f S(f, \mathfrak{D}).$$

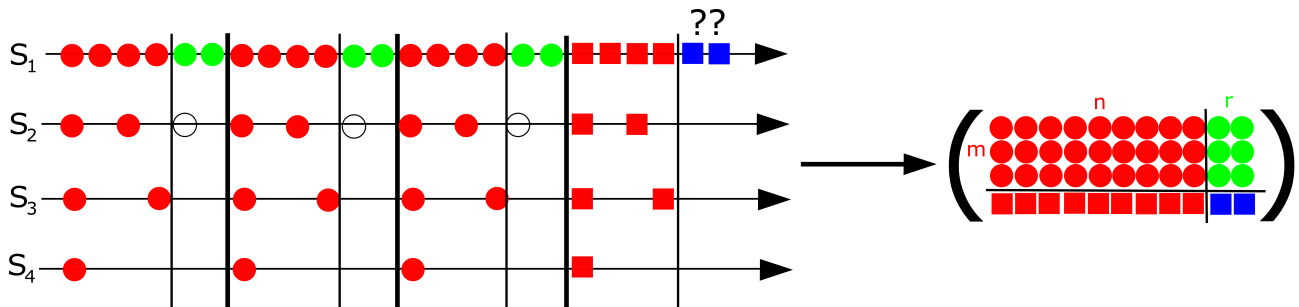


Рис. 1. Временные ряды разных масштабов и матрица авторегрессии

3 Авторегрессия

Приведём поставленную задачу к задаче векторной регрессии. Основная идея авторегрессии заключается в том, что значение временного ряда в данный момент времени зависит от значений ряда в предыдущие моменты времени:

$$x_{i,j} = f(x_{i,j-1}, x_{i,j-2}, \dots, x_{i,j-R}),$$

где R — порядок авторегрессии, т.е. количество учитываемых моментов времени.

В данной работе предлагается следующий подход, основанный на методе авторегрессии. Разобьём ряд на равные участки через равные промежутки времени. Затем каждый участок разобьём на две части — признаков и ответов так, чтобы часть «ответов» последнего отрезка совпала с отрезком прогнозирования. Теперь для всех отрезков, кроме последнего, выявим закономерность между этими двумя частями. Затем построим предсказание, применив эту закономерность к части признаков первого ряда.

Рассмотрим один ряд \mathbf{s}_i . Для удобства обозначим элемент ряда $s_i^j = s_i(j \cdot \Delta t_i)$.

Построим авторегрессионную матрицу следующим образом. Разобьём ряд на отрезки длительности AT и запишем их друг под другом. Согласно гипотезе о периодичности, можно считать, что последние l_i значений зависят от всех остальных. Тогда правую часть полученной матрицы можно рассматривать как матрицу «объект-признак», а левую часть — как матрицу «объект-ответ». Объектами в данном случае являются периоды.

$$\left(\begin{array}{ccc|cc} s_i^1 & \dots & s_i^{(A-1)l_1} & \dots & s_i^{Al_1} \\ \vdots & \ddots & \vdots & \vdots & \ddots \\ s_i^{(m-2)Al_1+1} & \dots & s_i^{((m-1)A-1)l_1} & \dots & s_i^{(m-1)Al_1} \\ \hline s_i^{(m-1)Al_1+1} & \dots & s_i^{(mA-1)l_1} & \dots & s_i^{mAl_1} \end{array} \right) = \left(\begin{array}{c|c} X_i & Y_i \\ \hline m-1 \times (A-1)l_1 & m-1 \times l_1 \\ X_i^0 & \hat{Y}_i^0 \\ \hline 1 \times (A-1)l_1 & 1 \times l_1 \end{array} \right)$$

Матрицы X, Y_i, X_i^0 известны, т.к. они содержат предысторию. Строка Y_i^0 - искомый прогноз. Предлагается решая задачу регрессии найти зависимость ответов в матрице Y от признаков в матрице X , а затем, используя эту зависимость, посчитать значение Y_i^0 , как функцию от X_0 .

Теперь добавим информацию о связанных рядах. Построим следующую авторегрессионную матрицу, для каждого периода в качестве признаков содержащую отсчёты-признаки всех рядов (см. рис. 1):

$$\left(\begin{array}{c|c} X_1 X_2 \dots X_p & Y_1 \\ \hline X_1^0 X_2^0 \dots X_p^0 & Y_1^0 \end{array} \right) = \left(\begin{array}{c|c} X & Y \\ \hline X^0 & Y^0 \\ \hline m-1 \times n & m-1 \times l_1 \\ 1 \times n & 1 \times r \end{array} \right), \quad (1)$$

где общее число признаков

$$n = (A-1) \cdot \sum_{i=1}^p l_i,$$

а $r = l_1$ — запрос прогнозирования, т.е. количество прогнозируемых отсчётов.

Итак, задача сводится к задаче регрессии, т.е. поиску функции $f : \mathbb{R}^n \rightarrow \mathbb{R}^r$, которая точно определяет закономерность на обучающей выборке:

$$f^* = \arg \min_f \sum_{j=1}^{m-1} \|f(X_j) - Y_j\|_2^2. \quad (2)$$

4 Линейная регрессия

Простейшая модель алгоритма f — однородная линейная функция

$$[f(\mathbf{x})]_j = \sum_{k=1}^n x_k w_{k,j},$$

или, более кратко,

$$f(\mathbf{x}) = \mathbf{x}W, \quad (3)$$

где W - матрица коэффициентов $n \times r$.

Для такой модели решение задачи (2) существует и единственно, если $X^T X$ - невырождена [14]:

$$W = X^+Y = (X^T X)^{-1} X^T Y.$$

Если матрица X мультиколлинеарна (что характерно для рассматриваемой задачи, т.к. близкие значения временных рядов могут зависеть друг от друга), данная модель может переобучаться и становится неустойчивой (чувствительной к шумам). Это существенный недостаток, и с ним можно бороться регуляризацией и отбором признаков [10].

В данной работе предлагаются алгоритмы композиции над линейными моделями, которые позволяют выявить нелинейные закономерности, а также устраняют переобучение. Модель линейной регрессии рассматривается как базовая модель, и её результаты сравниваются с результатами других алгоритмов.

5 Композиции моделей

Рассматривается композиция алгоритмов [14] - построение алгоритма f на основе нескольких базовых алгоритмов f_1, \dots, f_K :

$$f(\mathbf{x}) = \sum_{i=1}^K \pi_i(\mathbf{x}) f_i(\mathbf{x}).$$

Здесь f_1, \dots, f_K — базовые модели, которые в данной работе будут линейными моделями (3) с различными матрицами весов. π_1, \dots, π_K - веса, с которыми модели учитываются. Для них должно выполняться $\sum_{i=1}^K \pi_i = 1$. Для модели простого голосования $\pi_i = \frac{1}{K}$. В общем случае, эти веса могут зависеть от объекта \mathbf{x} . В таком случае им можно придать вероятностный смысл: $\pi_i(\mathbf{x})$ - вероятность того, что объект \mathbf{x} описывается моделью f_i .

5.1 Бэггинг и метод случайных подпространств

Бэггинг (Bootstrap aggregating) – это композиция алгоритмов, каждый из которых обучается независимо на некотором подмножестве объектов исходной выборки. Метод случайных подпространств (Random Subspace Method, Attribute Bagging)– это композиция алгоритмов, каждый из которых обучается независимо на исходной выборке, но учитывает лишь некоторое подмножество признаков.

5.2 Бустинг

Бустинг (Boosting) — это процедура последовательного построения композиции алгоритмов машинного обучения, когда каждый следующий алгоритм стремится компенсировать недостатки композиции всех предыдущих алгоритмов. Бустинг представляет собой жадный алгоритм построения композиции алгоритмов. Изначально понятие бустинга возникло в работах по вероятно почти корректному обучению в связи с вопросом: возможно ли, имея множество плохих (незначительно отличающихся от случайных) алгоритмов обучения, получить хороший.

В данной работе рассматривается алгоритм AdaBoost (Adaptive Boosting) [14, 15]. AdaBoost вызывает в цикле базовую модель. После каждого вызова обновляется распределение весов, которые отвечают важности каждого из объектов обучающего множества

для классификации. На каждой итерации возрастают веса объектов, для которых предыдущая модель дала наихудшее предсказаний. Таким образом новая модель «фокусирует своё внимание» на этих объектах.

Рассмотрим достоинства и недостатки этого алгоритма. AdaBoost обладает более высокой обобщающей способностью, чем базовая модель, которая не ухудшается с ростом числа моделей. Алгоритм эффективен: время его работы практически полностью определяется временем построения базовых моделей.

Однако этому алгоритму требуются длинные обучающие выборки. Кроме того, он склонен к переобучению на шумовых объектах.

5.3 Кластерная регрессия

Ниже предлагается эвристический алгоритм для построения композиции линейных моделей.

При построении алгоритма используется идея итерационного алгоритма кластеризации K-means [14]: в пространстве объектов выделяются некоторые группы объектов и центры, затем объекты распределяются между центрами так, чтобы каждый объект был ближе к «своему» центру, чем к остальным. Изначально объекты разбиваются на группы случайным образом, а затем на каждой итерации центры вычисляются как центры масс в каждой группе и перераспределяются объекты.

Здесь группы задаются не центрами, а описывающими их моделями. Вместо вычисления центра масс производится обучение модели. «Ближайшую» модель для объектов находим как ту, которая даёт самую меньшую ошибку на этом объекте.

Обучение. Опишем итерационный алгоритм, который строит модели f_i и находит веса π_i . Он строит такое разбиение множества всех объектов на K непересекающихся подмножеств (классов)

$$\overline{1, m} = \bigsqcup_{j=1}^K C_j,$$

что объекты i -го класса описываются i -й моделью. При запуске алгоритма это разбиение строится случайно, затем за несколько итераций происходит перераспределение объектов и изменение моделей.

На первом шаге каждой итерации будем обучать модели на объектах соответствующих им классов.

На втором шаге будем вычислять для каждой пары объекта и модели $S(i, j)$ — ошибку предсказания ответа для i -го объекта j -й моделью, и на основании полученных значений перераспределять объекты между классами: объект поместим в j -класс, если j -я модель описывает его лучше остальных, т.е. даёт наименьшую ошибку.

После перераспределения объектов может возникнуть одна из двух исключительных ситуаций. В некоторый класс может не попасть ни один объект. Тогда, скорее всего, начальное количество классов K слишком большое, и надо удалить из рассмотрения эти пустые классы, уменьшив K . Может оказаться, что перераспределения не произошло. Тогда нужно остановить алгоритм, т.к. на следующих итерациях перераспределения, скорее всего, происходить тоже не будет.

Итак, алгоритм останавливается, если произошло заданное количество итераций или достигнуто стационарное состояние.

Предсказание. Пусть теперь нам надо найти ответ для объекта x . Для этого надо определить веса $\pi_i(x)$. Построим предварительный ответ для объекта, используя единственную линейную модель f_0 , обученную на всей выборке. Затем предскажем ответ с помощью каждой модели. Сравнивая предварительный ответ и ответы моделей, найдём веса. Для этого введём монотонно убывающую функцию $\mathfrak{E}(S_i(x))$, которая будет для ошибки $S_i(x) = \frac{\|f_i(x) - f_0(x)\|}{\|f_0(x)\|}$ вычислять меру правдоподобия. Вычислим правдоподобие для каждого класса, нормируем — и получим веса моделей:

$$\pi_i(x) = \frac{\mathfrak{E}(S_i(x))}{\sum_{i=1}^K \mathfrak{E}(S_i(x))}.$$

Примеры возможных функций $\mathfrak{E}(S)$: $\frac{1}{S+\varepsilon}$, e^{-S} , $1 - \frac{1}{1+e^{-S}}$.

Запишем алгоритм формально.

Вход:

$(X \in \mathbb{R}^{m \times n}, Y \in \mathbb{R}^{m \times r}, X^0 \in \mathbb{R}^n)$ — входные данные задачи авторегрессии;

K — число моделей;

N_{it} — число итераций;

\mathfrak{E} — функция преобразования погрешности в правдоподобие.

Выход: Y^0 — ответ задачи авторегрессии

- 1: $\forall j \in \overline{1, m} \ C_j = \text{rand}(1, m)$ — начальное разбиение на классы
- 2: для $iter \in \overline{1, N_{it}}$
- 3: для $j \in \overline{1, K}$
- 4: Обучаем алгоритм f_j на объектах с индексами C_j .
- 5: для $i \in \overline{1, m}$
- 6: для $j \in \overline{1, K}$
- 7: $S(i, j) = \|f_j(X_i) - Y_i\|$ — вычисление ошибок
- 8: для $i \in \overline{1, m}$
- 9: для $j \in \overline{1, K}$
- 10: $C_j = \{i \in \overline{1, m} \mid \underset{k}{\operatorname{argmin}}(S(i, k)) = j\}$ — перераспределение
- 11: если $\exists i : C_i = \emptyset$ то
- 12: $\{C_1, \dots, C_K\} = \{C_i \mid C_i \neq \emptyset\}$
- 13: если перераспределение не поменяло класс ни для одного объекта то
- 14: выйти из цикла
- 15: Обучаем алгоритм f_0 на всей выборке.
- 16: $\tilde{Y}^0 = f_0(X^0)$ — предварительный ответ.
- 17: $\forall j \in \overline{1, K} \ \pi'_j = \mathfrak{E}\left(\frac{\|f_j(X^0) - \tilde{Y}^0\|}{\|\tilde{Y}^0\|}\right)$ — вычисление вероятностей.
- 18: $\forall j \in \overline{1, K} \ \pi_j = \frac{\pi'_j}{\sum_{k=1}^K \pi'_k}$ — нормировка.
- 19: $Y^0 = \sum_{j=1}^K \pi_j \cdot f_j(X^0)$ — ответ.

У предложенного алгоритма имеется существенный недостаток. Ему требуются очень большие наборы данных. Так как набор данных разбивается на K классов, то теперь линейные модели обучаются на выборках, в среднем в K раз меньших, чем исходная. При этом необходимо, чтобы при обучении каждой модели число объектов превосходило число признаков. Иначе веса линейной модели будут подобраны так, что она будет абсолютно точно описывать обучающую выборку, и при перераспределении каждый объект будет

оставаться в своём классе, т.к. «своя» модель будет предсказывать его почти с нулевой ошибкой, а другие — нет. Тогда алгоритм остановится после первой итерации, и будет эквивалентен бэггингу.

Рассмотрим пример: имеется почасовой временной ряд, и предсказание строится по предыстории за 6 дней недели. Тогда признаков будет $6 \cdot 24 = 144$, значит в одном классе должно быть 144 объекта. Пусть $K = 10$. Тогда выборка должна охватывать 1440 недель, т.е. примерно 28 лет. Разумеется, на таком длительном периоде нарушаются многие закономерности.

Простой способ обойти эту проблему — уменьшить количество признаков, беря в рассмотрение не весь период, а, например, последний день. Тогда результат сильно ухудшается из-за потери информации.

6 Вычислительный эксперимент

6.1 Описание данных

В данной работе были построены предсказания для следующих временных рядов:

- 1) временные ряды потребления электроэнергии в Турции [16]. Период: с 1.01.2005 по 31.12.2005. Измерения — каждый час. Набор данных состоит из пяти временных рядов. Предсказывается один из них - «Decided Daily Production Program», на рис.2(a) — этот ряд за первые две недели 2005 года.
- 2) временные ряды почасовой загрузки энергетической системы Польши за 2002-2004гг. [17], на рис.2(b) - первые две недели 2002 года.
- 3) временные ряды рыночных цен на электроэнергию в Германии за 2003-2009 годы [18] (на рис.2(c) — этот ряд за первые четыре недели 2005 года) и ежедневные данные о погоде за тот же период [19].

6.2 Описание эксперимента

Для всех рядов предсказание проводилось для 24 часов ($r = 24$) на основании данных о предшествующих им 6 сутках ($n = 144$).

Для рядов строились матрицы авторегрессии (1) таким образом, что предсказывались данные на последние 24 часа ряда. Затем полученные предсказания сравнивались с известными значениями.

Сравниваются предсказания, полученные с помощью линейной регрессии, бустинга и бэггинга. Использовались соответственно реализации `LinearRegression`, `AdaBoostRegressor` и `BaggingRegressor` из библиотеки `sklearn`. Последние два алгоритма запускались с 40 базовыми моделями.

Для данных Турции проведено два эксперимента: предсказание ряда на основании данных только этого ряда («Турция 1»), а также с учётом показаний сопутствующих четырёх рядов («Турция 2»). Для данных Германии также было построено предсказание на основании только данных ряда цен («Германия 1») и с учётом средней температуры из ряда погоды («Германия 2»). Для данных Польши был проведён один эксперимент.

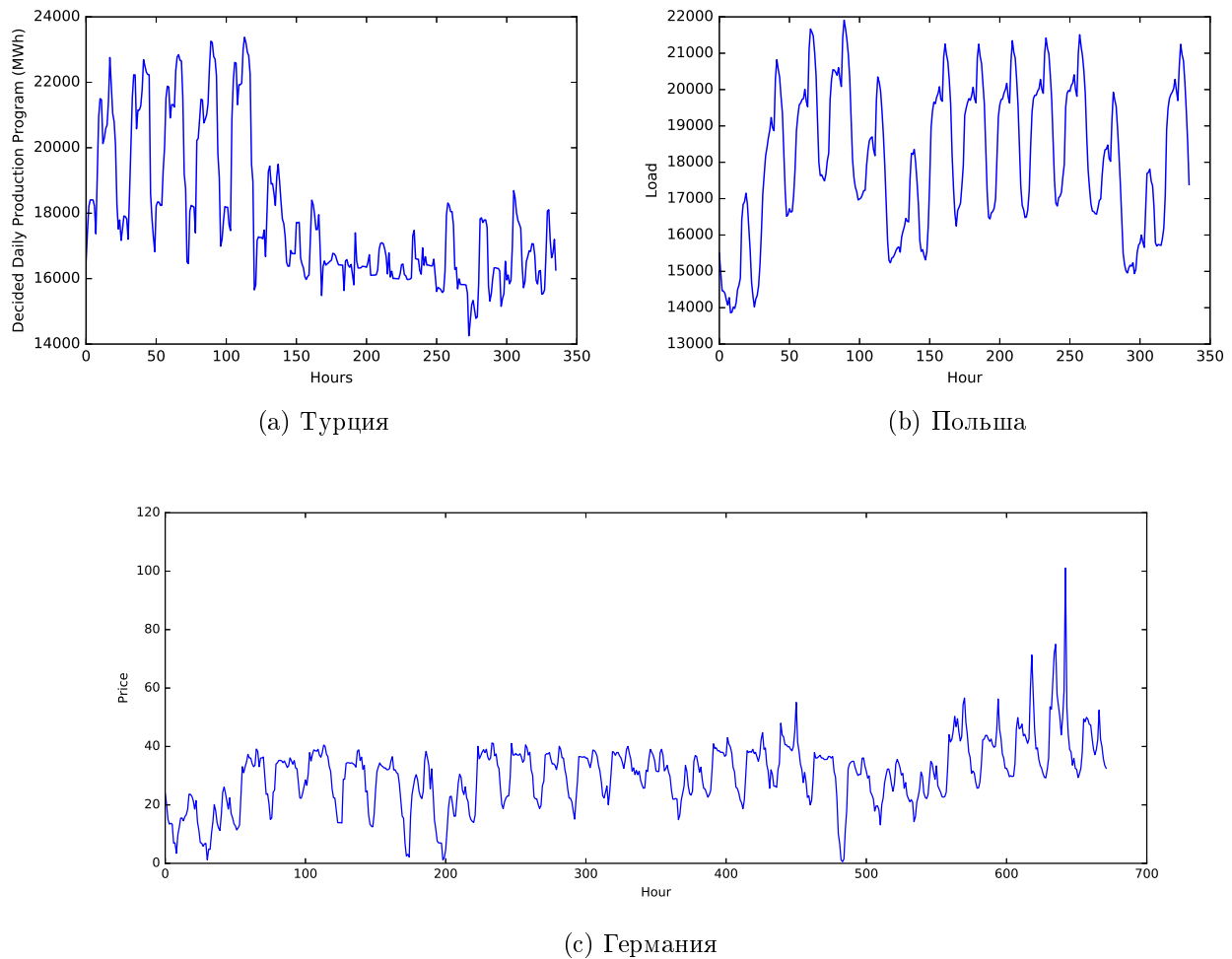


Рис. 2. Временные ряды

6.3 Результаты

На рис. 3 представлены результаты всех экспериментов. Для сравнения приводится график действительных значений ряда за этот период. В качестве функции ошибки используется среднеквадратичная ошибка, отнесённая к истинному среднему значению ряда на отрезке прогнозирования:

$$MSE(\hat{s}) = \frac{\sqrt{\langle (s_1 - \hat{s}_1)^2 \rangle}}{\langle s_1 \rangle} = \frac{1}{\sqrt{r}} \sqrt{\frac{\sum_i (s_1^i - \hat{s}_1^i)^2}{\sum_i s_1^i}} \quad (4)$$

В следующей таблице представлена величина ошибки для всех экспериментов.

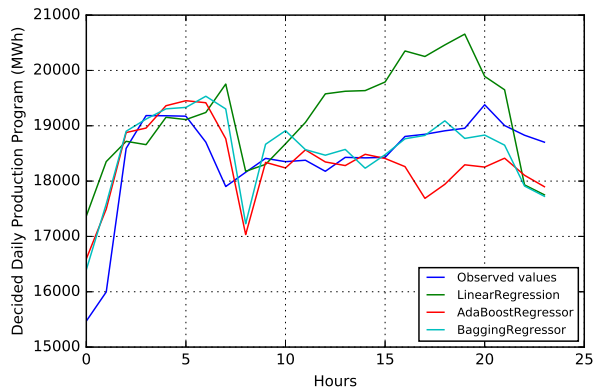
Эксперимент	Турция 1	Турция 2	Германия 1	Германия 2	Польша
Линейная регрессия	6.3%	4.0%	30.1%	28.8%	2.0%
Бустинг	3.8%	3.6%	24.1%	23.0%	3.6%
Бэггинг	3.6%	3.0%	27.5%	26.4%	1.7%

На наборе данных Турции исследовалась зависимость точности прогноза от количества базовых моделей (рис. 4). Ошибки усреднены по 25 независимым испытаниям. Можно

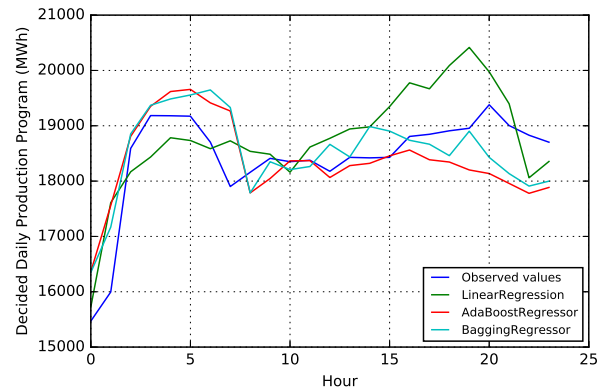
сделать вывод, что оптимальный результат достигается при небольшом числе базовых моделей (около 20).

Также был исследован предложенный алгоритм кластерной регрессии. По описанным выше причинам на данных выборках его тестировать бесполезно, т.к. для них первое случайное разбиения на классы не будет изменяться. Поэтому были построены выборки с сокращённым числом признаков. В качестве объектов рассматривались дни, строился прогноз на последние 8 часов на основании предыдущих 16 часов, как признаков. На таких выборках алгоритм давал результаты, близкие к результатам базового алгоритма — линейной регрессии. Этот результат почти не зависел от выбора функции ошибки \mathcal{E} , от количества классов и от количества итераций.

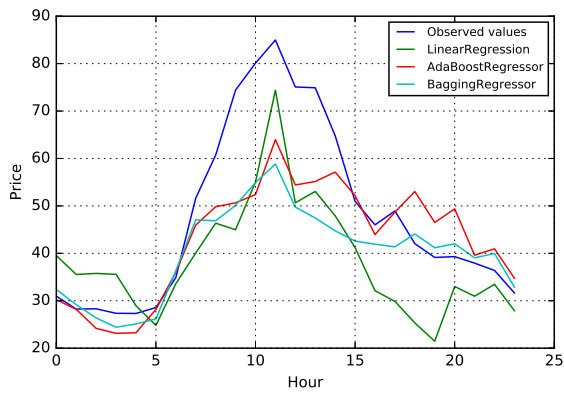
Такой результат может объясняться тем, что рассмотренные данные плохо поддаются кластеризации, т.е. не удаётся выделить несколько сильно отличающихся закономерностей в разных периодах временного ряда.



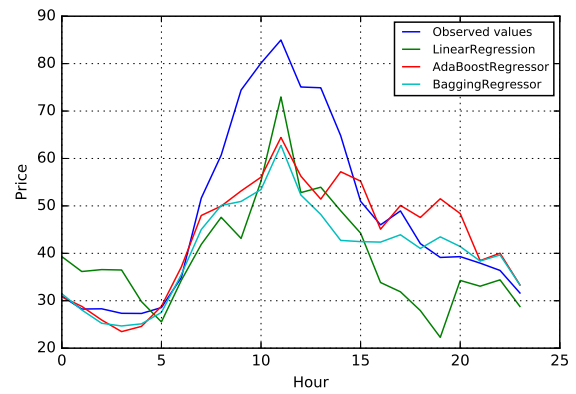
(a) Турция 1



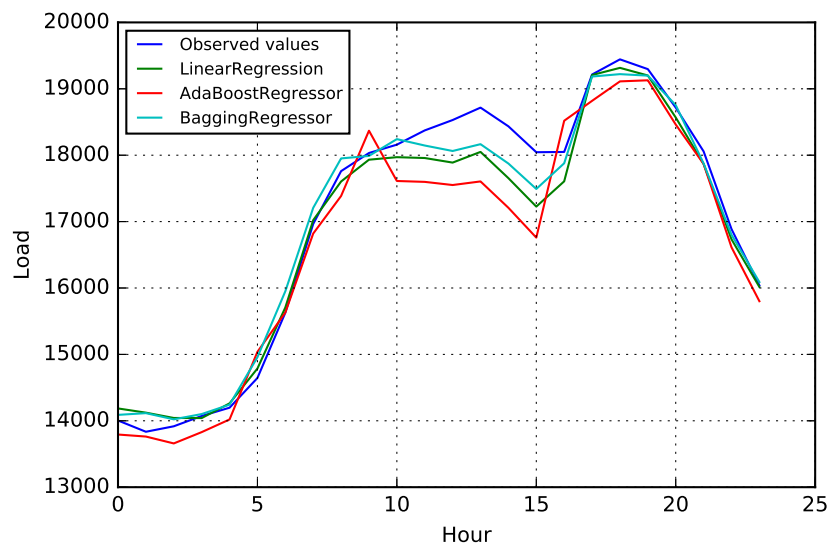
(b) Турция 2



(c) Германия 1



(d) Германия 2



(e) Польша

Рис. 3. Прогнозы

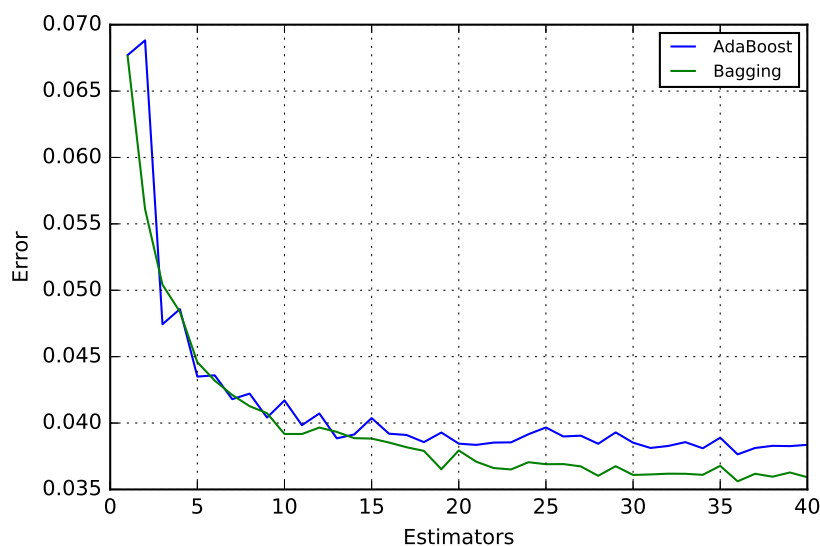


Рис. 4. Зависимость точности прогноза от числа базовых моделей

7 Заключение

Рассмотрена задача краткосрочного прогнозирования временных рядов. Исследовались временные ряды разного масштаба, связанные между собой и обладающие свойством периодичности. Задача прогнозирования была сведена к задаче регрессии, которая решалась с помощью линейной модели.

Для повышения точности предсказания были применены композиции моделей, которые строились двумя методами: бэггингом в сочетании с методом случайных подпространств и адаптивным бустингом. На реальных данных было показано, что такой подход в данной задаче применим, и что построение композиции моделей действительно повышает точность прогноза.

Также предложен эвристический итерационный алгоритм композиции моделей. Однако алгоритм не оказался эффективным в данной задаче из-за его требований к размеру обучающей выборки и из-за свойств временных рядов.

8 Список литературы

- [1] Бокс Дж. Дженкинс Г. *Анализ временных рядов: прогноз и управление. Выпуск 1.* М.: Мир, 1974.
- [2] Ю.П. Лукашин. *Адаптивные методы краткосрочного прогнозирования временных рядов.* М.: Финансы и статистика, 2003.
- [3] Н. Э. Голяндина. *Метод Гусеница-SSA: прогноз временных рядов.* СПбГУ, 2004.
- [4] Е.В. Дунаев Е.А. Ефремова. *Применение нейронных сетей для прогнозирования финансовых временных рядов. Доклады ТУСУРа,* 2004.
- [5] Дудаков В. П. Крючин О.В., Козадаев А. С. Прогнозирование временных рядов с помощью искусственных нейронных сетей и регрессионных моделей на примере прогнозирования котировок валютных пар. *Электронный научный журнал «ИССЛЕДОВАНО В РОССИИ»,* 2010.

- [6] Е.Ю. Клочков. Прогноз квазипериодических временных рядов непараметрическими методами. *Машинное обучение и анализ данных*, 1(3):328–334, 2012.
- [7] Э.Е.Тихонов. *Методы прогнозирования в условиях рынка*. 2006.
- [8] A. P. Motrenko and V. V. Strijov. Extracting fundamental periods to segment human motion time series. *Journal of Biomedical and Health Informatics*, PP(99):?, 2015.
- [9] Л. Н. Сандуляну and В. В. Стрижов. Выбор признаков в авторегрессионных задачах прогнозирования. *Информационные технологии*, 7:11–15, 2012.
- [10] Стрижов В.В. Нейчев Р.Г., Катруца А.М. Выбор оптимального набора признаков из мультикоррелирующего множества в задаче прогнозирования. 2015.
- [11] Mikhail Matrosov and Vadim Strijov. Short-term forecasting of musical compositions using chord sequences. In *Conference of the International Federation of Operational Research Societies*, page 229, 2014.
- [12] М. М. Стенина and В. В. Стрижов. Согласование прогнозов при решении задач прогнозирования иерархических временных рядов. *Информатика и ее применения*, 9(2):77–89, 2015.
- [13] Р. К. Газизуллина, М. М. Стенина, and В. В. Стрижов. Прогнозирование объемов железнодорожных грузоперевозок по парам веток. *Системы и средства информатики*, 25(1):144–157, 2015.
- [14] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [15] Schapire R. E. Freund Y. Experiments with a new boosting algorithm. */International Conference on Machine Learning*, 1996.
- [16] <https://svn.code.sf.net/p/dmba/code/data/turkelectricityconsumption.csv>.
- [17] <http://gdudek.el.pcz.pl/files/pl.xls>.
- [18] <https://svn.code.sf.net/p/dmba/code/data/germanspotprice.csv>.
- [19] <https://svn.code.sf.net/p/dmba/code/data/germanweather.xls>.