# Применение адаптивных ансамблей методов машинного обучения к задаче прогнозирования временных рядов

Т.А. Волошин, К.С. Зайцев, М.Е. Дунаев

**Аннотация - Целью данной работы является исследование процесса применения адаптивных ансамблей методов машинного обучения к задаче прогнозирования временных рядов, преимущественно для режима потоковой обработки данных. Для этого были проведены эксперименты с разными типами регрессионных алгоритмов, исследовались как классические методы ансамблирования деревьев решений (случайный лес, градиентный бустинг), так и нейронные сети, в частности сверточные и рекуррентные. Для проведения исследований использовались сгенерированные с помощью сервиса Apache Kafka временные ряды метрик работы приложения. Чтобы применить алгоритмы машинного обучения к подготовленным данным, задача прогнозирования рядов была преобразована к задаче обучения с учителем общего вида при помощи техники скользящего окна. В ходе экспериментов предсказания осуществлялись на продолжительный интервал, для чего была применена рекурсивная стратегия построения прогноза. Для улучшения результатов анализа в потоковом режиме была предложена модель ансамблирования нескольких методов, которая по мере поступления новых данных на основе ошибок пересчитывала веса, определявшие вклад, вносимый в итоговое предсказание каждым отдельным алгоритмом. Также была исследована возможность инициализации этих весов из предобучения модели на части тренировочной выборки. В результате проведения экспериментов с предложенными моделями ансамблевые методы показали хорошие с точки зрения точности результаты, а предложенная методика адаптивного взвешенного усреднения действительно оказалась способна улучшить эффективность прогнозирования. Результаты применения нейронных сетей впечатлили меньше, что может быть связано с использованием сравнительно небольшой выборки данных для обучения.**

***Ключевые слова — временные ряды, ансамбли методов, адаптивный подход, бустинг, случайный лес.***

## 1. ВВЕДЕНИЕ

В настоящее время многие компании в сферах индустрии и бизнеса, активно применяют методы мониторинга для собственных приложений и сервисов, собирая их метрики и логи. Одной из актуальных задач мониторинга является прогнозирование временных рядов на основе собранных данных для эффективного управления ресурсами систем. Решение этой задачи все чаще базируется на методах машинного обучения.

Задача прогнозирования временных рядов изучается уже достаточно долго в самых разных областях. Существует множество работ, исследующих применение различных методов для этой цели. Одним из широко распространенных методов является ARIMA [1, 2], основанный на авторегрессионной модели временного ряда, где значения ряда линейно зависят от предыдущих значений. Другим используемым методом является экспоненциальное сглаживание [3], которое основывается на учете влияния прошлых значений ряда на текущее значение, при этом вес каждого значения определяется экспоненциально убывающим коэффициентом. Также распространено использование таких нелинейных ансамблевых методов, как градиентный бустинг [4] и случайный лес [5]. Кроме того, в моделировании временных рядов зачастую применяются и нейронные сети [6], особо широко распространены сверточные [7] и рекуррентные [8, 9] сети, хорошо подходящие для анализа данных с последовательной структурой.

Настоящая работа посвящена применению ансамблевых методов градиентного бустинга и случайного леса к прогнозированию временных рядов. При этом для повышения эффективности прогнозирования предлагается использовать эти алгоритмы в ансамбле. А, для решения проблемы возможной изменчивости (волатильности) временных рядов и необходимости адаптации к этим изменениям исследуется подход «адаптивного» взвешенного усреднения результатов прогноза каждой моделей в отдельности. Идея этого подхода заключается в постоянном пересчете ошибок моделей по мере поступления новых данных. На основании полученных значений можно подобрать весовые коэффициенты так, чтобы больший вклад в итоговый результат вносила более точная на текущем интервале временного ряда

Статья получена 24 марта 2023.
Волошин Тарас Андреевич, Национальный Исследовательский Ядерный Университет МИФИ, магистрант, tvoloshin38@gmail.com; Зайцев Константин Сергеевич, Национальный Исследовательский Ядерный Университет МИФИ, профессор, KSZajtsev@mephi.ru; Дунаев Максим Евгеньевич, Национальный Исследовательский Ядерный Университет МИФИ, аспирант, ст. инженер ИИКС, преподаватель кафедры №132, MEDunaev@mephi.ru

модель. Если в какой-то момент времени один из базовых алгоритмов станет предсказывать некорректные значения, модель сможет перестроиться на ходу, уменьшив влияние такого алгоритма на итоговый ответ.

## II. АЛГОРИТМЫ ПРЕДСКАЗАНИЯ ЗНАЧЕНИЙ

В задачах предсказания временных рядов широко распространены ансамблевые алгоритмы машинного обучения [10]. Среди них чаще других применяются, такие как:

a) *случайный лес*. Этот метод используется для объединения в ансамбль нескольких деревьев принятия решений. Каждое дерево строится на своей обучающей подвыборке, которая может включать повторяющиеся примеры, за счёт чего предотвращается переобучение и уменьшается влияние выбросов.

b) *градиентный бустинг*. Этот алгоритм ансамблирования также часто применяется с деревьями решений, но, что характерно, обучение моделей происходит последовательно. Каждая следующая модель обучается на ошибках предыдущей итерации ансамбля, поэтому добавление ее предсказаний уменьшает общую ошибку ансамбля.

Кроме этого, для предсказания временных рядов могут применяться нейронные сети.

a) *сверточные нейронные сети (CNN)*. Этот тип нейронных сетей особенно популярен в задачах компьютерного зрения и обработки изображений [11]. Но с таким же успехом CNN могут быть применены к временным рядам, если в качестве входных данных использовать не двумерные изображения, а одномерные последовательности значений ряда. Ключевую роль в архитектуре CNN выполняют сверточные слои, в которых производится математическая операция свертки. Матрица оптимизируемых параметров такого слоя, называемая ядром, поэлементно умножается на фрагменты матрицы входных данных, результат суммируется и записывается матрицу выходных данных. Такая операция позволяет выделять паттерны, которые могут возникать в любых местах обрабатываемых изображений или временных рядов. Также в архитектуру сверточных сетей обычно входят слои пулинга, служащие для снижения размерности данных и предотвращения переобучения, и полносвязные слои для получения итогового предсказания.

b) *рекуррентные нейронные сети (RNN)*. Такие нейронные сети специально предназначены для обработки упорядоченных данных. Все элементы входных данных последовательно проходят через рекуррентные слои, которые при этом способны запоминать предыдущее состояние и передавать его вместе со следующим входным элементом. Благодаря этой внутренней памяти RNN могут быть эффективны в задачах предсказания временных рядов [12]. Простые рекуррентные сети зачастую подвержены проблеме затухающего градиента, и потому редко применятся в чистом виде. Более популярной является модификация сетей с долгой краткосрочной памятью (Long short-term memory; LSTM) [13]. В них состояние ячейки, передаваемое на следующий шаг, контролируется тремя фильтрами (gates): входным (“input gate”), определяющим какую новую информацию запомнить, фильтром забывания (“forget gate”), служащим для удаления части информации, и выходным (“output gate”), определяющим следующее значение состояния.

## III. ПОДГОТОВКА ДАННЫХ

Для проведения исследований с моделями был сгенерирован набор данных, представляющий собой временные ряды метрик. В качестве сервиса, для которого осуществлялся мониторинг показателей, был взят брокер сообщений с открытым исходным кодом Apache Kafka. Производителем и потребителем сообщений было приложение на языке Java. Оно с определенной периодичностью отправляло сообщения различной длины в заранее заданную тему (topic) Kafka.

Для отслеживания метрик запущенного сервиса Kafka была использована утилита JMX Exporter, которая снимала показатели и экспортировала их в мониторинговый сервис Prometheus, после чего данные можно было сохранить в базу данных или в файл, и тестировать на них различные прогнозирующие модели.

Для проведения экспериментов была взята метрика kafka_consumer_fetch_manager_bytes_consumed_rate, описывающая среднее количество байтов, получаемых потребителем в секунду, как одна из метрик, напрямую связанных с количеством и частотой отправляемых сообщений. На рис. 1 приведен график тренировочной выборки данных временного ряда этой метрики.

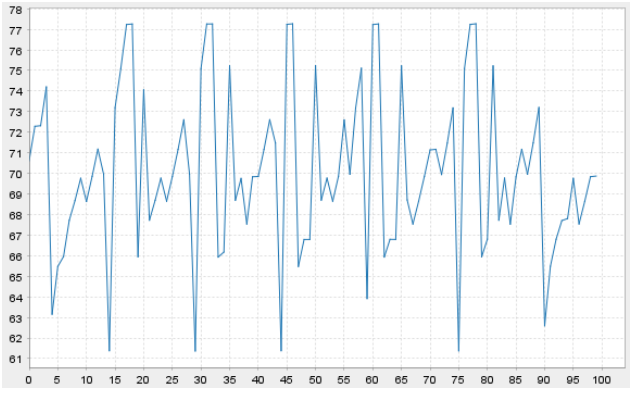


Рисунок 1. График тренировочной выборки.

## IV. ОБРАБОТКА ДАННЫХ

Для применения универсальных методов решения задач регрессии к предсказыванию временных рядов можно поставить задачу определения значения ряда в следующий шаг по времени как задачу обучения с учителем, где тренировочная выборка представляет собой матрицу признаков и вектор значений целевой переменной [14].

В случае временных рядов целевой переменной является очередное значение, а ее признаковым описанием – множество значений ряда в $n$ предыдущих моментах времени. Тогда если временной ряд представляет собой последовательность $S = (y_1, \dots, y_N)$, то для применения алгоритмов машинного обучения необходимо построить матрицу входных данных размерностью $[(N - n - 1) \times n]$ вида

$$X = \begin{bmatrix} y_{N-1} & y_{N-2} & \cdots & y_{N-n-1} \\ y_{N-2} & y_{N-3} & \cdots & y_{N-n-2} \\ \vdots & \vdots & \vdots & \vdots \\ y_n & y_{n-1} & \cdots & y_1 \end{bmatrix}$$

И вектор выходных данных размерностью $[(N - n - 1) \times 1]$:

$$Y = \begin{bmatrix} y_N \\ y_{N-1} \\ \vdots \\ y_{n+1} \end{bmatrix}$$

С практической точки зрения более интересной является возможность предсказывания на более продолжительные интервалы, то есть на некоторое число временных шагов вперед. Простой, но широко распространенной и успешно применяемой, техникой для решения такой задачи является рекурсивная стратегия прогнозирования.

После обучения на вышеописанных данных предсказывающей модели $\hat{f}$ первое предсказание может быть получено из уже имеющихся данных:

$$\hat{y}_{N+1} = \hat{f}(y_N, y_{N-1}, \dots, y_{N-n})$$

Полученное значение может быть использовано для дальнейшего прогнозирования, и далее снова можно будет применять предсказания для построения следующего вектора признаков:

$$\hat{y}_{N+2} = \hat{f}(\hat{y}_{N+1}, y_N, \dots, y_{N-n+1})$$

$$\hat{y}_{N+3} = \hat{f}(\hat{y}_{N+2}, \hat{y}_{N+1}, \dots, y_{N-n+2})$$

...

## V. МЕТОДЫ АНСАМБЛИРОВАНИЯ МОДЕЛЕЙ

Для объединения концептуально отличающихся регрессионных моделей с целью улучшения общих показателей могут применяться разные техники ансамблирования, одна из таких техник – голосование [15]. Предположим, есть набор предсказывающих моделей $\{\hat{f}_i,\ i = 1..n\}$. Тогда в случае простого голосования итоговым ответом ансамбля будет среднее арифметическое решений всех моделей:

$$\hat{f}_f = \frac{1}{n}\sum_{i=1}^{n} \hat{f}_i$$

Однако, такой ансамбль придает одинаковое значение всем моделям, что не всегда может быть эффективно, поэтому более обобщающим методом является взвешенное голосование. При таком подходе каждому алгоритму назначается вес $\alpha_i$, с которым его ответ будет входить в итоговую сумму:

$$\hat{f}_f = \sum_{i=1}^{n} \alpha_i \cdot \hat{f}_i$$

При этом сами веса нормируются на единицу:

$$\sum_{i=1}^{n} \alpha_i = 1$$

Один из вариантов подбора весов – обратно пропорционально ошибке соответствующей модели. Такой метод хорошо подходит к задаче исследования временных рядов, особенно если анализ происходит в потоковом режиме. При появлении новых данных можно будет ретроспективно высчитывать ошибку предсказаний каждой модели и обновлять веса для прогноза на следующий промежуток времени, пока снова не будут доступны реальные значения.

С учетом условия нормировки коэффициентов на единицу веса могут быть рассчитаны по формуле:

$$\alpha_i = \frac{(\varepsilon_i)^{-1}}{\sum_{i=1}^{n}(\varepsilon_i)^{-1}},$$

Где $\varepsilon_i$ – ошибка i-ой модели. В данной работе для пересчета весов была использована метрика корня из среднего квадрата ошибки:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$$

С таким подходом присутствует проблема холодного старта. После изначального обучения моделей на тестовой выборке не остается данных для расчета начальных весов. В работе были рассмотрены два подхода к их определению:

1) равные веса для всех моделей, то есть простое усреднение ответов;
2) предварительный процесс обучения на части тестовой выборки и определение ошибок на остатке, после чего модель снова обучалась уже на всех тестовых данных.

## VI. СРАВНЕНИЕ РЕЗУЛЬТАТОВ

Исследования ансамблевых алгоритмов проводились при прогнозировании на всю тестовую выборку по рекурсивной стратегии использования собственных предсказаний. Графики результатов приведены на рис. 2. В качестве метрик оценки эффективности и сравнения взяты корень из среднего квадрата ошибки и средняя абсолютная ошибка. Полученные значения метрик представлены в табл. 1. Как видно, в данной задаче более эффективной оказалась модель случайного леса деревьев решений, а простое усреднение результатов только увеличило значение ошибок.

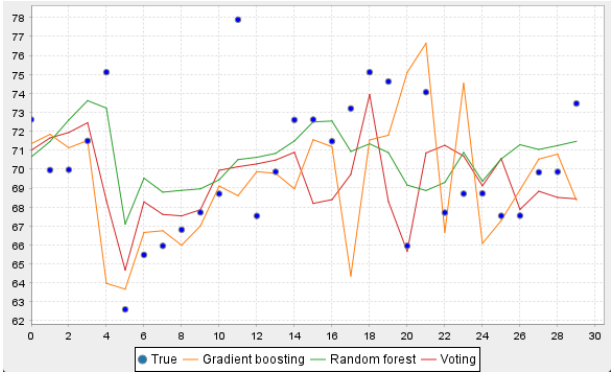


Рисунок 2. Графики предсказаний ансамблевых алгоритмов.

Таблица 1. Метрики предсказаний ансамблевых алгоритмов.

| | RMSE | MAE |
|---|---|---|
| Градиентный бустинг | 4.09 | 2.73 |
| Случайный лес | 2.88 | 2.44 |
| Голосование моделей | 3.11 | 2.44 |

При исследовании был проведен ряд экспериментов для испытания модели адаптивного взвешенного усреднения. Поскольку такая модель подразумевает потоковый вход данных, методика тестирования здесь подразумевала постепенное поступление пакетов (batch) тестовых точек. Поэтому прогнозирование осуществлялось не на всю тестовую выборку, а интервалами по 10 точек, после которых становились доступными истинные значения, а значит, мог быть произведен расчет метрик и весовых коэффициентов. Базовой (baseline) моделью в данном эксперименте была модель простого усреднения: веса были равными и не пересчитывались после каждого интервала прогнозирования. Результаты испытания приведены на рис. 3 и в табл. 2. Две тестируемые модели адаптивного взвешенного усреднения отличаются способом решения проблемы холодного старта. В одной (equal init) начальные веса берутся равными, в другой (pretrain init) – рассчитываются на отложенной части тренировочной выборки. Как видно из результатов эксперимента, применение адаптивной модели улучшает показатели метрик, а расчет начальных весов на тренировочных данных позволяет еще больше уменьшить средний квадрат ошибки, так как на основе именно этой метрики происходит расчет.

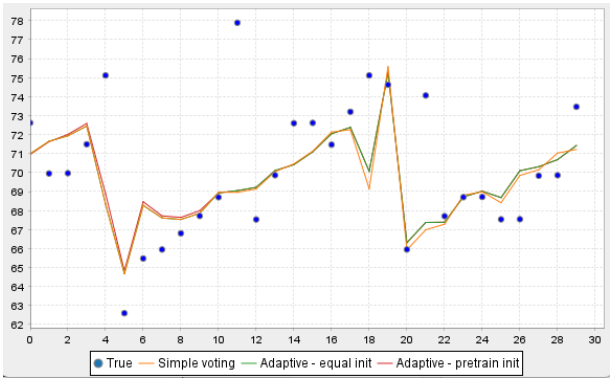


Рисунок 3. Результаты предсказаний модели адаптивного взвешенного усреднения в сравнении с простым голосованием.

Таблица 2. Метрики предсказаний модели адаптивного взвешенного усреднения в сравнении с простым голосованием.

| | RMSE | MAE |
|---|---|---|
| Простое голосование | 2.95 | 1.95 |
| Адаптивная модель – равные начальные веса | 2.85 | 1.90 |
| Адаптивная модель – начальные веса из предобучения | 2.83 | 1.91 |

Далее было проведено сравнение нейронных сетей с вышеописанными ансамблевыми алгоритмами, объединенными в модель простого голосования. В этом эксперименте прогнозирование осуществлялось на один временной шаг при каждом предсказании на векторе признаков из истинных исторических значений ряда. Из результатов на рис. 4 и из табл. 3 видно, что модель усреднения предсказываний градиентного бустинга и случайного леса оказывается эффективнее, чем и сверточная, и рекуррентная сети.

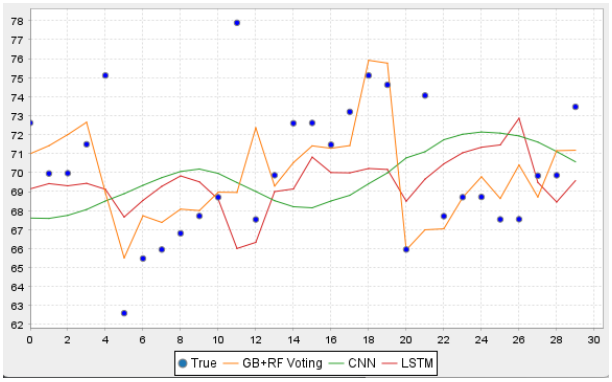


Рисунок 4. Результаты предсказаний нейронных сетей в сравнении с ансамблевыми методами.

Таблица 3. Метрики предсказаний нейронных сетей в сравнении с ансамблевыми методами.

| | RMSE | MAE |
|---|---|---|
| Голосование ансамблевых методов | 2.88 | 1.99 |
| Сверточная сеть | 4.08 | 3.74 |
| Рекуррентная сеть | 3.80 | 3.06 |

## VII. ДИСКУССИЯ ПО ТЕМЕ ИССЛЕДОВАНИЙ

К настоящему моменту такие статистические модели как ARIMA, GARCH уже являются тщательно изученными классическими методами. Последнее время более широкое распространение получают нелинейные методы машинного обучения, такие как ансамбли деревьев решений или нейронные сети [16]. Так в [17] показано, что сети могут оказаться гораздо эффективнее в анализе сезонности и нерегулярностей в данных. При сопоставлении нейронных сетей с ансамблевыми методами, в частности, с градиентным бустингом в реализациях XGBoost [18], LGBM [19], последние показывают сравнимые, а иногда и превосходящие в плане точности прогнозирования результаты, при этом будучи менее требовательными к вычислительным ресурсам и затрачивая меньше времени на обучение и предсказания. А когда речь заходит о прогнозировании временных рядов в потоковом режиме, ресурсоемкость становится особенно значимым фактором.

В этом исследовании градиентный бустинг, а также другой ансамблевый алгоритм решающих деревьев – случайный лес, действительно показали более удовлетворительные результаты по сравнению со сверточными и рекуррентными нейронными сетями, при этом требуя меньше времени на обучение. Стоит отметить также, что возможным ограничением для применения сетей был небольшой размер тренировочной выборки данных, а возможно и сама их специфика (волатильность, периодичность).

Помимо этого, в разных исследованиях поднимается тема онлайнового машинного обучения (online learning) [20], при котором модели можно дообучать на ходу, используя поступающие потоковые данные. Такой подход может помочь решить проблему требования больших ресурсов для обучения алгоритмов с нуля, однако далеко не все методы могут быть адаптированы соответствующим образом. Так, например, показывающие хорошие результаты прогнозирования методы бустинга в своих классических реализациях предназначены для обучения на всех имеющихся данных сразу (offline) [21].

Как видно из результатов этой работы, одним из возможных направлений исследования в области прогнозирования “в потоке” может быть техника адаптивного ансамблирования нескольких базовых моделей. Она показала некоторое улучшение метрики качества, которая была выбрана целевой (RMSE), и возможно, это улучшение может быть еще более значимым при обработке больших выборок данных.

## VIII. ЗАКЛЮЧЕНИЕ

В данной работе были исследованы различные методы машинного обучения для прогнозирования временных рядов метрик приложений и сервисов. При этом подразумевалась необходимость в проведении такого мониторинга и получении предсказаний в режиме реального времени.

В качестве датасетов для проведения исследований были использованы метрики брокера сообщений Apache Kafka. Сами сообщения с некоторой периодичностью генерировались в Java-приложении, и отправлялись в определенный топик. На подготовленных таким путем данных были протестированы такие методы машинного обучения, как градиентный бустинг и случайный лес. Для их ансамблирования была разработана модель адаптивного взвешенного усреднения, которая позволяет получать средние результаты алгоритмов в зависимости от вносимого в итоговый результат вклада, обратно пропорционального ошибкам соответствующей модели, и при этом постоянно обновлять веса для каждого алгоритма. Это повышает точность прогнозирования. Такой подход позволяет анализировать потоковые данные и адаптироваться к их изменениям.

Разработанная модель показала определенные улучшения в сравнении с базовыми алгоритмами, предсказания которых усреднялись с весовыми коэффициентами. Получилось добиться улучшения целевой метрики – среднего квадрата ошибки, на основе которой как раз и пересчитывались веса отдельных алгоритмов. Стоит все же отметить, что улучшения нельзя назвать значительными, однако эффективность такого подхода может быть более выраженной при работе с более объемными массивами данных.

Также в работе были рассмотрены нейронные сети для решения задачи прогнозирования метрик. В частности, рассматривались сверточные сети, способные к распознаванию паттернов в пространственных данных, и потому применимые к временным рядам, представляющим собой одномерную структуру, и рекуррентные сети, непосредственно предназначенные для решения задач с данными, имеющими определенную последовательность.

Оказалось, что в проведенном исследовании ансамблевые методы оказались более эффективными для предсказания временных рядов метрик сервисов, чем нейронные сети. Причиной этого может быть то, что ансамблевые методы позволяют учитывать более широкий спектр

факторов, влияющих на динамику метрик, в том числе различные корреляции и сезонности, а также справляются с выбросами и аномалиями данных. В то время как нейронные сети часто требуют более тонкой настройки параметров и обучения на больших объемах данных. Однако, необходимо отметить, что результаты могут различаться в зависимости от конкретной задачи и корпусов обрабатываемых данных, поэтому в каждом случае следует проводить тщательный анализ и выбирать подходящий метод прогнозирования с учетом конкретных требований и условий.

В итоге можно сказать, что предложенные методы, могут быть использованы в реальных системах мониторинга и прогнозирования метрик приложений и сервисов, а рассмотренная модель адаптивного взвешенного усреднения может позволить добиться еще более точных результатов при обработке данных в потоковом режиме.

## БЛАГОДАРНОСТИ

Авторы выражают благодарность Высшей инжиниринговой школе НИЯУ МИФИ за помощь в возможности опубликовать результаты выполненной работы.

## БИБЛИОГРАФИЯ

1. Chyon, F. A., Suman, M. N. H., Fahim, M. R. I., & Ahmmed, M. S. (2022). Time series analysis and predicting COVID-19 affected patients by ARIMA model using machine learning. Journal of virological methods, 301, 114433.
2. McClymont, H., Si, X., & Hu, W. (2023). Using weather factors and google data to predict COVID-19 transmission in Melbourne, Australia: A time-series predictive model. Heliyon, 9(3), e13782.
3. Seong, Byeongchan. (2020). Smoothing and forecasting mixed-frequency time series with vector exponential smoothing models. Economic Modelling. 91.
4. Necati Aksoy, Istemihan Genc. (2023). Predictive models development using gradient boosting based methods for solar power plants. Journal of Computational Science, 67, 101958
5. Ibrahim, Ahmed & Kashef, Rasha & Corrigan, Liam. (2021). Predicting market movement direction for bitcoin: A comparison of time series modeling methods. Computers & Electrical Engineering. 89. 106905.
6. Dercole, Fabio & Sangiorgio, Matteo & Schmirander, Yunus. (2020). An empirical assessment of the universality of ANNs to predict oscillatory time series. IFAC-PapersOnLine. 53. 1255-1260.
7. Cantero-Chinchilla, Sergio & Simpson, Chris & Ballisat, Alexander & Croxford, Anthony & Wilcox, Paul. (2022). Convolutional neural networks for ultrasound corrosion profile time series regression. NDT & E International. 133. 102756.
8. Alassafi, Madini & Jarrah, Mu'tasem & Al-Otaibi, Reem. (2021). Time Series Predicting of COVID-19 based on Deep Learning. Neurocomputing. 468.
9. Tessoni, V., Amoretti, M. (2022). Advanced statistical and machine learning methods for multi-step multivariate time series forecasting in predictive maintenance. Procedia Computer Science. 200. 748–757.
10. Zhang, Lingyu & Bian, Wenjie & Qu, Wenyi & Tuo, Liheng & Wang, Yunhai. (2021). Time series forecast of sales volume based on XGBoost. Journal of Physics: Conference Series. 1873.
11. Wibawa, A.P., Utama, A.B.P., Elmunsyah, H. et al. Time-series analysis with smoothed Convolutional Neural Network. J Big Data 9, 44 (2022).
12. Sako, K., Mpinda, B. N., & Rodrigues, P. C. (2022). Neural Networks for Financial Time Series Forecasting. Entropy (Basel, Switzerland), 24(5), 657.
13. Rajagukguk RA, Ramadhan RAA, Lee H-J. A Review on Deep Learning Models for Forecasting Time Series Data of Solar Irradiance and Photovoltaic Power. Energies. 2020; 13(24):6623.
14. Bontempi, Gianluca & Ben Taieb, Souhaib & Le Borgne, Yann-Aël. (2013). Machine Learning Strategies for Time Series Forecasting.
15. Shikun, Chen & Luc, Nguyen. (2022). RRMSE Voting Regressor: A weighting function based improvement to ensemble regression.
16. Yajiao Tang & Zhenyu Song & Yulin Zhu & Huaiyu Yuan & Maozhang Hou & Junkai Ji & Cheng Tang & Jianqiang Li. (2022). A survey on machine learning models for financial time series forecasting. Neurocomputing, 512, 363-380.
17. Aggarwal, Akarsh & Alshehri, Mohammed & Kumar, Manoj & Alfarraj, Osama & Sharma, Purushottam & Pardasani, Kamal. (2020). Landslide data analysis using various time-series forecasting models. Computers & Electrical Engineering. 88. 106858.
18. Runge, Jason & Saloux, Etienne. (2023). A comparison of prediction and forecasting artificial intelligence models to estimate the future energy demand in a district heating system. Energy. 269. 126661.
19. Hewamalage, Hansika & Bergmeir, Christoph & Bandara, Kasun. (2020). Global Models for Time Series Forecasting: A Simulation Study. Pattern Recognition. 124. 108441.
20. Hoi, Steven & Sahoo, Doyen & Lu, Jing & Zhao, Peilin. (2018). Online Learning: A Comprehensive Survey. Neurocomputing.
21. Alina Beygelzimer, Elad Hazan, Satyen Kale, and Haipeng Luo. (2015). Online gradient boosting.

# Application of adaptive ensembles of machine learning methods to the problem of time series forecasting

T.A. Voloshin, K.S. Zaytsev, M.E. Dunaev

**Abstract - The purpose of this work is to study the process of applying adaptive ensembles of machine learning methods to the problem of time series forecasting, mainly for streaming data processing. For this, experiments with different types of regression algorithms were carried out, both classical methods for ensembling decision trees (random forest, gradient boosting) and neural networks, in particular convolutional and recurrent ones, were studied. For research, time series of application performance metrics, generated with the Apache Kafka service, were used. To apply machine learning algorithms to the trained data, the series prediction problem was converted to a general supervised learning problem using the sliding window technique. During the experiments, predictions were made for a long interval, for this a recursive forecasting strategy was applied. To improve the results of analysis in streaming mode, an ensemble model was proposed, which, as new data became available, based on errors, recalculate the weights that determine the contribution made to the final prediction by each individual algorithm. The possibility of initializing these weights by pre-training the model on a part of the training set was explored. During the experiments, ensemble methods showed good results in terms of accuracy, and the proposed adaptive weighted averaging technique really turned out to be able to improve the forecasting efficiency. The results of using neural networks were less impressive, which may be due to the use of a relatively small sample of data for training.**

***Keywords* — time series, methods ensembles, adaptive approach, boosting, random forest**

## REFERENCES

1. Chyon, F. A., Suman, M. N. H., Fahim, M. R. I., & Ahmmed, M. S. (2022). Time series analysis and predicting COVID-19 affected patients by ARIMA model using machine learning. Journal of virological methods, 301, 114433.
2. McClymont, H., Si, X., & Hu, W. (2023). Using weather factors and google data to predict COVID-19 transmission in Melbourne, Australia: A time-series predictive model. Heliyon, 9(3), e13782.
3. Seong, Byeongchan. (2020). Smoothing and forecasting mixed-frequency time series with vector exponential smoothing models. Economic Modelling. 91.
4. Necati Aksoy, Istemihan Genc. (2023). Predictive models development using gradient boosting based methods for solar power plants. Journal of Computational Science, 67, 101958
5. Ibrahim, Ahmed & Kashef, Rasha & Corrigan, Liam. (2021). Predicting market movement direction for bitcoin: A comparison of time series modeling methods. Computers & Electrical Engineering. 89. 106905.
6. Dercole, Fabio & Sangiorgio, Matteo & Schmirander, Yunus. (2020). An empirical assessment of the universality of ANNs to predict oscillatory time series. IFAC-PapersOnLine. 53. 1255-1260.
7. Cantero-Chinchilla, Sergio & Simpson, Chris & Ballisat, Alexander & Croxford, Anthony & Wilcox, Paul. (2022). Convolutional neural networks for ultrasound corrosion profile time series regression. NDT & E International. 133. 102756.
8. Alassafi, Madini & Jarrah, Mu'tasem & Al-Otaibi, Reem. (2021). Time Series Predicting of COVID-19 based on Deep Learning. Neurocomputing. 468.
9. Tessoni, V., Amoretti, M. (2022). Advanced statistical and machine learning methods for multi-step multivariate time series forecasting in predictive maintenance. Procedia Computer Science. 200. 748–757.
10. Zhang, Lingyu & Bian, Wenjie & Qu, Wenyi & Tuo, Liheng & Wang, Yunhai. (2021). Time series forecast of sales volume based on XGBoost. Journal of Physics: Conference Series. 1873.
11. Wibawa, A.P., Utama, A.B.P., Elmunsyah, H. et al. Time-series analysis with smoothed Convolutional Neural Network. J Big Data 9, 44 (2022).
12. Sako, K., Mpinda, B. N., & Rodrigues, P. C. (2022). Neural Networks for Financial Time Series Forecasting. Entropy (Basel, Switzerland), 24(5), 657.
13. Rajagukguk RA, Ramadhan RAA, Lee H-J. A Review on Deep Learning Models for Forecasting Time Series Data of Solar Irradiance and Photovoltaic Power. Energies. 2020; 13(24):6623.
14. Bontempi, Gianluca & Ben Taieb, Souhaib & Le Borgne, Yann-Aël. (2013). Machine Learning Strategies for Time Series Forecasting.
15. Shikun, Chen & Luc, Nguyen. (2022). RRMSE Voting Regressor: A weighting function based improvement to ensemble regression.
16. Yajiao Tang & Zhenyu Song & Yulin Zhu & Huaiyu Yuan & Maozhang Hou & Junkai Ji & Cheng Tang & Jianqiang Li. (2022). A survey on machine learning models for financial time series forecasting. Neurocomputing, 512, 363-380.
17. Aggarwal, Akarsh & Alshehri, Mohammed & Kumar, Manoj & Alfarraj, Osama & Sharma, Purushottam & Pardasani, Kamal. (2020). Landslide data analysis using various time-series forecasting models. Computers & Electrical Engineering. 88. 106858.
18. Runge, Jason & Saloux, Etienne. (2023). A comparison of prediction and forecasting artificial intelligence models to estimate the future energy demand in a district heating system. Energy. 269. 126661.
19. Hewamalage, Hansika & Bergmeir, Christoph & Bandara, Kasun. (2020). Global Models for Time Series Forecasting: A Simulation Study. Pattern Recognition. 124. 108441.
20. Hoi, Steven & Sahoo, Doyen & Lu, Jing & Zhao, Peilin. (2018). Online Learning: A Comprehensive Survey. Neurocomputing.
21. Alina Beygelzimer, Elad Hazan, Satyen Kale, and Haipeng Luo. (2015). Online gradient boosting.