International Scientific Conference 20 UNITECH 26

INTERNATIONAL SCIENTIFIC CONFERENCE
19-21 November 2026
GABROVO

РЕАЛИЗАЦИЯ И ОЦЕНКА НА МОДУЛНА ПРОЦЕДУРА ЗА ПРОГНОЗИРАНЕ НА ИКОНОМИЧЕСКИ ПОКАЗАТЕЛИ

Десислава Д. Ангелова

Технически университет – Варна, ул. „Студентска“ 1, Варна, България
e-mail: d.zhejnova@tu-varna.bg

IMPLEMENTATION AND EVALUATION OF A MODULAR FORECASTING PROCEDURE FOR ECONOMIC INDICATORS

Desislava D. Angelova

Technical University of Varna, 1 Studentska Str., Varna, Bulgaria
e-mail: d.zhejnova@tu-varna.bg

Abstract

Economic forecasting requires consistent handling of observation frequencies, publication delays and sequential updates. A modular MATLAB implementation integrates data alignment, feature construction, forecasting, evaluation and storage. ARIMA, a local-level Kalman model and gradient boosting provide forecasts through common interfaces. Chronological training and training-based imputation prevent the use of future target values. Applications to weekly volatility and monthly inflation and production growth demonstrate operation across frequencies. Forecasts and execution states are stored to support traceability and resumption after interruption. Approximate publication delays and short monthly test samples restrict the accuracy assessment.

Keywords: economic forecasting; modular design; time series; MATLAB; reproducibility.

ВЪВЕДЕНИЕ

При прогнозирането на икономически показатели моделите се обучават последователно, като във всеки момент се използва само информацията, налична дотогава. Входните времеви редове може да се различават както по честота, така и по срок на публикуване, а стойностите за минали периоди понякога се ревизират. Ако в обучението попаднат данни, публикувани след момента на прогнозата, оценката на точността се изкривява [1].

За един и същ показател могат да се прилагат различни прогнозни модели. Линейните модели на времеви редове и моделите в пространството на състоянията описват времевите зависимости, докато методите за машинно обучение могат да улавят и нелинейни връзки между входните променливи и прогнозирания показател [2]. За да работят в обща процедура, моделите трябва да използват съгласувани входни данни и формат на прогнозата, а грешките им да се актуализират в правилния момент.

Изборът или комбинирането на модели може да се основава на входните променливи и на резултатите от вече направени прогнози [3, 4]. При реализацията трябва да се предвидят и липсващи данни, недостатъчна обучаваща извадка и невалидни прогнозни стойности. Необходимо е междинните резултати да се съхраняват така, че изчисленията да могат да продължат след прекъсване. Същите изисквания важат както за седмични, така и за месечни данни.

Библиотеките Darts и aeon предлагат общи интерфейси за работа с времеви редове. Darts позволява обучението на

Print ISSN: 1313-230X, Online ISSN: 3033-1404

различни модели и изчисляването на прогнози чрез унифицирани операции. В aeon са включени алгоритми за анализ на времеви редове и преобразуване на данните [5, 6].

В тази работа е разработена модулна реализация в MATLAB за последователно прогнозиране с икономически данни с различна честота и срокове на публикуване. Данните се подравняват по време, след което се подготвят входните променливи за три прогнозни модела с общ интерфейс за обучение и прогнозиране. Отделен модул избира една от прогнозите или ги комбинира. Проверяват се валидността на входните данни и на получените прогнози, а междинните резултати и състоянието на изчисленията се записват периодично, за да може работата да продължи след прекъсване. Реализацията е приложена към три икономически показателя със седмична и месечна честота. Изчислени са прогнозните грешки, но не е провеждан отделен експеримент за измерване на влиянието на модулната организация върху точността.

ПОСТАНОВКА НА ПРОГНОЗНАТА ЗАДАЧА

При изготвянето на всяка прогноза се използват само данни, които вече са били достъпни към съответния момент. Затова за всяка стойност трябва да се разграничават периодът, за който се отнася, и датата, на която е публикувана. Например стойност за даден месец може да стане достъпна едва през следващия. Това разграничение се запазва и при съчетаване на месечни и седмични редове.

Множеството от данни, допустими за прогноза в момент t, е зададено с (1). С $x_j(s)$ е означена стойността на входната променлива j за момент s, а с $a_j(s)$ — моментът, от който тази стойност е налична.

$$\Omega_t = \{\boldsymbol{x}_j(s) : a_j(s) \leq t\} \qquad (1)$$

Прогнозният хоризонт определя колко стъпки напред се изчислява прогнозата. При хоризонт една стъпка напред се прогнозира стойността за следващата седмица или месец според честотата на показателя. При подготовката на входните данни една месечна стойност може да участва в няколко седмични реда, но това не я превръща в няколко отделни месечни наблюдения. Това се отчита при създаването на входните променливи от минали стойности, определянето на размера на обучаващата извадка и изчисляването на прогнозните грешки.

Грешката на прогнозата може да се изчисли едва след като стане известна действителната стойност. Дотогава изборът на модел се основава само на грешките от по-ранни прогнози. След получаване на действителната стойност се определя моделът с най-малка абсолютна грешка; неговият номер служи като клас при обучението на класификатора за следващите прогнозни стъпки [7].

При предварителната обработка също се използва само обучаващата извадка. Ако липсващите стойности се попълват или входните променливи се избират с участие на тестовите данни, точността може да бъде надценена [8]. Затова параметрите на обработката се изчисляват единствено от наличните обучаващи наблюдения.

Модулът за данни подрежда наблюденията и съгласува датите, а модулът за входни променливи подготвя данните за моделите. Всеки прогнозен модул връща една числова стойност. При оценяването прогнозите се сравняват с действителните стойности и се изчисляват грешките. Общият интерфейс позволява даден прогнозен компонент да бъде заменен, без да се променят обработката на датите и записът на резултатите. Точността на моделите се оценява поотделно за всеки показател.

Таблица 1. Изисквания и механизми на реализацията

Изискване	**Механизъм**
Хронологичен ред	Обучаващите редове предхождат целевата стойност на текущата прогноза
Съгласуваност на честотите	Отделни индекси за седмичните и месечните задачи
Заменяемост на моделите	Една числова прогноза от всеки модел
Липсващи входове	Медиани, изчислени от обучаващите наблюдения
Проследимост	Прогнози и записи за решенията по моменти на прогнозиране
Възобновяване след прекъсване	Отделен запис на резултатите и състоянието за базовите прогнози и избора

Таблица 1 свързва основните изисквания с операциите, чрез които са реализирани. Проверяват се както датите на входните данни, така и числовите резултати. Валидната числова стойност сама по себе си не гарантира коректна прогноза: ако при изчисляването ѝ е използвана бъдеща информация, оценката е невалидна. Възможно е и обратното — данните да са подредени правилно, но обучението на модела да не завърши успешно.

МОДУЛНА СТРУКТУРА НА РЕАЛИЗАЦИЯТА

Фигура 1 показва модулите, обмена на данни между тях и обратната връзка, формирана от грешките на предходните прогнози. Получаването на данните и записът на резултатите са отделени от обучението на моделите. Независимо дали задачата е седмична или месечна, се изпълнява една и съща последователност от стъпки.

Данните са от базата FRED [9]. Седмичната пазарна волатилност се получава като средна стойност на дневните наблюдения на индекса VIX. Индексите на потребителските цени и промишленото производство се преобразуват в годишни процентни изменения чрез (2). При месечните задачи се включват само редовете, съответстващи на нова месечна публикация.



Фиг. 1. *Модули, входни и изходни данни и обратна връзка чрез прогнозните грешки*

$$g_t = 100\left(\frac{I_t}{I_{t-12}} - 1\right) \qquad (2)$$

В (2) I_t означава стойността на индекса, а g_t — процентното изменение спрямо същия месец на предходната година.

За входни променливи се използват предходните стойности на прогнозирания показател и изчислени от тях статистики. При седмичните задачи се включват стойности от 1, 2, 4 и 8 наблюдения назад, а при месечните — от 1, 2, 3, 6 и 12 наблюдения назад. Допълнителните икономически променливи се подравняват по общите дати на задачата. Липсващите стойности се заменят с медиа-

ната на съответната колона, изчислена от обучаващата извадка; за текущия входен ред се използват същите медиани. Действителната стойност за прогнозирания период не участва нито в обучението, нито в попълването на липсващи данни.

За ARIMA(1,0,1) и модела на локално ниво с филтър на Калман се използват предходните стойности на прогнозирания показател. LSBoost използва и допълнителни икономически променливи. Всеки от трите модела връща прогноза за следващото наблюдение, а прогнозите се съхраняват в обща матрица. Регресионните дървета позволяват да се моделират нелинейни зависимости при същия формат на изхода [10].

Моделът ARIMA(1,0,1) е зададен с (3). Тук c е константата, φ — авторегресионният коефициент, θ — коефициентът на плъзгащата средна, а ε_t — случайният шум (иновацията).

$$y_t = c + \varphi y_{t-1} + \varepsilon_t + \theta \varepsilon_{t-1} \qquad (3)$$

Моделът на локално ниво е описан със системата (4). Състоянието μ_t се оценява чрез филтъра на Калман, а η_t и ε_t означават съответно шума в състоянието и шума в наблюдението.

$$\mu_t = \mu_{t-1} + \eta_t, \; y_t = \mu_t + \varepsilon_t \qquad (4)$$

При LSBoost прогнозата се получава като сума от началната функция и последователно обучените дървета съгласно (5). F_0 е началната функция, а b_m — m-тото регресионно дърво. Броят на дърветата е M = 75, скоростта на обучение е ν = 0,05, а x означава вектора от входни променливи.

$$F_M(\boldsymbol{x}) = F_0(\boldsymbol{x}) + \nu \sum_{m=1}^{M} b_m(\boldsymbol{x}) \qquad (5)$$

Отделен модул избира една от компонентните прогнози или ги комбинира. Като вход използва прогнозите, грешките от последните оценени наблюдения и признаци за текущите данни. Изборът се извършва чрез ансамбъл от класификационни дървета, всяко от които се обучава върху случайна извадка с възстановяване от обучаващите данни. Ако оценката за избрания клас достигне зададения праг и прогнозата е валидна, тя се използва директно. В противен случай се изчислява претеглена средна на валидните прогнози, като теглата са обратно пропорционални на скорошната MAE. Оценката на класификатора се сравнява с прага, но не се интерпретира като калибрирана вероятност.

Ако ARIMA или моделът на Калман не успее да изчисли прогноза, се използва последната налична стойност. Същото правило важи и когато обучаващата извадка за LSBoost е недостатъчна. Липсващите оценки на скорошната MAE се заменят с медианата на наличните неотрицателни оценки; ако такива няма, теглата се задават равни. Преди изчисляването на обратните стойности се прилага долна граница 10^{-8}. След това теглата на моделите с валидни числови прогнози се нормализират така, че сборът им да е единица. Ако няма валидна прогноза, резултатът се записва като липсваща стойност.

Прогнозите се обединяват чрез претеглена сума. Теглата се изчисляват с (6), а крайната прогноза — с (7). V_t е множеството от моделите с валидна числова прогноза, а $r_{j,t}$ — скорошната MAE след попълване на липсващите оценки. Използва се долна граница $\delta = 10^{-8}$. Ако липсват всички оценки на грешката, теглата на компонентите във V_t се задават равни.

$$w_{j,t} = \frac{1 \,/\, max(r_{j,t}, \delta)}{\sum_{k \in V_t} 1 \,/\, max(r_{k,t}, \delta)} \qquad (6)$$

$$\hat{y}_{t+1|t} = \sum_{j \in V_t} w_{j,t} \hat{y}_{j,t+1|t} \qquad (7)$$

Зависимости (3)–(7) описват изчисленията в прогнозните модули и в модула за комбиниране. Приносът на разработката е в програмното обединяване на тези компоненти, а не във въвеждането

на нови базови прогнозни модели.

Базовите прогнози и решенията за избор или комбиниране се записват отделно. За сравнение се изчисляват и наивна прогноза, равна на последната наблюдавана стойност, и проста средна на трите компонентни прогнози. Статията разглежда програмната реализация; подробна оценка на избора на модел и на чувствителността към прага не е включена.

ПРОГРАМНА РЕАЛИЗАЦИЯ И ПОСЛЕДОВАТЕЛНО ИЗПЪЛНЕНИЕ

Реализацията е разработена с MATLAB R2025b. За иконометричното моделиране е използван Econometrics Toolbox, а за статистическия анализ и машинното обучение — Statistics and Machine Learning Toolbox. При всяка нова прогнозна стъпка обучаващата извадка се разширява с вече наличните наблюдения. LSBoost е конфигуриран със 75 регресионни дървета и скорост на обучение 0,05. Класификаторът също използва 75 дървета, обучавани върху случайни извадки с възстановяване. Минималният размер на обучаващата извадка и прозорецът за изчисляване на скорошната грешка се задават в конфигурацията на задачата.

Функцията buildTaskData подготвя данните за конкретната прогнозна задача от таблица тип timetable — MATLAB структура за наблюдения с времеви отметки, подредени по дата. Последователните стъпки се изпълняват от runSingleTask. Функцията medianImputeTrainAndTest попълва липсващите стойности, като изчислява медианите само от обучаващата извадка. След това LSBoost се обучава чрез fitrensemble, а класификаторът за избор на модел — чрез fitcensemble. При различна честота на прогнозирания показател форматът на изходните прогнози остава един и същ.

Таблица 2 показва броя наблюдения, използвани за обучение и оценяване: седмични при седмичните задачи и месечни при месечните. Прагът за избор

Таблица 2. Конфигурация според честотата, изразена в наблюдения

Настройка	Седмична	Месечна
Минимален брой наблюдения за LSBoost	156	60
Минимален брой наблюдения за класификатора	104	36
Прозорец за скорошни грешки	26	12
Кратък подвижен прозорец	4	3
Дълъг подвижен прозорец	8	12

на отделен прогнозен модел е определен с данни от период преди 2020 г. Началната година е 2018 г. за седмичната задача и 2016 г. за месечните задачи. Проверени са стойности от 0,34 до 0,60 през интервали от 0,02. Избран е прагът с най-ниска MAE, при който отделен модел се използва в поне 20% от прогнозните стъпки, когато това условие е изпълнимо. Получените прагове са 0,40 за волатилността и 0,48 за двете месечни задачи. Те остават фиксирани през оценъчния период. Моделите се обучават отново при постъпване на нови данни, а грешките се добавят след получаване на действителната стойност.

При всяка прогнозна стъпка се определят наличните обучаващи редове, подготвят се входните данни и се изчисляват базовите прогнози. Едва след като те бъдат записани, се извършва изборът или комбинирането им. Класификаторът се обучава само с етикети от прогнози, за които действителните стойности вече са известни. Грешката на текущата прогноза се добавя след получаването на съответната действителна стойност. При обработката на историческите данни този ред се спазва чрез индексите на редовете.

За повторяемост на резултатите генераторът на случайни числа се инициализира със стойност 20260812. При всяко повторно обучение на дърветата се задава начална стойност, определена от про-

гнозната стъпка. Заедно с междинните резултати се съхраняват настройките, идентификаторът на задачата, имената на входните променливи, индексите на прогнозните редове и информацията кои стъпки са завършени. Прогнозите и таблиците с грешки се записват във файлове за последващо зареждане и автоматична обработка.

За да може даден резултат да бъде възпроизведен, трябва да са известни използваните данни и настройки. Тази информация може да се съхранява в структуриран вид и да се обработва автоматично [11]. Разработената реализация записва междинните резултати и част от настройките, но не поддържа пълна история на версиите на всички входни данни.

Междинните резултати и състоянието на изчисленията се записват поотделно за базовите прогнози и решенията на класификатора. Запис се прави на всеки 25 прогнозни стъпки и при приключване. Първо се създава временен MAT файл — форматът на MATLAB за съхраняване на променливи — след което той заменя основния файл. При възобновяване се сравняват настройките, обобщените характеристики на данните, индексите на наблюденията, имената на входните променливи и размерите на масивите. Вече завършените стъпки се пропускат, което ограничава риска да бъдат използвани резултати, получени при несъвместими настройки.

При възобновяване се проверяват избрани настройки и обобщени статистики на прогнозирания времеви ред, но не и всяка отделна входна стойност. Пълната проверка на възпроизводимостта би изисквала фиксирани копия на данните, контролни суми на входните файлове и описание на софтуерната среда. Възобновяването от записано състояние е реализирано, но не е изпитано в отделен експеримент с умишлено прекъсване.

ЕКСПЕРИМЕНТАЛНИ РЕЗУЛТАТИ

Реализацията е приложена към три показателя: седмична пазарна волатилност, месечна инфлация и месечен растеж на промишленото производство. Използвани са редовете VIXCLS, CPIAUCSL и INDPRO [9]. За волатилността се прогнозира нивото на индекса, а за останалите два показателя — годишното процентно изменение. При трите задачи се използват едни и същи програмни интерфейси, но различни входни редове и настройки.

Моделите се обучават отделно за всеки показател. Следователно общата програмна структура не представлява един глобален модел, обучен върху всички времеви редове. Това разграничение съответства на разликата между локалното и глобалното прогнозиране на групи от времеви редове [12].

Резултатите са оценени за периода 2020–2025 г. и за наличните наблюдения през 2026 г. За волатилността са изчислени съответно 313 и 31 прогнози, а за всяка месечна задача — 72 и 6. Данните за 2026 г. обхващат само част от годината. За всяка задача са изчислени средната абсолютна грешка (MAE) и коренът от средната квадратична грешка (RMSE) [13]. В (8) и (9) $e_i = y_i - \hat{y}_i$ означава прогнозната грешка, а N — броя на оценените прогнози. Грешките на различните показатели не се обединяват, тъй като са изразени в различни мерни единици.

Подобно предварително определяне на оценъчните извадки и показателите за грешка се използва и в прогнозни състезания, например M5 [14]. За разглежданите задачи е приложен един и същ ред на обучение и оценяване, а резултатите са представени поотделно за всеки показател.

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |e_i| \tag{8}$$

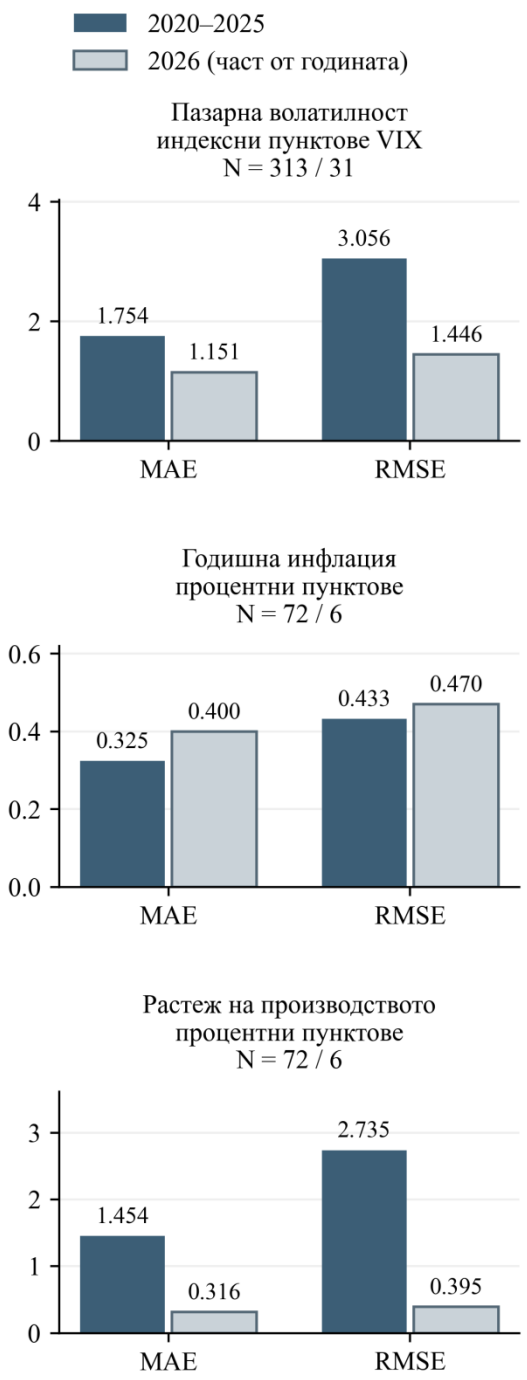
$$\mathrm{RMSE} = \sqrt{2\frac{1}{N}\sum_{i=1}^{N} {e_i}^2} \tag{9}$$

Таблица 3. Прогнозни грешки за разглежданите задачи

Показател и период	N	MAE	RMSE
Волатилност 2020–2025	313	1.7541	3.0562
Волатилност 2026	31	1.1505	1.4456
Инфлация 2020–2025	72	0.3246	0.4328
Инфлация 2026	6	0.3995	0.4697
Растеж на производството 2020–2025	72	1.4541	2.7350
Растеж на производството 2026	6	0.3163	0.3952

Таблица 3 представя грешките на крайните прогнози. При инфлацията и растежа на промишленото производство те са в процентни пункта, а при волатилността — в пунктове на индекса VIX. Резултати са получени и за двете честоти. Самите обобщени стойности обаче не позволяват да се изолира влиянието на модулната организация върху точността.

На фигура 2 RMSE е по-висока от MAE при всяка задача. Това е очаквано, тъй като при RMSE големите грешки имат по-силно влияние. Разликата между двата показателя е най-изразена при волатилността и растежа на промишленото производство през 2020–2025 г. За тези показатели отчетените грешки са по-ниски през наличната част от 2026 г. Периодите и броят наблюдения обаче се различават, затова от сравнението не може да се заключи, че точността се е подобрила.



Фиг. 2. *MAE и RMSE по показатели и периоди. Използвани са отделни скали. Данните са от таблица 3.*

През 2020–2025 г. MAE за волатилността е 1.7541, при 1.7470 за LSBoost. За инфлацията MAE е 0.3246, при 0.3100 за ARIMA. При растежа на промишленото производство MAE е 1.4541 — с 1,03% над тази на простата средна. Следователно изборът и комбинирането на прогнозите не осигуряват най-ниска MAE във всички случаи. Причините за различията не могат да се установят само от обобщените грешки. През наличната част от 2026 г. крайните прогнози за волатилността съвпадат с тези на LSBoost.

Месечната оценка включва 72 прогнози за 2020–2025 г. и шест налични прогнози за 2026 г.; стойностите за последния период са представени като допълнителна проверка. При историческата оценка са използвани наличните ревизирани данни и приблизителни срокове на публикуване. Пълното възстановяване на информацията, налична във всеки минал момент, не е обект на настоящото изследване.

ЗАКЛЮЧЕНИЕ

Разработена е модулна реализация в MATLAB, която обединява подготовката на икономическите данни, обучението на прогнозни модели, избора или комбинирането на прогнозите и оценката на грешките. Общите интерфейси позволяват статистически модели и регресионни дървета да участват в една изчислителна процедура. При всяка прогнозна стъпка се проверява наличността на входните данни, а грешките се добавят едва след получаване на действителните стойности.

Реализацията е приложена към три икономически показателя със седмична и месечна честота. За всеки от тях са получени прогнози и оценки на грешките. Благодарение на общите интерфейси прогнозните модели могат да се добавят или заменят, без да се променя останалата структура. Реализирано е и съхраняване на междинните резултати и състоянието на изчисленията, което позволява изпълнението да бъде възобновено след прекъсване.

Получените резултати показват, че разработената структура може да се приложи към различни прогнозни задачи. За по-надеждна оценка при месечните показатели е необходим по-дълъг тестов период. В бъдеща проверка могат да бъдат измерени времето за изпълнение и възстановяването след контролирано прекъсване.

ЛИТЕРАТУРА

[1] Petropoulos F. et al. Forecasting: theory and practice. International Journal of Forecasting 2022;38(3):705–871. doi:10.1016/j.ijforecast.2021.11.001.

[2] Goulet Coulombe P., Leroux M., Stevanovic D., Surprenant S. How is machine learning useful for macroeconomic forecasting? Journal of Applied Econometrics 2022;37(5):920–964. doi:10.1002/jae.2910.

[3] Wang X., Hyndman R.J., Li F., Kang Y. Forecast combinations: An over 50-year review. International Journal of Forecasting 2023;39(4):1518–1547. doi:10.1016/j.ijforecast.2022.11.005.

[4] Talagala T.S., Hyndman R.J., Athanasopoulos G. Meta-learning how to forecast time series. Journal of Forecasting 2023;42(6):1476–1501. doi:10.1002/for.2963.

[5] Herzen J. et al. Darts: User-Friendly Modern Machine Learning for Time Series. Journal of Machine Learning Research 2022;23(124):1–6. Available at: https://jmlr.org/papers/v23/21-1177.html (accessed 6 October 2026).

[6] Middlehurst M. et al. aeon: a Python Toolkit for Learning from Time Series. Journal of Machine Learning Research 2024;25(289):1–10. Available at: https://jmlr.org/papers/v25/23-1444.html (accessed 6 October 2026).

[7] Cerqueira V., Torgo L., Mozetič I. Evaluating time series forecasting models: An empirical study on performance estimation methods. Machine Learning 2020;109:1997–2028. doi:10.1007/s10994-020-05910-7.

[8] Kapoor S., Narayanan A. Leakage and the reproducibility crisis in machine-learning-

based science. Patterns 2023;4(9):100804. doi:10.1016/j.patter.2023.100804.
[9] Federal Reserve Bank of St. Louis. FRED Economic Data. Available at: https://fred.stlouisfed.org/ (accessed 5 October 2026).
[10] Januschowski T., Wang Y., Torkkola K., Erkkilä T., Hasson H., Gasthaus J. Forecasting with trees. International Journal of Forecasting 2022;38(4):1473–1481. doi:10.1016/j.ijforecast.2021.10.004.
[11] Wagner A.S. et al. FAIRly big: A framework for computationally reproducible processing of large-scale data. Scientific Data 2022;9:80. doi:10.1038/s41597-022-01163-2.
[12] Montero-Manso P., Hyndman R.J. Principles and algorithms for forecasting groups of time series: Locality and globality. International Journal of Forecasting 2021;37(4):1632–1653. doi:10.1016/j.ijforecast.2021.03.004.
[13] Hyndman R.J., Athanasopoulos G. Forecasting: Principles and Practice. 3rd ed. Melbourne: OTexts, 2021. Available at: https://otexts.com/fpp3/ (accessed 5 October 2026).
[14] Makridakis S., Spiliotis E., Assimakopoulos V. M5 accuracy competition: Results, findings, and conclusions. International Journal of Forecasting 2022;38(4):1346–1364. doi:10.1016/j.ijforecast.2021.11.013.