TSB-FCST: A Billion-Scale Time-Series Forecasting Benchmark with Taxonomy-Specific Evaluation: [Experiments & Analysis]

Anonymous Author(s)

Abstract

Time series, defined as sequences of ordered observations, have become ubiquitous across diverse domains. Among the core tasks in this field, time-series forecasting—using historical patterns to predict future outcomes—has gained significant attention due to its broad practical impact. Accordingly, numerous forecasting methods and evaluation benchmarks have been developed to track and drive progress. Despite decades of advances, existing evaluation frameworks still exhibit critical limitations: (i) many studies rely on homogeneous or imbalanced datasets that lack sufficient diversity or quality assurance across dynamic environments and domains; (ii) most benchmarks focus on only a limited categories of models, frequently neglecting simple yet highly effective baselines; and (iii) existing studies often provide limited statistical and taxonomy-specific analysis, leading to superficial interpretations of model performance. To address these limitations, we introduce TSB-FCST, a *billion-scale* time-series forecasting benchmark curated with rigorous data cleaning and quality assurance for comprehensive and reliable evaluation. Notably, TSB-FCST provides a standardized evaluation of 100 algorithms, ranging from classical statistical and machine learning methods to deep learning-based and recently emerged foundation models. Our extensive evaluation reveals: (i) While foundation models demonstrate strong generalist zero-shot capabilities, they still exhibit "Achilles' heels" in specific data domains such as retail data, supported by our comprehensive domain coverage; (ii) Simple yet highly effective classical and deep learning baselines are frequently overlooked, even though many complex state-of-the-art architectures fail to strongly outperform them; (iii) Relying solely on aggregated metrics is insufficient, as forecasting difficulty and model design efficacy are fundamentally shaped by underlying dataset characteristics; and (iv) Pre-training scaling gains plateau beyond 100B observations, suggesting that future progress requires shifting focus towards tackling task-specific challenges. Finally, we release all source code to ensure reproducibility. In summary, TSB-FCST offers a rigorous roadmap for understanding recent developments and advancing future forecasting research.

ACM Reference Format:

Anonymous Author(s). 2026. TSB-FCST: A Billion-Scale Time-Series Forecasting Benchmark with Taxonomy-Specific Evaluation: [Experiments & Analysis]. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 21 pages. https://doi.org/XXXXXXX.XXXXXXX

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

Table 1: Comparison with existing time-series forecasting benchmarks. TSB-FCST provides (i) the largest data scale and composition, (ii) broader model and metric coverage, and (iii) reliable data quality assurance (e.g., systematic data cleaning process, human inspection and leakage report especially for pretrained foundation model) and evaluation protocol (Hyperparameter Optimization (HPO) set, statistical test, and comprehensive evaluation metrics).

Benchmark	Dataset				Model					Quality Assurance			Eval Protocol		
	#Obs.	UTS	MTS	Tot.	Stat	ML	DL	FM	Tot.	Data Clean	Human Inspect	Leak. Report	HPO Set	Stats. Test	Eval. Met.
Monash [36]	182M	34	16	50	5	2	5	0	12	✗	✗	✗	✗	✗	4
TSLib [126]	30M	0	14	14	0	0	13	0	13	✗	✗	✗	✗	✗	2
TFB [103]	198M	28	25	53	3	3	15	0	21	✗	✗	✗	✗	✗	8
GIFT-Eval [2]	157M	35	20	55	5	0	8	4	17	✗	✗	✓	✗	✗	2
fev-bench [113]	761M	65	35	100	6	0	0	8	14	✗	✗	✓	✓	✗	2
TSFM-bench [54]	40M	0	21	21	0	0	7	14	21	✗	✗	✗	✗	✗	2
TSB-FCST (Ours)	**1B**	**64**	**64**	**128**	**19**	**15**	**47**	**19**	**100**	✓	✓	✓	✓	✓	**8**

1 Introduction

The recent explosion in sensor networks and data acquisition frameworks has led to an unprecedented accumulation of high-fidelity signal data. These sequential measurements, characterized as ordered, real-valued observations, are formally referred to as *time series* in both scientific research and industrial applications [95]. The effective extraction and analysis of such temporal information is crucial for uncovering latent patterns and making strategic decisions in a wide range of real-world applications. Consequently, the field has experienced a substantial expansion in the range and number of downstream tasks, ranging from time-series classification [60, 79, 107, 110] and clustering [90, 91, 96] to anomaly detection [67, 89, 93, 112], similarity discovery [28, 56, 94, 108, 114], and forecasting tasks [4, 62–65, 82, 141–143].

Among these tasks, time-series forecasting—which targets the prediction of future values based on historical observations—remains a fundamental domain with widespread applications. For example, weather forecasting supports planning in agriculture and aviation [11], while demand forecasting helps enterprises align inventory with market needs, reducing operational costs and preventing stock-outs in supply chain [78]. Beyond direct prediction, forecasting also supports higher-level tasks: forecasting models can learn discriminative representations for downstream applications such as clustering [96], and prediction residuals can be used to identify anomalies [67], where large deviations from expected behavior often indicate faults or process shifts. Over the past two decades, forecasting methods have rapidly evolved. Traditional statistical methods, such as ARIMA [11], model temporal dependencies through structured assumptions and often provide efficient training and inference. Machine learning methods, especially tree-based methods, formulate forecasting as supervised regression problems [13]. More recently, deep learning models have advanced forecasting by

capturing nonlinear temporal dynamics through diverse architectures [82, 104, 132], while newly emerged foundation models have demonstrated promising zero-shot capabilities by pre-training from large-scale, cross-domain time-series corpora [3, 24, 61].

In the meantime, substantial efforts have been made to build time-series forecasting benchmarks for comparing models across different scales and domains [2, 36, 54, 103, 113], which have established important foundations for standardized evaluation in the past decade. However, existing practices still exhibit several limitations that hinder a comprehensive understanding of model progress: (i) many studies rely on constrained or imbalanced data collections without quality assurance that do not fully reflect the complexity of multi-domain forecasting scenarios; (ii) some benchmarks focus on only a limited subset of model families, overlooking holistic comparisons between traditional models and newly emerged architectures; and (iii) many evaluations rely primarily on aggregate average metrics, with limited statistical and taxonomy-specific analysis, leading to potentially superficial interpretations of model performance. These limitations collectively obscure the true landscape of modern forecasting models, highlighting the demand for a more balanced and rigorous evaluation standard.

To address these challenges, we introduce `TSB-FCST`, a large-scale benchmark for comprehensive and rigorous evaluation of time-series forecasting models to date. To ensure diverse and heterogeneous data representation, this `TSB-FCST` benchmark incorporates 128 datasets across eight diverse domains (e.g., Energy, Nature, Healthcare, Mobility, Retail, Cloud, and Economics), balanced between 64 univariate (UTS) and 64 multivariate (MTS) datasets with diverse data patterns and temporal granularities (as shown in Figure 1 and Figure 2). To the best of our knowledge, `TSB-FCST` is the first *billion*-scale archive specifically curated for systematic forecasting evaluation. To ensure data quality, we design a multi-stage curation pipeline that combines automated screening with human inspection to identify and correct anomalies, artifacts, and low-quality observations that could otherwise compromise the robustness of model training and evaluation. Additionally, we provide a standardized evaluation of 100 forecasting models, ranging from classical statistical and machine learning methods to modern deep learning architectures and recent emerged foundation models. To facilitate rigorous parameter tuning and prevent underrepresenting model performance, we introduce `TSB-FCST-50M`, a carefully curated Hyperparameter Optimization (HPO) set that maintains a balanced distribution and comprehensive coverage across all eight domains. Finally, as a practical contribution, we also release an integrated library with unified pipelines and open-source implementations of all evaluated methods [1]. Table 1 summarizes the characteristics of `TSB-FCST` against existing benchmarks.

Our comprehensive experimental study across the 128 `TSB-FCST` datasets yields four pivotal insights that complement and deepen our understanding of current paradigms: (i) *The "Achilles' Heels" of Foundation Models*: While foundation models (FMs) demonstrate remarkable generalist and zero-shot capabilities, they are still underperformed by traditional statistical and machine learning methods in specific data domains with localized statistical properties, such as Nature and Retail datasets. This suggests that generic pre-trained representations still struggle with highly specialized, erratic domain-specific structures. (ii) *The Value of Simple yet Effective Baselines*: Simple statistical and machine learning (e.g., *Exponential Smoothing* and *Huber*) and learning-based methods (e.g., simple RNNs) are widely overlooked in previous studies, yet many complex state-of-the-art architectures fail to significantly outperform them. When coupled with proper hyperparameter tuning and straightforward designs like Reversible Instance Normalization (RevIN) [51] to handle distribution shifts, simpler baselines remain exceptionally competitive. (iii) *Taxonomy-specific Evaluation Paradigms*: Relying solely on aggregated metrics is insufficient, as forecasting difficulty and architectural efficacy (e.g., channel dependency) are fundamentally governed by underlying dataset characteristics rather than model complexity alone. Selecting a suitable evaluation dataset pool and following a taxonomy-specific evaluation protocol is key to producing reliable conclusions. (iv) *Pre-training Scaling Limits*: Zero-shot performance gains plateau beyond 100 billion observations, demonstrating distinct diminishing marginal returns to brute-force data scaling. Future progress requires shifting the research focus toward tackling specific temporal patterns and deeply understanding data characteristics, rather than merely chasing aggregated metrics and a hypothetical "champion" on global leaderboards.

We start with the preliminaries and related work in the field of time-series forecasting (Section 2). Then we highlight our contributions of this work, detailed as follows:

- We construct `TSB-FCST`, a 1B-scale time-series forecasting benchmark curated with rigorous cleaning and quality assurance. Notably, it ensures balanced data representation across a total of eight diverse domains to provide an unbiased and comprehensive evaluation platform.
- We evaluate over 100 representative algorithms, spanning traditional statistical and machine learning methods to state-of-the-art deep learning and foundation models.
- We implement a rigorous training and evaluation framework. To ensure both efficiency and reliability, we utilize a dedicated `TSB-FCST-50M` subset for systematic parameter tuning.
- We provide a systematic study uncovering the domain-specific nature of forecasting performance across different categories, establishing a roadmap for future taxonomy-specific research.

Finally, we discuss the conclusions of the experimental benchmark on `TSB-FCST` and potential directions for future time-series forecasting model advancement.

2 Preliminaries and Related Work

In this section, we start with an introduction of notation and concepts, as well as related studies on time-series forecasting models and the limitations of current benchmark studies.

2.1 Notation and Definitions

Definition 1 (Time Series). *Let $\mathcal{X} = \{X_1, X_2, \ldots, X_n\}$ denote a repository of n time-series instances. Each instance $X_i \in \mathbb{R}^{T \times d}$ consists of T temporal observations across d channels.*

Based on the dimensionality of the observation space, a time-series is classified as *univariate* (UTS) if $d = 1$ and *multivariate* (MTS) if $d > 1$. While UTS analysis primarily focuses on capturing temporal dependencies, MTS introduces additional complexity by requiring models to account for inter-channel correlations and high-dimensional dynamics and dependencies.

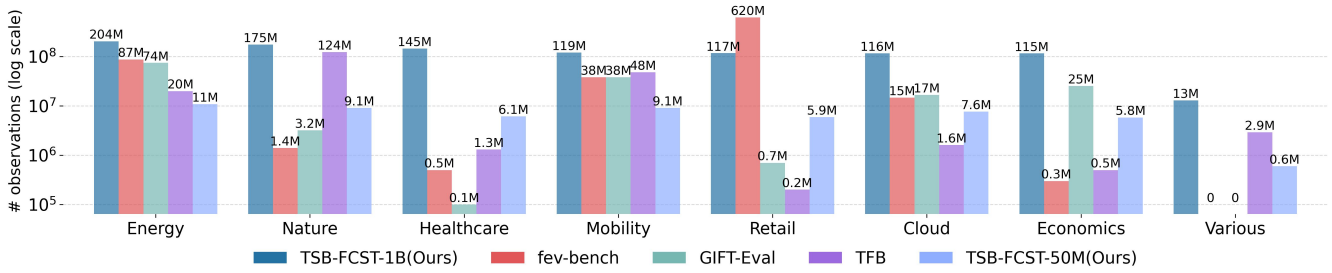

Figure 1: Domain-wise distribution of observations across time series forecasting benchmarks. Our proposed benchmarks demonstrate a more balanced and comprehensive coverage across domains.

Definition 2 (Subsequence). *Given a time-series $X \in \mathbb{R}^{T\times d}$, a subsequence $X_{s,e}$ is defined as a contiguous segment of observations starting from offset s and ending at offset e. Formally, $X_{s,e} = (\vec{x}_s, \vec{x}_{s+1}, \ldots, \vec{x}_e)$, where $1 \le s \le e \le T$. Each $\vec{x}_t \in \mathbb{R}^d$ represents the d-dimensional observation at time t. The length of the subsequence is given by $|X_{s,e}| = e - s + 1$.*

Definition 3 (Sliding Window). *Based on the definition of subsequences, a sliding window of width w is a specific subsequence $X_{i,i+w-1}$. By traversing the series X with a fixed stride s, we extract a collection of such windows: $\mathcal{W} = \{X_{1+(k-1)s,w+(k-1)s}\}_{k=1}^{K}$. Here, $K = \lfloor \frac{T-w}{s} \rfloor + 1$ represents the total number of extracted windows, where each window is a local segment of shape (w, d).*

Definition 4 (Time Series Forecasting). *Given a time-series $X \in \mathbb{R}^{T\times d}$ and a look-back window of length L, the task of forecasting is to learn a mapping $f : \mathbb{R}^{L\times d} \to \mathbb{R}^{H\times d}$ that generates a forecast $\hat{X}_{t+1,t+H} = f(X_{t-L+1,t})$ of a horizon H. The model is optimized to minimize the discrepancy between the prediction and the ground truth $X_{t+1,t+H}$, formally defined by a objective function $\mathcal{L}$:*

$$\mathcal{L} = \psi(X_{t+1,t+H}, \hat{X}_{t+1,t+H})$$

where $\psi(\cdot)$ denotes a distance metric or loss function (e.g., ℓ_1 or ℓ_2 norm) that quantifies the prediction error over the forecast horizon H. In our study, we employ a diverse set of evaluation metrics to provide a comprehensive assessment of this discrepancy (Section 4.4).

In the following, we go through the details of time-series data patterns and related work of time-series forecasting research.

2.2 Time-series Data Patterns

We also characterize several fundamental time-series patterns such as trend, seasonality, stationarity, transition, and correlation. The formal definitions and mathematical formulations are detailed as follows. Representative time-series examples are shown in Figure 2.

Definition 5 (Trend). *The trend of a given time series represents the long-term patterns that occur over time. Formally, a Seasonal-Trend Decomposition Procedure Based on Loess (STL) [20] is used to decompose a time series X into three terms:*

$$X = X_S + X_T + X_R$$

where X_T, X_S, X_R refer to trend, seasonality, and the remainder, respectively. The strength of trend is then computed as follows.

$$Trend = max\left(0, 1 - \frac{var\,(X_R)}{var\,(X - X_S)}\right) \tag{1}$$

Definition 6 (Seasonality). *Seasonality is defined as the recurring pattern in a time series that repeats at regular intervals. Similar to trend, the strength of seasonality in a time series X can be calculated:*

$$Seasonality = \max\left(0, 1 - \frac{var\,(X_R)}{var\,(X - X_T)}\right) \tag{2}$$

Definition 7 (Stationarity). *Stationarity of a time series X is defined by three key properties: the mean of any observation is constant; for all observations, the variance is finite; the distance between any two observations exclusively determines the covariance. Augmented Dick-Fuller (ADF) test statistic [30] is employed to measure stationarity, computed as follows.*

$$Stationarity = \begin{cases} True, & ADF(X) \le 0.05 \\ False, & Otherwise \end{cases} \tag{3}$$

Definition 8 (Correlation). *Correlation measures the degree to which different variables (i.e., channels) within a time-series instance share similar structural patterns. Let $X_i \in \mathbb{R}^{T\times d}$ denote a time-series instance with d variables. For each variable $j \in \{1, \ldots, d\}$, extract its Catch22 [76] feature vector $F_i^j = \text{Catch22}(X_i^{(:,j)})$, and collect all feature vectors from each channel*

$$F_i = [F_i^1, \ldots, F_i^d] \in \mathbb{R}^{22\times d}. \tag{4}$$

We compute all pairwise Pearson correlation coefficients between variables in the formula as:

$$P_i = \{r(F_i^j, F_i^k) \mid 1 \le j < k \le d\}. \tag{5}$$

The correlation of instance X_i is defined as

$$\text{Corr}(X_i) = \text{mean}(P_i) + \frac{1}{1 + \text{var}(P_i)}. \tag{6}$$

Detailed definition and formulations of other time-series patterns such as transition, shifting, ADI, and demand patterns are provided in the Appendix B. More details of feature distribution on TSB-FCST can be found in Figure 5 and Figure 13.

2.3 Related Work

In this section, we briefly review representative forecasting models as well as existing benchmark and evaluation studies that motivate the design of our TSB-FCST benchmark.

Time-Series Forecasting Models. Time series forecasting, as one of the most critical tasks in time series analysis, has been continuously evolving over the past few decades [50]. Traditional time series forecasting methods can broadly be categorized into statistical and machine learning approaches. Statistical models[5, 15, 25, 31, 98,

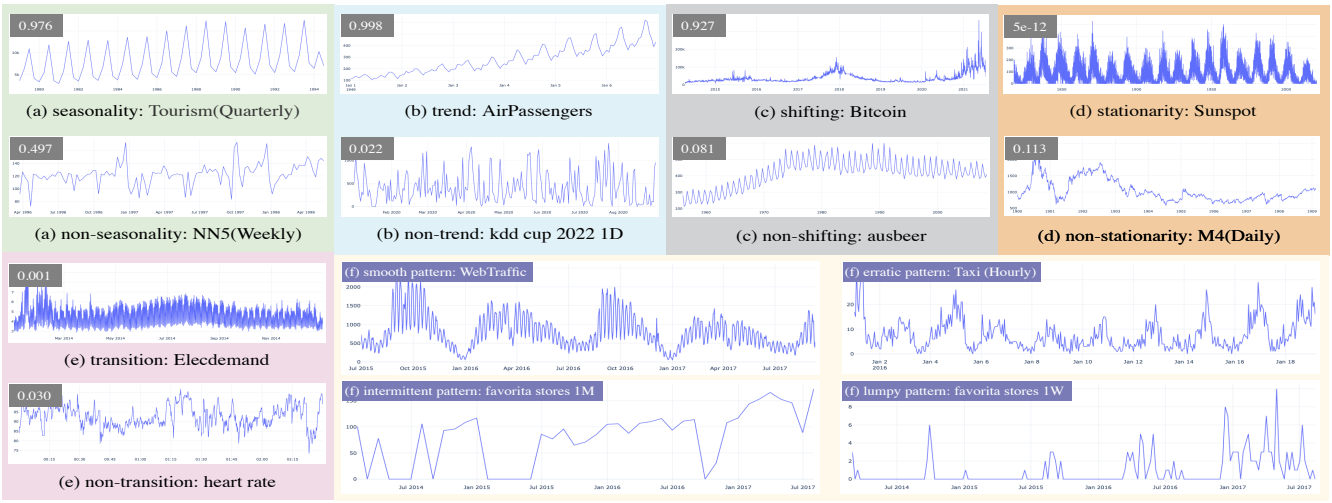

Figure 2: Qualitative examples of common time series patterns. We compare real-world datasets based on a total of 6 time series patterns: seasonality, trend, shifting, stationarity, transition, and 4 different demand patterns.

100] are usually contingent on explicit, structured, and principled probabilistic assumptions; Machine learning methods [9, 12, 18] formulate time series forecasting as a supervised learning problem. More recently, the past decade has witnessed that deep learning method and emergent foundation models have been dominating the majority of breakthrough [24, 41, 48, 55, 70, 82, 109, 115, 127].

Time-Series Forecasting Benchmark and Evaluation Studies. The continual advances in developing new time series forecasting algorithms have been accompanied by a fast-growing community of benchmarking studies [2, 36, 103, 113, 126]. Monash [36], one of the first time series forecasting benchmarks, collects 20 publicly available time series datasets. To account for advances in deep learning architectures [70, 131, 140, 146], Time Series Library (TSLib) [126] evaluates 13 advanced deep learning-based methods. TFB [103] proposes to perform a thorough evaluation of 21 forecasting algorithms spanning statistical learning, machine learning, and deep learning categories. Recently, GIFT-Eval [2] encompasses 44,000 time series and 177 million data points to improve the evaluation comprehensiveness. fev-bench [113] further offers a collection of 46 tasks with covariates, representing the largest curated time-series dataset collection prior to TSB-FCST.

However, we identify several common limitations in these existing benchmarks. Left unaddressed, these issues can lead to biased evaluation results and potentially misleading conclusions regarding overall model efficacy. We go through these details as follows.

3 Common Limitations in Existing Time-Series Forecasting Benchmark

In this section, we summarize common limitations from two perspectives: dataset construction and model evaluation. These limitations motivate the design of TSB-FCST, which aims to provide a large-scale, high-quality, and taxonomy-aware evaluation platform.

Limitations in Benchmark Datasets. Public time-series datasets often suffer from data quality issues such as missing values, abnormal spikes, and long inactive regions, all of which can distort evaluation and lead to misleading predictability. For instance, abnormal spikes in the test region may disproportionately penalize otherwise robust models, while inactive regions can artificially lower forecasting errors because even naive predictors perform well on nearly constant signals. Likewise, purely stochastic series provide limited meaningful signal for evaluating forecasting capability. Beyond data quality, existing benchmarks also exhibit substantial domain imbalance, with observations concentrated in only a few application areas while others remain underrepresented (Figure 1). Such imbalance can bias evaluation results, since models effective in domains like energy or mobility may not generalize to healthcare, cloud, economics, or retail forecasting. Accordingly, a robust forecasting benchmark should integrate clearly defined quality assurance protocols and ensure extensive, balanced coverage of relevant domains towards rigorous and representative evaluation. Establishing such standardized guardrails is therefore essential to prevent misleading architectural conclusions and ensure that performance gains generalize reliably to unseen real-world scenarios.

Limitations in Model Collection and Evaluation Protocols. Existing forecasting benchmarks may also provide incomplete coverage of forecasting methodologies, potentially leading to oversimplified conclusions about model progress. Although modern architectures have achieved strong performance, traditional statistical and machine learning methods remain highly competitive in many forecasting settings. Without representative coverage across statistical, machine learning, deep learning, and foundation model families, it becomes difficult to determine whether improvements stem from genuinely stronger forecasting ability or merely from comparisons against limited baselines. Moreover, fair evaluation remains challenging because forecasting models differ substantially in their sensitivity to hyperparameters, normalization strategies, and training procedures. While default settings may underestimate certain models, exhaustive per-dataset tuning is often computationally impractical; therefore, standardized yet efficient tuning protocols, such as representative Hyperparameter Optimization (HPO)

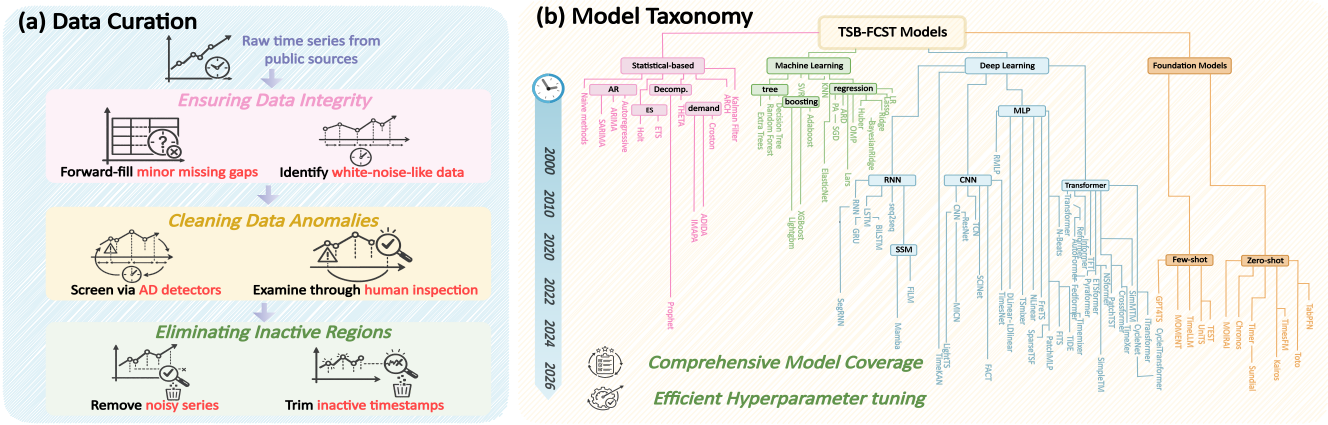

Figure 3: Overview of TSB-FCST. (a) Data curation pipeline: raw public time series are processed through integrity checking, anomaly cleaning, and inactive-region elimination. (b) Model taxonomy: TSB-FCST provides comprehensive coverage of forecasting paradigms of 100 models across statistical machine learning techniques and neural network methods.

subsets, are necessary to balance fairness and scalability. Finally, heavy reliance on aggregate average scores can obscure substantial heterogeneity across domains, forecasting horizons, temporal patterns, and multivariate dependency structures [106]. In particular, the comparative efficacy of Channel-Dependent (CD) versus Channel-Independent (CI) modeling paradigms in multivariate forecasting is often strongly contingent on dataset-specific correlation structures, thereby underscoring the necessity of taxonomy-aware, fine-grained evaluation and analyses.

4 TSB-FCST: Our Proposed Benchmark

We propose TSB-FCST, a large-scale time-series forecasting benchmark designed for comprehensive and rigorous evaluation. In the following, we go through (i) dataset acquisition and standardization; (ii) data curation and quality control; (iii) model search space and optimization protocol; and (iv) evaluation pipeline and metrics. An overview of TSB-FCST dataset curation and forecasting model taxonomy can be seen from Figure 3.

4.1 Data Preparation and Quality Assurance

To facilitate a comprehensive evaluation of time-series forecasting, we initiated an extensive data acquisition process aimed at capturing the vast diversity inherent in real-world temporal dynamics. We initially assembled a massive repository of raw time-series candidates in TSB-FCST, spanning a broad spectrum of domains including energy, meteorology, cloud computing, healthcare, transportation, economics, and retail. These sources encompass a wide range of temporal resolutions—from high-frequency seconds to yearly intervals—challenging a model's adaptability across varying scales and complex season patterns.

Following this initial collection, we first conducted and implemented a rigorous preliminary check process to ensure the basic integrity of the benchmark. This stage involved the systematic removal of "low-quality" datasets characterized by an excessive portion of missing values, severe temporal misalignment across multivariate channels, or purely stochastic properties (e.g., *white noise* and *random walk signals*), which are not suitable for model evaluation. After this preliminary filtering, we maintained a high-quality archive of 128 datasets with a total scale of *over one billion* observations, balanced between 64 univariate (UTS) and 64 multivariate (MTS) datasets. While this version represents an expansive and diverse archive for broad exploration, it serves as the foundation for the subsequent refined and curated 1B-scale "clean" version used in our primary benchmarking (detailed in Section 3.2).

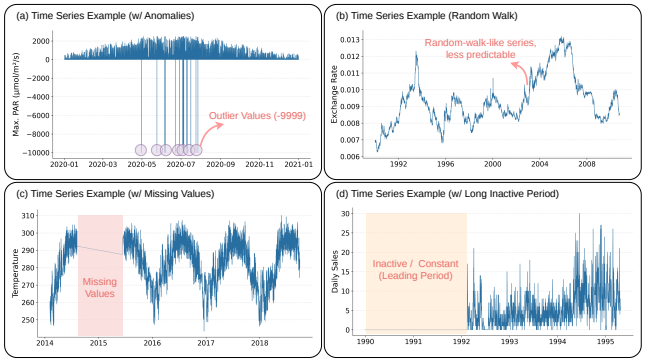

Figure 4: Examples of low-quality time series in public datasets. (a) Anomalies, extreme outlier values (e.g., −9999) in the nature-related dataset; (b) Random walk, noisy and unpredictable dynamics in the economics dataset; (c) Missing values, dates missing in the mobility dataset; (d) Constant segments, a long inactive period in the retail dataset.

To streamline the utilization of such a large-scale archive, all datasets have been unified into a standardized tabular format. Considering the heterogeneous original source or file structures, each dataset in TSB-FCST is transformed into a uniform schema with clearly distinguishable fields for Sample ID, Timestamp, and Channel identifiers. This unified representation significantly reduces the engineering overhead for data loading and preprocessing, enabling researchers to efficiently navigate this multi-billion scale archive without the burden of complex data wrangling.

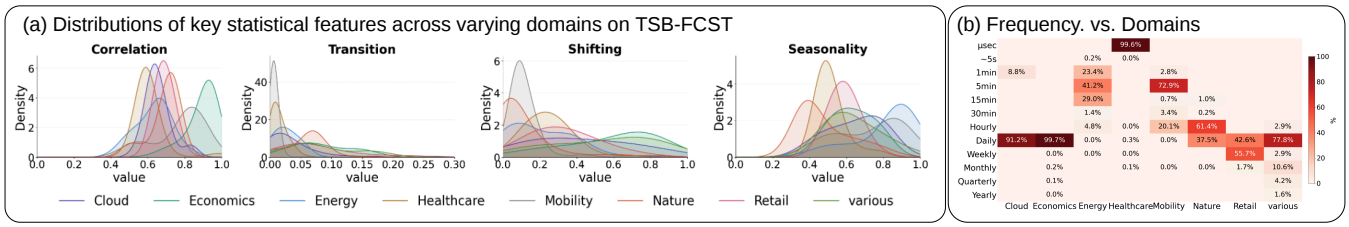

Figure 5: Data diversity and domain-specific characteristics of TSB-FCST. (a) Distribution of key temporal features across eight data domains. (b) Heatmap of sampling frequencies, representing various granularities. (c) Classification of demand patterns categorized by domain. A full version of all feature distribution can be seen in the Appendix (Figure 13).

4.2 Data Curation and Quality Control

As outlined in Section 4.1, we implement a rigorous, multi-stage curation and cleaning pipeline to ensure the integrity and predictability of the TSB-FCST archive. This process culminates in a ***one-billion-scale*** curated dataset, serving as a comprehensive and ready-to-use archive for large-scale research. Given the inherent heterogeneity of large data sources, we systematically address several critical issues commonly observed in public datasets through an automated screening process, complemented by meticulous human inspection. Our pipeline specifically targets: (i) Missing Value and Irregular Sampling Handling, (ii) Anomaly and Artifact Mitigation, (iii) Stochasticity and Random Walk Detection, and (iv) Constant and Inactive Region Filtering. Detailed descriptions of each stage are provided below. Examples of time-series are shown in Figure 4.

Missing Value and Irregular Sampling Handling: Missing values in time-series data often arise from various real-world constraints, such as sensor malfunctions, network transmission interruptions, or maintenance-related downtime. While certain advanced architectures possess an inherent capacity to process missing data, the majority of conventional and contemporary forecasting methodologies operate under the assumption of continuous and complete input sequences. To bridge this gap, we carefully handle missing entries while maintaining experimental integrity. Specifically, we employ a forward-fill imputation method for datasets with minor gaps, as it effectively preserves the temporal continuity of the signal while strictly preventing temporal data leakage—ensuring that information from the future does not leak to past observations. We discard datasets with such high levels of missing data to preserve a strong signal-to-noise ratio in the benchmark.

Furthermore, we rigorously verify the temporal regularity of the sampled intervals. To ensure a standardized and reliable baseline, the vast majority of the TSB-FCST archive is curated to be regularly sampled, facilitating the evaluation of models that rely on consistent seasonal patterns. However, we acknowledge that irregular sampling is a prevalent challenge in event-driven recording protocols. Instead of forcing these sequences into a uniform grid through aggressive interpolation—which might distort the ground-truth seasonality—we explicitly categorize a specialized subset that retains its original non-uniform interval nature. This subset allows for a targeted assessment of model robustness against irregular temporal dynamics without compromising the overall consistency. Finally, for both regular and irregular cases, we perform cross-channel temporal alignment so that all channels are synchronized across timestamps, resulting in a standardized data loading pipeline.

Anomaly and Artifact Mitigation: While the majority of real-world temporal signals exhibit relatively smooth transitions, we observed nonnegligible anomalies across various source data, often attributable to sensor malfunctions or data logging artifacts. A recurring debate in the forecasting community is whether models should be trained to be robust against such noise; however, the presence of these anomalies—especially within the test partitions—can significantly distort the overall evaluation.

Based on this, we implement a systematic protocol to detect and mitigate these artifacts. To ensure objectivity, we leverage an ensemble of three widely-used anomaly detection algorithms [67] and flag sequences that exhibit high consensus anomaly scores. We acknowledge that the precise identification of anomalies often necessitates substantial domain-specific prior knowledge, as certain "anomalous" fluctuations may represent rare but legitimate physical events. Consequently, our mitigation efforts focus primarily on "spiky" anomalies—extreme deviations (e.g., erroneous values like $-9,999$ or unrealistic spikes) that are highly likely to be the result of sensor failure or a human typo error. Once identified under such automated screening process, these localized artifacts are treated using the same imputation rationale applied to missing values to ensure temporal consistency. Finally, to prevent the unintended removal of critical signals, all flagged anomalies undergo a final round of meticulous human inspection. This hybrid approach ensures that our cleaning actions are both necessary and justified.

Stochasticity and Random Walk Detection: To ensure the benchmark provides meaningful signals for evaluating forecasting efficacy, we systematically exclude datasets that exhibit purely stochastic or unpredictable behavior. Specifically, we target two categories: (1) White Noise, identified via the Ljung-Box test [74]; these sequences lack any temporal autocorrelation, meaning past observations provide zero information about future values. (2) Random Walk processes, characterized by the form $y_t = y_{t-1} + \epsilon_t$, where ϵ_t is a noise term. Such a process is fundamentally unpredictable because its best unbiased estimator is simply the most recent observation (the naive forecast). We employ the Augmented Dickey-Fuller (ADF) test complemented by the Variance Ratio (VR) test.

We validate all flagged candidates by running both naive and SOTA models, ensuring that the final removed series cannot offer genuine predictive challenges for benchmarking purpose. Consequently, we remove datasets with a high proportion of random walk sequences, such as the ExchangeRate dataset, which is widely recognized as an unsuitable benchmark dataset [75].

Constant and Inactive Region Filtering: In contrast to stochastic signals that lack predictability, certain sequences exhibit a lack of temporal variance, rendering them "too simple" to provide meaningful evaluative value. This category primarily includes constant and inactive regions—extended intervals where the signal remains unchanged. Such phenomena are typically artifacts of sensor cold-starts, device malfunctions, or periods of system inactivity rather than legitimate seasonal patterns (e.g., zero-demand during scheduled off-hours). If left unaddressed, these regions can bias evaluation metrics by artificially deflating error rates, as even a naive model can achieve near-zero error on a "flatline" signal.

To mitigate this, we rigorously scan the archive for datasets characterized by long inactive "heads" or "tails." We perform precision trimming on these sequences, removing the stagnant prefixes or suffixes while preserving the high-quality, dynamic segments in the middle. Furthermore, we examine series with a significant portion of inactive intervals in their interior. For these cases, we evaluate the structural integrity of the remaining data: if the "active" segments before or after the gap are of sufficient length and quality, they are retained as independent sub-series; otherwise, the entire flagged series is discarded to prevent models from learning or being evaluated on trivial, non-representative patterns. This curation ensures that the benchmark focuses exclusively on active, dynamic temporal behaviors that challenge a model's performance.

Consistent with the aforementioned quality control philosophy, all sequences processed through the automated filters—whether involving imputation, trimming, or removal—undergo a final stage of meticulous human inspection to ensure the automated operations accurately preserve the underlying physical semantics of the data.

4.3 Model and Optimization

Model Search Space and Optimization Protocol. TSB-FCST evaluates 100 representative forecasting models across five major methodological families: naive methods, statistical methods, machine learning models, deep learning architectures, and time-series foundation models. This coverage spans classical baselines, feature-based models, modern neural architectures, and recent pre-trained foundation models[1], enabling unified comparison between traditional and modern forecasting paradigms. To bridge the performance gap between disparate model classes, we rigorously benchmark 100 widely-adopted models, grouped into four architectural paradigms to reflect the evolution of forecasting capabilities:

- ***Statistical Baselines***: We include **18** classical methods ranging from fundamental benchmarks to sophisticated automated frameworks such as *Naive Moving Average*, *Naive Seasonal*, *Naive Mean*, *Naive Drift*, *Autoregressive*, *ARIMA* [11], *SARIMA* [100], *Theta* [5], *ETS* [47], *Simple Exponential Smoothing* [45], *Croston* [45], *Prophet* [119], *ADIDA* [83], *IMAPA* [102], *ARCH* [31], *GARCH* [10], *Kalman Filter* [99], and *HOLT* [45]. These models provide a baseline for understanding the inherent predictability of the data.
- ***Machine Learning Models***: This category involves **16** representative algorithms, including *RandomForest* [13], *LightGBM* [49], *DecisionTree* [14], *K-Neighbors* [22], *Extratrees* [35], SVR [8], linear regression models like *Linear Regression*, *Lasso* [120], *Ridge* [42], *Huber* [88], *ElasticNet* [80], *SGDRegressor* [69], *BayesianRidge* [121], *ARDRegression* [129], *Lars* [29], and boosting models such as XGBoost [18]. These models test the archive's suitability for feature-based methods.
- ***Deep Learning Architectures***: We evaluate **47** state-of-the-art deep architectures, covering diverse structural designs such as Transformers (*Autoformer* [133], *PatchTST* [82], *Crossformer* [140], *iTransformer* [70], *Pyraformer* [68], *Non-stationary Transformer* [72], *Reformer* [52], *Informer* [144], *Transformer* [122], *FEDformer* [146], *TFT* [55], *ETSformer* [131], *CycleNet* [57], *CycleiTransformer* [57], *DUET* [104], *SimMTM* [27], *SimpleTM* [16], and *TimeXer* [128]), Linear-based networks (*DLinear* [137], *TSMixer* [17], *NLinear* [138], *LDlinear* [137], *RMLP* [86], *FreTS* [136], *TimeMixer* [125], *PatchMLP* [118], *FITS* [134], *TiDE* [23], *N-Beats* [85], *SparseTSF* [58]), convolution-based networks (*CNN* [87], *TimesNet* [132], *MICN* [123], *LightTS* [139], *SCINet* [66], *ResNet* [39], *FACT* [124], and *TCN* [7]), recurrent networks (*RNN* [109], *GRU* [19], *LSTM* [41], *BiLSTM*[105], *Seq2Seq* [117], *SegRNN* [59]), State Space Models (*Mamba* [38] and *FiLM* [145]), and emerging Kolmogorov-Arnold Networks (*TimeKAN* [44]). This unprecedentedly comprehensive scope is intended to reduce systematic bias in favor of specific deep learning architectures.
- ***Time-Series Foundation Models***: To assess the frontier of the field, we incorporate **19** recently emerged foundation models, including *zero-shot* forecasters such as *MOIRAI v1* [130], *MOIRAI v2* [61], *Chronos t5* [4], *Chronos bolt* [4], *Chronos v2* [3], *Timer* [73], *SunDial* [71], *Kairos* [32], *TiRex* [6], *TimesFM v1* [24], *TimesFM v2* [24], *TimesFM v2.5* [24], *Toto* [21], *TabPFN* [43] and *few-shot* learners like *GPT4TS* [147], *MOMENT* [37], *TimeLLM* [48], *UniTS* [34], and *TEST* [116]). This allows for a rigorous cross-comparison between domain-specific training and zero-shot transfer capabilities. For simplicity in subsequent sections, model names without explicit version suffixes refer to their latest respective versions (e.g., *Chronos* for v2, *TimesFM* for v2.5, and *MOIRAI* for v2).

Detailed model descriptions are provided in Section C. Such broad coverage avoids narrow interpretations of forecasting progress, since traditional statistical and machine learning methods remain competitive across many domains and temporal patterns despite recent advances in deep learning and foundation models. To ensure fair and scalable evaluation, we further adopt an efficient hyperparameter optimization protocol as follows.

Efficient Hyper-parameter Optimization: We acknowledge that optimizing 100 different models on the full 128 TSB-FCST archive is computationally infeasible—tuning multiple parameters for each model across 128 datasets would easily exceed 100k experimental runs, with large-scale datasets often requiring days or weeks to converge. Furthermore, per-dataset parameterization risks "overfitting" a model's potential to specific noise patterns rather than generalizable dynamics. To address this, we introduce TSB-FCST-50M, a strategically downsampled version from TSB-FCST-1B that preserves the original data distributions with proper validation sets while facilitating efficient tuning. This dataset and framework design ensures that each model reach a performance level that is

[1]For time-series foundation models, we report zero-shot forecasting performance when supported and few-shot performance otherwise, while all other task-specific models are evaluated in the full-shot setting.

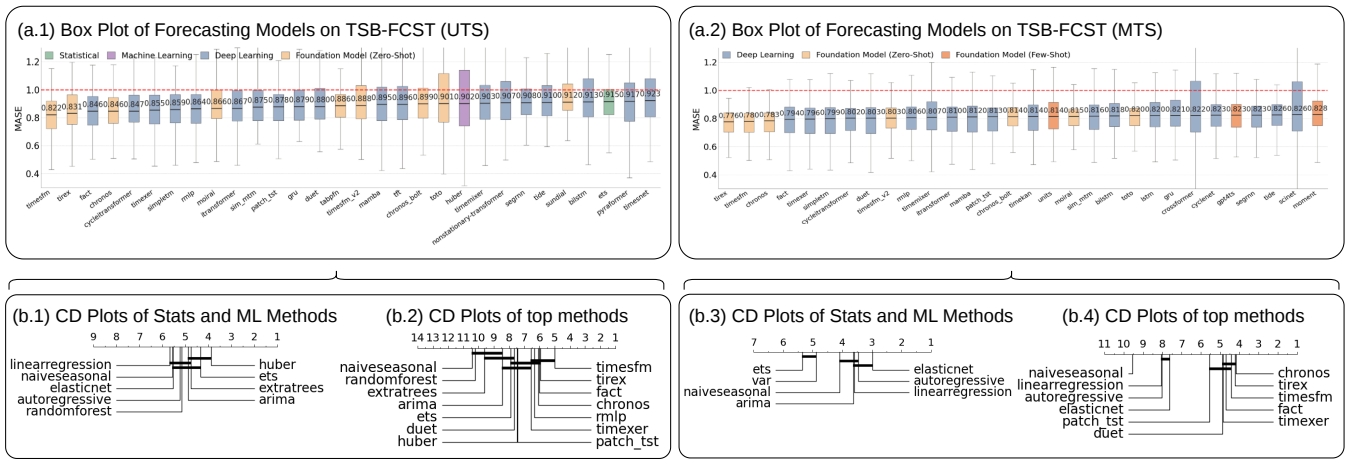

Figure 6: Global performance on TSB-FCST across model categories. (a) Box plots detailing the forecasting error of top 30 models across UTS and MTS. (b) Critical Difference (CD) diagrams illustrating the average model rankings with the Friedman-Nemenyi test. Models connected by a solid horizontal bar show no significant performance difference. For simplicity, model names without explicit version suffixes refer to their latest versions (e.g., *Chronos* for v2, *TimesFM* for v2.5, and *MOIRAI* for v2).

comparable to or exceeds default settings, while maintaining safeguards against dataset-specific over-parameterization, resulting in a more representative assessment of model generalizability.

4.4 Evaluation Pipeline and Metrics

To ensure a robust and statistically sound comparison across 100 disparate models, we implement a standardized evaluation pipeline inspired by the protocols in TFB [103] and TSLib [126]. This pipeline addresses `train/val/test data` partitioning, normalization-related preprocessing, and frequency-aware horizon settings.

Data Partitioning and Forecasting Schemes: We employ both fixed and rolling window forecasting schemes to comprehensively assess model stability. In the fixed forecasting setup, models predict a single future window following the training and validation set. For rolling forecasting, which provides a more rigorous evaluation of long-term predictive consistency across more time steps, we ensure sufficient sequence length by adopting a default 6:2:2 split for training, validation, and test sets. We employ overlapping windows for shorter time series and adopt non-overlapping windowing strategies for longer series in order to improve efficiency.

Data Pre-processing and Metrics: Following established practices in large-scale forecasting, all input series are normalized using a Standard Scaler (z-score normalization) to ensure numerical stability during training. For evaluation, we select metrics tailored to the dimensionality of the data: (i) Univariate Time-Series (UTS): We utilize Mean Absolute Scaled Error (MASE) and Mean Symmetric Mean Absolute Percentage Error (MSMAPE). These scale-invariant metrics are particularly suited for UTS as they allow for meaningful aggregation across datasets with vastly different magnitudes; and (ii) Multivariate Time-Series (MTS): we employ the standard Mean Squared Error (MSE) and Mean Absolute Error (MAE) to quantify reconstruction and forecasting accuracy across variables.

Frequency-Aware Evaluation Horizons: Rather than assigning arbitrary forecasting lengths, we define a specific set of horizons (H) for each dataset based on its inherent sampling frequency. These horizons are carefully curated to represent short, medium, and long-term forecasting tasks that align with real-world decision-making requirements (see Table 2 and 3 for details in Appendix). For instance, hourly datasets may feature horizons ranging from 48, 168 to 336 hours, which match 2 days, 1 week and 2 weeks in practical applications. This intuition-driven selection procedure ensures that the benchmark assesses models on tasks that are both practically relevant and challenging within their respective domains.

Mean Absolute Error (MAE) MAE measures the average absolute difference between the forecast and the ground truth. It is easy to interpret because it has the same unit as the original time series. A lower MAE indicates that the model's predictions are globally closer to the true values. Let y_i denote the ground-truth value, $\hat{y}_i$ denote the predicted value, and n denote the number of observations being predicted. MAE can be calculated as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|, \text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2. \tag{7}$$

Mean Squared Error (MSE) MSE measures the average squared forecasting error. Because the errors are squared, large mistakes would receive heavy penalties than small mistakes: therefore, MSE is a desired metric when large forecasting errors are strongly discouraged. MSE is computed in Eq. 7:

Mean Absolute Scaled Error (MASE). MASE measures the forecasting error relative to the error of a *naive seasonal* forecasting baseline [46]. Unlike percentage-based metrics such as MAPE, MASE is scale-independent and remains well-defined even when the time series contains zeros. Because of its robustness and comparability across datasets, MASE is widely used in large-scale forecasting benchmarks [2, 103, 113]. To enhance interpretability and facilitate a direct comparison against a standard baseline, we adopt a normalized variant of this metric to compute relative performance. Specifically, we formulate it as the ratio of the current model's

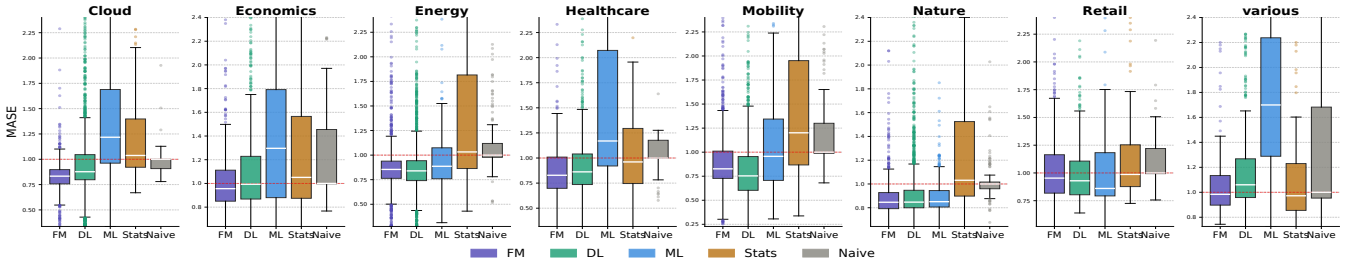

Figure 7: Performance comparison across diverse domains (measured in MASE), where the top performing methods are reported for each category. The result underscores the necessity of taxonomy-specific evaluation, as no single model category consistently dominates all scenarios on diverse TSB-FCST datasets.

metric against the naive baseline:

$$\text{MASE}(u) = \frac{\text{MAE}_{\text{model}}(u)}{\text{MAE}_{\text{S-Naive}}(u)} = \frac{\frac{1}{H}\sum_{i=1}^{H}|y_i - \hat{y}_i|}{\frac{1}{H}\sum_{i=1}^{H}|y_i - y_{i-m}|}, \tag{8}$$

where H denotes the forecasting horizon, and m represents the seasonal period of the time series. Under this formulation, a normalized score smaller than 1 directly indicates that the forecasting model outperforms the seasonal naive baseline, while a value greater than 1 implies inferior performance.

Modified Symmetric Mean Absolute Percentage Error (MSMAPE) MSMAPE is a modified version of SMAPE designed to improve numerical stability. By adding a small constant ϵ to the denominator, it avoids excessive penalties when both the true and predicted values are very small. This makes it more suitable for sparse, intermittent, or low-volume time series.

$$\text{MSMAPE} = \frac{100}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{|y_i| + |\hat{y}_i| + \epsilon}. \tag{9}$$

Details of additional evaluation metrics (RMSE, MAPE, SMAPE, WAPE) are presented in Appendix Section D.

In addition, we define frequency-aware forecasting horizons corresponding to short-, medium-, and long-term forecasting tasks aligned with each dataset's sampling frequency and practical context. Together with taxonomy-specific analysis and statistical significance testing, this evaluation pipeline provides a reliable framework for identifying model strengths and remaining challenges.

5 Experimental Settings

In this section, we introduce the experiment settings applied in this benchmark study, including the platform and implementation, statistical test and model training and evaluation details.

Platform and Implementation. We use a cluster equipped with 2xAMD EPYC 7713 64-Core processors on the Ubuntu 22.04.3 LTS operating system with 4 NVIDIA A100 80GB GPU cards to conduct our benchmarking experiments. We implement all forecasting models in Python 3.11 to ensure the fairness and consistency of evaluation. We also list the primary dependencies as follows: torch 2.4.1, numpy 1.26.4, pandas 2.1.4, scikit-learn 1.4.2, statsmodels 0.14.1, and scipy 1.11.4. To promote the reproducibility of our evaluation results and analysis, we release our source code repository and collected datasets [1]. For all model experiments, we run 3 random seeds and take average to reduce the effect of randomness. For each model, we perform hyper-parameter searches across multiple sets, with a limit of 4 sets to improve evaluation efficiency.

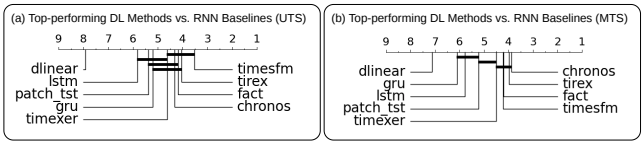

Figure 8: Forecasting performance comparison of the state-of-the-art methods against simple RNN baselines. Statistical testing indicates that despite architectural innovation, many state-of-the-art learning-based methods do not significantly outperform RNN + RevIN with proper hyperparameter tuning, particularly in univariate settings.

Statistical Test. To more reliably measure each model's overall performance, following [26, 92, 135], we employ Friedman test [33] and post-hoc Nemenyi test [81] with 95% confidence level. In this paper, we present the average ranking results with Critical Difference (CD) diagrams (shown in Figure 6 and 8), where models connected by a solid horizontal line indicates there is no significant performance difference between compared methods.

Model Training and Evaluation. To facilitate a rigorous benchmark study, we employ full-shot training for all task-specific models. However, to balance computational efficiency with the demands of full-shot fine-tuning for foundation models, we report zero-shot performance where publicly accessible (e.g., via public model weights or APIs). Otherwise, if fine-tuning was supported at the time of benchmark preparation, we report 10% few-shot performance

6 Experimental Results

In this section, we comprehensively evaluate the forecasting performance of diverse model categories across the 128 datasets within the TSB-FCST archive. Our analyses are structured around three core dimensions: (i) a macroscopic evaluation highlighting the top-performing methods across model categories under both UTS and MTS settings; (ii) a granular, taxonomy-specific comparison exploring domain-level variations, intrinsic feature impacts, multivariate (MTS) dynamics, and varying forecasting horizons; and (iii) accuracy versus efficiency with retrospect and prospect.

6.1 Global Benchmarking: The Overall Performance Landscape

In this section, we present a global overview of model performance across the TSB-FCST benchmark. Following established benchmarking protocols, we first establish an overall performance landscape to evaluate the general efficacy and robustness of various model categories across diverse datasets.

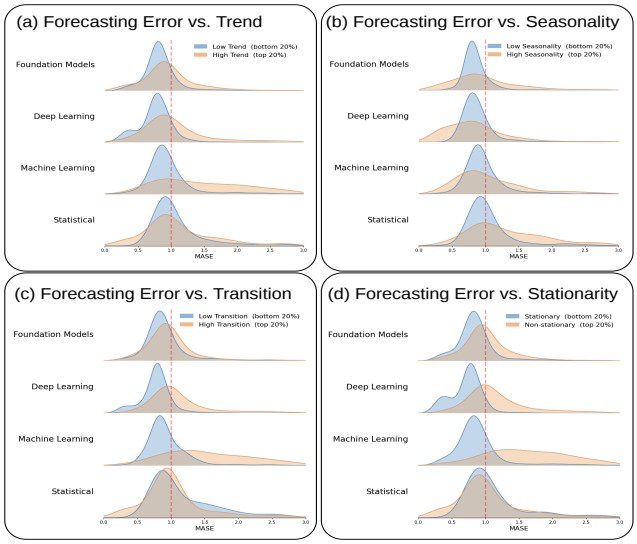

Figure 9: Performance sensitivity to data characteristics, visualized by ridge plots, where magnitude of forecasting performance degradation varies across model families.

Global view of Model Perfomance acorss Categories. Our analysis begins with Univariate Time Series (UTS) forecasting, providing a granular comparison of traditional and modern methodologies. A distinguishing feature of our study is the inclusion of a vast array of statistical and machine learning (ML) methods, which are commonly overlooked in contemporary forecasting literature. As illustrated in the Critical Difference (CD) diagram for the UTS top-performing statistical/ML methods (Figure 6, b.1), the majority of these classical approaches significantly outperform the *NaiveSeasonal* baseline across the 64 UTS datasets. Notably, regression-based methods and robust estimators such as *Huber* exhibit a statistically significant performance gain over the naive baseline. This observation is further corroborated by the performance distribution in Figure 6(a.1), where two statistical and ML models successfully secure positions within the top 30 out of the 100 evaluated algorithms on our global leaderboard, outperforming a wide range published forecasting methods and establishing a strong baseline in the univariate forecasting scenarios.

When evaluating all categories (Figure 6(b.2)), Foundation Models (FMs) and Deep Learning (DL) architectures—such as *TimesFM* and *TiReX*, followed by *FACT* and *Chronos*—occupy the leading positions, though many top-tier modern models do not significantly outperform the best classical ML methods (e.g., *Huber*) in univariate settings. Conversely, the landscape shifts in Multivariate Time Series (MTS) forecasting (Figure 6(b.3, b.4)), where FMs and DL models exhibit a clear, statistically significant advantage over classical baselines[2]. This performance gap stems from modern architectures' superior ability to handle high-dimensional and complex temporal dependencies—properties that are highly prevalent in MTS datasets. Notably, these top-performing FMs achieve their

[2]We select top-performing and scalable statistical and ML methods on MTS tasks. As most natively support only univariate forecasting, we apply a Channel-Independent (CI) strategy to evaluate them on multivariate tasks as baselines.

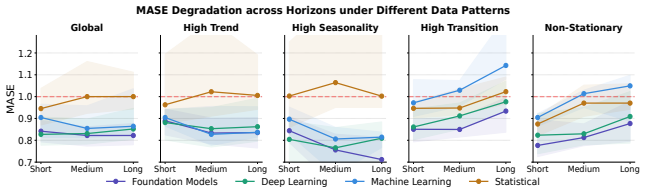

Figure 10: Performance degradation across varying forecasting horizons under distinct data patterns and model categories. Each panel corresponds to a different data pattern.

rankings in a strictly zero-shot setting, indicating substantial room for future improvement via domain-specific fine-tuning.

The Undervalued Efficacy of Simple Baselines. A major limitation of current benchmarks lies in their systemic omission of well-established statistical methods and simpler deep learning architectures, which are often prematurely dismissed in favor of overly complex models. However, a global, cross-domain view of our results reveals that these approaches remain exceptionally simple, computationally efficient, and highly competitive. Traditional statistical models like *Exponential Smoothing (ETS)* and robust machine learning estimators like *Huber* consistently provide strong, reliable baselines across diverse scenarios, serving as valuable reference points that prevent overestimating the progress of more complex architectural designs in applications.

This oversight is not restricted to statistical models alone. As illustrated in the Critical Difference diagram (Figure 8), several highly publicized, state-of-the-art architectures struggle to strongly outperform a basic, RNN-based model when coupled with simple, proven design components—such as Reversible Instance Normalization (RevIN) to handle distribution shift [51, 53, 82, 142]. Despite its simplicity and effectiveness, this class of modified classic networks is widely neglected in contemporary literature. Ultimately, these findings underscore that a truly reliable evaluation framework must not only track the frontier of foundation models but also actively maintain these straightforward, highly efficient baselines to measure genuine, incremental algorithmic progress.

In the following section, we move beyond the global winner and conduct a more fine-grained, taxonomy-specific analysis.

6.2 Taxonomy-Specific Evaluation: Domains, Dynamics, and Dependencies

Domain-Specific Efficacy and Feature Sensitivity. To provide a more granular understanding of model capabilities and move beyond global averages, we dissect performance across distinct application domains (Figure 7) and intrinsic data characteristics (Figure 9). Figure 7 reveals that model dominance is highly domain-dependent. While Deep Learning (DL) architectures and Foundation Models (FMs) demonstrate strong superiority in the Cloud and Mobility domains, traditional Statistical and ML methods remain highly competitive in Energy, Nature, and Retail domains. Notably, in the Retail domain, ML methods actually surpass modern architectures consistently with a top-ranked median MASE.

The performance variation of different categories of methods may be explained from the rich diversity of data patterns in TSB-FCST. Figures 2 and 5 further illustrate the remarkable heterogeneity of

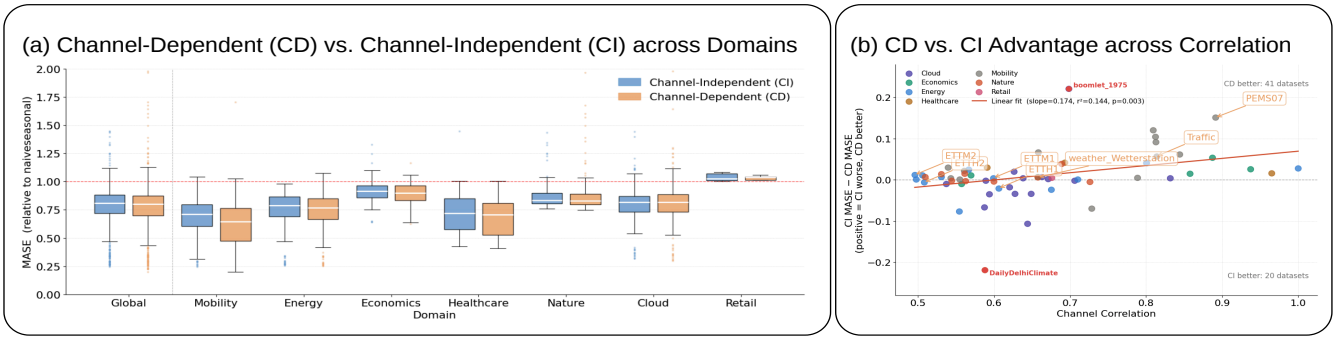

Figure 11: Channel-Dependent (CD) vs. Channel-Independent (CI) model performance on MTS datasets. To compare, we select the top-performing deep learning-based forecasting methods individually with either CD or CI model design. (a) MASE distribution across domains, sorted in descending order of CD advantage. (b) Scatter plot of CI-CD MASE difference against channel correlation. Widely used MTS datasets (e.g., ETT datasets) are also highlighted for further discussion.

TSB-FCST. Figure 2 presents representative examples spanning classical temporal properties (e.g., seasonality, trend, shifting, stationarity, and transition) as well as diverse demand behaviors (smooth, erratic, intermittent, and lumpy), highlighting the wide spectrum of forecasting challenges encountered in practice. Figure 5 also complements these examples quantitatively by showing that such characteristics are unevenly distributed across domains. Broadly, human-centric and operational domains (e.g., Cloud, Economics, and Mobility) tend to exhibit smoother temporal dependencies with strong trends or erratic behaviors at relatively fine temporal resolutions, whereas resource- and demand-driven domains (e.g., Energy, Retail, and Nature) are dominated by lumpy demand patterns and greater variability in temporal dynamics. In contrast, Healthcare datasets stand out as an extreme case, consisting primarily of ultra-high-frequency observations with sparse behaviors. Collectively, these results demonstrate that TSB-FCST captures substantial diversity not only in temporal patterns, but also in sampling and domain-specific forecasting challenges. Furthermore, the performance degradation and shifts in model rankings across certain data domains reveal a failure to capture specific data patterns, offering valuable insights to guide future methodological improvements of task-specific or foundation models.

To further interpret these dynamics, Figure 9 examines model sensitivity under different data characteristics. The ridge plots show that forecasting errors can shift substantially between low- and high-regime datasets, but the direction and magnitude of these shifts vary across model families. In particular, high transition and non-stationary regimes tend to induce broader or more right-skewed error distributions for several model categories, indicating increased risk of large forecasting errors. This effect is especially visible for some machine learning methods, while foundation models and deep learning models generally exhibit more concentrated error distributions but are still affected by challenging dynamics. Seasonality and trend also lead to heterogeneous behavior: they do not uniformly benefit or degrade all models, suggesting that their impact depends on both the data regime and the modeling assumptions. Overall, these results reinforce that model performance is strongly conditioned on temporal characteristics, and such feature-level sensitivities can be obscured by aggregate benchmark scores.

Horizons Meet Data Dynamics. Figure 10 further shows that forecasting degradation across horizons is strongly shaped by temporal data patterns and model-family characteristics. Under high trend and seasonality, foundation models maintain strong robustness and even improve as the horizon increases, suggesting their ability to exploit recurring structures over longer prediction windows. However, this advantage does not transfer uniformly to all regimes. In high-transition and non-stationary datasets, most model families exhibit increasing forecasting errors as the horizon becomes longer. This degradation is more pronounced for machine learning and statistical methods, which deteriorate further over longer prediction windows given complex data patterns. This pattern helps explain their lower average rankings across varying datasets versus horizons from the global evaluation perspective. Foundation models and deep learning models are generally more stable, but they also suffer from clear long-horizon degradation under challenging dynamics. More broadly, future forecasting evaluation should not focus solely on increasingly long horizons or aggregated accuracy scores. Instead, benchmarks are suggested to account for the intrinsic challenges induced by data characteristics, such as seasonality, trend, and non-stationary behaviors.

Identifying Datasets Suitable for Channel-Dependency Evaluation. Next, we revisit the long-standing discussion on Channel-Independent (CI) versus Channel-Dependent (CD) modeling strategies for multivariate time-series forecasting, as shown in Figure 11. Building on prior studies that relate the effectiveness of cross-channel modeling to dataset characteristics [103, 142], our benchmark extends this analysis to a larger and more diverse evaluation space, providing a clearer picture across domains and data characteristics regimes. Figure 11(a) shows that the relative advantage of CD models is not uniform across application domains. For example, mobility datasets exhibit a more visible CD advantage, suggesting that explicitly modeling inter-channel interactions can be beneficial when channels are strongly coupled. In contrast, other domains show much smaller differences between CI and CD strategies, indicating that cross-channel modeling is not always necessary.

Figure 11(b) further connects this observation to inter-channel correlation. The CI–CD MASE difference increases with channel

correlation, suggesting that CD models become more advantageous when the channels contain stronger shared structure. Meanwhile, several commonly used MTS benchmark datasets, such as ETTh and ETTm variants, are highlighted in the plot and lie in regions where the CI–CD performance gap is negligible. This implies that repeatedly debating channel dependency on a narrow set may provide limited diagnostic insight. To solve this problem, TSB-FCST covers a broader spectrum of datasets with diverse data patterns and channel dependency structures, enabling a more systematic evaluation of when channel-dependent modeling is truly beneficial. These results suggest that reliable MTS benchmarks should explicitly account for dependency structure rather than treating all multivariate datasets equally for task-specific challenges.

6.3 Forecasting Model Accuracy vs. Scaling: Retrospect and Prospect

To better understand the practical trade-offs among forecasting model families, Figure 12 compares the forecasting error and computational efficiency of representative models across categories on TSB-FCST, including both the trade-off analysis of accuracy versus runtime, as well as the pretrained dataset scale.

Accuracy vs. Model Efficiency. A key revelation from this multi-dimensional comparison is the competitive of tradition forecasting methods. Classic approaches such as *exponential smoothing (ets)* establish highly simple and robust baselines, consistently achieving a competitive forecasting accuracy across a wide variety of data domains. While these lightweight models are generally confined to CPU execution and require sequential retraining for each individual dataset, their minimal parameter footprint and reliable performance make them practical options to serve as a baseline. In contrast, foundation models shift the heavy computational burden entirely to the offline pre-training phase (see varying pre-trained dataset scale). By leveraging highly parallelized GPU architectures during deployment, models like *TimesFM*, *Chronos*, and *Moirai* deliver outstanding forecasting accuracy while maintaining highly competitive zero-shot accuracy and inference speeds. However, as demonstrated in our taxonomy-specific analyses in Section 6.2, foundation models are not *universally dominant*. Despite their scale, they still exhibit pronounced weaknesses when encountering specific data domains characterized by irregular temporal patterns, such as Retail, where local statistical properties or specialized domain knowledge still hold an advantage. Meanwhile, deep learning models occupy an intermediate regime, bridging the gap by balancing strong representation capacity with moderate, practical computational overhead.

Pre-training Scaling Laws and Data Leakage Controllability. To explore the scaling behavior of foundation models, Figure 12 analyzes the relationship between forecasting error, pre-training data scale, and data leakage rates. Our empirical analysis reveals a clear scaling trend at the starting point: within a scale of 100 billion observations, increasing the pre-training data scale yields consistent gains in zero-shot forecasting performance. Beyond this 100B threshold, however, we observe distinct diminishing marginal returns, where further scaling the pre-training corpus struggles to yield proportional accuracy gains. This performance plateau suggests that simply expanding pre-training volume is alone be insufficient, and and future progress likely hinges on addressing specific model vulnerabilities under diverse data patterns rather than purely

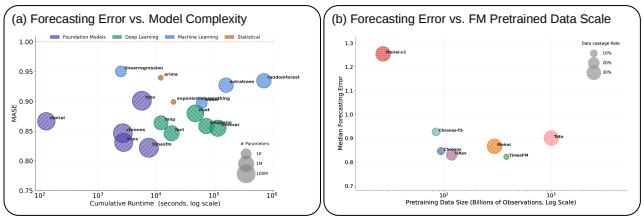

Figure 12: Forecasting error across top performing models on TSB-FCST. Each bubble circle represents a model. (a) MASE vs. cumulative runtime across model categories, where the circle size represents the model parameter counts (i.e., model complexity); (b) MASE vs. pretrained dataset scale, where the circle size represents the leakage rate of each pretrained foundation models compared with TSB-FCST collections.

scaling up data quantities. Furthermore, as foundation models scale, data leakage represents a major obstacle to reliable evaluation. We introduce TSB-FCST, a comprehensive evaluation benchmark that targets on both a diverse testing suite and a controlled environment designed for reliable benchmarking. Even under rigorous evaluation, we observe a genuine performance uplift driven by model scaling, while the data leakage rate of top-performing models (such as *TimesFM* and *Chronos*) remains approximately 10%. This demonstrates that the observed algorithmic progress on this benchmark is substantial and less likely an artifact of training-set contamination.

Overall, our experimental analysis suggests that forecasting evaluation must transcend a single, aggregate accuracy ranking. A reliable, future-proof benchmarking paradigm should jointly evaluate predictive accuracy, model scale and efficiency, while explicitly accounting for the domain-specific data characteristics that collectively govern real-world forecasting behavior.

7 Conclusion

In this work, we present TSB-FCST, a billion-scale time-series forecasting benchmark designed for comprehensive and reliable model evaluation. We curate 128 high-quality time-series datasets across eight application domains and evaluate 100 forecasting models spanning statistical, machine learning, deep learning, and recently emerged foundation-model paradigms. Our results show that although foundation models demonstrate strong overall and zero-shot performance, traditional methods and simple deep-learning-based methods remain highly competitive in a global view. We further show that forecasting performance is heavily influenced by application domains, temporal dynamics, forecasting horizons, and multivariate dependency structures, highlighting the limitations of relying solely on aggregate metrics. Besides, we also go through the discussion of forecasting accuracy and the model scaling, which reveals the current situation of foundation model scaling laws. Nevertheless, we acknowledge several limitations: this work primarily focuses on point forecasting, while probabilistic forecasting may be more suitable for certain application scenarios. In addition, forecasting with exogenous variables represents another important direction that is not fully explored right now. We leave these extensions for future work. Overall, TSB-FCST provides a rigorous dataset benchmark and evaluation framework for future comprehensive, robust, and taxonomy-aware time-series forecasting research.

References

[1] 2026. Anonymized GitHub. https://anonymous.4open.science/r/TSB-FCST-Benchmark. Accessed: 2026-05-06.

[2] Taha Aksu, Gerald Woo, Juncheng Liu, Xu Liu, Chenghao Liu, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. 2024. GIFT-Eval: A Benchmark For General Time Series Forecasting Model Evaluation. arXiv:2410.10393 [cs.LG] https://arxiv.org/abs/2410.10393

[3] Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. 2025. Chronos-2: From Univariate to Universal Forecasting. arXiv:2510.15821 [cs.LG] https://arxiv.org/abs/2510.15821

[4] Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Syndar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. 2024. Chronos: Learning the Language of Time Series. *arXiv preprint arXiv:2403.07815* (2024).

[5] V. Assimakopoulos and K. Nikolopoulos. 2000. The theta model: a decomposition approach to forecasting. *International Journal of Forecasting* 16, 4 (Oct. 2000), 521–530. doi:10.1016/s0169-2070(00)00066-2

[6] Andreas Auer, Patrick Podest, Daniel Klotz, Sebastian Böck, Günter Klambauer, and Sepp Hochreiter. 2025. TiRex: Zero-Shot Forecasting Across Long and Short Horizons with Enhanced In-Context Learning. arXiv:2505.23719 [cs.LG] https://arxiv.org/abs/2505.23719

[7] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. 2018. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. arXiv:1803.01271 [cs.LG] https://arxiv.org/abs/1803.01271

[8] Debasish Basak, Srimanta Pal, and Dipak Patranabis. 2007. Support Vector Regression. *Neural Information Processing – Letters and Reviews* 11 (11 2007).

[9] Perceval Beja-Battais. 2023. Overview of AdaBoost : Reconciling its views to better understand its dynamics. arXiv:2310.18323 [cs.LG] https://arxiv.org/abs/2310.18323

[10] Tim Bollerslev. 1986. Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics* 31, 3 (1986), 307–327. doi:10.1016/0304-4076(86)90063-1

[11] George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung. 2015. *Time Series Analysis: Forecasting and Control* (5 ed.). John Wiley & Sons, Hoboken, NJ. 712 pages.

[12] Leo Breiman. 2001. *Machine Learning* 45, 1 (2001), 5–32. doi:10.1023/a:1010933404324

[13] Leo Breiman. 2001. Random Forests. *Machine Learning* 45 (2001), 5–32. doi:10.1023/A:1010933404324

[14] Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. 1984. *Classification and Regression Trees*. Wadsworth International Group, Belmont, CA.

[15] C. Chatfield. 1978. The Holt-Winters Forecasting Procedure. *Applied Statistics* 27, 3 (1978), 264. doi:10.2307/2347162

[16] H. Chen, V. Luong, L. Mukherjee, and V. Singh. 2025. SimpleTM: A Simple Baseline for Multivariate Time Series Forecasting. In *The Thirteenth International Conference on Learning Representations (ICLR)*. https://openreview.net/forum?id=oANkBaVci5

[17] Si-An Chen, Chun-Liang Li, Nate Yoder, Sercan O. Arik, and Tomas Pfister. 2023. TSMixer: An All-MLP Architecture for Time Series Forecasting. arXiv:2303.06053 [cs.LG] https://arxiv.org/abs/2303.06053

[18] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. ACM, 785–794. doi:10.1145/2939672.2939785

[19] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. arXiv:1409.1259 [cs.CL] https://arxiv.org/abs/1409.1259

[20] Robert B. Cleveland, William S. Cleveland, Jean E. McRae, and Irma Terpenning. 1990. STL: A Seasonal-Trend Decomposition Procedure Based on Loess (with Discussion). *Journal of Official Statistics* 6 (1990), 3–73.

[21] Ben Cohen, Emaad Khwaja, Kan Wang, Charles Masson, Elise Ramé, Youssef Doubli, and Othmane Abou-Amal. 2024. Toto: Time Series Optimized Transformer for Observability. arXiv:2407.07874 [cs.LG] https://arxiv.org/abs/2407.07874

[22] Pádraig Cunningham and Sarah Jane Delany. 2021. k-Nearest Neighbour Classifiers - A Tutorial. *ACM Comput. Surv.* 54, 6, Article 128 (July 2021), 25 pages. doi:10.1145/3459665

[23] Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. 2024. Long-term Forecasting with TiDE: Time-series Dense Encoder. arXiv:2304.08424 [stat.ML] https://arxiv.org/abs/2304.08424

[24] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. 2024. A decoder-only foundation model for time-series forecasting. arXiv:2310.10688 [cs.CL] https://arxiv.org/abs/2310.10688

[25] Alysha De Livera, Rob Hyndman, and Ralph Snyder. 2010. Forecasting Time Series With Complex Seasonal Patterns Using Exponential Smoothing. *J. Amer. Statist. Assoc.* 106 (01 2010), 1513–1527. doi:10.1198/jasa.2011.tm09771

[26] Janez Demšar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* 7 (Dec. 2006), 1–30.

[27] Jiaxiang Dong, Haixu Wu, Haoran Zhang, Li Zhang, Jianmin Wang, and Mingsheng Long. 2023. SimMTM: A Simple Pre-Training Framework for Masked Time-Series Modeling. arXiv:2302.00861 [cs.LG] https://arxiv.org/abs/2302.00861

[28] Jens E d'Hondt, Haojun Li, Fan Yang, Odysseas Papapetrou, and John Paparrizos. 2025. A Structured Study of Multivariate Time-Series Distance Measures. In *Proceedings of the 2025 ACM SIGMOD international conference on management of data*.

[29] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. 2004. Least angle regression. *The Annals of Statistics* 32, 2 (April 2004). doi:10.1214/009053604000000067

[30] Graham Elliott, Thomas J. Rothenberg, and James H. Stock. 1996. Efficient Tests for an Autoregressive Unit Root. *Econometrica* 64, 4 (1996), 813–836. http://www.jstor.org/stable/2171846

[31] Robert F. Engle. 1982. Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica* 50, 4 (July 1982), 987. doi:10.2307/1912773

[32] Kun Feng, Shaocheng Lan, Yuchen Fang, Wenchao He, Lintao Ma, Xingyu Lu, and Kan Ren. 2026. Kairos: Toward Adaptive and Parameter-Efficient Time Series Foundation Models. arXiv:2509.25826 [cs.LG] https://arxiv.org/abs/2509.25826

[33] Milton Friedman. 1937. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *J. Amer. Statist. Assoc.* 32 (1937), 675–701. https://api.semanticscholar.org/CorpusID:120581754

[34] Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. 2024. UniTS: A Unified Multi-Task Time Series Model. arXiv:2403.00131 [cs.LG] https://arxiv.org/abs/2403.00131

[35] Pierre Geurts, Damien Ernst, and Louis Wehenkel. 2006. Extremely Randomized Trees. *Machine Learning* 63, 1 (2006), 3–42. doi:10.1007/s10994-006-6226-1

[36] Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I. Webb, Rob J. Hyndman, and Pablo Montero-Manso. 2021. Monash Time Series Forecasting Archive. arXiv:2105.06643 [cs.LG] https://arxiv.org/abs/2105.06643

[37] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. 2024. MOMENT: A Family of Open Time-series Foundation Models. arXiv:2402.03885 [cs.LG] https://arxiv.org/abs/2402.03885

[38] Albert Gu and Tri Dao. 2024. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. arXiv:2312.00752 [cs.LG] https://arxiv.org/abs/2312.00752

[39] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs.CV] https://arxiv.org/abs/1512.03385

[40] Julien Herzen, Francesco Lässig, Samuele Giuliano Piazzetta, Thomas Neuer, Léo Tafti, Guillaume Raille, Tomas Van Pottelbergh, Marek Pasieka, Andrzej Skrodzki, Nicolas Huguenin, Maxime Dumonal, Jan Kościsz, Dennis Bader, Frédérick Gusset, Mounir Benheddi, Camila Williamson, Michal Kosinski, Matej Petrik, and Gaël Grosch. 2022. Darts: User-Friendly Modern Machine Learning for Time Series. arXiv:2110.03224 [cs.LG] https://arxiv.org/abs/2110.03224

[41] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780. doi:10.1162/neco.1997.9.8.1735

[42] Arthur E. Hoerl and Robert W. Kennard. 2000. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* 42, 1 (2000), 80–86. http://www.jstor.org/stable/1271436

[43] Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. 2023. TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second. arXiv:2207.01848 [cs.LG] https://arxiv.org/abs/2207.01848

[44] Songtao Huang, Zhen Zhao, Can Li, and Lei Bai. 2025. TimeKAN: KAN-based Frequency Decomposition Learning Architecture for Long-term Time Series Forecasting. arXiv:2502.06910 [cs.LG] https://arxiv.org/abs/2502.06910

[45] Rob J. Hyndman and George Athanasopoulos. 2018. *Forecasting: Principles and Practice* (2nd ed.). OTexts, Melbourne, Australia. https://otexts.com/fpp2/

[46] Rob J Hyndman and Anne B Koehler. 2006. Another look at measures of forecast accuracy. *International journal of forecasting* 22, 4 (2006), 679–688.

[47] Rob J. Hyndman, Anne B. Koehler, J. Keith Ord, and Ralph D. Snyder. 2008. *Forecasting with Exponential Smoothing: The State Space Approach*. Springer, Berlin, Germany. doi:10.1007/978-3-540-71918-2

[48] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. 2024. Time-LLM: Time Series Forecasting by Reprogramming Large Language Models. arXiv:2310.01728 [cs.LG] https://arxiv.org/abs/2310.01728

[49] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: a highly efficient gradient boosting

decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 3149–3157.

[50] Jongseon Kim, Hyungjoon Kim, HyunGi Kim, Dongjun Lee, and Sungroh Yoon. 2025. A Comprehensive Survey of Deep Learning for Time Series Forecasting: Architectural Diversity and Open Challenges. arXiv:2411.05793 [cs.LG] https://arxiv.org/abs/2411.05793

[51] Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. 2021. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International conference on learning representations*.

[52] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The Efficient Transformer. arXiv:2001.04451 [cs.LG] https://arxiv.org/abs/2001.04451

[53] Zhe Li, Shiyi Qi, Yiduo Li, and Zenglin Xu. 2023. Revisiting long-term time series forecasting: An investigation on linear mapping. *arXiv preprint arXiv:2305.10721* (2023).

[54] Zhe Li, Xiangfei Qiu, Peng Chen, Yihang Wang, Hanyin Cheng, Yang Shu, Jilin Hu, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, and Bin Yang. 2025. TSFM-Bench: A Comprehensive and Unified Benchmark of Foundation Models for Time Series Forecasting. arXiv:2410.11802 [cs.LG] https://arxiv.org/abs/2410.11802

[55] Bryan Lim, Sercan O. Arik, Nicolas Loeff, and Tomas Pfister. 2020. Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting. arXiv:1912.09363 [stat.ML] https://arxiv.org/abs/1912.09363

[56] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. 2003. A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*. 2–11.

[57] Shengsheng Lin, Weiwei Lin, Xinyi Hu, Wentai Wu, Ruichao Mo, and Haocheng Zhong. 2024. CycleNet: Enhancing Time Series Forecasting through Modeling Periodic Patterns. arXiv:2409.18479 [cs.LG] https://arxiv.org/abs/2409.18479

[58] Shengsheng Lin, Weiwei Lin, Wentai Wu, Haojun Chen, and Junjie Yang. 2024. SparseTSF: Modeling Long-term Time Series Forecasting with 1k Parameters. arXiv:2405.00946 [cs.LG] https://arxiv.org/abs/2405.00946

[59] Shengsheng Lin, Weiwei Lin, Wentai Wu, Feiyu Zhao, Ruichao Mo, and Haotong Zhang. 2023. SegRNN: Segment Recurrent Neural Network for Long-Term Time Series Forecasting. arXiv:2308.11200 [cs.LG] https://arxiv.org/abs/2308.11200

[60] Jason Lines, Sarah Taylor, and Anthony Bagnall. 2016. HIVE-COTE: The Hierarchical Vote Collective of Transformation-Based Ensembles for Time Series Classification. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. 1041–1046. doi:10.1109/ICDM.2016.0133

[61] Chenghao Liu, Taha Aksu, Juncheng Liu, Xu Liu, Hanshu Yan, Quang Pham, Silvio Savarese, Doyen Sahoo, Caiming Xiong, and Junnan Li. 2026. Moirai 2.0: When Less Is More for Time Series Forecasting. arXiv:2511.11698 [cs.LG] https://arxiv.org/abs/2511.11698

[62] Haoxin Liu, Harshavardhan Kamarthi, Lingkai Kong, Zhiyuan Zhao, Chao Zhang, and B. Aditya Prakash. 2024. Time-Series Forecasting for Out-of-Distribution Generalization Using Invariant Learning. In *Proceedings of the 41st International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 235)*, Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (Eds.). PMLR, 31312–31325. https://proceedings.mlr.press/v235/liu24ae.html

[63] Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Prabhakar Kamarthi, Aditya Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, et al. 2024. Time-mmd: Multi-domain multimodal dataset for time series analysis. *Advances in Neural Information Processing Systems* 37 (2024), 77888–77933.

[64] Haoxin Liu, Zhiyuan Zhao, Shiduo Li, and B Aditya Prakash. 2025. Evaluating System 1 vs. 2 Reasoning Approaches for Zero-Shot Time-Series Forecasting: A Benchmark and Insights. *arXiv preprint arXiv:2503.01895* (2025).

[65] Haoxin Liu, Zhiyuan Zhao, Jindong Wang, Harshavardhan Kamarthi, and B. Aditya Prakash. 2024. LSTPrompt: Large Language Models as Zero-Shot Time Series Forecasters by Long-Short-Term Prompting. In *Findings of the Association for Computational Linguistics: ACL 2024*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 7832–7840. doi:10.18653/v1/2024.findings-acl.466

[66] Minhao Liu, Ailing Zeng, Muxi Chen, Zhijian Xu, Qiuxia Lai, Lingna Ma, and Qiang Xu. 2022. SCINet: Time Series Modeling and Forecasting with Sample Convolution and Interaction. arXiv:2106.09305 [cs.LG] https://arxiv.org/abs/2106.09305

[67] Qinghua Liu and John Paparrizos. 2024. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. *Advances in Neural Information Processing Systems* 37 (2024), 108231–108261.

[68] Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X. Liu, and Schahram Dustdar. 2022. Pyraformer: Low-Complexity Pyramidal Attention for Long-Range Time Series Modeling and Forecasting. In *International Conference on Learning Representations*. https://api.semanticscholar.org/CorpusID:251649164

[69] Yanli Liu, Yuan Gao, and Wotao Yin. 2020. An Improved Analysis of Stochastic Gradient Descent with Momentum. arXiv:2007.07989 [math.OC] https://arxiv.org/abs/2007.07989

[70] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. 2024. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. arXiv:2310.06625 [cs.LG] https://arxiv.org/abs/2310.06625

[71] Yong Liu, Guo Qin, Zhiyuan Shi, Zhi Chen, Caiyin Yang, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. 2025. Sundial: A Family of Highly Capable Time Series Foundation Models. arXiv:2502.00816 [cs.LG] https://arxiv.org/abs/2502.00816

[72] Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. 2023. Non-stationary Transformers: Exploring the Stationarity in Time Series Forecasting. arXiv:2205.14415 [cs.LG] https://arxiv.org/abs/2205.14415

[73] Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. 2024. Timer: Generative Pre-trained Transformers Are Large Time Series Models. arXiv:2402.02368 [cs.LG] https://arxiv.org/abs/2402.02368

[74] Greta M Ljung and George EP Box. 1978. On a measure of lack of fit in time series models. *Biometrika* 65, 2 (1978), 297–303.

[75] Andrew W Lo and A Craig MacKinlay. 1988. Stock market prices do not follow random walks: Evidence from a simple specification test. *The review of financial studies* 1, 1 (1988), 41–66.

[76] Carl H Lubba, Sarab S Sethi, Philip Knaute, Simon R Schultz, Ben D Fulcher, and Nick S Jones. 2019. catch22: CAnonical Time-series CHaracteristics. arXiv:1901.10200 [cs.IR] https://arxiv.org/abs/1901.10200

[77] Carl H Lubba, Sarab S Sethi, Philip Knaute, Simon R Schultz, Ben D Fulcher, and Nick S Jones. 2019. catch22: CAnonical Time-series CHaracteristics. arXiv:1901.10200 [cs.IR] https://arxiv.org/abs/1901.10200

[78] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. 2022. M5 accuracy competition: Results, findings, and conclusions. *International journal of forecasting* 38, 4 (2022), 1346–1364.

[79] Matthew Middlehurst, Patrick Schäfer, and Anthony Bagnall. 2024. Bake off redux: a review and experimental evaluation of recent time series classification algorithms. *Data Mining and Knowledge Discovery* 38, 4 (2024), 1958–2031.

[80] C. De Mol, E. De Vito, and L. Rosasco. 2008. Elastic-Net Regularization in Learning Theory. arXiv:0807.3423 [stat.ML] https://arxiv.org/abs/0807.3423

[81] PETER B. NEMENYI. 1963. *DISTRIBUTION-FREE MULTIPLE COMPARISONS*. Ph. D. Dissertation. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2023-02-17. https://proxy.lib.ohio-state.edu/login?url=https://www.proquest.com/dissertations-theses/distribution-free-multiple-comparisons/docview/302256074/se-2

[82] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *International Conference on Learning Representations*.

[83] Konstantinos Nikolopoulos, Aris A. Syntetos, John E. Boylan, Fotios Petropoulos, and Vassilios Assimakopoulos. 2011. An Aggregate-Disaggregate Intermittent Demand Approach (ADIDA) to Forecasting: An Empirical Proposition and Analysis. *Journal of the Operational Research Society* 62, 3 (2011), 544–554. doi:10.1057/jors.2010.32

[84] Nixtla. 2026. Nixtla/statsforecast. https://github.com/Nixtla/statsforecast. Python package. Original work published 2021.

[85] Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. 2020. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. arXiv:1905.10437 [cs.LG] https://arxiv.org/abs/1905.10437

[86] Joel Valdivia Ortega, Lorenz Lamm, Franziska Eckardt, Benedikt Schworm, Marion Jasnin, and Tingying Peng. 2025. Randomized-MLP Regularization Improves Domain Adaptation and Interpretability in DINOv2. arXiv:2511.05509 [cs.CV] https://arxiv.org/abs/2511.05509

[87] Keiron O'Shea and Ryan Nash. 2015. An Introduction to Convolutional Neural Networks. arXiv:1511.08458 [cs.NE] https://arxiv.org/abs/1511.08458

[88] Art B. Owen. 2006. A robust hybrid of lasso and ridge regression. https://api.semanticscholar.org/CorpusID:1617819

[89] John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron Elmore, and Michael J Franklin. 2022. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *Proceedings of the VLDB Endowment* 15, 11 (2022), 2774–2787.

[90] John Paparrizos and Michael J Franklin. 2019. Grail: efficient time-series representation learning. *Proceedings of the VLDB Endowment* 12, 11 (2019), 1762–1777.

[91] John Paparrizos and Luis Gravano. 2015. k-shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. ACM, 1855–1870.

[92] John Paparrizos and Luis Gravano. 2016. k-Shape: Efficient and Accurate Clustering of Time Series. *SIGMOD Rec.* 45, 1 (June 2016), 69–76. doi:10.1145/2949741.2949758

[93] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S Tsay, Themis Palpanas, and Michael J Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. *Proceedings of the VLDB Endowment* 15, 8

(2022), 1697–1711.

[94] John Paparrizos, Haojun Li, Fan Yang, Kaize Wu, Jens E d'Hondt, and Odysseas Papapetrou. 2024. A survey on time-series distance measures. *arXiv preprint arXiv:2412.20574* (2024).

[95] John Paparrizos, Ryen W White, and Eric Horvitz. 2016. Screening for pancreatic adenocarcinoma using signals from web search logs: Feasibility study and results. *Journal of oncology practice* 12, 8 (2016), 737–744.

[96] John Paparrizos, Fan Yang, and Haojun Li. 2024. Bridging the gap: A decade review of time-series clustering methods. *arXiv preprint arXiv:2412.20582* (2024).

[97] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* 12, null (Nov. 2011), 2825–2830.

[98] Yan Pei, Swarnendu Biswas, Donald S. Fussell, and Keshav Pingali. 2019. An Elementary Introduction to Kalman Filtering. arXiv:1710.04055 [eess.SY] https://arxiv.org/abs/1710.04055

[99] Yan Pei, Swarnendu Biswas, Donald S. Fussell, and Keshav Pingali. 2019. An Elementary Introduction to Kalman Filtering. arXiv:1710.04055 [eess.SY] https://arxiv.org/abs/1710.04055

[100] Adhistya Erna Permanasari, Indriana Hidayah, and Isna Alfi Bustoni. 2013. SARIMA (Seasonal ARIMA) implementation on time series to forecast the number of Malaria incidence. In *2013 International Conference on Information Technology and Electrical Engineering (ICITEE)*. 203–207. doi:10.1109/ICITEED.2013.6676239

[101] Aladdin Persson. 2026. aladdinpersson/machine-learning-collection. https://github.com/aladdinpersson/Machine-Learning-Collection. Python package. Original work published 2020.

[102] Fotios Petropoulos and Nikolaos Kourentzes. 2015. Forecast combinations for intermittent demand. *Journal of the Operational Research Society* 66, 6 (2015), 914–924. doi:10.1057/jors.2014.62

[103] Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. 2025. TFB: Towards Comprehensive and Fair Benchmarking of Time Series Forecasting Methods. arXiv:2403.20150 [cs.LG] https://arxiv.org/abs/2403.20150

[104] Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. 2025. DUET: Dual Clustering Enhanced Multivariate Time Series Forecasting. arXiv:2412.10859 [cs.LG] https://arxiv.org/abs/2412.10859

[105] Dung Quoc Nguyen, Minh Nguyet Phan, and Ivan Zelinka. 2021. Periodic Time Series Forecasting with Bidirectional Long Short-Term Memory: Periodic Time Series Forecasting with Bidirectional LSTM. In *Proceedings of the 2021 5th International Conference on Machine Learning and Soft Computing* (Da Nang, Viet Nam) *(ICMLSC '21)*. Association for Computing Machinery, New York, NY, USA, 60–64. doi:10.1145/3453800.3453812

[106] Raeid Saqur, Christoph Bergmeir, Blanka Horvath, Daniel Schmidt, Frank Rudzicz, and Terry Lyons. 2026. Seeking SOTA: Time-Series Forecasting Must Adopt Taxonomy-Specific Evaluation to Dispel Illusory Gains. *arXiv preprint arXiv:2603.15506* (2026).

[107] Patrick Schäfer. 2015. The BOSS is concerned with time series classification in the presence of noise. *Data Min. Knowl. Discov.* 29, 6 (nov 2015), 1505–1530. doi:10.1007/s10618-014-0377-7

[108] Patrick Schäfer and Mikael Högqvist. 2012. SFA: a symbolic fourier approximation and index for similarity search in high dimensional datasets. In *Proceedings of the 15th International Conference on Extending Database Technology* (Berlin, Germany) *(EDBT '12)*. Association for Computing Machinery, New York, NY, USA, 516–527. doi:10.1145/2247596.2247656

[109] Robin M. Schmidt. 2019. Recurrent Neural Networks (RNNs): A gentle Introduction and Overview. arXiv:1912.05911 [cs.LG] https://arxiv.org/abs/1912.05911

[110] Patrick Schäfer and Ulf Leser. 2017. Fast and Accurate Time Series Classification with WEASEL. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM '17)*. ACM. doi:10.1145/3132847.3132980

[111] Skipper Seabold and Josef Perktold. 2010. Statsmodels: Econometric and Statistical Modeling with Python. https://github.com/statsmodels/statsmodels. Original work published 2011.

[112] Pavel Senin, Jessica Lin, Xing Wang, Tim Oates, Sunil Gandhi, Arnold P Boedihardjo, Crystal Chen, Susan Frankenstein, and Manfred Lerner. 2014. Grammarviz 2.0: a tool for grammar-based pattern discovery in time series. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part III 14*. Springer, 468–472.

[113] Oleksandr Shchur, Abdul Fatir Ansari, Caner Turkmen, Lorenzo Stella, Nick Erickson, Pablo Guerron, Michael Bohlke-Schneider, and Yuyang Wang. 2026. fev-bench: A Realistic Benchmark for Time Series Forecasting. arXiv:2509.26468 [cs.LG] https://arxiv.org/abs/2509.26468

[114] Jin Shieh and Eamonn Keogh. 2008. iSAX: indexing and mining terabyte sized time series. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Las Vegas, Nevada, USA) *(KDD '08)*. Association for Computing Machinery, New York, NY, USA, 623–631. doi:10.1145/1401890.1401966

[115] Olivier Sprangers, Sebastian Schelter, and Maarten de Rijke. 2021. Parameter Efficient Deep Probabilistic Forecasting. arXiv:2112.02905 [cs.LG] https://arxiv.org/abs/2112.02905

[116] Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. 2024. TEST: Text Prototype Aligned Embedding to Activate LLM's Ability for Time Series. arXiv:2308.08241 [cs.CL] https://arxiv.org/abs/2308.08241

[117] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. arXiv:1409.3215 [cs.CL] https://arxiv.org/abs/1409.3215

[118] Peiwang Tang and Weitai Zhang. 2024. Unlocking the Power of Patch: Patch-Based MLP for Long-Term Time Series Forecasting. arXiv:2405.13575 [cs.LG] https://arxiv.org/abs/2405.13575

[119] Sean J. Taylor and Benjamin Letham. 2018. Forecasting at Scale. *The American Statistician* 72, 1 (Jan. 2018), 37–45. doi:10.1080/00031305.2017.1380080

[120] Robert Tibshirani. 1996. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58, 1 (1996), 267–288. doi:10.1111/j.2517-6161.1996.tb02080.x

[121] Michael E. Tipping. 2001. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.* 1 (Sept. 2001), 211–244. doi:10.1162/15324430152748236

[122] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. arXiv:1706.03762 [cs.CL] https://arxiv.org/abs/1706.03762

[123] Huiqiang Wang, Jian Peng, Feihu Huang, Jince Wang, Junhui Chen, and Yifei Xiao. 2023. MICN: Multi-scale Local and Global Context Modeling for Long-term Series Forecasting. (2023).

[124] H. Wang, J. Shi, and L. Qing. 2026. FACT: Fine-grained Across-Variable Convolution for Multivariate Time Series Forecasting. In *The Fourteenth International Conference on Learning Representations (ICLR)*. https://openreview.net/forum?id=j3gNYqrHtl

[125] Shiyu Wang, Jiawei Li, Xiaoming Shi, Zhou Ye, Baichuan Mo, Wenze Lin, Shengtong Ju, Zhixuan Chu, and Ming Jin. 2025. TimeMixer++: A General Time Series Pattern Machine for Universal Predictive Analysis. arXiv:2410.16032 [cs.LG] https://arxiv.org/abs/2410.16032

[126] Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Chen Wang, Mingsheng Long, and Jianmin Wang. 2025. Deep Time Series Models: A Comprehensive Survey and Benchmark. arXiv:2407.13278 [cs.LG] https://arxiv.org/abs/2407.13278

[127] Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. 2024. TimeXer: Empowering Transformers for Time Series Forecasting with Exogenous Variables. arXiv:2402.19072 [cs.LG] https://arxiv.org/abs/2402.19072

[128] Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. 2024. TimeXer: Empowering Transformers for Time Series Forecasting with Exogenous Variables. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=INAeUQ04lT

[129] David Wipf and Srikantan Nagarajan. 2007. A new view of automatic relevance determination. In *Proceedings of the 21st International Conference on Neural Information Processing Systems* (Vancouver, British Columbia, Canada) *(NIPS'07)*. Curran Associates Inc., Red Hook, NY, USA, 1625–1632.

[130] Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. 2024. Unified Training of Universal Time Series Forecasting Transformers. arXiv:2402.02592 [cs.LG] https://arxiv.org/abs/2402.02592

[131] Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. 2022. ETSformer: Exponential Smoothing Transformers for Time-series Forecasting. arXiv:2202.01381 [cs.LG] https://arxiv.org/abs/2202.01381

[132] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. 2023. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. arXiv:2210.02186 [cs.LG] https://arxiv.org/abs/2210.02186

[133] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. 2022. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. arXiv:2106.13008 [cs.LG] https://arxiv.org/abs/2106.13008

[134] Zhijian Xu, Ailing Zeng, and Qiang Xu. 2024. FITS: Modeling Time Series with $10k$ Parameters. arXiv:2307.03756 [cs.LG] https://arxiv.org/abs/2307.03756

[135] Fan Yang and John Paparrizos. 2025. SPARTAN: Data-Adaptive Symbolic Time-Series Approximation. *Proc. ACM Manag. Data* 3, 3, Article 220 (June 2025), 30 pages. doi:10.1145/3725357

[136] Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Defu Lian, Ning An, Longbing Cao, and Zhendong Niu. 2023. Frequency-domain MLPs are More Effective Learners in Time Series Forecasting. arXiv:2311.06184 [cs.LG] https://arxiv.org/abs/2311.06184

[137] Guoqi Yu, Jing Zou, Xiaowei Hu, Angelica I Aviles-Rivero, Jing Qin, and Shujun Wang. 2024. Revitalizing Multivariate Time Series Forecasting: Learnable Decomposition with Inter-Series Dependencies and Intra-Series Variations Modeling. In *Forty-first International Conference on Machine Learning*.

https://openreview.net/forum?id=87CYNyCGOo

[138] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. 2022. Are Transformers Effective for Time Series Forecasting? arXiv:2205.13504 [cs.AI] https://arxiv.org/abs/2205.13504

[139] Tianping Zhang, Yizhuo Zhang, Wei Cao, Jiang Bian, Xiaohan Yi, Shun Zheng, and Jian Li. 2022. Less Is More: Fast Multivariate Time Series Forecasting with Light Sampling-oriented MLP Structures. arXiv:2207.01186 [cs.LG] https://arxiv.org/abs/2207.01186

[140] Yunhao Zhang and Junchi Yan. 2023. Crossformer: Transformer Utilizing Cross-Dimension Dependency for Multivariate Time Series Forecasting. In *International Conference on Learning Representations*.

[141] Zhiyuan Zhao, Haoxin Liu, and B Aditya Prakash. 2025. Tackling Time-Series Forecasting Generalization via Mitigating Concept Drift. *arXiv preprint arXiv:2510.14814* (2025).

[142] Zhiyuan Zhao, Juntong Ni, Shangqing Xu, Haoxin Liu, Wei Jin, and B Aditya Prakash. 2025. TimeRecipe: A Time-Series Forecasting Recipe via Benchmarking Module Level Effectiveness. *arXiv preprint arXiv:2506.06482* (2025).

[143] Zhiyuan Zhao, Alexander Rodriguez, and B Aditya Prakash. 2023. Performative time-series forecasting. *arXiv preprint arXiv:2310.06077* (2023).

[144] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. arXiv:2012.07436 [cs.LG] https://arxiv.org/abs/2012.07436

[145] Tian Zhou, Ziqing Ma, Xue wang, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, and Rong Jin. 2022. FiLM: Frequency improved Legendre Memory Model for Long-term Time Series Forecasting. arXiv:2205.08897 [cs.LG] https://arxiv.org/abs/2205.08897

[146] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. 2022. FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting. arXiv:2201.12740 [cs.LG] https://arxiv.org/abs/2201.12740

[147] Tian Zhou, PeiSong Niu, Xue Wang, Liang Sun, and Rong Jin. 2023. One Fits All:Power General Time Series Analysis by Pretrained LM. arXiv:2302.11939 [cs.LG] https://arxiv.org/abs/2302.11939