Failing the Test of Time: Do Text and Time-Series Benchmarks Measure Time-Series Understanding?

Anonymous authors
Paper under double-blind review

Abstract

Recent advancements in foundation models, particularly Large Language Models (LLMs), has provided new opportunities for time-series analysis, where languages are natural media of parallel information accompanying time-series—substantial pioneering research has recently begun to jointly model time-series and text. While the recent works are of fundamental importance, we observed a somewhat surprising phenomenon: in the widely used benchmarks, the evaluated LLMs can often achieve high accuracy, *without actually consulting the time-series*, raising essential concerns on whether the existing benchmarks for joint time-series and language analysis actually measure the intended model capabilities. We provide comprehensive studies on a sphere of widely deployed LLMs, across publicly available benchmarks, and under different input conditions with original input, masked conditions, noisy corruption, and modified time-series values. We leverage Bloom's-taxonomy analysis to reveal the relationship with benchmark complexity. Our study carried on complementary analyses provides comprehensive evidence and insights on the potential misalignment between the existing benchmarks and their objectives. We hope the research helps redirect the research to focus on more rigorous assessment.

1 Introduction

Time-series data underpins decision-making across many domains of modern life. Financial markets generate continuous streams of price and volume data on which large numbers of trades depend, clinical settings rely on physiological signals to monitor patient health, climate science depends on long-running observational records to characterize a changing planet, and industrial operations track sensor measurements to maintain critical infrastructure. The economic and societal stakes of correctly interpreting temporal numerical signals are difficult to overstate. Recent advancements in foundation models, particularly Large Language Models (LLMs), has provided new opportunities for time-series analysis, where languages are natural media of parallel information accompanying time-series. Accurate joint reasoning over numeric series and natural language has become a prerequisite for deploying LLMs in finance (Chen et al., 2025), weather (Li et al., 2025), healthcare (Chan et al., 2024), and many other settings (Kong et al., 2025c). Time-series reasoning that combines world knowledge with quantitative analysis has accordingly become an increasingly popular evaluation target for LLMs (Merrill et al., 2024; Gruver et al., 2023; Quinlan et al., 2026), with LLMs also being used to improve the explainability of time-series analysis systems (Jiang et al., 2025). Recently, several benchmarks for multimodal time-series question answering (QA) have been proposed, aiming to test an LLM's ability to answer time-series related questions in a zero-shot setting. Each LLM uses a combination of text and time-series data to answer multiple choice questions in natural language with varying degrees of complexity.

Given the fundamental importance of joint modeling, we observed a rather surprising phenomenon: in the widely used benchmarks, the evaluated LLMs can often achieve high accuracy, *without actually consulting the time-series*, raising essential concerns on whether the existing benchmarks for joint time-series and language analysis actually measure the intended model capabilities. For example, on MTBench-Finance, GPT-4.1-nano achieves 91.7% accuracy when the time-series is completely masked (replaced with placeholder tokens) versus 89.7% accuracy with the full time-series present—the model actually performs comparable or slightly

better without the numeric data it is purportedly reasoning over (example shown in Figure 1). Across three of four benchmarks, masking out the time-series or even replacing it with random noise results in only a minor drop, or sometimes an *increase*, in accuracy. The counterintuitive result suggests that the models are exploiting shortcuts or clues in text, rather than performing the desired numerical reasoning over time-series.

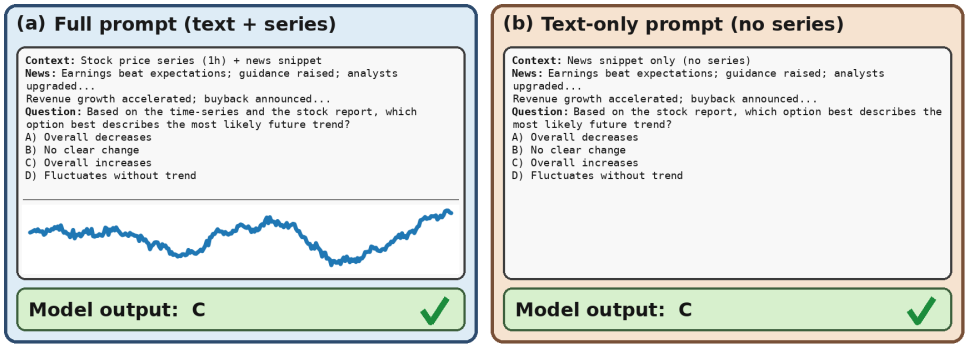


Figure 1: Model produces the same correct answer with and without the time-series. Note that we find this trend even for examples with limited context such as those in the MCQ2TS dataset (Merrill et al., 2024).

We hypothesize that **as time-series reasoning datasets become more complex, they transition from testing time-series-specific reasoning (that is reasoning that is tied to the analysis of the time-series) to testing general reasoning abilities (reasoning only using the provided text query and solution options)**. Simple questions that require directly using time-series values and time-series specific knowledge about the series prove difficult for LLMs. Meanwhile, more complex questions that incorporate rich textual context inadvertently provide alternative solution paths through pure language understanding, allowing models to bypass the time-series entirely. This misalignment between benchmark intent and model behavior has serious implications: current evaluations may be measuring general language comprehension and reasoning skills rather than the targeted multimodal time-series reasoning capabilities.

In this paper, we evaluate recent LLMs on various time-series QA benchmarks under different conditions, with and without time-series data. We explore the decisive factors, showing the key role of question complexity in how these benchmarks behave — the level of cognitive demand that a question places on the reader can directly shapes whether the numeric series is needed to answer it: simple questions can force genuine reliance on the series, while more complex questions tend to supply richer textual context that opens alternative reasoning paths and allows models to bypass the numeric data entirely. To provide a comprehensive study, we develop a complexity-aware evaluation that combines a masking-based ablation approach with metrics for model confidence, calibration, and answer consistency under different input conditions. We show that the benchmarks we study often fail to robustly test time-series reasoning under the text-serialized encoding that dominates current practice, particularly as complexity increases, and that under this encoding the time-series data has little impact on model responses for the models we evaluate. Our analysis is organized around the following research questions:

- **RQ1**: Do LLMs rely on time-series data when answering multimodal time-series questions in the existing benchmarks, and to what extent?
- **RQ2**: Do benchmarks that pose more complex questions actually demand stronger time-series reasoning?
- **RQ3**: How confidence, answer consistency, and ablations reveal whether existing benchmarks genuinely measure joint time-series and language reasoning?

Our work provides insights into current problems with time-series reasoning from both a benchmark design and model behavior and calibration perspective for the broader research community. We provide a complete

package of code and a consolidated download link for all datasets to help ensure the easy reproduction and extension of our results at [GITHUB_PLACEHOLDER_For_ANONYMITY].

2 Related Work

Multimodal Time-Series Question Answering. There has been growing interest in combining time-series data with natural language for QA and reasoning tasks. MT Bench (Chen et al., 2025) pairs stock time-series with financial news and weather time-series with reports to create cross-modal QA tasks. Related benchmark efforts include Time-MQA (Kong et al., 2025a), TS-Exam (Cai et al., 2024), and ChatTime (Wang et al., 2025), which span domains such as finance, weather, and healthcare and cover question types ranging from trend identification and anomaly detection to higher-level causal or explanatory prompts. Merrill et al. (2024) introduce a set of controlled datasets for time-series reasoning, including counterfactual questions over *two* time-series (MCQ2TS), and highlight that some *single*-series constructions can be compromised by strong language-only baselines. These datasets provide a diverse set of time-series reasoning benchmarks for evaluating general time-series reasoning capabilities.

Text serialization as the default encoding for LLM-based time-series QA. A key practical design choice in LLM-based time-series QA is how the numeric sequence is represented in the prompt. Across most LLM-centric time-series reasoning and QA work, the predominant approach is to *serialize the time-series into text tokens* (e.g., timestamp–value pairs, delimited lists, or compact textual templates) and concatenate it with the natural-language context and question (Merrill et al., 2024; Chen et al., 2025; Cai et al., 2024; Xie et al., 2025; Chow et al., 2024; Jin et al., 2024b). Even in the closely related forecasting literature that uses LLMs, a common strategy is to cast sequences into token strings (or lightly transformed textual sequences) to exploit the LLM's native interface (Gruver et al., 2023; Jin et al., 2024a). Exploring alternative encodings (e.g., learned numeric embeddings or dedicated time-series tokenizers) is therefore best viewed as *orthogonal*: it may improve how models ingest sequences, but it does not remove the need for benchmarks whose answers cannot be recovered from text priors alone.

Shortcut learning, dataset artifacts, and partial-input baselines. Exploiting spurious correlations is well documented in both unimodal and multimodal QA. In visual question answering (VQA), models can often answer from question priors with limited image use, motivating balanced constructions and diagnostic evaluation (Goyal et al., 2017; Agrawal et al., 2018). In natural language inference and reading-comprehension style datasets, hypothesis-only or question-only baselines can be surprisingly strong due to annotation artifacts (Poliak et al., 2018; Gururangan et al., 2018). These behaviors are instances of *shortcut learning*, where models latch onto non-causal but predictive features that hold in-distribution (Geirhos et al., 2020). Accordingly, evaluation methodologies that probe robustness, such as behavioral test suites (Ribeiro et al., 2020) and contrast sets that minimally perturb examples (Gardner et al., 2020)—have been proposed to detect when accuracy overstates intended capability.

Confidence and calibration under modality ablations. Finally, reliability metrics complement accuracy when diagnosing whether models are grounded in evidence. Expected calibration error (ECE) is a standard summary statistic connecting predicted confidence to empirical correctness (Guo et al., 2017). Prior work has shown that pre-trained transformers exhibit varying calibration properties across in-domain and out-of-domain settings (Desai & Durrett, 2020), and that models may remain overconfident even when answer quality degrades (Jiang et al., 2021). In multimodal settings, several diagnostic approaches have been proposed to test whether models genuinely rely on visual information: Shekhar et al. (2017) introduced foiled captions with single-word substitutions to probe vision-language grounding, Thomason et al. (2019) demonstrated that unimodal baselines can be surprisingly strong on visual QA tasks, and Parcalabescu et al. (2022) built a task-independent benchmark requiring models to ground linguistic phenomena in images. Our work extends these methods to time-series QA, using masking-based ablations alongside calibration and consistency metrics to measure how much models actually depend on the temporal numerical data they are purportedly reasoning over.

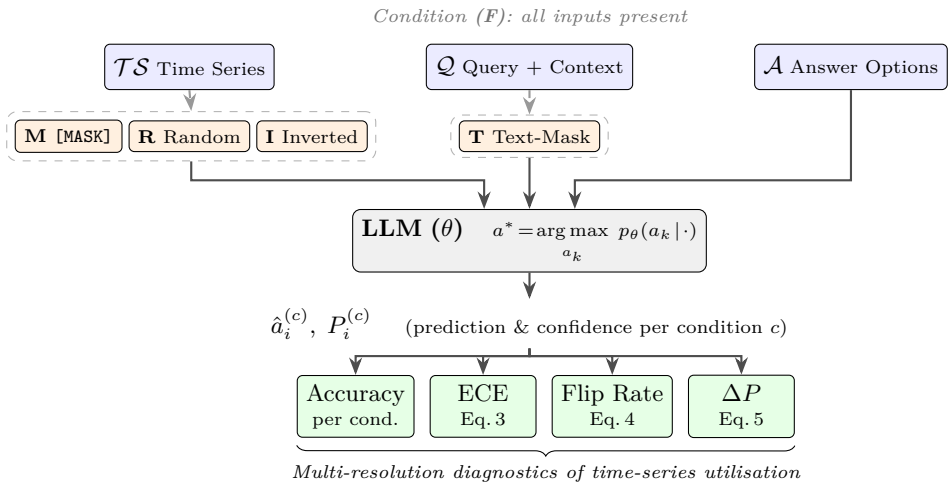


Figure 2: **Evaluation pipeline.** Each column shows an input modality and its ablated variants (dashed boxes). Under the Full **(F)** condition the original inputs are used; under ablation, $\mathcal{TS}$ is replaced by a masked (**M**), randomised (**R**), or inverted (**I**) variant, and $\mathcal{Q}$ is reduced to a minimal prompt (**T**). $\mathcal{A}$ is never modified. For every question and condition $c \in \{F, M, R, I, T\}$ we record the predicted answer $\hat{a}_i^{(c)}$ and its token-level confidence $P_i^{(c)}$, from which we derive four complementary diagnostics.

3 Problem Setup

3.1 The Problem

Time-series analysis encompasses a wide range of tasks, including forecasting, anomaly detection, classification, and question answering. Following the prior work (Merrill et al., 2024; Chen et al., 2025; Cai et al., 2024), we set up our study in a Question-Answering (QA) setting, in which a model reasons over a numeric time series and textual context in order to identify a correct answer to a natural-language question from a set of options.

Foramally, let $\mathcal{TS} = \{(\tau_t, v_t)\}_{t=1}^{N}$ denote a time series of N observations, where τ_n is a timestamp and $v_n \in \mathbb{R}$ is the corresponding numeric value. Let $\mathcal{Q}$ denote a natural-language query together with any supplementary text (e.g. news articles, weather reports), and let $\mathcal{A} = \{a_1, \ldots, a_K\}$ denote a set of K candidate answers. A model parameterised by θ assigns a probability to each candidate, conditioned jointly on the query, the answer options, and the time series; the model must reason over $\mathcal{A}$ as well as $\mathcal{Q}$ and $\mathcal{TS}$, so we retain $\mathcal{A}$ explicitly in the conditioning set. The objective function for the model is as follows.

$$\hat{a} \;=\; \arg\max_{a_k \in \mathcal{A}} \; p_\theta\big(a_k \mid \mathcal{Q},\, \mathcal{A},\, \mathcal{TS}\big), \tag{1}$$

where $p_\theta(a_k \mid \cdot)$ is the softmax probability the model assigns to the first token of option label a_k (see Section 3.1). Of the four inputs to p_θ, two are held fixed across all experimental conditions: the parameters θ (inherent to the model) and the answer set $\mathcal{A}$ (required for the multiple-choice format and itself part of what the model reasons over). The remaining two inputs, $\mathcal{Q}$ and $\mathcal{TS}$, can be selectively ablated.

Inference conditions. To measure the causal contribution of each modality we define five input conditions, each modifying exactly one component of Eq. equation 1 (see Figure 2). In every condition we evaluate $\arg\max_{a_k \in \mathcal{A}} p_\theta(a_k \mid \widetilde{\mathcal{Q}}, \mathcal{A}, \widetilde{\mathcal{TS}})$ for the specified choice of $\widetilde{\mathcal{Q}}$ and $\widetilde{\mathcal{TS}}$:

(F) Full. $\widetilde{\mathcal{Q}} = \mathcal{Q}$, $\widetilde{\mathcal{TS}} = \mathcal{TS}$. Unmodified input.

(M) Series-Masked. $\widetilde{\mathcal{TS}} = \mathcal{TS}_{\text{mask}}$, where each numeric value v_t is replaced by a placeholder `[MASK]` token. Timestamps and prompt scaffolding are preserved so total token count is approximately unchanged; only the numeric content is removed.

(R) Noise. $\widetilde{\mathcal{TS}} = \mathcal{TS}_{\text{rand}}$, a random numeric sequence of equal length and surface format. Structural appearance is preserved but semantic content is destroyed.

(I) Inverted. $\widetilde{\mathcal{TS}} = \mathcal{TS}_{\text{inv}}$ with $\tilde{v}_n = 2\bar{v} - v_n$, reflecting values about their mean. Directional trends are reversed while scale and sequence length are preserved.

(T) Text-Masked. $\widetilde{\mathcal{Q}} = \mathcal{Q}_{\text{min}}$, where all contextual text (context passages, question stems, news/weather reports) is replaced by `[MASK]` tokens; the numeric series is kept verbatim and the `A`–`D` option lines are retained so the model can still select an answer. This isolates the contribution of the numeric signal.

If a benchmark genuinely requires time-series reasoning, accuracy under conditions **M**, **R**, and **I** should approach the chance baseline $1/K$. A model that retains high accuracy despite corrupted or absent series reveals one of two failure modes: *(i)* the benchmark admits textual shortcuts, or *(ii)* the model fails to incorporate numeric evidence even when it is present. These different conditions will help us to address research questions **RQ1**.

Per-item quantities. Let j denote the number of questions in an evaluation set, indexed by $i = 1, \ldots, j$, and let $a_i^\star \in \mathcal{A}_i$ be the gold answer for question i. For each condition $c \in \{F, M, N, I, T\}$ and question i we record:

- The **predicted answer** $\hat{a}_i^{(c)} = \arg\max_{a_k \in \mathcal{A}_i} p_\theta^{(c)}(a_k \mid \cdot)$ and its correctness $\mathbf{1}[\hat{a}_i^{(c)} = a_i^\star]$;
- The **token-level confidence** $P_i^{(c)} = \max_{a_k} p_\theta^{(c)}(a_k \mid \cdot)$, the softmax probability assigned to the first token of the emitted option label. When the provider returns only top-k logprobs and the emitted label lies outside the top-k, the item is excluded from analyses that depend on $P_i^{(c)}$ (ECE and ΔP); accuracy and flip-rate analyses always include all items.

Diagnostic metrics. From the per-item quantities above we derive four complementary diagnostics that together characterise how much the model relies on $\mathcal{TS}$ versus $\mathcal{Q}$.

(i) Accuracy under condition c:

$$\text{Acc}^{(c)} \;=\; \frac{1}{j} \sum_{i=1}^{j} \mathbf{1}\big[\hat{a}_i^{(c)} = a_i^\star\big]. \tag{2}$$

(ii) Expected Calibration Error (ECE). Partition items by their confidence $P_i^{(c)}$ into $B = 10$ equal-width bins $\{\mathcal{B}_b\}_{b=1}^{B}$ over $[0, 1]$, and let $\text{acc}(\mathcal{B}_b)$ and $\text{conf}(\mathcal{B}_b)$ be the empirical accuracy and mean confidence within bin $\mathcal{B}_b$:

$$\text{ECE}^{(c)} \;=\; \sum_{b=1}^{B} \frac{|\mathcal{B}_b|}{j} \,\big|\text{acc}(\mathcal{B}_b) - \text{conf}(\mathcal{B}_b)\big|. \tag{3}$$

(iii) Flip rate between two conditions. The fraction of items whose predicted option differs between conditions c and c':

$$\text{Flip}(c, c') \;=\; \frac{1}{j} \sum_{i=1}^{j} \mathbf{1}\big[\hat{a}_i^{(c)} \neq \hat{a}_i^{(c')}\big]. \tag{4}$$

We primarily report $\text{Flip}(F, M)$ and $\text{Flip}(F, T)$.

(iv) Confidence shift relative to the Full condition. For each ablation $c \in \{M, R, I, T\}$ we measure how much removing the corresponding modality moves the model's belief in the answer it emits under Full:

$$\Delta P_i^{(c)} \;=\; P_i^{(F)} - P_i^{(c)}. \tag{5}$$

For brevity we write $\Delta P^{\mathrm{TS}} := \Delta P^{(M)}$ (series removed) and $\Delta P^{\mathrm{TXT}} := \Delta P^{(T)}$ (text removed). A distribution of $\Delta P^{(c)}$ concentrated at zero indicates the ablated modality has negligible causal effect on the model's confidence; systematic positive or negative shifts reveal modality reliance. Comparing ΔP^{TS} against ΔP^{TXT} on matched instances directly contrasts the contribution of the series with that of the text.

Together, accuracy under ablation quantifies *whether* the model needs the series to answer correctly, the flip rate captures *how often* individual predictions change, and $\Delta P^{(c)}$ reveals *how much* the model's internal confidence shifts, providing a multi-resolution view of time-series utilisation (Figure 2). The measured confidence, answer consistency, and ablation results will help us to address research question **RQ3**.

3.2 Query Complexity

To address the research question **RQ2**, we investigate the key role of question complexity in affecting LLMs' behavior in the existing benchmarks. The cognitive demand that a question places on the reader can shape LLMs exploration on the numeric series — in our preliminary study we observed that benchmarks with different complexity profiles produce very different patterns of reliance on the time-series. To systematically study this, we propose to characterize each question on a coarse complexity scale grounded in Bloom's Taxonomy (Bloom, 1956; Adams, 2015). Bloom's hierarchy was originally developed to contextualize objectives and skills for student learning, and consists of six levels: (1) Remember, (2) Understand, (3) Apply, (4) Analyze, (5) Evaluate, and (6) Create. We propose collapsing these six levels into three clear and distinct categories suitable for time-series QA. *C1* corresponds to the lower end of Bloom's taxonomy (Remember/Understand): questions that involve basic identification or comprehension of time-series information (e.g., reading a value, recognizing a trend). *C2* aligns with the middle levels (Apply/Analyze): questions that require applying knowledge of time-series concepts or performing multi-step reasoning, possibly combining text and data (e.g., explaining an anomaly, comparing segments). *C3* corresponds to the highest levels (Evaluate/Create): questions that demand drawing new conclusions from multiple distinct sources or synthesis of information (e.g., predicting an outcome or hypothesizing about a causal relationship). We assign each question in the labeled subset a complexity label (C1, C2, or C3) based on this schema, and use these labels to evaluate how model performance correlates with complexity.

Annotation methodology. Assigning complexity to time-series QA items is non-trivial: the rubric requires simultaneously parsing a numeric series, a textual context, and a question, and then mapping them to the Bloom's-taxonomy category that most tightly describes the cognitive demand. In a pilot study, we found that non-expert annotators produced low inter-annotator agreement on this task, primarily because reliably distinguishing C2 from C3 requires familiarity with both time-series reasoning and the relevant textual domain (finance, weather, etc.). Rather than scale up non-expert annotation, we adopted a two-stage protocol: an LLM grader produces candidate labels for all items at temperature 0, and an author with domain expertise audits a random subset for agreement. We uniformly sample 200 items from each dataset (800 total) for labeling, and present the grader with the full example (text context, serialized time-series, question, and answer options) alongside a concise rubric with definitions and examples for C1/C2/C3 derived from Bloom's taxonomy. The grader is run deterministically (temperature = 0) and returns a single label per item. On a stratified 60-item audit (15 per dataset), author–grader agreement was 78.3%, with unweighted $\kappa = 0.67$ and linear-weighted $\kappa = 0.73$ (substantial agreement on the Landis–Koch scale). We treat the audited subset as a calibration check on the LLM grader; per-dataset agreement, label distributions, and the full rubric, prompt, and audit procedure are provided in Appendix B.3. We then use the grader's per-item labels for all complexity analyses (RQ2).

We emphasize that neither human nor LLM-based annotation is free of bias: expert human raters bring their own disciplinary priors, and an LLM grader inherits any systematic tendencies of its training distribution. Our audit establishes agreement between the grader and expert judgment on the sampled population but does not rule out correlated errors on unsampled items. We view the resulting labels as a coarse instrument

suitable for the population-level trends reported in Section 5.2 rather than as ground truth for any individual item.

4 Experiment Setup

4.1 Model & Dataset Selection

We consider several multimodal datasets where each example consists of: (1) a time-series (a sequence of numeric values with timestamps), (2) accompanying textual information (such as a description, news snippet, or explanatory context), and (3) a question that the model must answer (we focus on multiple-choice QA rather than open-ended generation). We focus on four representative benchmark datasets: MCQ2TS (Merrill et al., 2024), TS-Exam (Cai et al., 2024), MTBench-Finance (Chen et al., 2025), and MTBench-Weather (Chen et al., 2025). We chose these datasets due to both their popularity, varying curation methods and complexity. These cover a range of domains (counterfactual reasoning with synthetic data, exam-style QA on time-series properties, financial stock data with news context, and weather data with report context, respectively) and include both synthetic and real-world scenarios. Each dataset contains between 700–1000 question-answer pairs. In the case of MCQ2TS, we randomly sample 1000 instances from the full set. For more information on dataset selection and other datasets we considered please refer to Appendix B.

We evaluate multiple proprietary LLMs from frontier research labs on these datasets. In particular, we test GPT-4.1-nano (OpenAI et al., 2024), DeepSeekV3 (DeepSeek-AI, 2025), Gemini-2.5-flash-lite (Team et al., 2024) and Cohere Command-R (Cohere Labs, 2024) . These models vary in size and training origin, providing a diverse set of reasoning capabilities. In keeping with recent time-series reasoning works and implementations from the original dataset evaluations, we use text-based encoding for each time-series (Merrill et al., 2024; Xie et al., 2025; Chow et al., 2024; Jin et al., 2024b; Cai et al., 2024; Chen et al., 2025).

4.2 Time-Series Encoding & Masking

Because *text encoding is the de facto standard* for LLM-based evaluation pipelines, we adopt the same family of encodings as prior work (Merrill et al., 2024; Chen et al., 2025; Cai et al., 2024; Xie et al., 2025; Chow et al., 2024; Jin et al., 2024b; Cai et al., 2024) to ensure comparability in line with previous works. Crucially, our main conclusions are drawn from time-series encoding modulations (masking, noise substitution, and inversion) that preserve prompt structure while selectively breaking time-series representation. This isolates (i) whether benchmark items admit text-only solution paths and (ii) whether model confidence and decisions are sensitive to the time-series under the representation used by most prior LLM-based work (Merrill et al., 2024; Chen et al., 2025; Gruver et al., 2023).

For masking experiments we replace all numeric time-series values with a placeholder token (e.g., a generic `[MASK]` symbol). The goal is to remove the actual time-series information without significantly changing the length or structure of the input. This is important especially for prompts where text and time-series data are interleaved, as we want to preserve the overall context format. One concern is that by removing the time-series, we might inadvertently make it easier for the model to ignore that modality. To check this, we also evaluate the model under a condition where the time-series is replaced with random numeric noise. This random sequence maintains the appearance of a time-series (a list of numbers in sequence) but carries no relevant information, serving as a control to ensure that the model isn't simply using the presence of tokens or their positions. Finally, we also test an *inverted* condition where the time-series values are inverted around the mean, which disrupts the numeric trends while preserving the sequence structure.

5 Results

5.1 RQ1: Do LLMs rely on time-series data when answering multimodal time-series questions in the existing benchmarks, and to what extent?

Table 1 summarizes each model's accuracy (% of correct questions answered) on each dataset under four conditions: Full (F), Masked (M), Noise (R), and Inverted (I). We also include the random baseline accuracy

as the number of options in the Time-Series Exam baseline is variable while for the other datasets there are always four options. Uncertainty estimates for each accuracy value are computed using non-parametric bootstrap resampling (Efron & Tibshirani, 2006). For a given (model, dataset, condition) with n test samples, we generate B=1,000 bootstrap resamples by repeatedly drawing n samples with replacement from the observed binary outcomes $\{1[\hat{y}_i = y_i]\}_{i=1}^{n}$. We compute the accuracy for each of the B resamples and report the 2.5th and 97.5th percentiles of the resulting distribution as a 95% confidence interval. The reported "±" value is the half-width of this interval.

In line with our hypothesis we observe an interesting trend. For the TS-Exam benchmark, the simplest of the time-series reasoning benchmarks, removing the time-series causes the models to drop to near-random accuracy on this benchmark.

In almost every other case, the performance under M or R is comparable to (and occasionally even slightly higher than) performance under the full input. Even when the time-series values are entirely replaced with random noise or negated (I), the accuracy remains essentially unchanged. We do not see the dramatic accuracy collapse one would expect if the time-series information were truly crucial to answering the questions. In fact, in many settings the model does marginally *better* without the genuine time-series, suggesting the original data might have introduced distractions or inconsistencies that hindered the model.

Table 1: Per–model accuracy (in %), shown as mean ± 95% bootstrap confidence intervals under Full (F), Mask (M), Noise (R), Inverted (I). Bold (M/R/I) indicates within 2 pp of F; red indicates M/R/I > F.

Time-Series Exam				
Model	**F**	**M**	**R**	**I**
Random Baseline		.4013		
command-r-08-2024	.448 ± .036	.387 ± .034	.393 ± .036	.416 ± .036
Deepseek-V3	.539 ± .034	.398 ± .036	.395 ± .035	.511 ± .037
gemini-2.5-flash-lite	.488 ± .037	.394 ± .036	.378 ± .034	**.480 ± .036**
gpt-4.1-nano	.436 ± .036	.377 ± .037	.375 ± .035	**.428 ± .034**

MCQ2TS				
Model	**F**	**M**	**R**	**I**
Random Baseline		.25		
command-r-08-2024	.380 ± .031	.343 ± .030	**.363 ± .030**	.336 ± .030
Deepseek-V3	.510 ± .032	.623 ± .031	.635 ± .029	.465 ± .032
gemini-2.5-flash-lite	.399 ± .030	.484 ± .031	.480 ± .032	.363 ± .029
gpt-4.1-nano	.367 ± .029	.507 ± .030	.502 ± .031	.339 ± .028

MTBench-Finance				
Model	**F**	**M**	**R**	**I**
Random Baseline		.25		
command-r-08-2024	.740 ± .039	**.729 ± .037**	.705 ± .039	.713 ± .040
Deepseek-V3	.890 ± .026	**.890 ± .027**	**.895 ± .026**	.797 ± .035
gemini-2.5-flash-lite	.857 ± .029	**.851 ± .032**	**.849 ± .030**	.800 ± .033
gpt-4.1-nano	.897 ± .025	**.917 ± .024**	**.913 ± .024**	.872 ± .029

MTBench-Weather				
Model	**F**	**M**	**R**	**I**
Random Baseline		.25		
command-r-08-2024	.471 ± .035	**.486 ± .034**	**.482 ± .036**	.515 ± .036
Deepseek-V3	.571 ± .035	**.574 ± .035**	**.566 ± .034**	**.563 ± .034**
gemini-2.5-flash-lite	.543 ± .037	**.523 ± .037**	**.537 ± .035**	**.537 ± .037**
gpt-4.1-nano	.613 ± .034	**.602 ± .037**	**.595 ± .035**	**.607 ± .035**

These results highlight two important points. First, model performance remains surprisingly high even when the time-series data is not actually available to the model. Second, the difference between having the real time-series versus none (or random/inverted data) is minimal across the board. For GPT-4.1-nano, for example, the accuracy difference between full input and masked input is only on the order of 2–3 percentage points (and in some cases the masked variant is higher). For DeepSeekV3, the differences are often under 1 percentage point. In several conditions (marked in red in the table), the model without true time-series data actually outperforms the model with it. This counterintuitive phenomenon suggests that, under text serialization, the time-series content provides little marginal signal to these models as textual context becomes richer. We take a more detailed look at time-series complexity with relation to performance in Section 5.2.

A natural hypothesis is that models are exploiting artifacts or biases in the datasets. Perhaps many questions contain clues in the text or answer options that allow the model to guess correctly without real data. We explore this next by trying to filter out such "cheatable" questions.

5.1.1 Filtering Out "Cheatable" Questions Does Not Solve the Problem

To sanity-check whether our main findings are driven by a subset of easily-guessed questions, we construct a *hard subset* of each dataset by removing every question that GPT-4.1-nano answered correctly in the Masked condition. The intent of this test is to ask whether removing items that any single model can solve without the series changes the qualitative picture for the other models. We chose GPT-4.1-nano as the filter model

because it exhibited among the strongest masked-condition performance in Table 1, making it a conservative filter: any item it solves without the series is a plausible candidate for a text-only shortcut.

This procedure filtered out 1,699 of the 2,990 total items (56.8%). As expected, the pruning is highly uneven across datasets (e.g., 91.7% of MTBench-Finance versus 37.7% of TS-Exam), which itself is informative: it indicates that MTBench-Finance admits text-only solutions at scale, whereas TS-Exam does not.

Table 2: Hard–subset per–model accuracy (proportion) under Full (F), Mask (M), Noise (R), Inverted (I). Values are Hard–subset with Δ vs. Original in parentheses. Samples excluded: 1699 (solvable by `gpt-4.1-nano` without time-series). Time-Series Exam hard–subset random baseline: .478; others: .25.

Time-Series Exam (Hard Subset)				
Model	**F**	**M**	**R**	**I**
Random Baseline		.478		
command-r-08-2024	.368 (-.080)	.305 (-.082)	.308 (-.085)	.335 (-.081)
Deepseek-V3	.456 (-.083)	.295 (-.103)	.295 (-.100)	.398 (-.113)
gemini-2.5-flash-lite	.398 (-.090)	.305 (-.089)	.277 (-.101)	.389 (-.091)

MTBench-Finance (Hard Subset)				
Model	**F**	**M**	**R**	**I**
Random Baseline		.25		
command-r-08-2024	.488 (-.252)	.442 (-.287)	.372 (-.333)	.372 (-.341)
Deepseek-V3	.744 (-.146)	.651 (-.239)	.651 (-.244)	.488 (-.309)
gemini-2.5-flash-lite	.674 (-.183)	.558 (-.293)	.651 (-.198)	.651 (-.149)

MCQ2TS (Hard Subset)				
Model	**F**	**M**	**R**	**I**
Random Baseline		.25		
command-r-08-2024	.288 (-.092)	.252 (-.091)	.272 (-.091)	.256 (-.080)
Deepseek-V3	.363 (-.147)	.438 (-.185)	.465 (-.170)	.310 (-.155)
gemini-2.5-flash-lite	.280 (-.119)	.325 (-.159)	.333 (-.147)	.231 (-.132)

MTBench-Weather (Hard Subset)				
Model	**F**	**M**	**R**	**I**
Random Baseline		.25		
command-r-08-2024	.314 (-.157)	.300 (-.186)	.324 (-.158)	.348 (-.167)
Deepseek-V3	.317 (-.254)	.303 (-.271)	.293 (-.273)	.310 (-.253)
gemini-2.5-flash-lite	.300 (-.243)	.293 (-.230)	.310 (-.227)	.276 (-.261)

Pruning statistics. Total samples: 2990; Total pruned: 1699 (56.8%). Pruned per dataset: Time-Series Exam 281/746 (37.7%), MCQ2TS 507/1000 (50.7%), MTBench-Finance 473/516 (91.7%), MTBench-Weather 438/728 (60.2%). *Note:* Hard–subset generally preserves the qualitative ordering among F/M/R/I while lowering absolute accuracy.

Table 2 shows the results on this hard subset. As expected, accuracy in the Full condition drops for all models (since we removed the easier questions). However, the qualitative trend remains the same: for the other models, masked and noised performance is still close to full performance, and in some cases still higher. For example, on the filtered MCQ2TS, DeepSeekV3 is 36.3% with full input vs 46.5% with random noise (only slightly lower than before filtering), and on the filtered Weather data the differences between full, masked, noise, and inverted are within a few points for most models. The one dataset where masked/noised still exceeds full is MCQ2TS (e.g., DeepSeekV3 goes from 36.3% F to 46.5% N), suggesting that even among the questions that *weren't* trivial for GPT-4.1-nano, many can still be handled by other models without the time-series. Overall, removing the questions which may be solved without time-series reduces absolute scores but does not fundamentally change the observation that models can do almost as well without the time-series.

5.2 RQ2: Do benchmarks that pose more complex questions actually demand stronger time-series reasoning?

We examine how question complexity correlates with model performance and reliance on the time-series. Figure 3 shows accuracy for each model on questions binned by complexity level (C1, C2, C3), under both full input and masked conditions. This analysis is performed on the labeled subset of 200 examples per dataset described in Section 3.2.

The clearest pattern is that accuracy rises monotonically with complexity across every model: all four models are weakest on C1 items (ranging from 0.23 to 0.55) and strongest on C3 items (0.62 to 0.73). On a benchmark that genuinely measured time-series reasoning, we would expect the opposite, with lookup-style questions easier than multi-step synthesis. Instead, the questions humans would deem simplest are the hardest for these models, and the questions humans would deem most complex are the easiest—consistent with our broader finding that richer textual context provides alternative solution paths rather than harder reasoning targets.

A second observation is that the Full–Masked gap is small at every complexity level. At C2 and C3 this is unsurprising given the RQ1 results: on MTBench-Finance and MTBench-Weather, which contribute nearly

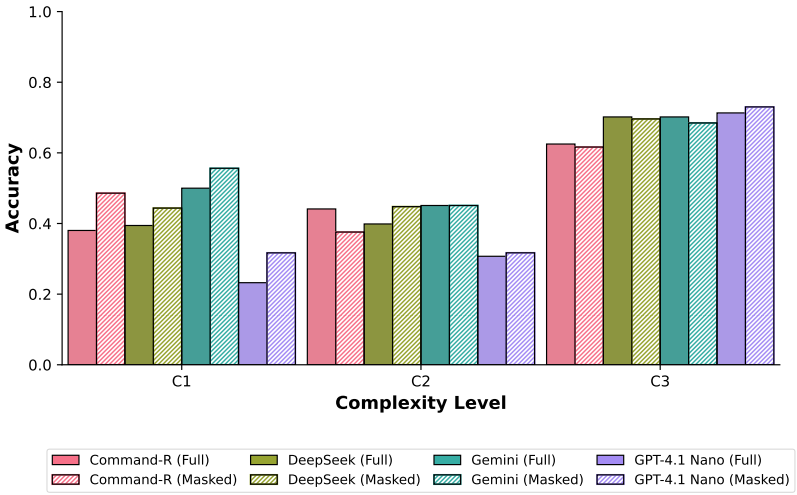


Figure 3: Accuracy by question complexity level (C1–C3) on the per-item labeled subset (200 items per dataset). Solid bars show accuracy with full input; hatched bars show accuracy with the time-series masked out. Accuracy rises monotonically with rubric complexity across all four models, and the Full–Masked gap is small at every level.

all of the C3 items and most of the C2 items in our labeled subset (Table 4), masking the series barely changes accuracy. The narrative context in these datasets supplies enough signal to answer most questions without the numeric series.

The C1 result requires more careful interpretation. Across all four models, Masked accuracy at C1 is slightly *higher* than Full accuracy, but this aggregate hides a split between datasets. TS-Exam contributes the majority of series-dependent C1 items: on that dataset, masking drops accuracy toward the random baseline (Table 1), exactly as one would expect for a benchmark that forces reliance on the numeric data. MCQ2TS contributes the remaining C1 items, and on that dataset masking *improves* accuracy for three of four models. The aggregate C1 bars in Figure 3 therefore mix a dataset where the series matters with one where text-only shortcuts are available even for simple questions, and the MCQ2TS contribution is large enough to push the mean Masked bar above the mean Full bar. We read this as evidence that shortcut availability is not confined to high-complexity items: the counterfactual framing of MCQ2TS appears to admit text-only heuristics even at the lowest complexity levels, while the property-identification framing of TS-Exam does not.

From a benchmark design perspective, these findings are concerning on two fronts. The questions that most faithfully test time-series reasoning (TS-Exam-style C1 items) are ones that large pre-trained models struggle with, and providing the series does not close the gap by much. Meanwhile, questions that incorporate rich textual context, whether at the C1 level in MCQ2TS or at C3 in MTBench, can often be answered without the series, meaning they are not measuring the intended capability. In sum, current benchmarks inadvertently reward general reasoning and test-taking skills. Making questions simpler forces reliance on data only when the option set and framing are themselves carefully constructed (as in TS-Exam); otherwise simple questions can admit their own shortcuts. Making questions more complex supplies richer text and widens the space of shortcuts available to language models.

5.3 RQ3: How confidence, answer consistency, and ablations reveal whether existing benchmarks genuinely measure joint time-series and language reasoning?

Text-masked condition. To contrast the contribution of text vs. series, we also construct a *text-masked* variant (T). We preserve the numeric time-series tokens and the multiple-choice answer options verbatim, but replace all other textual tokens in the prompt (context passages, question stems, news/weather reports) with placeholder `[MASK]` tokens. To avoid changing sequence length and position-dependent behavior, each masked textual span is replaced by a number of `[MASK]` tokens approximately matching the original text

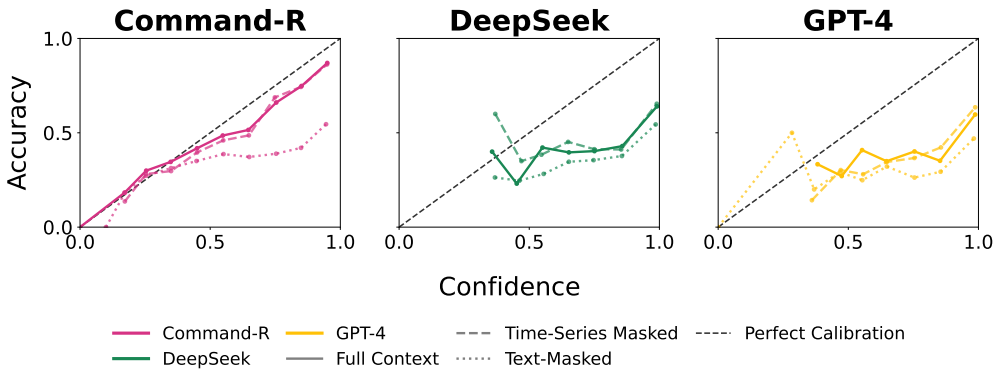


Figure 4: **Calibration with and without the series.** Reliability curves and ECE across datasets. The dashed diagonal is perfect calibration. Curves cluster above the diagonal and ECE changes minimally between Full and Masked, showing that model confidence scarcely depends on the time-series. The "Text-Masked" line indicates that we dropped text context with the exception of the answer options required to solve the problem.

length, so the total token count remains approximately unchanged. This yields three input regimes used throughout: Full (F), Series-masked (M), and Text-masked (T).

We quantify modality usage via three different analysis techniques. **Calibration:** we compute expected calibration error (ECE) with K=10 equal-width bins in $[0, 1]$. Confidence is the model-reported probability of the emitted answer token (the first token of the chosen option). We report ECE under Full, Series-masked, and Text-masked conditions. **Consistency:** we measure flip rates between Full and masked inputs, Flip($F \to M$) and Flip($F \to T$), to capture instance-level instability masked by similar aggregate accuracy. **Confidence shift:** we compute two token-level shifts,

$$\Delta P^{\mathrm{TS}} = P_{\mathrm{emit}}(\text{answer} \mid F) - P_{\mathrm{emit}}(\text{answer} \mid M), \quad \Delta P^{\mathrm{TXT}} = P_{\mathrm{emit}}(\text{answer} \mid F) - P_{\mathrm{emit}}(\text{answer} \mid T),$$

which isolate how much the series or the text moves belief in the emitted answer. We summarize distributions of ΔP^{TS} and ΔP^{TXT}, compare them directly per-instance, and also report per-dataset means $\overline{\Delta P}^{\mathrm{TS}}$ and $\overline{\Delta P}^{\mathrm{TXT}}$. Systems with degenerate probabilities (e.g., constant 1.0 for every emitted token) are excluded from ECE and ΔP analyses.

5.3.1 Calibration under Masking

Figure 4 plots reliability curves and reports ECE for the evaluated models under Full (solid) and Masked (dashed) inputs across all datasets. Models are consistently overconfident, and ECE trends remain similar when the series is removed. However the calibration improves considerably for datasets with higher proportions of complex reasoning tasks such as MTBench Finance and Weather. This holds even where masking slightly perturbs accuracy (Table 1). The insensitivity of calibration to masking indicates that confidence is grounded in textual cues and priors rather than in the numeric series. We can see that the Text-Masked (T) condition results in noticeably worse calibration than the Series-Masked (M) condition, consistent with the interpretation that the model's confidence is rooted in textual rather than numeric evidence.

5.3.2 Instance Consistency: Flip Rate

Accuracy aggregates can hide which examples are solved. We therefore compute the flip rate Flip($F \to M$), the fraction of instances whose predicted option changes when masking the series. Figure 5 shows substantial flip rates in every dataset. Even when Full and Masked accuracies are close, the set of items solved changes, especially in MCQ2TS and TS-Exam. This points to a dataset-side issue: questions admit heuristics that allow success without numeric reasoning. It is important to note that significant flip rates do not necessarily mean that the time-series is contributing in a meaningful way to the performance. For example the models often perform better on the MCQ2TS under the masked setting than with the full time-series.

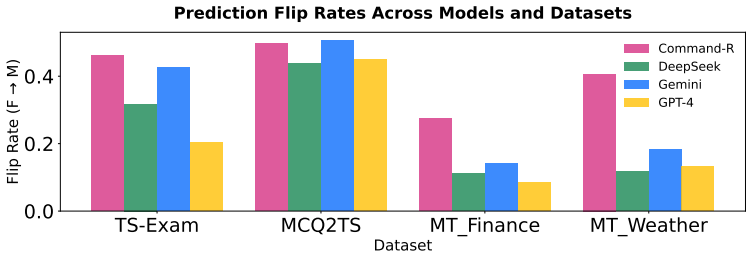


Figure 5: **Flip rate ($F \to M$).** The model often changes *which* items it solves when the series is masked, despite similar aggregate accuracy, indicating reliance on alternative text-only heuristics.

5.3.3 Confidence Shift

We examine whether the time-series or the text meaningfully increases confidence in the emitted answer. For each instance we compute $\Delta P^{\text{TS}} = P_{\text{emit}}(\text{answer} \mid F) - P_{\text{emit}}(\text{answer} \mid M)$ and $\Delta P^{\text{TXT}} = P_{\text{emit}}(\text{answer} \mid F) - P_{\text{emit}}(\text{answer} \mid T)$. Figure 6 summarizes three complementary views in a model-by-model layout. The *top row* reproduces the distribution of ΔP^{TS} (time-series drop) as in our earlier analysis under differing permutation settings and shows it is tightly centered around 0, implying that removing the time-series leaves confidence in the chosen answer essentially unchanged. The *middle row* directly compares ΔP^{TXT} and ΔP^{TS} on the same instances; the volume of points clustered around $\Delta P = 0$ is much higher for the time-series masked condition indicating that in comparison the time-series has a much smaller effect on performance. The *bottom row* reports dataset-level means $\overline{\Delta P}^{\text{TS}}$ vs. $\overline{\Delta P}^{\text{TXT}}$, which further show that the average effect of text removal dominates series removal on every dataset. Small negative means for ΔP^{TS} in some settings suggest the series can even slightly distract.

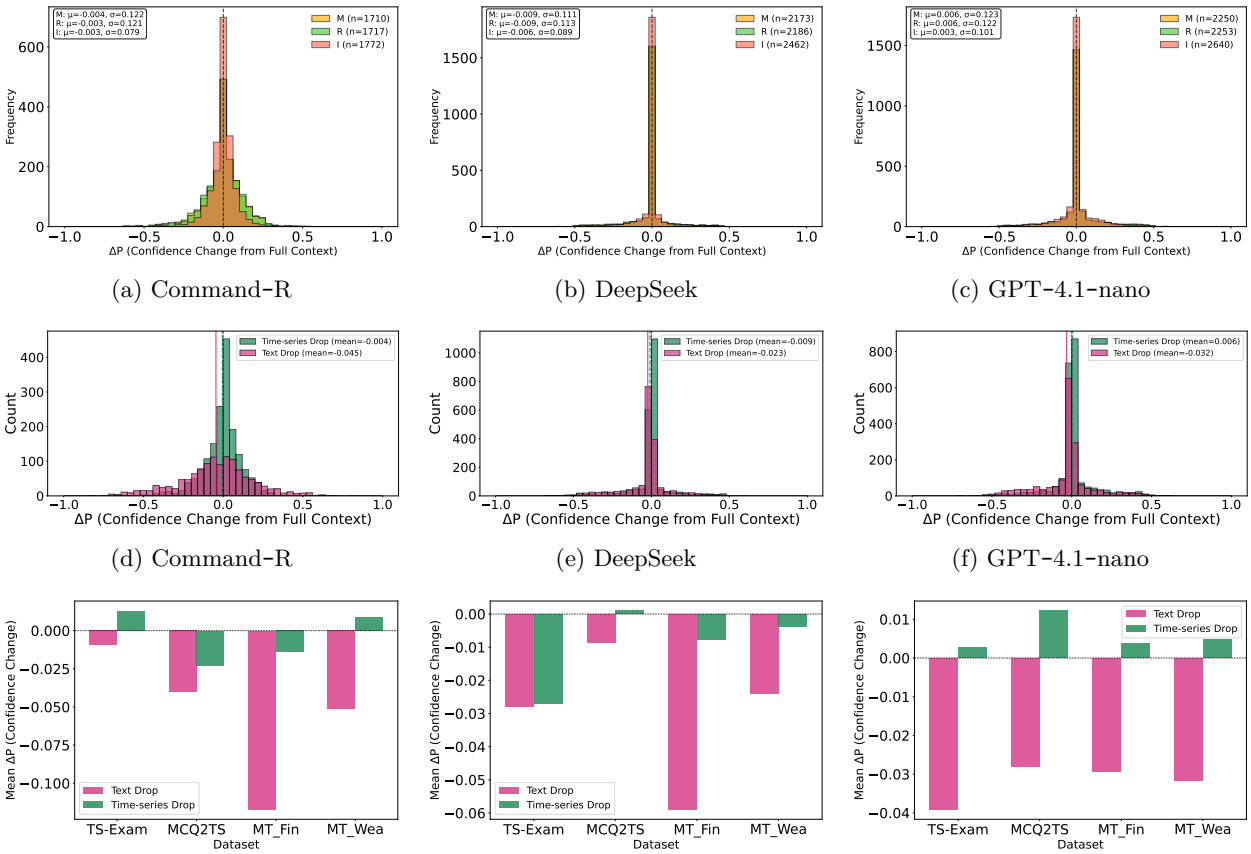
Across models, ΔP^{TS} concentrates near zero, while ΔP^{TXT} is substantially larger and dominates at the dataset mean level. Combined with the calibration and flip-rate findings, this indicates that the LLMs we evaluate, under text-serialized input, root their beliefs almost entirely in textual cues and priors; the numeric series rarely serves as decisive evidence and can occasionally distract.

6 Limitations and Future Work

Our study is scoped to multiple-choice question answering with short textual contexts and relatively short time-series; we did not explore open-ended reasoning, which would be considerably more difficult to evaluate reliably. Our input perturbations (masking, noise, inversion) are deliberately simple and information-removing, and many other perturbation families could yield complementary evidence. Our findings are further scoped to text-serialized encoding evaluated on widely deployed non-reasoning LLMs; native numeric-sequence encoders (e.g., dedicated time-series foundation models) and reasoning-augmented models may exhibit different behavior and remain important targets for follow-up evaluation. A further limitation is our reliance on an LLM-assisted complexity labeling scheme: while we validated it against an expert audit with substantial Cohen's κ (Appendix B.3), the grader can inherit systematic biases of its training distribution, the audit sample is modest, and the C2-versus-C3 boundary retains some subjectivity even under a principled Bloom's-taxonomy rubric. In future work, we plan to investigate training and prompting techniques that force models to attend to the time-series (e.g., specialized numeric encodings or intermediate questions that elicit specific series properties), and to construct new benchmarks designed to require genuine cross-modal reasoning by eliminating textual shortcuts or demanding precise numeric computation.

7 Conclusion

We asked whether current multimodal time-series QA benchmarks, as evaluated with text-serialized encoding on widely deployed LLMs, truly measure time-series reasoning. Across four diverse datasets and four state-of-the-art models, we found that accuracy remains nearly as high when the time-series data is hidden or



(a) Command-R (b) DeepSeek (c) GPT-4.1-nano

(d) Command-R (e) DeepSeek (f) GPT-4.1-nano

(g) Command-R (h) DeepSeek (i) GPT-4.1-nano

Figure 6: **Confidence shift with series vs. text masking (per row).** *Row 1 — ΔP under series ablations.* For each model, we show the distribution of the token-level confidence shift when we remove or corrupt the *time-series*: $\Delta P^{\mathrm{TS}} = P_{\mathrm{emit}}(\text{answer} \mid \text{Full}) - P_{\mathrm{emit}}(\text{answer} \mid \text{Series Ablation})$, where the ablation is one of: Mask **(M)** (series tokens replaced by `[MASK]`), Noise **(R)** (series replaced by random numeric noise), or Invert **(I)** (reflect values around the mean). The mass concentrates tightly near 0 across all ablations, indicating that belief in the emitted answer barely changes when the numeric series is removed or corrupted. *Row 2 — Text vs. series: per-instance comparison.* For the same instances, we compare confidence loss when dropping the *text* vs. dropping the *series* by contrasting $\Delta P^{\mathrm{TXT}} = P_{\mathrm{emit}}(\text{answer} \mid \text{Full}) - P_{\mathrm{emit}}(\text{answer} \mid \text{Text Masked})$ with ΔP^{TS}. The distributions/pairwise summaries show markedly larger shifts under text masking, revealing that confidence is anchored in textual cues rather than numeric evidence. *Row 3 — Mean effect sizes by dataset.* We aggregate per-dataset means $\overline{\Delta P}^{\mathrm{TS}}$ and $\overline{\Delta P}^{\mathrm{TXT}}$ for each model. In every benchmark, $\overline{\Delta P}^{\mathrm{TXT}} \gg \overline{\Delta P}^{\mathrm{TS}}$, confirming at the dataset level that removing text depresses confidence far more than removing the series. Systems with degenerate token probabilities (e.g., constant 1.0) are excluded from all ΔP summaries.

even replaced with random noise. Model predictions, both the selected answer and the confidence in that answer, change very little with the inclusion or exclusion of the numeric time-series, while changes to the textual context produce markedly larger shifts. Interestingly, the questions of simplest complexity (C1) proved hardest for the LLMs we evaluate because those require truly analyzing the time-series and little world knowledge or multi-step reasoning, whereas more complex questions could be answered by analysis from the text alone. In sum, under the encoding and model class dominant in today's literature, LLMs often succeed on time-series QA benchmarks without really using the time-series data, revealing a misalignment between benchmark intent and model behavior. We emphasize that our findings are scoped to this setting: reasoning-augmented models and native numeric-sequence encoders (e.g., dedicated time-series foundation

models) may exhibit different behavior, and evaluating them with the diagnostics introduced here is an important direction for future work.

Our findings suggest that both benchmarks and models need to evolve to address this gap. Benchmark creators should design tasks that cannot be solved without proper use of the time-series modality, perhaps by including specific properties of the time-series which must be elicited before reasoning steps required for more complex problems. On the model side, techniques to ensure multimodal attention, whether through architecture, training, or prompting, are necessary so that LLMs treat time-series as first-class data even when encoding time-series as text as has become the standard in the literature for many dataset benchmarks. By identifying the pitfalls in current datasets and models, we hope this work inspires more robust evaluation frameworks and improved multimodal reasoning capabilities for temporal data.

References

Nancy E Adams. Bloom's taxonomy of cognitive learning objectives. *J. Med. Libr. Assoc.*, 103(3):152–153, July 2015.

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering, 2018. URL https://arxiv.org/abs/1712.00377.

B.S. Bloom. *Taxonomy of Educational Objectives: The Classification of Educational Goals.* Number v. 1 in Taxonomy of Educational Objectives: The Classification of Educational Goals. D. McKay, 1956. URL https://books.google.ca/books?id=hos6AAAAIAAJ.

Yifu Cai, Arjun Choudhry, Mononito Goswami, and Artur Dubrawski. Timeseriesexam: A time series understanding exam, 2024. URL https://arxiv.org/abs/2410.14752.

Nimeesha Chan, Felix Parker, William Bennett, Tianyi Wu, Mung Yao Jia, James Fackler, and Kimia Ghobadi. Medtsllm: Leveraging llms for multimodal medical time series analysis, 2024. URL https://arxiv.org/abs/2408.07773.

Jialin Chen, Aosong Feng, Ziyu Zhao, Juan Garza, Gaukhar Nurbek, Cheng Qin, Ali Maatouk, Leandros Tassiulas, Yifeng Gao, and Rex Ying. Mtbench: A multimodal time series benchmark for temporal reasoning and question answering, 2025. URL https://arxiv.org/abs/2503.16858.

Winnie Chow, Lauren Gardiner, Haraldur T. Hallgrímsson, Maxwell A. Xu, and Shirley You Ren. Towards time series reasoning with llms, 2024. URL https://arxiv.org/abs/2409.11376.

Cohere Labs. c4ai-command-r-08-2024, 2024. URL https://huggingface.co/CohereLabs/c4ai-command-r-08-2024.

DeepSeek-AI. Deepseek-v3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Shrey Desai and Greg Durrett. Calibration of pre-trained transformers. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 295–302, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.21. URL https://aclanthology.org/2020.emnlp-main.21/.

Bradley Efron and Robert J. Tibshirani. *An introduction to the bootstrap.* Chapman & Hall, 2006.

Matt Gardner, Yoav Artzi, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, et al. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP*, 2020. doi: 10.18653/v1/2020.findings-emnlp.117. URL https://aclanthology.org/2020.findings-emnlp.117/.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *arXiv preprint arXiv:2004.07780*, 2020. URL https://arxiv.org/abs/2004.07780.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew Gordon Wilson. Large language models are zero-shot time series forecasters. *arXiv preprint arXiv:2310.07820*, 2023. doi: 10.48550/arXiv.2310.07820. URL https://arxiv.org/abs/2310.07820.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017. URL https://arxiv.org/abs/1706.04599.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2017. URL https://aclanthology.org/N18-2017/.

Yushan Jiang, Wenchao Yu, Geon Lee, Dongjin Song, Kijung Shin, Wei Cheng, Yanchi Liu, and Haifeng Chen. TimeXL: Explainable multi-modal time series prediction with LLM-in-the-loop. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=WRwr2YZ4zt.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021. doi: 10.1162/tacl_a_00407. URL https://aclanthology.org/2021.tacl-1.57/.

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. Time-llm: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations (ICLR)*, 2024a. doi: 10.48550/arXiv.2310.01728. URL https://arxiv.org/abs/2310.01728.

Ming Jin, Yifan Zhang, Wei Chen, Kexin Zhang, Yuxuan Liang, Bin Yang, Jindong Wang, Shirui Pan, and Qingsong Wen. Position: What can large language models tell us about time series analysis, 2024b. URL https://arxiv.org/abs/2402.02713.

Yaxuan Kong, Yiyuan Yang, Yoontae Hwang, Wenjie Du, Stefan Zohren, Zhangyang Wang, Ming Jin, and Qingsong Wen. Time-mqa: Time series multi-task question answering with context enhancement, 2025a. URL https://arxiv.org/abs/2503.01875.

Yaxuan Kong, Yiyuan Yang, Yoontae Hwang, Wenjie Du, Stefan Zohren, Zhangyang Wang, Ming Jin, and Qingsong Wen. Time-MQA: Time series multi-task question answering with context enhancement. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2025b. URL https://aclanthology.org/2025.acl-long.1437/.

Yaxuan Kong, Yiyuan Yang, Shiyu Wang, Chenghao Liu, Yuxuan Liang, Ming Jin, Stefan Zohren, Dan Pei, Yan Liu, and Qingsong Wen. Position: Empowering time series reasoning with multimodal llms, 2025c. URL https://arxiv.org/abs/2502.01477.

Haobo Li, Zhaowei Wang, Jiachen Wang, Yueya Wang, Alexis Kai Hon Lau, and Huamin Qu. Cllmate: A multimodal benchmark for weather and climate events forecasting, 2025. URL https://arxiv.org/abs/2409.19058.

Mike A. Merrill, Mingtian Tan, Vinayak Gupta, Tom Hartvigsen, and Tim Althoff. Language models still struggle to zero-shot reason about time series, 2024. URL https://arxiv.org/abs/2404.11757.

OpenAI, Josh Achiam, Steven Adler, and Sandhini Agarwal et al. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Letitia Parcalabescu, Michele Cafagna, Lilitta Muradjan, Anette Frank, Iacer Calixto, and Albert Gatt. VALSE: A task-independent benchmark for vision and language models centered on linguistic phenomena. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8253–8280, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.567. URL https://aclanthology.org/2022.acl-long.567/.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis only baselines in natural language inference. In Malvina Nissim, Jonathan Berant, and Alessandro Lenci (eds.), *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pp. 180–191, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2023. URL https://aclanthology.org/S18-2023/.

Paul Quinlan, Qingguo Li, and Xiaodan Zhu. Chat-ts: Enhancing multi-modal reasoning over time-series and natural language data, 2026. URL https://arxiv.org/abs/2503.10883.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020. doi: 10.18653/v1/2020.acl-main.442. URL https://aclanthology.org/2020.acl-main.442/.

Ravi Shekhar, Sandro Pezzelle, Yauhen Klimovich, Aurélie Herbelot, Moin Nabi, Enver Sangineto, and Raffaella Bernardi. FOIL it! find one mismatch between image and language caption. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 255–265, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1024. URL https://aclanthology.org/P17-1024/.

Gemini Team, Rohan Anil, Sebastian Borgeaud, and Jean-Baptiste Alayrac et al. Gemini: A family of highly capable multimodal models, 2024. URL https://arxiv.org/abs/2312.11805.

Jesse Thomason, Daniel Gordon, and Yonatan Bisk. Shifting the baseline: Single modality performance on visual navigation & QA. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 1977–1983, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1197. URL https://aclanthology.org/N19-1197/.

Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, Lei Zhang, and Jianxin Liao. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *AAAI Conference on Artificial Intelligence*, 2025.

Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tieying Zhang, Jianjun Chen, Rui Shi, and Dan Pei. Chatts: Aligning time series with llms via synthetic data for enhanced understanding and reasoning, 2025. URL https://arxiv.org/abs/2412.03104.

A Experimental Details

A.1 Models and Inference Settings

We evaluate a set of frontier and lightweight LLMs via their official APIs. All models are instructed to output *exactly one token* from {A, B, C, D} (no explanation, punctuation, or additional text). We score an item as correct if the first non-whitespace character matches the gold option label; non-conforming outputs are treated as incorrect unless a valid option letter can be unambiguously extracted.

Table 3: Evaluated API models.

Model ID	Provider
`gpt-4.1-nano`	OpenAI
`gemini-2.5-flash-lite`	Google
`command-r-08-2024`	Cohere
`deepseek-chat`	DeepSeek

Generation parameters. Unless otherwise specified, we use a fixed random seed (42), temperature = 0.0 for standard evaluation runs, and an output budget sufficient to return a single option token.

Answer format and scoring. All models are prompted to answer *only* with a single option label from {`A`, `B`, `C`, `D`} (no explanation, no punctuation). We score an item as correct if the first non-whitespace character of the model response matches the gold option label. All models evaluated have strong instruction- following capabilities, and we did not observe any non-compliant responses in our runs.

Token probabilities for confidence analyses. Some providers expose token probabilities only for a limited top-k set, and some do not return token probabilities for all requests. For calibration (ECE) and confidence-shift (ΔP) analyses, we therefore use the *provider-returned probability of the emitted option token* when available, denoted P_{emit}. When token probabilities are not available for a given example, that example is excluded from ECE and ΔP computation. All models remain included in accuracy and flip-rate analyses regardless of probability availability.

B Datasets and Prompt Standardization

We evaluate four public multiple-choice time-series QA benchmarks spanning synthetic and real-world domains: TS-Exam (Cai et al., 2024), MCQ2TS (Merrill et al., 2024), MTBench-Finance (Chen et al., 2025), and MTBench-Weather (Chen et al., 2025). For MCQ2TS, we cap the evaluation set at 1000 examples for efficiency, consistent with our main experiments.

B.1 Dataset Information

The benchmark comprises four primary datasets spanning three broad complexity regimes (C1–C3). Rather than assigning a single *dataset-level* complexity label, we report the *modal* (most common) complexity label on our per-item labeled subset (200 items per dataset; Section 3.2). This provides a concise characterization of the dominant cognitive demand in each benchmark while still allowing item-level variation.

Table 4: Dataset composition and complexity distribution. We sample 200 items from each dataset with n total samples and assign a Bloom's Taxonomy complexity label. **C1** (Remember/Understand) questions are answerable by inspection of the series; **C2** (Apply/Analyze) requires measuring, computing, or decomposing series properties; **C3** (Evaluate/Create) requires predicting, causal attribution, or jointly interpreting time series and external textual context.

Dataset	n	C1	C2	C3	# of Options	Random
TSExam (AutonLab)	746	66	125	9	2/3/4	40.1%
MCQ2TS	1000	68	128	4	4	25.0%
MTBench-Finance	516	6	27	167	4	25.0%
MTBench-Weather	728	2	26	172	4	25.0%
Total	2990	142	306	352		

In terms of dataset structure, TS-Exam primarily targets time-series trait and property identification (often answerable by directly reading or recognizing simple patterns), while MCQ2TS emphasizes counterfactual

reasoning over two time-series. MTBench-Finance and MTBench-Weather pair serialized time-series with domain text (news or weather reports), which yields longer natural-language context but, as our main results show, also creates opportunities for text-only shortcuts. Across all datasets, we normalize prompts to a consistent multiple-choice format and require single-token answers in {A,B,C,D} (Section A.1).

We exclude a few other datasets from this study. In particular, ChatTS (Xie et al., 2025), ChatTime (Wang et al., 2025), and Time-MQA (Kong et al., 2025b) were not included. ChatTS mixes multiple-choice and open-ended questions, complicating evaluation as it requires fuzzy matching metrics intended for retrieval-augmented generation. ChatTime and Time-MQA were excluded because their training and evaluation splits derive from the same synthetic source and lacked a manual assessment of quality.

Prompt standardization. Across datasets, we serialize time-series as text and wrap them in explicit tags `[TS]... [/TS]`. All prompts terminate with a strict instruction to answer with only a single letter from `A/B/C/D`. For datasets whose original question strings include embedded options, we remove embedded option text and re-append normalized `A)`–`D)` lines to ensure consistent formatting.

Representative prompt templates. Below we show abbreviated templates illustrating the common structure used in our loaders.

```
TS-Exam:
<question text>
Time Series 1: [TS] ... [/TS]
Time Series 2: [TS] ... [/TS]   (if present)
A) ...
B) ...
C) ...
D) ...
Please answer with only the letter (e.g. A, B, C, D). Do not explain.

MCQ2TS:
<question text>
Original Time Series:   [TS] ... [/TS]
Counter-factual Series: [TS] ... [/TS]
A) ...
B) ...
C) ...
D) ...
Please answer with only the single letter (A, B, C or D). Do not explain.

MTBench-Finance:
... stock price of <TICKER> between <t1> to <t2>, interval <...>:
[TS] ... [/TS]
News published at <timestamp>:
<news text>
Question: <question text>
A) ...
B) ...
C) ...
D) ...
Please answer with only the letter (e.g. A, B, C, D). Do not explain.

MTBench-Weather:
... temperature series between <t1> to <t2>, interval 1 h:
[TS] ... [/TS]
weather event report:
```

```
<report text>
<question text>
A) ...
B) ...
C) ...
D) ...
Please answer with only the letter (e.g. A, B, C, D). Do not explain.
```

B.2 Masking, Noise, Inversion, and Text-Masking

We evaluate each dataset under multiple input conditions designed to preserve prompt structure while selectively breaking evidence.

Series-masked (M). We replace time-series values with a placeholder token `[MASK]`. Unless otherwise noted, Series-masked uses 100% masking (all numeric values are replaced), leaving timestamps and prompt scaffolding intact.

Noise (N). We replace the time-series values with a random numeric sequence of the same length and surface format as the original serialization. This retains the appearance and token budget of a time-series while removing relevant signal.

Inverted (I). We invert the numeric values around the series mean, i.e., $x'_t = 2\bar{x} - x_t$, which reverses trends while preserving sequence length and scale.

Text-masked (T). We preserve the serialized time-series and the `A-D` options verbatim, while replacing the remaining textual content (context passages, question stem, reports) with `[MASK]` tokens. This yields a complementary ablation to Series-masked and is used in confidence-shift analyses.

B.3 LLM-Assisted Complexity Annotation

We assign per-item complexity labels (C1/C2/C3) on a labeled subset of 800 items (200 per dataset) to support RQ2 analyses. Labels are produced by an LLM rubric grader run deterministically (temperature = 0) and audited against an author with domain expertise on a stratified 60-item subsample (15 per dataset). Overall author–grader agreement on the audit is 78.3%, with unweighted Cohen's $\kappa = 0.67$ and linear-weighted Cohen's $\kappa = 0.73$. Because the C1/C2/C3 scale is ordinal, the linear-weighted κ is the more appropriate headline statistic, as it gives partial credit for adjacent-category disagreements (e.g., C2 vs. C3) over non-adjacent ones (C1 vs. C3). Both values fall in the "substantial agreement" range on the Landis–Koch scale.

Label distribution. Table 5 reports the grader's label distribution on the 800 labeled items. The four datasets differ sharply in their cognitive-demand profile: TS-Exam and MCQ2TS are dominated by C1/C2 items (property identification and counterfactual reasoning), whereas MTBench-Finance and MTBench-Weather are overwhelmingly C3, reflecting the narrative text contexts that pair with each numeric series in those benchmarks.

Table 5: LLM grader label distribution across the 800-item labeled subset (200 per dataset).

Dataset	**C1**	**C2**	**C3**	**Total**
TS-Exam	66	125	9	200
MCQ2TS	68	128	4	200
MTBench-Finance	6	27	167	200
MTBench-Weather	2	26	172	200
Total	142	306	352	800

Per-dataset audit agreement. Table 6 breaks down audit agreement by dataset. TS-Exam, MCQ2TS, and MTBench-Finance all show substantial-to-perfect agreement. MTBench-Weather is the outlier: raw agreement is 53.3% and linear-weighted $\kappa = 0.12$.

Table 6: Per-dataset author–grader agreement on the 60-item audit (15 items per dataset). κ_{unwtd} is unweighted Cohen's κ; κ_{lin} is linear-weighted κ.

Dataset	n	**Agreement**	$\kappa_{\textbf{unwtd}}$	$\kappa_{\textbf{lin}}$
TS-Exam	15	100.0%	1.00	1.00
MCQ2TS	15	86.7%	0.80	0.84
MTBench-Finance	15	73.3%	0.60	0.67
MTBench-Weather	15	53.3%	0.03	0.12
Overall	60	78.3%	0.67	0.73

Disagreement structure. Of the 13 total mismatches across the 60-item audit, all 13 are cases in which the grader assigned a *lower* complexity label than the author. No instances of the grader over-estimating complexity relative to the author occurred. The breakdown is:

- author C2 → grader C1: 7 items
- author C3 → grader C2: 5 items
- author C3 → grader C1: 1 item

Because disagreements are exclusively one-directional and concentrated at adjacent categories, the grader's labels are better understood as a consistently conservative reading of the rubric rather than as noise.

Caveat on MTBench-Weather. The MTBench-Weather audit deserves particular scrutiny. The disagreements in this dataset were concentrated at the C2/C3 boundary, with the grader tending to label items C2 that the author would label C3. Crucially, C2-labeled items are a small minority of the full Weather population (26/200 = 13%; Table 5), but by chance the 15-item random audit for Weather drew a larger share of grader-C2 items than the 13% base rate would predict. Because the grader's disagreements with the author are concentrated precisely at grader-C2 items, the audit over-sampled the hardest boundary cases, and the resulting κ is a worst-case estimate of agreement on the full Weather set rather than a representative one. A population-weighted audit would place most of its probability mass on grader-C3 items, where the two raters largely agree. We nonetheless retain the unweighted per-dataset κ in Table 6 as the transparent statistic, and we read complexity-conditioned results for MTBench-Weather (Section 5.2) with this caveat in mind. We note further that the absolute-accuracy findings of Section 5.2 do not hinge on the exact C2-versus-C3 boundary within MTBench-Weather, since the headline result—that models perform as well under masking as under the full series—holds across all C-levels within this dataset.

Rubric (Bloom-aligned). C1 (Remember/Understand): Answerable by directly reading or recognizing basic properties of the time-series (e.g., identify a value, detect an obvious monotonic trend).
C2 (Apply/Analyze): Requires applying time-series concepts or multi-step reasoning (e.g., compare segments, relate two series, connect a described change to a numeric pattern).
C3 (Evaluate/Create): Requires synthesis across multiple sources or higher-level judgment (e.g., causal attribution using text + series, counterfactual reasoning beyond direct reading, forecasting-like judgments conditioned on narrative context).

Labeling prompt template. We provide the labeler with the full example (serialized series, text context, question, and options) and request one label only.

```
You are grading the cognitive complexity of a multiple-choice time-series QA item
```

```
using Bloom's Taxonomy.

1 (Remember / Understand): The answer requires only recognising or naming an
observable property of the series -- identifying trend direction, naming a wave
type, judging whether a mean or variance is stable, spotting an obvious anomaly,
or comparing two series visually to see if they share the same shape. No
arithmetic, decomposition, or inference is needed; a reader could answer by
inspection.

2 (Apply / Analyze): The answer requires a concrete operation that goes beyond
inspection -- measuring or estimating a numeric quantity (e.g. period, slope,
lag, amplitude difference), decomposing a series into components, determining
whether one series is a transformed or lagged copy of another by testing
alignment, or comparing statistical properties (e.g. noise level, variance
ratio) that cannot be read off directly. The question is about what the data IS
or WAS, derivable mechanically from the series values.

3 (Evaluate / Create): The answer requires reasoning beyond what the series
directly shows:
  * Predicting or forecasting -- what will happen next or as a result of an event.
  * Causal attribution -- explaining WHY a pattern, anomaly, or break occurred.
  * Evaluating a claim, hypothesis, or conclusion against the evidence.
  * Drawing a conclusion that requires combining the series with external
    textual context (a news article, weather report, financial note, or event
    description).

Key distinctions:
  * Naming/recognising a property by eye -> 1
  * Measuring or decomposing to obtain a value -> 2
  * Explaining, predicting, or evaluating using the data -> 3

Respond with exactly one digit: 1, 2, or 3. Do not output anything else.
```

C Qualitative Examples

This section provides representative examples from each dataset. For each dataset, we reserve space for **(i)** a rendered time-series plot and **(ii)** the corresponding text block (context, question, options, ground truth) loaded directly from a matching `.txt` file.

C.1 TS-Exam

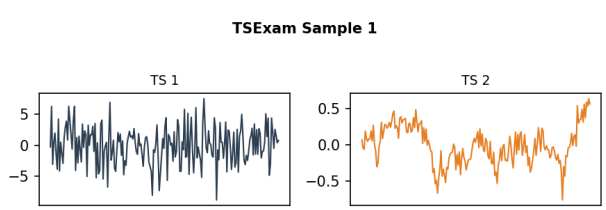


Figure 7: TS-Exam: Example 1 time-series plot.

Example 1.

```
--- CONTEXT ---
```

```
Concepts: Gaussian White Noise, Red Noise

--- QUESTION ---
Are the given two time series likely to have the same underlying distribution?
A. No, they have different underlying distribution
B. Yes, they have the same underlying distribution: Gaussian White Noise

--- GROUND TRUTH ---
Answer: A
```

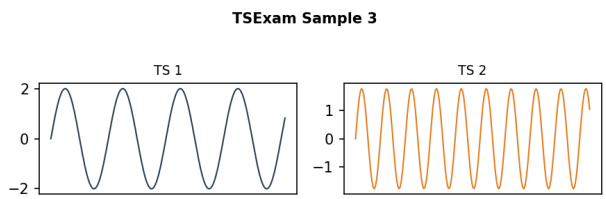


Figure 8: TS-Exam: Example 2 time-series plot.

Example 2.

```
--- CONTEXT ---
Concepts: Sine Wave, Square Wave

--- QUESTION ---
Does the given two time series have similar pattern?
A. Yes, they have similar seasonal pattern
B. No, they have different seasonable pattern

--- GROUND TRUTH ---
Answer: A
```

C.2 MCQ2TS

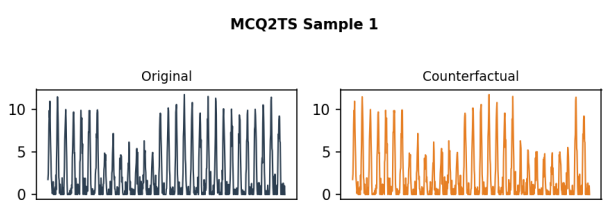


Figure 9: MCQ2TS: Example 1 time-series plot(s).

Example 1.

```
--- CONTEXT ---
 The scenario involves tracking the quantity of electricity generated from a solar panel in a
 residential building. The sample rate is each hour over a month (720 observations). The
 external event is a week-long cloudy spell drastically reducing the solar output.

--- QUESTION ---
In which part of the month does the second cloudy weather period occur in the updated time
series?
```

```
A. The updated series has the second cloudy period in the first week.
B. The second cloudy weather period in the updated time series appears in the fourth week.
C. The second cloudy period occurs in the third week in the updated series.
D. Both time series have the second cloudy period starting in the second week.

--- GROUND TRUTH ---
Answer: B
```

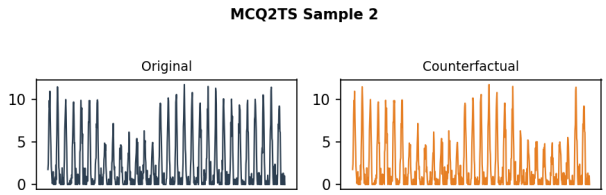


Figure 10: MCQ2TS: Example 2 time-series plot(s).

Example 2.

```
--- CONTEXT ---
 The scenario involves tracking the quantity of electricity generated from a solar panel in a
 residential building. The sample rate is each hour over a month (720 observations). The
 external event is a week-long cloudy spell drastically reducing the solar output.

--- QUESTION ---
How does the duration of cloudy periods within the month compare between the original and
updated time series?
A. Both series have the same duration of cloudy periods.
B. The updated series has a total of two weeks of cloudy periods whereas the original series has
only one week.
C. The original series has two weeks of cloudy periods.
D. The updated series has a shorter duration of cloudy periods.

--- GROUND TRUTH ---
Answer: B
```

C.3 MTBench-Finance

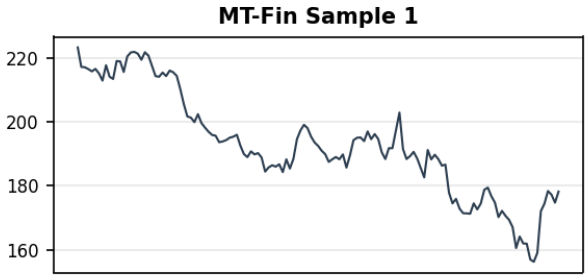


Figure 11: MTBench-Finance: Example 1 stock-price time-series plot.

Example 1.

```
--- CONTEXT ---
```

```
NEWS REPORT:
Title: Gina Sanchez explains why these two mega cap stocks are a buy
 Content: ['In a market that continues to punish the mega cap stocks, Gina Sanchez âĂŞ the chief
 market strategist at Lido Advisors, remains bullish on the following two. \xa0', '\nAre you
 looking for fast-news, hot-tips and market analysis?\n\nSign-up for the Invezz newsletter,
 today.\n\n\n', 'The retail giant did fairly well during the pandemic and now Sanchez is
 convinced itâĂŹll turn out to be an even better reopening stock. This afternoon on CNBCâĂŹs
 âĂThe ExchangeâĂİ, she said:', '\n', 'Now theyâĂŹre playing the reopening game. If anybody can
 handle supply chain problems, itâĂŹs Walmart. They boast one of the top logistics team in the
 world. So, this is a stock we think can manage and navigate the risks and still give you
 defensive returns.', 'The American multinational is expected to report its Q1 results on May
 17th. Ahead of earnings, the stock is down nearly 8.0% from its year-to-date high. \xa0', 'The
 semiconductor giant is down almost 45% year-to-date, which, as per Gina Sanchez, makes up for a
 fantastic opportunity to buy a market leader at a deep discount. Speaking with CNBCâĂŹs Kelly
 Evans, she noted:', '\n\n', 'The underpinning for almost all demand going forward seems to be
 in the digital space. And thatâĂŹs reliant on the Cloud, which is reliant on semiconductors.
 Nvidia is easily the top player in this space right now. The Cloud trend is just not going
 away.', 'Nvidia will also be reporting its quarterly results before month end. According to
 Motiur Rahman âĂŞ financial analyst at Invezz, the Santa Clara-headquartered company is headed
 for $280 a share.']

--- QUESTION ---
Which of the following statements about NVDA's stock price and the given financial analysis is
incorrect?
A. The stock price of NVDA experienced a significant decline of nearly 45% year-to-date prior to
the publication of the news, indicating a potential opportunity for investors to buy at a
discounted price.
B. Following the news publication, NVDA's stock displayed some volatility, with price
fluctuations indicating market uncertainty as the stock traded between 162.81 and 180.8 in the
subsequent week.
C. Analysts predict that NVDA's stock will reach $280 as stated by Motiur Rahman in the
immediate next week after the news, disregarding the recent downtrend leading up to that point.
D. Gina Sanchez's bullish outlook on NVDA suggests that the anticipated demand for
semiconductors, particularly in the cloud sector, could lead to potential recovery in the stock
price in the medium to long term.

--- GROUND TRUTH ---
Answer: C
```

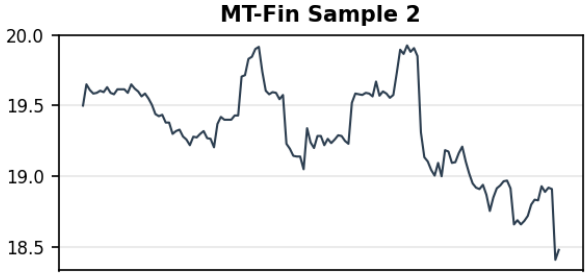


Figure 12: MTBench-Finance: Example 2 stock-price time-series plot.

Example 2.

```
--- CONTEXT ---
NEWS REPORT:
```

```
Title: BKEAY or IBN: Which Is the Better Value Stock Right Now?
 Content: ["Investors with an interest in Banks - Foreign stocks have likely encountered both
 The Bank of East Asia Ltd. (BKEAY Quick QuoteBKEAY  -  Free Report)     and ICICI Bank Limited
 (IBN Quick QuoteIBN  -  Free Report)    . But which of these two companies is the best option
 for those looking for undervalued stocks? Let's take a closer look.", 'We have found that the
 best way to discover great value opportunities is to pair a strong Zacks Rank with a great
 grade in the Value category of our Style Scores system. The proven Zacks Rank puts an emphasis
 on earnings estimates and estimate revisions, while our Style Scores work to identify stocks
 with specific traits.', 'Right now, The Bank of East Asia Ltd. is sporting a Zacks Rank of #2
 (Buy), while ICICI Bank Limited has a Zacks Rank of #3 (Hold). The Zacks Rank favors stocks
 that have recently seen positive revisions to their earnings estimates, so investors should
 rest assured that BKEAY has an improving earnings outlook. But this is just one piece of the
 puzzle for value investors.', 'Value investors also tend to look at a number of traditional,
 tried-and-true figures to help them find stocks that they believe are undervalued at their
 current share price levels.', 'Our Value category grades stocks based on a number of key
 metrics, including the tried-and-true P/E ratio, the P/S ratio, earnings yield, and cash flow
 per share, as well as a variety of other fundamentals that value investors frequently use.',
 "BKEAY currently has a forward P/E ratio of 7.43, while IBN has a forward P/E of 23.05. We also
 note that BKEAY has a PEG ratio of 0.36. This metric is used similarly to the famous P/E ratio,
 but the PEG ratio also takes into account the stock's expected earnings growth rate. IBN
 currently has a PEG ratio of 0.89.", "Another notable valuation metric for BKEAY is its P/B
 ratio of 0.31. The P/B is a method of comparing a stock's market value to its book value, which
 is defined as total assets minus total liabilities. By comparison, IBN has a P/B of 2.78.",
 'Based on these metrics and many more, BKEAY holds a Value grade of B, while IBN has a Value
 grade of C.', 'BKEAY is currently sporting an improving earnings outlook, which makes it stick
 out in our Zacks Rank model. And, based on the above valuation metrics, we feel that BKEAY is
 likely the superior value option right now.', 'ICICI Bank Limited (IBN) - free report >>']

--- QUESTION ---
Which of the following statements about IBN's stock price and the given financial analysis is
correct?
A. The news article's emphasis on BKEAY's improving earnings outlook relative to IBN likely
contributed to investors shifting their interest away from IBN towards BKEAY.
B. IBN's higher PEG ratio of 0.89 suggests it is a more attractive investment compared to
BKEAYâĂŹs PEG ratio of 0.36.
C. The decline in IBNâĂŹs stock price after the news was minimal, as it fluctuated between 19.5
and 19.65 without any significant downward trend.
D. The stock price of IBN increased following the news publication on October 6, 2021,
indicating a positive market reaction.

--- GROUND TRUTH ---
Answer: A
```

C.4 MTBench-Weather

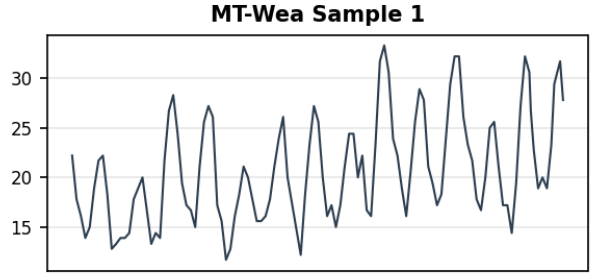


Figure 13: MTBench-Weather: Example 1 temperature time-series plot.

Example 1.

```
--- CONTEXT ---
WEATHER REPORT:
The following events were reported: Flash Flood, Hail, Thunderstorm Wind, Tornado. These
occurred near station USW00003017, approximately 25.2091 km away, between 2015-06-24 15:45:00
and 2015-06-24 20:15:00. The events included records with hailstones measuring 8.682222222222222
inches.Thankfully, no injuries or fatalities were reported. The events caused property damage
valued at 75000.0 and crop damage of 0.0. Episode Narrative: Two colliding outflow boundaries
merged over east Denver and northwest Aurora at the height of rush hour. The collision quickly
spawned a severe thunderstorm that produced an EF1 tornado, damaging hail, torrential rain and
flash flooding. The tornado touched down in east Denver and west Aurora. The tornado first
touched down near Quebec and 6th Ave. It then moved east northeast across the Lowry Campus into
the west part of Aurora. The tornado then lifted near Mount Nebo Memorial Park. Some homes had
minor roof damage with one former apartment building on the Lowry Campus had more significant
roof damage. The tornado and intense thunderstorm winds uprooted trees, damaging vehicles and
blocking roads. ||The storm produced torrential rain, 2 to 2.5 inches, much of which fell in
less than 30 minutes and resulted in flooded intersections and power outages. Flash flooding
forced the evacuation of a theater at the Cherry Creek Shopping Center, where drifts of hail
formed in the parking lot, and flooding set off alarms at the University of Denver's Ritchie
Center. Numerous water rescues were reported as vehicles stalled flooded intersections. Many
stoplights were knocked out. The water was reportedly 3-feet deep on the South Broadway ramp to
Interstate 25. The bike path along Cherry Creek was inundated with several feet of water at the
height of the storm. The South Platte River crested above flood stage for one hour. Employees
still at work were urged to stay inside but others waded across flooded intersections downtown.
About 30 flights had to be diverted from Denver International Airport. Two colliding outflow
boundaries merged over east Denver and northwest Aurora at the height of rush hour. Event
Narrative:         Large trees toppled near 23rd and Galena. A tornado touched town near Quebec
and 6th Ave and lifted near Mount Nebo Memorial Park. This tornado was a non-supercell tornado.
Two outflow boundaries from earlier thunderstorms intersected over eastern Denver and western
Aurora. This produced localized convergence and wind shear which likely contributed to the rapid
formation of the tornado. Torrential rainfall, from 2 to 2.5 inches, flooded intersections up to
3 feet of standing water at several intersections including:  6th Ave and Lowell, Broadway and
Interstate 25, as well as parts of Speer Blvd along Cherry Creek.

--- QUESTION ---
Based on the 14-day temperature time series A and the reported weather events, what can be
inferred about the implications of a flash flood and tornado on the temperature trends occurring
shortly after these events? Specifically, considering the temperature data for the next three
days in time series B, which option correctly describes the expected temperature changes
following the severe weather reports?

A) The temperatures are expected to rise sharply above 30ÃřC as a direct consequence of the heat
from the aftermath of these severe events due to increased solar insolation after the storm.
B) The heavy rainfall and resulting flash flood are likely to lead to a cooling trend initially,
with temperatures dropping below 15ÃřC for at least the first 48 hours post-event due to cloud
cover and water accumulation.
C) Following the tornado and flash flooding, temperatures typically stabilize around a
consistent level of about 20ÃřC or slightly above, driven by high humidity levels that persist
in the area.
D) The temperature is expected to drop significantly immediately after the events, but then
rebound to near pre-storm levels shortly afterward due to high pressure settling in.

--- GROUND TRUTH ---
Answer: B
```

Example 2.

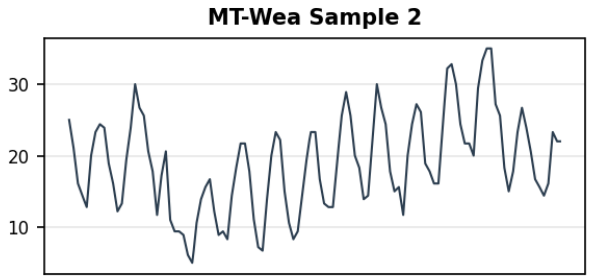


Figure 14: MTBench-Weather: Example 2 temperature time-series plot.

```
--- CONTEXT ---
WEATHER REPORT:
The following events were reported: Flash Flood, Hail, Thunderstorm Wind, Tornado. These
occurred near station USW00003017, approximately 32.4879 km away, between 2004-06-09 13:00:00
and 2004-06-09 20:15:00. The events included records with hailstones measuring 10.2704
inches.The events resulted in 2 injuries and 0 fatalities. The events caused property damage
valued at 148400000.0 and crop damage of 0.0. Episode Narrative:
Event Narrative: Several roads were damaged from flooding and flash flooding over south-central
Weld County. Most of the damage occured from County Road 87 eastward. Weld County Road 91 was
completely washed out. Heavy rain damaged several county roads over southeastern Weld County.
Anywhere from 1 to 2 feet covered sections of some county roads, about 5 to 10 miles west of
State Highway 59. A tornado touched down in downtown Sterling, causing extensive damage. At
least two commercial buildings, including a car dealership, and one college dormitory lost
roofs. Several trees were uprooted; one caused major damage to a home when it fell on it. A
semi-truck was also knocked on its side. Downed power lines caused scattered electrical outages
which affected about 6,000 customers. A total of 150 buildings received some degree of damage.
Four families were left homeless and two residents received minor injuries. Some of the damage
to property and structures included:  $550,000 for damaged vehicles, $500,000 to the county
fairgrounds including the stadium and arena, $200,000 to Sterling High School, $70,000 to
Northeastern Junior College, and $20,000 to city property. The total economic impact to Sterling
which also involved cleanup and repair was estimated to be between $5 million and $7 million.
Locally heavy rain caused flash flooding in the Hayman burn area. Up to a foot of water covered
a portion of Westcreek Road. Debris also washed onto State Highway 67. A small tornado touched
down but did no damage. Strong thunderstorm winds damaged a mobile home. Severe thunderstorms
producing very large hail caused extensive damage across southern sections of the Denver
metropolitan area. Parts of Aurora, Lakewood, Littleton and South Denver incurred much of the
damage. The combined damage to homes and automobiles, not including commercial buildings, was
estimated at $146.5 million, making it the 4th costliest insurance disaster in the state's
history. Very heavy rain caused flooding and flash flooding problems in and around Sterling.
Several streets in town were inundated with up to a foot of standing water.

--- QUESTION ---
Given that a significant weather event occurred on June 9th, 2004, involving hail, tornadoes,
and flash floods in the vicinity of USW00003017, which of the following statements about the
temperature variations in time series A (leading up to the event) and subsequent time series B
is most accurate regarding atmospheric conditions and their likely impact on these temperature
patterns?

A) The prolonged heavy rainfall and flash flooding on June 9th led to an immediate drop in
temperatures as high humidity levels typically result in a rapid cooling effect in subsequent
days.
```

```
B) The temperature on June 10th, 2004, increased significantly due to residual heat from the
previous dayâĂŹs severe weather, indicating a typical warming pattern after storm events.

C) Following the sequence of extreme weather events, temperature readings showed a general
decline due to the atmospheric instability often associated with such occurrences, coupled with
the cooling effects of the heavy rain.

D) Temperatures during time series B reflected a steady increase, suggesting that prevailing
high pressure resulted from the absence of storm systems after June 9th.

--- GROUND TRUTH ---
Answer: C
```