DISCREPANCY-RANKED TEMPORAL VALIDATION FOR DEEP TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Validation-set selection in time-series forecasting has largely converged on the computationally scalable chronological tail holdout used in modern deep forecasting benchmarks. This default concentrates validation on recent targets, assuming that the latest regime best represents the test period. Earlier work on time-series performance estimation used repeated holdouts across multiple test periods to reduce dependence on any one period. The modern tail holdout instead uses a single validation period, trading temporal coverage for computational efficiency. We therefore propose *Discrepancy-Ranked Temporal Validation* (DRTV), a validation-set construction rule for checkpoint selection under a single-training-run budget. DRTV distributes validation positions across the development pool using a deterministic discrepancy-based rank ordering while preserving a separate test set. We formalize three desirable properties of validation-set construction: determinism, nestedness, and temporal coverage. The standard tail holdout satisfies the first two, whereas DRTV satisfies all three. Across 11 datasets and four forecasting models, DRTV achieved lower MAE than the tail holdout in 32 of 44 cases and lower MSE in 33 of 44. Across all 44 cases, the median relative error reductions were 3.50% for MAE and 6.62% for MSE. The gains varied substantially across dataset types; among cases without improvement for each metric, median degradation was 0.74% for MAE and 0.37% for MSE. By including eligible tail observations in the initial training run, DRTV can avoid a separate post-selection retraining step undertaken solely to incorporate them. These results position DRTV as a practical validation strategy for efficient model selection and deployment. Code is available at `https://anonymous.4open.science/r/ValSelect-A565/README.md`.

1 INTRODUCTION

Modern deep time-series forecasters select a checkpoint using validation loss. The validation set therefore determines which learned state reaches the test set even when the architecture, optimizer, and evaluation data are unchanged. In the long-term forecasting benchmark lineage established by Informer and Autoformer, the common protocol reserves a contiguous block immediately before the test period for validation (Zhou et al., 2021; Wu et al., 2021; Zhou et al., 2022; Zeng et al., 2023; Nie et al., 2023). This chronological tail holdout fixes validation position as inherited benchmark infrastructure rather than an explicit model-selection choice.

Earlier time-series evaluation sought evidence beyond a single testing period, both by adapting cross-validation to temporal data and by using rolling-origin or repeated-holdout tests across multiple periods (Bergmeir & Benítez, 2012; Tashman, 2000; Cerqueira et al., 2020). Applying such multi-period evaluation to deep checkpoint selection would require refitting models at each split, compounding training cost across the datasets and forecasting horizons of a benchmark. The chronological tail holdout offers a practical compromise: it supports checkpoint selection from one training run and places validation immediately before test. This placement is consistent with the assumption that the most recent regime best reflects test conditions (Wang et al., 2025). The trade-off is selection evidence from only one period and the exclusion of recent targets from the initial fit. Some forecasting workflows reincorporate those targets through post-selection retraining (Oreshkin et al., 2020). In adjacent temporal tabular learning, reducing training–test lag improved performance with validation and test periods held fixed (Cai & Ye, 2025). These observations motivate the possibility that recent

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

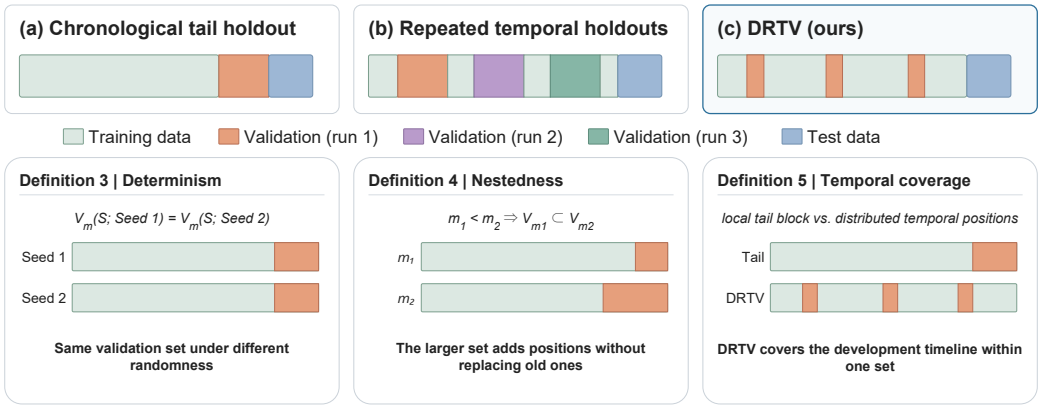


Figure 1: **Validation-position designs and structural properties.** A chronological tail holdout uses one terminal validation block, repeated temporal holdouts use different positions in separate runs, and DRTV distributes one validation set across the development pool. The lower panels summarize determinism, nestedness, and temporal coverage.

data may sometimes be more valuable for model fitting than for selection on one tail period. Figure 1 summarizes this design space and the structural properties that motivate our method.

We therefore seek a validation-position selection rule that draws evidence from across the pre-test period under a single-training-run budget. A replacement for the tail holdout should retain its determinism and nestedness as the validation set grows while adding temporal coverage. To develop such a rule, we revisit conventional machine-learning cross-validation, which has often used random partitioning or stratified sampling. For validation-set selection, however, random sampling can spread validation positions but introduces split-seed variation and may leave gaps at small validation-set sizes; uniform grids rebuilt independently at each size can improve coverage but cannot preserve nestedness across all sizes. Best-discrepancy systematic cross-validation (BDSCV) offers a useful starting point by replacing pseudo-random fold construction with a deterministic low-discrepancy ordering (Guo et al., 2021).

We call all observations preceding the test set the *development pool*; the tail holdout divides this pool into training and validation data. We propose *Discrepancy-Ranked Temporal Validation* (DRTV), which turns the best-discrepancy sequence into a deterministic ranking of eligible forecasting positions across the development pool. DRTV uses the same ranking at every validation-set size, retaining previously selected positions while distributing validation evidence beyond the tail. We remove training windows containing validation targets; the scaler and forecasting model are fitted on the remaining windows. Eligible tail observations can therefore participate in the initial training run while a separate validation set selects checkpoints. DRTV retains the tail holdout's determinism and nestedness but adds temporal coverage. DRTV can reduce the need for post-selection retraining aimed at incorporating recent data, since eligible tail observations already enter its initial fit.

Our contributions are:

- We introduce DRTV, a deterministic validation-position rule that draws checkpoint-selection evidence from across the development pool in one training run. Eligible tail observations can remain in the initial fit because only training windows containing validation targets are removed.
- We formalize determinism for reproducible validation positions, nestedness for extending a validation set without replacing earlier positions, and temporal coverage for drawing checkpoint-selection evidence beyond a single recent regime. The tail holdout satisfies the first two; DRTV satisfies all three.
- Across 11 datasets and four forecasting models, DRTV achieved lower MAE than the tail holdout in 32 of 44 cases and lower MSE in 33 of 44. Across all 44 cases, the median relative error reductions were 3.50% for MAE and 6.62% for MSE. Among cases without

improvement for each metric, the median relative degradation was 0.74% for MAE and 0.37% for MSE.

2 Related Work

2.1 Time-Series Validation

Time-series validation depends not only on temporal dependence but also on where evaluation occurs. Block-and-gap designs primarily control dependence and leakage (Racine, 2000; Bergmeir & Benítez, 2012). Rolling-origin and repeated-holdout protocols instead evaluate multiple periods, reducing reliance on one local regime (Tashman, 2000; Cerqueira et al., 2020). In their respective settings, repeated holdout yielded more accurate estimates for non-stationary series, while a single out-of-sample holdout was the most uncertain of three estimators in deep groundwater forecasting (Cerqueira et al., 2020; Doll et al., 2026). Together, these findings show that evidence from one temporal position can be unstable.

These studies do not directly solve checkpoint selection: an estimator of future error need not rank candidates equally well (Cerqueira et al., 2023). A single holdout also creates a positional dilemma. Withholding the most recent data may keep models from learning the latest information, whereas distant-past validation may select suboptimal parameters (Kuznetsov & Mohri, 2016). Repeated training across splits could recover multi-period evidence but exceed our single-run budget. We therefore study how one validation set can distribute checkpoint-selection evidence across the development period.

2.2 Validation in Deep Forecasting

Tail validation predates modern LTSF. N-BEATS holds out the latest observations for selection but reincorporates them before final test comparison (Oreshkin et al., 2020). Informer and Autoformer then establish a one-shot chronological train/validation/test protocol, with Autoformer calling 6:2:2 or 7:1:2 standard (Zhou et al., 2021; Wu et al., 2021). FEDformer, DLinear, and PatchTST subsequently treat this split as benchmark infrastructure (Zhou et al., 2022; Zeng et al., 2023; Nie et al., 2023). The same default persists across newer branches: lightweight FITS, SparseTSF, and TimeBase (Xu et al., 2024; Lin et al., 2024; Huang et al., 2025); multiscale MICN, TimeMixer, and Pathformer (Wang et al., 2023; 2024; Chen et al., 2024b); and channel-dependency CCM, DUET, and TimeFilter (Chen et al., 2024a; Qiu et al., 2025; Hu et al., 2025). Architectures have diversified, but selection evidence still comes from one chronological tail period.

Earlier forecasting studies provide the training-side premise: recent observations can be especially valuable under structural change, motivating adaptive weighting or online updating (Giraitis et al., 2013; Zhang et al., 2023). More recent adjacent work examines validation position directly. QuantBench compares tail, random, and fragmented validation and studies post-selection retraining (Wang et al., 2025); Cai and Ye separate validation bias from the training lag caused by chronological splits under temporal shift (Cai & Ye, 2025). Together, these studies show that validation position shapes not only selection evidence but also the recency of the information available for training.

2.3 Low-Discrepancy Construction

Low-discrepancy designs distribute a finite number of points over an ordered domain. They underpin quasi-Monte Carlo integration and simulation, including high-dimensional settings (Niederreiter, 1992; Dick et al., 2013), and have also been used to cover hyperparameter spaces, where the points are candidate configurations rather than validation positions (Bergstra & Bengio, 2012). Guo et al. introduce best-discrepancy systematic cross-validation (BDSCV) for ordered folds of i.i.d. classification samples (Guo et al., 2021). DRTV adapts its fractional-part rank idea to the chronological axis of forecasting windows. It forms one validation set from a shared rank prefix, excludes training windows containing validation targets, and uses validation loss for checkpoint selection. DRTV spreads validation evidence across the development pool through a deterministic ordering. This ordering eliminates Random's split-seed variation and extends earlier selections when the validation set grows, unlike independently rebuilt Uniform grids.

3 PROBLEM FORMULATION

We retain each dataset's standard test split and use all preceding observations as the development pool. Within this pool, each validation rule designates a subset of candidate start positions as validation points. Each selected point anchors an input–target forecasting window, and the resulting windows constitute the validation set. The training set consists of the remaining eligible windows after excluding any window that contains a validation-target time point.

3.1 VALIDATION-POINT SELECTION

Let $\boldsymbol{Y} \in \mathbb{R}^{T\times d}$ be a multivariate series and denote the last point before the test set by T_{dev}. With input length L and horizon H, the candidate start positions are indexed by $s \in \mathbb{S} = \{1, \ldots, N\}$, where $N = T_{\mathrm{dev}} - L - H + 1$. Therefore, each validation method can be viewed as a point-selection rule

$$\begin{aligned} &h \in \{\mathrm{DRTV}, \mathrm{Random}, \mathrm{Uniform}, \mathrm{Tail}\}, \\ &h : (\mathbb{S}, m) \longmapsto \mathbb{V}_m^{(h)} \subseteq \mathbb{S}. \end{aligned} \tag{1}$$

The key distinction among these rules lies in the temporal placement of validation points within the development pool. The standard tail holdout confines them to a contiguous block at the end of the pool, whereas DRTV, Random, and Uniform select points across its full temporal extent.

3.2 VALIDATION SET CONSTRUCTION

Each candidate point $s \in \mathbb{S}$ anchors an input sequence of length L followed by a target horizon of length H. The windows indexed by $\mathbb{V}_m^{(h)}$ constitute the validation set. To ensure that no validation target appears in training, we remove every candidate training window that contains a validation-target time point:

$$\mathbb{T}_m^{(h)} = \left\{ s \in \mathbb{S} \setminus \mathbb{V}_m^{(h)} : s \notin [v - H + 1, v + L + H - 1],\ \forall v \in \mathbb{V}_m^{(h)} \right\}. \tag{2}$$

The retained starts constitute the training set. Because Random can select nearby points, the resulting input–target windows and their exclusion spans can overlap. Consequently, Random can retain a different number of training windows than DRTV and Uniform, even though all three methods use m validation pairs. In contrast, tail holdout uses a contiguous block of $m(L + 2H)$ validation positions. To simplify the experimental setup, we restrict m to multiples of five and, for each dataset, select the value for which this block most closely matches the validation length under the benchmark's default split. Table 1 reports the selected m and the resulting retained-training proportions. All methods use exactly the same test set.

4 METHOD

Figure 2 visualizes the deterministic generation, rank-to-time mapping, and prefix selection used by DRTV. We first formalize temporal coverage through discrepancy. We then present DRTV as a discrepancy-based rule for selecting validation positions and establish its determinism, nestedness, and temporal coverage. Finally, we describe how DRTV fits into a standard deep forecasting pipeline.

4.1 DISCREPANCY AND DRTV CONSTRUCTION

Discrepancy is a classical measure of how uniformly a finite sequence covers a domain (Kuipers & Niederreiter, 1974). For a sequence $X = (x_i)_{i\geq 1} \subset [0, 1)$, we use the following interval form.

Definition 1 (Interval discrepancy)**.** The discrepancy of the first n terms of X is

$$D_n(X) = \sup_{0\leq a<b<1} \left| \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{x_i \in [a,b)\}} - (b - a) \right|. \tag{3}$$

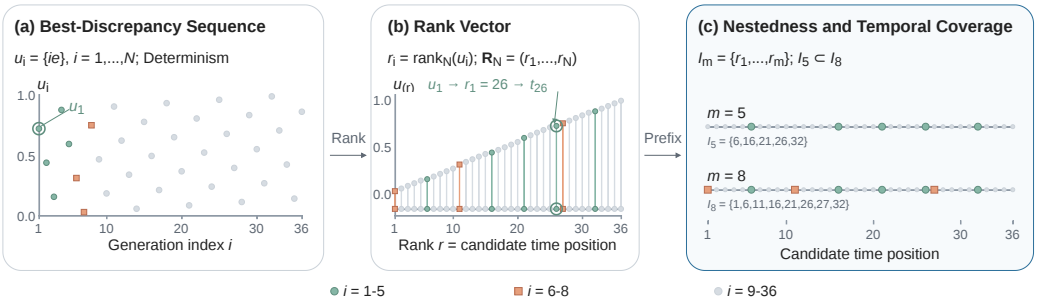


Figure 2: **From a best-discrepancy sequence to nested temporal positions.** (a) For fixed $N = 36$, DRTV deterministically generates $u_i = \{ie\}$. (b) Sorting the same N values assigns each original point its rank r_i, which indexes the chronologically ordered candidate position t_{r_i}. (c) Prefixes of the fixed rank vector produce $\mathcal{I}_5 \subset \mathcal{I}_8$ while retaining dispersed temporal coverage. DRTV maps ranks to candidate positions ordered by time.

A sequence X is a low-discrepancy sequence (LDS) if $D_n(X)$ remains small as n grows. Constructing an LDS aims to distribute a finite set of points as uniformly as possible over the domain, reducing errors from uneven sampling. Applied to positions along the time axis, this principle corresponds to temporal coverage, which we formalize in this paper.

The LDS concept specifies the desired coverage only qualitatively, without prescribing a discrepancy rate or a concrete sequence. Guo et al. (2021) make this requirement quantitative through the following best-discrepancy sequence (BDS) criterion.

Definition 2 (Best-discrepancy sequence)**.** A sequence X is a best-discrepancy sequence if, for every $\varepsilon > 0$, there is a constant $c_\varepsilon > 0$, independent of n, such that

$$D_n(X) \le c_\varepsilon n^{-1+\varepsilon}. \tag{4}$$

Guo et al. (2021, Theorem 2.2) show that the fractional-part sequence $X(e) = (\{ie\})_{i\ge1}$ is a BDS, where e is the base of the natural logarithm.

DRTV construction. The candidate time axis $\mathbb{S}_N := \mathbb{S} = \{1, \dots, N\}$ is already in chronological order and can therefore serve directly as the sorted population in the construction of Guo et al. (2021, Section 2.3). DRTV then sorts the first N terms of $X(e)$ and maps their ranks to the corresponding temporal positions. Let

$$X_N(e) = (u_1, \dots, u_N), \qquad u_i = \{ie\} = ie - \lfloor ie \rfloor, \quad i = 1, \dots, N, \tag{5}$$

and let $\widetilde{X}_N(e) = (\tilde{u}_1, \dots, \tilde{u}_N)$ denote its ascending ordering. The rank vector $\boldsymbol{r}_N = (r_1, \dots, r_N)$ is defined by

$$\tilde{u}_1 < \cdots < \tilde{u}_N, \qquad \tilde{u}_{r_i} = u_i, \quad i = 1, \dots, N. \tag{6}$$

Because e is irrational, the values u_i are distinct, so $\boldsymbol{r}_N$ is a permutation of $\mathbb{S}_N$. Each r_i thus indexes a unique temporal position. For a validation-set size $m \le N$, DRTV selects the validation points

$$\mathbb{V}_m^{(\mathrm{DRTV})} = \{r_1, \dots, r_m\} \subseteq \mathbb{S}, \tag{7}$$

For a fixed candidate population, the length-m prefix of this rank vector uniquely determines $\mathbb{V}_m^{(\mathrm{DRTV})}$, which establishes the determinism of DRTV; the selected points are subsequently windowed into input sequences of length L followed by target horizons of length H to form the validation set.

4.2 Structural Properties

We consider three desirable properties of a validation-point selection rule: determinism makes the selected positions reproducible, nestedness makes increases in validation size cumulative, and temporal coverage prevents the validation signal from being confined to a restricted part of the development period.

4.2.1 DETERMINISM

Definition 3 (Determinism)**.** Let $\omega \in \Omega$ denote all randomness in validation sampling and model training. A validation-point selection rule h is deterministic if, for every dataset-induced candidate population $\mathbb{S}$ and fixed m,

$$\mathbb{V}_m^{(h)}(\mathbb{S};\omega) = \mathbb{V}_m^{(h)}(\mathbb{S};\omega'), \qquad \forall\, \omega, \omega' \in \Omega.$$

Thus, $\mathbb{V}_m^{(h)}$ is determined solely by $\mathbb{S}$ and m and is invariant to randomness.

Determinism ensures that models with different initializations use the same validation set. The standard tail-holdout rule satisfies this requirement because its validation positions are fixed by the dataset split. Since e is fixed and irrational, the scores $u_i = \{ie\}$ are distinct and uniquely determine the rank vector $\boldsymbol{r}_N$. Consequently, $\mathbb{V}_m^{(\mathrm{DRTV})}$ is uniquely determined by N and m, and DRTV is deterministic.

4.2.2 NESTEDNESS

Definition 4 (Nestedness)**.** For a fixed candidate population $\mathbb{S}$, consider the family of validation-point sets $\{\mathbb{V}_m^{(h)}\}_{m=1}^N$ produced by a rule h as the realized validation size m varies. The rule is nested if this family is monotone under set inclusion:

$$\mathbb{V}_{m_1}^{(h)} \subseteq \mathbb{V}_{m_2}^{(h)}, \qquad \forall\, 1 \le m_1 < m_2 \le N.$$

Equivalently, increasing the validation size retains every previously selected point and introduces only additional points.

Proposition 1 (Nestedness of DRTV)**.** *For a fixed candidate population, let $\boldsymbol{r}_N = (r_1, \ldots, r_N)$ be the rank vector used by DRTV. For any $1 \le m_1 < m_2 \le N$,*

$$\mathbb{V}_{m_2}^{(\mathrm{DRTV})} = \mathbb{V}_{m_1}^{(\mathrm{DRTV})} \cup \{r_{m_1+1}, \ldots, r_{m_2}\}, \qquad \mathbb{V}_{m_1}^{(\mathrm{DRTV})} \subset \mathbb{V}_{m_2}^{(\mathrm{DRTV})}.$$

Thus, the family $\{\mathbb{V}_m^{(\mathrm{DRTV})}\}_{m=1}^N$ is nested in the sense of Definition 4.

Appendix A.1 proves the result. Nestedness keeps validation data consistent across sizes: the smaller set is retained within the larger, so changing m only adds or removes points. Tail holdout satisfies this property naturally because each larger terminal block extends the same chronological suffix. DRTV instead uses prefixes of a fixed low-discrepancy ordering, preserving consistency across sizes while extending coverage over time.

4.2.3 TEMPORAL COVERAGE

Mapping the ranks of BDS values to temporal positions transfers the BDS's uniform coverage to the development period. We call this property temporal coverage to distinguish it from Uniform, which is a deterministic baseline that selects evenly spaced positions.

Definition 5 (Temporal coverage)**.** Let $\mathbb{S}_N = \{1, \ldots, N\}$ denote the chronologically ordered candidate positions. By mapping each of the first N BDS values to its rank among these values, we transfer the uniform coverage of the BDS from its numerical domain to the candidate positions along the time axis. We refer to this coverage in the temporal domain as *temporal coverage*.

Determinism and nestedness alone still permit validation positions to remain concentrated in one part of the timeline, as in a tail holdout. Temporal coverage instead follows from the uniformity of the underlying BDS.

Proposition 2 (Temporal coverage of BDS prefixes)**.** *Let $X = (x_i)_{i\ge1}$ be a BDS whose first N values are distinct, and let r_i be the rank of x_i among these values. For $1 \le m \le N$, define*

$$\mathcal{I}_m = \{r_1, \ldots, r_m\}.$$

Then

$$\max_{q\in\{1,\ldots,N\}} \left| \frac{|\mathcal{I}_m \cap \{1,\ldots,q\}|}{m} - \frac{q}{N} \right| \le D_m^* + D_N^*,$$

where D_ℓ^* denotes the star discrepancy of the first ℓ BDS values.

Since $D_\ell^* \leq D_\ell(X)$, the BDS rate in Definition 2 implies that the coverage error converges to zero as $m \to \infty$ with $N \geq m$. Appendix A.2 proves the result. Under the DRTV construction, $\mathcal{I}_m = \mathbb{V}_m^{(\mathrm{DRTV})}$, so the proposition establishes temporal coverage for every DRTV prefix. Together with Proposition 1, it shows that increasing m appends points from one ordering while tightening the coverage bound asymptotically. It does not imply that realized coverage improves at every increment of m.

4.3 Using DRTV in Deep Forecasting

DRTV changes only the composition of the validation set. Whereas tail holdout selects validation positions from a contiguous block at the end of the development period, DRTV distributes them across the development pool to provide temporal coverage.

Determinism makes DRTV as reproducible as tail holdout, while nestedness keeps its validation sets consistent across sizes by retaining previously selected positions as m increases. Consequently, DRTV can be incorporated into existing deep-learning forecasting frameworks without modifying their training procedures.

5 Experiments

We ask three questions: whether DRTV selects better checkpoints than the chronological tail holdout (RQ1), whether its structural properties hold in the realized splits and how it compares with other globally distributed rules (RQ2), and how performance varies with nominal size (RQ3).

5.1 Experimental Setup

We evaluate DRTV on 11 widely used benchmarks for multivariate time-series forecasting: ETTm1, ETTm2, ETTh1, ETTh2, Electricity, Weather, PEMS03, PEMS04, PEMS07, PEMS08, and Traffic. We use four widely adopted forecasting baselines: DLinear (Zeng et al., 2023), PatchTST (Nie et al., 2023), STID (Shao et al., 2022), and iTransformer (Liu et al., 2024). Unless otherwise stated, the input length and forecasting horizon are $L = H = 96$.

We compare four validation-position methods: tail holdout, DRTV, Random sampling without replacement, and a deterministic endpoint-inclusive Uniform grid. For every method, we exclude from training any window whose target overlaps a validation target. RQ1 and RQ2 use seeds 1–5 at frozen dataset-specific nominal sizes, whereas RQ3 compares DRTV with tail holdout at four nominal sizes using seed 1. Because the selected validation positions map to overlapping input–target windows, the realized validation and retained-training set sizes cannot be matched exactly across methods. Table 1 reports the configurations used and the resulting retained-training proportions. All experiments are implemented in BasicTS v0.5.8 (Shao et al., 2025). Within each comparison, all validation methods use the same test set and identical model-training hyperparameters.

5.2 RQ1: DRTV versus Tail Holdout

Figure 3 compares checkpoints selected by DRTV and tail holdout. DRTV achieved lower MAE in 32 of 44 cases and lower MSE in 33 of 44, with median relative error reductions of 3.50% and 6.62%, respectively. In the remaining cases, the median relative degradations were 0.74% for MAE and 0.37% for MSE. The gains were concentrated in the ETT family, where all 16 cases favored DRTV for both metrics. By contrast, PEMS showed no consistent family-level direction, with 7 of 16 MAE cases and 6 of 16 MSE cases favoring DRTV.

In real-world forecasting applications, test targets correspond to future observations and are therefore unavailable during model development, so model selection must rely on a training–validation partition of the observed history. By distributing validation positions throughout the development pool rather than withholding a contiguous tail segment, DRTV retains eligible recent observations for model fitting. The results indicate that this reallocation is beneficial overall, delivering clear gains in some settings without a statistically detectable systematic disadvantage in others.

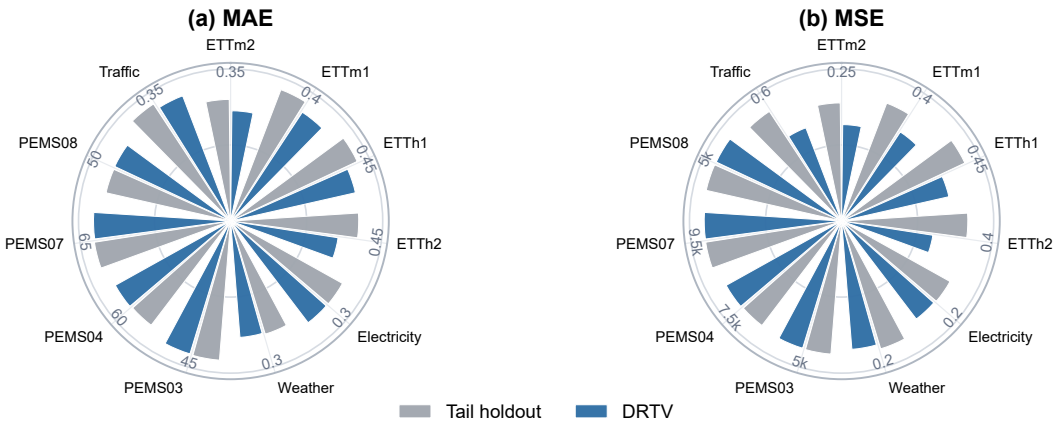


Figure 3: **DRTV versus tail holdout across 11 datasets.** Adjacent bars summarize four models and five paired seeds. Panel (a) reports overall test MAE and panel (b) reports overall test MSE. Lower is better for both metrics.

5.3 RQ2: Realized Properties and Global Sampling Rules

RQ2 assesses the importance of the three properties formalized above through comparisons with three representative alternatives. Random serves as the contrast for determinism; tail holdout provides the reference without global temporal coverage, whereas an endpoint-inclusive Uniform grid serves as the global-coverage baseline. Figure 4 summarizes the results on 11 datasets. For each dataset and method, the 20 test errors from four forecasting baselines and five seeds are divided by the same-seed Random error and multiplied by 100; the bars and error bars show the mean and sample standard deviation. Figure 6 reports the same comparison separately for each forecasting baseline. Tables 3, 4, and 5 provide the detailed numerical values.

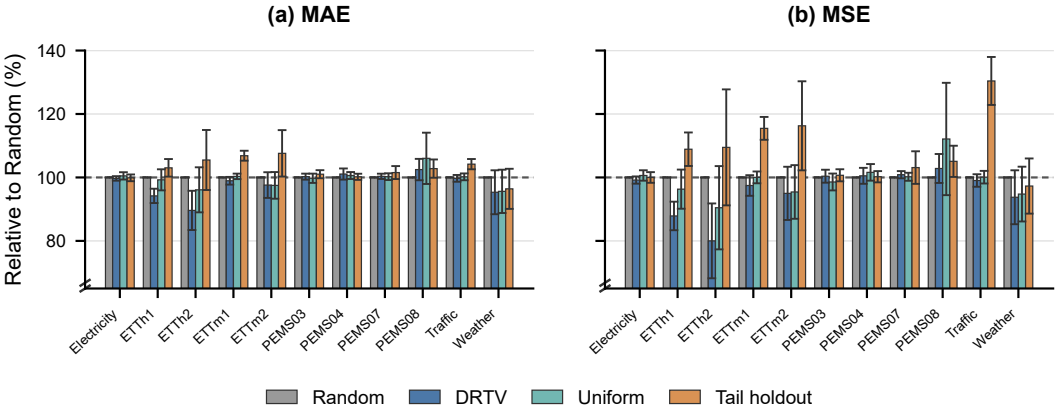


Figure 4: **Performance at a fixed nominal size relative to paired Random.** Each value is divided by the same-seed Random result and multiplied by 100. Bars and error bars are the mean and sample standard deviation across 20 runs (four models and five seeds). Panels (a) and (b) show overall MAE and MSE; lower is better for both.

Compared with tail holdout, the globally covering Uniform baseline yields lower MAE and MSE in 33 of 44 model–dataset cases, supporting the practical value of global temporal coverage. DRTV further improves on Uniform, with lower MAE in 30 of 44 cases and lower MSE in 31 of 44. Unlike Uniform, DRTV preserves selected positions as the nominal size increases and is therefore nested across sizes. DRTV is also seed-independent and outperforms Random on both metrics in 7 of 11 datasets, although Random performs better on all four PEMS datasets. These mixed results indicate that validation-set selection from the full development pool requires further study.

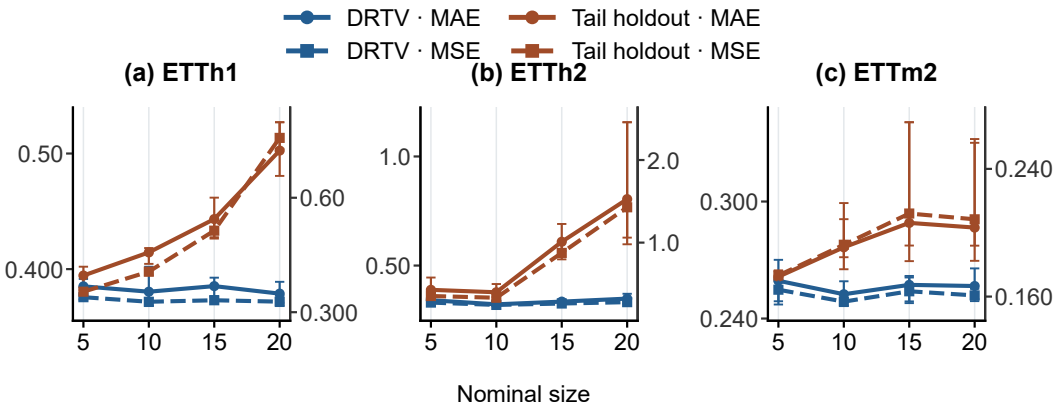


Figure 5: **DRTV selects checkpoints with broadly comparable test errors across nominal sizes, while its advantage over tail holdout narrows at smaller m as more windows are retained for training.** Panels (a)–(c) show ETTh1, ETTh2, and ETTm2 at $m \in \{5, 10, 15, 20\}$. At each nominal size, lines show the equal-weight mean of raw overall test errors across four forecasting models (seed 1). Blue denotes DRTV and brown denotes tail holdout; solid lines show MAE (left axis), and dashed lines show MSE (right axis). Vertical whiskers show the range across models, not confidence intervals or across-seed variation. The two vertical scales are independent, and lower error is better. Complete model-level relative effects for all 11 datasets are reported in Appendix Figure 7.

5.4 RQ3: Sensitivity to Nominal Size

RQ3 addresses a practical question left open by the default validation protocol: how much observed data should be allocated to validation? We compare DRTV and tail holdout at $m \in \{5, 10, 15, 20\}$ on all datasets. Varying m indirectly examines the mismatch in realized validation and retained-training sizes across validation rules; RQ3 focuses on DRTV and tail holdout because both preserve nested validation sets as m increases, ensuring that larger validation sets retain all positions selected at smaller sizes.

Under both methods, reducing m allocates fewer windows to validation and retains more for training. Figure 5 shows that DRTV selects checkpoints with broadly comparable test errors across the four sizes on the three ETT datasets. Although the performance gap between DRTV and tail holdout narrows as m decreases and the retained training set grows, the appropriate validation size cannot be known a priori in practice. DRTV therefore offers a more practical strategy across plausible validation sizes. However, the relationship between validation size and relative method performance is not consistent across model–dataset cases: the two methods often perform similarly, and varying the validation size does not systematically favor either method. This heterogeneity may reflect differences in model design and dataset characteristics; complete results are reported in Appendix Figure 7.

6 Conclusion

We introduced DRTV as a deterministic validation rule for deep time-series forecasting. It distributes validation positions across the development pool and preserves nestedness across sizes. Across 11 datasets and four models, DRTV achieved lower MAE in 32 of 44 cases and lower MSE in 33 of 44. Median relative error reductions were 3.50% and 6.62%, respectively; for cases without improvement under each metric, median degradation was only 0.74% for MAE and 0.37% for MSE. We formalized three properties of validation-set construction: determinism, temporal coverage, and nestedness. Random and Uniform comparisons and validation-size experiments support the practical importance of these properties. DRTV satisfies all three and therefore offers reproducible validation with global temporal coverage when future test targets are unavailable and validation size is uncertain. It also retains eligible tail observations for training and can avoid retraining solely to reincorporate them.

AI USE STATEMENT

Generative AI tools were used to aid and polish writing, retrieve and discover relevant literature, draft portions of the manuscript, and assist in developing proofs of mathematical claims. They were not used for research ideation or execution or for generating synthetic datasets. The authors independently checked all AI-assisted text, verified cited sources against the original works, and verified the mathematical arguments and proofs. The authors take full responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

Anonymous code and experiment configurations are available at `https://anonymous.4open.science/r/ValSelect-A565/README.md`. All experiments use BasicTS v0.5.8 (Shao et al., 2025); the anonymous repository provides the corresponding implementations and configurations. Computations were performed on a 40 GB NVIDIA A100 GPU.

REFERENCES

Christoph Bergmeir and José M. Benítez. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 2012.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305, 2012.

Haorun Cai and Han-Jia Ye. Understanding the limits of deep tabular methods with temporal shift. In *International Conference on Machine Learning*, 2025.

Vitor Cerqueira, Luís Torgo, and Igor Mozetič. Evaluating time series forecasting models: An empirical study on performance estimation methods. *Machine Learning*, 2020.

Vitor Cerqueira, Luís Torgo, and Carlos Soares. Model selection for time series forecasting: An empirical analysis of multiple estimators. *Neural Processing Letters*, 2023.

Jialin Chen, Jan Eric Lenssen, Aosong Feng, Weihua Hu, Matthias Fey, Leandros Tassiulas, Jure Leskovec, and Rex Ying. From similarity to superiority: Channel clustering for time series forecasting. In *Advances in Neural Information Processing Systems*, volume 37, 2024a. URL `https://proceedings.neurips.cc/paper_files/paper/2024/hash/eb9b18ccb76a1156af5779ffdca1d91f-Abstract-Conference.html`.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *International Conference on Learning Representations*, 2024b. URL `https://openreview.net/forum?id=lJkOCMP2aW`.

Josef Dick, Frances Y. Kuo, and Ian H. Sloan. High-dimensional integration: The quasi-monte carlo way. *Acta Numerica*, 22:133–288, 2013. doi: 10.1017/S0962492913000044.

Fabienne Doll, Tanja Liesch, Maria Wetzel, Stefan Kunz, and Stefan Broda. Validation strategies for deep learning-based groundwater level time series prediction using exogenous meteorological input features. *Geoscientific Model Development*, 19:2657–2675, 2026. doi: 10.5194/gmd-19-2657-2026.

Liudas Giraitis, George Kapetanios, and Simon Price. Adaptive forecasting in the presence of recent and ongoing structural change. *Journal of Econometrics*, 177(2):153–170, 2013. doi: 10.1016/j.jeconom.2013.04.003.

Liang Guo, Jianya Liu, and Ruodan Lu. Subsampling bias and the best-discrepancy systematic cross validation. *Science China Mathematics*, 64(1):197–210, 2021. doi: 10.1007/s11425-018-9561-0.

Yifan Hu, Guibin Zhang, Peiyuan Liu, Disen Lan, Naiqi Li, Dawei Cheng, Tao Dai, Shu-Tao Xia, and Shirui Pan. TimeFilter: Patch-specific spatial-temporal graph filtration for time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*,

volume 267 of *Proceedings of Machine Learning Research*, pp. 24893–24911, 2025. URL https://proceedings.mlr.press/v267/hu25ac.html.

Qihe Huang, Zhengyang Zhou, Kuo Yang, Zhongchao Yi, Xu Wang, and Yang Wang. TimeBase: The power of minimalism in efficient long-term time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 26227–26246, 2025. URL https://proceedings.mlr.press/v267/huang25az.html.

Lauwerens Kuipers and Harald Niederreiter. *Uniform Distribution of Sequences*. Wiley, New York, 1974. ISBN 0471510459.

Vitaly Kuznetsov and Mehryar Mohri. Time series prediction and online learning. In *Proceedings of the 29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pp. 1190–1213, 2016. URL https://proceedings.mlr.press/v49/kuznetsov16.html.

Shengsheng Lin, Weiwei Lin, Wentai Wu, Haojun Chen, and Junjie Yang. SparseTSF: Modeling long-term time series forecasting with 1k parameters. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 30211–30226, 2024. URL https://proceedings.mlr.press/v235/lin24n.html.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/2ea18fdc667e0ef2ad82b2b4d65147ad-Abstract-Conference.html.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Harald Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*, volume 63 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992. ISBN 9780898712957. doi: 10.1137/1.9781611970081.

Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, 2020.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. DUET: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1185–1196, 2025. doi: 10.1145/3690624.3709325.

Jeff Racine. Consistent cross-validatory model-selection for dependent data: hv-block cross-validation. *Journal of Econometrics*, 2000.

Zezhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *ACM International Conference on Information and Knowledge Management*, 2022. URL https://arxiv.org/abs/2208.05233.

Zezhi Shao, Fei Wang, Yongjun Xu, Wei Wei, Chengqing Yu, Zhao Zhang, Di Yao, Tao Sun, Guangyin Jin, Xin Cao, Gao Cong, Christian S. Jensen, and Xueqi Cheng. Exploring Progress in Multivariate Time Series Forecasting: Comprehensive Benchmarking and Heterogeneity Analysis. *IEEE Transactions on Knowledge and Data Engineering*, 37(1):291–305, January 2025. ISSN 2326-3865.

Leonard J. Tashman. Out-of-sample tests of forecasting accuracy: An analysis and review. *International Journal of Forecasting*, 2000.

Huiqiang Wang, Jian Peng, Feihu Huang, Jince Wang, Junhui Chen, and Yifei Xiao. MICN: Multi-scale local and global context modeling for long-term series forecasting. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=zt53IDUR1U.

Saizhuo Wang, Hao Kong, Jiadong Guo, Fengrui Hua, Yiyan Qi, Wanyun Zhou, Jiahao Zheng, Xinyu Wang, Lionel M. Ni, and Jian Guo. QuantBench: Benchmarking AI methods for quantitative investment. *arXiv preprint arXiv:2504.18600*, 2025. doi: 10.48550/arXiv.2504.18600.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=7oLshfEIC2.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, 2021.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with 10k parameters. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=bWcnvZ3qMb.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *AAAI Conference on Artificial Intelligence*, 2023.

Yi-Fan Zhang, Qingsong Wen, Xue Wang, Weiqi Chen, Liang Sun, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. OneNet: Enhancing time series forecasting models under concept drift by online ensembling. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-3066. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/dd6a47bc0aad6f34aa5e77706d90cdc4-Abstract.html.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *AAAI Conference on Artificial Intelligence*, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, 2022.

A DRTV Details

A.1 Proof of Nestedness

Proof of Proposition 1. For $m_1 < m_2$, both DRTV sets are prefixes of the same permutation:

$$\mathbb{V}_{m_2}^{(\mathrm{DRTV})} = \mathbb{V}_{m_1}^{(\mathrm{DRTV})} \cup \{r_{m_1+1}, \ldots, r_{m_2}\}. \tag{8}$$

The added ranks are distinct from the first m_1, giving strict containment. □

A.2 Proof of Temporal Coverage

Proof of Proposition 2. Write the sorted BDS values as $x_{(1)} < \cdots < x_{(N)}$. For each $q \in \{1, \ldots, N-1\}$, choose $t_q \in (x_{(q)}, x_{(q+1)})$. By construction,

$$r_i \le q \quad \Longleftrightarrow \quad x_i < t_q, \qquad \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\{x_i < t_q\}} = \frac{q}{N}.$$

Table 1: Dataset scale and effective training proportions. Length and Vars. are the numbers of raw time steps and variables; Test is the shared number of test forecasting windows; and m is the nominal validation size. The four method columns report retained train windows divided by Length (%), while Default is the original raw-time-point training percentage.

Dataset	Length	Vars.	Test	m	Train / Length (%)				Default (%)
					Random	Uniform	DRTV	Tail holdout	
ETTm2	57,600	7	11,329	40	61.91	60.23	59.74	59.50	60.0
ETTm1	57,600	7	11,329	40	61.91	60.23	59.74	59.50	60.0
ETTh1	14,400	7	2,689	10	60.62	60.73	58.74	58.01	60.0
ETTh2	14,400	7	2,689	10	60.62	60.73	58.74	58.01	60.0
Electricity	26,304	321	5,069	10	69.33	69.45	68.37	67.97	70.0
Weather	52,696	21	10,348	20	69.31	69.29	68.75	68.53	70.0
PEMS03	26,208	358	5,050	20	60.03	58.46	57.37	56.93	60.0
PEMS04	16,992	307	3,207	15	57.88	55.23	53.54	52.90	60.0
PEMS07	28,224	883	5,453	20	61.33	60.00	58.99	58.58	60.0
PEMS08	17,856	170	3,380	15	58.83	56.42	54.82	54.21	60.0
Traffic	17,544	862	3,317	10	64.07	64.19	62.56	61.96	70.0

Consequently, the triangle inequality gives

$$\begin{aligned}\left|\frac{|\mathcal{I}_m \cap \{1,\dots,q\}|}{m} - \frac{q}{N}\right| &= \left|\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}_{\{x_i<t_q\}} - \frac{q}{N}\right| \\ &\le \left|\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}_{\{x_i<t_q\}} - t_q\right| + \left|t_q - \frac{q}{N}\right| \\ &\le D_m^* + D_N^*.\end{aligned}$$

The term for $q = N$ is zero, so taking the maximum over $q \in \{1,\dots,N\}$ proves the bound. □

B Detailed DRTV–Tail-Holdout Results

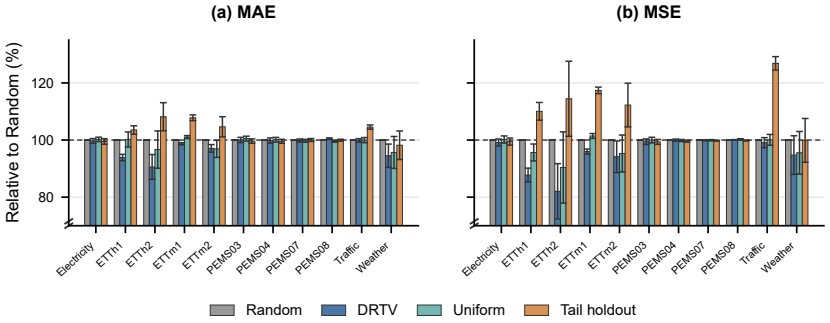
Table 1 reports the configurations used and the realized retained-training proportions for all four validation-position methods.

Table 2 reports the RQ1 model–dataset MAE and MSE summaries. Table 3 gives, for each dataset and method, the arithmetic mean and sample standard deviation across the four models and five seeds ($n = 20$), computed before rounding. These MAE/MSE means underlie Figure 3; the corresponding model-specific five-seed summaries appear in Table 2.

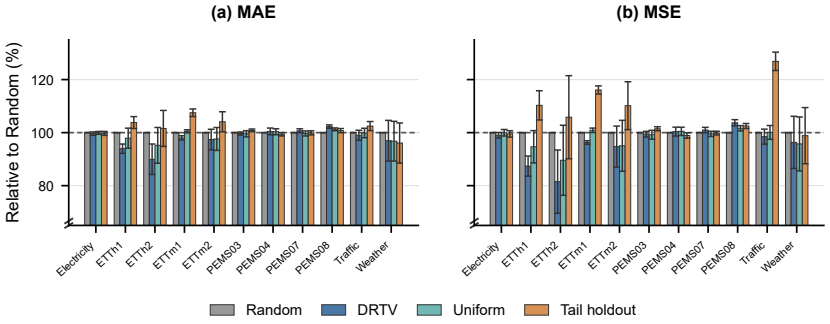
C Model-Specific Fixed-Nominal-Size Results

Figure 6 groups the four forecasting models. Within each model plot, panel (a) shows MAE and panel (b) shows MSE. Each panel divides a method–seed value by the same-seed Random result before summarizing five normalized runs. Tables 4 and 5 report the MAE and MSE comparisons numerically, with each dataset spanning four model rows.

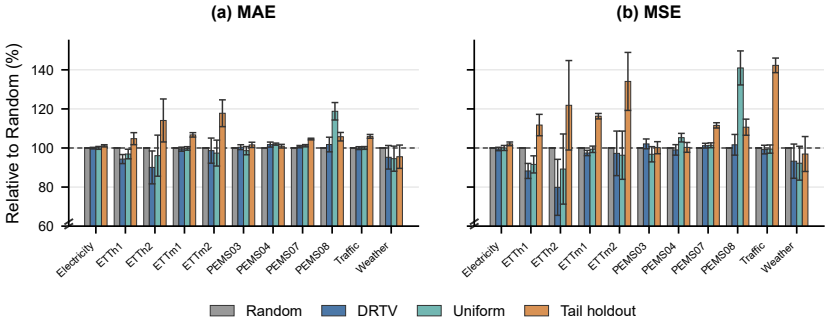
Detailed RQ2 observations. DRTV's paired-Random-normalized test error is lower than tail holdout in 25 of 28 non-PEMS model–dataset cases for MAE and 27 of 28 for MSE, including all 16 ETT cases on both metrics. Uniform also yields lower error than tail holdout in 33 of 44 cases for each metric, supporting the practical value of distributing validation positions across time. DRTV further improves on Uniform in 30 of 44 cases for MAE and 31 of 44 cases for MSE while retaining previously selected positions as the nominal size increases. Performance nevertheless varies across dataset types: Random has the lowest mean among the four validation rules in 7 of 16 PEMS cases for MAE and 6 of 16 for MSE.



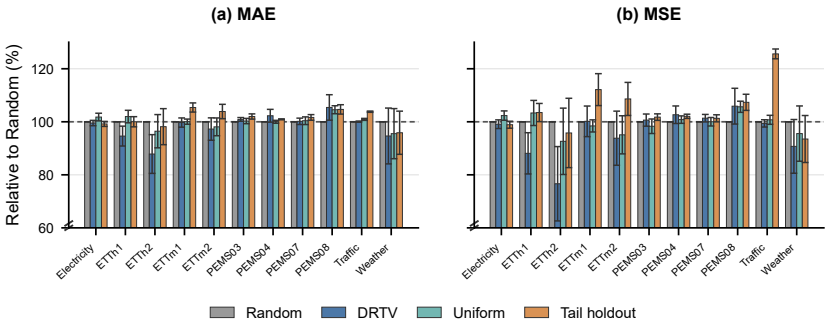
(i) DLinear



(ii) PatchTST



(iii) STID



(iv) iTransformer

Figure 6: **Model-specific performance relative to paired Random.** Bars and error bars show the mean and sample standard deviation over five paired seeds. Within each model plot, (a) shows MAE and (b) shows MSE.

Table 2: Test MAE and MSE under tail-holdout and DRTV validation at dataset-specific nominal sizes. Results are mean ± sample standard deviation over five paired training seeds; the lower mean in each model–dataset comparison is bold.

Dataset (m)	Split	DLinear	PatchTST	STID	iTransformer
Panel A: Test MAE					
ETT datasets					
ETTm2 (40)	Tail holdout	0.2753 ± 0.0023	0.2665 ± 0.0017	0.3113 ± 0.0086	0.2669 ± 0.0022
	DRTV	$\mathbf{0.2555 \pm 0.0040}$	$\mathbf{0.2493 \pm 0.0018}$	$\mathbf{0.2605 \pm 0.0034}$	$\mathbf{0.2498 \pm 0.0034}$
ETTm1 (40)	Tail holdout	0.3696 ± 0.0023	0.3648 ± 0.0054	0.3690 ± 0.0026	0.3769 ± 0.0041
	DRTV	$\mathbf{0.3384 \pm 0.0011}$	$\mathbf{0.3326 \pm 0.0019}$	$\mathbf{0.3436 \pm 0.0019}$	$\mathbf{0.3566 \pm 0.0050}$
ETTh1 (10)	Tail holdout	0.4039 ± 0.0019	0.4143 ± 0.0024	0.4179 ± 0.0047	0.4199 ± 0.0017
	DRTV	$\mathbf{0.3663 \pm 0.0012}$	$\mathbf{0.3750 \pm 0.0048}$	$\mathbf{0.3762 \pm 0.0025}$	$\mathbf{0.3971 \pm 0.0095}$
ETTh2 (10)	Tail holdout	0.3826 ± 0.0072	0.3552 ± 0.0022	0.4109 ± 0.0054	0.3670 ± 0.0117
	DRTV	$\mathbf{0.3204 \pm 0.0038}$	$\mathbf{0.3146 \pm 0.0029}$	$\mathbf{0.3244 \pm 0.0028}$	$\mathbf{0.3281 \pm 0.0109}$
PEMS datasets					
PEMS03 (20)	Tail holdout	$\mathbf{72.88 \pm 0.13}$	39.27 ± 0.05	21.95 ± 0.13	31.10 ± 0.18
	DRTV	73.17 ± 0.37	$\mathbf{38.80 \pm 0.22}$	$\mathbf{21.70 \pm 0.15}$	$\mathbf{30.76 \pm 0.25}$
PEMS04 (15)	Tail holdout	$\mathbf{85.40 \pm 0.13}$	$\mathbf{56.69 \pm 0.19}$	$\mathbf{23.63 \pm 0.14}$	$\mathbf{41.06 \pm 0.11}$
	DRTV	85.63 ± 0.31	57.29 ± 0.45	23.82 ± 0.34	41.61 ± 1.09
PEMS07 (20)	Tail holdout	107.21 ± 0.08	$\mathbf{59.29 \pm 0.14}$	26.75 ± 0.13	42.46 ± 0.41
	DRTV	$\mathbf{107.00 \pm 0.15}$	59.77 ± 0.42	$\mathbf{25.76 \pm 0.11}$	$\mathbf{41.84 \pm 0.31}$
PEMS08 (15)	Tail holdout	$\mathbf{76.96 \pm 0.05}$	$\mathbf{40.31 \pm 0.25}$	19.78 ± 0.06	$\mathbf{30.93 \pm 0.19}$
	DRTV	77.36 ± 0.11	40.93 ± 0.09	$\mathbf{19.03 \pm 0.65}$	31.14 ± 1.03
Other datasets					
Electricity (10)	Tail holdout	$\mathbf{0.2781 \pm 0.0010}$	0.2510 ± 0.0004	0.2412 ± 0.0007	$\mathbf{0.2403 \pm 0.0006}$
	DRTV	0.2786 ± 0.0007	$\mathbf{0.2508 \pm 0.0003}$	$\mathbf{0.2384 \pm 0.0003}$	0.2411 ± 0.0007
Weather (20)	Tail holdout	0.2599 ± 0.0035	$\mathbf{0.2124 \pm 0.0009}$	0.2223 ± 0.0009	0.2386 ± 0.0042
	DRTV	$\mathbf{0.2503 \pm 0.0029}$	0.2143 ± 0.0012	$\mathbf{0.2217 \pm 0.0023}$	$\mathbf{0.2350 \pm 0.0038}$
Traffic (10)	Tail holdout	0.4046 ± 0.0002	0.2892 ± 0.0007	0.3222 ± 0.0011	0.2746 ± 0.0010
	DRTV	$\mathbf{0.3866 \pm 0.0013}$	$\mathbf{0.2794 \pm 0.0012}$	$\mathbf{0.3039 \pm 0.0005}$	$\mathbf{0.2647 \pm 0.0008}$
W/T/L	DRTV vs. tail holdout	7/0/4	7/0/4	10/0/1	8/0/3
Median improvement (%)	DRTV vs. tail holdout	+3.73%	+1.19%	+3.77%	+1.51%
Panel B: Test MSE					
ETT datasets					
ETTm2 (40)	Tail holdout	0.1838 ± 0.0007	0.1837 ± 0.0020	0.2241 ± 0.0099	0.1854 ± 0.0025
	DRTV	$\mathbf{0.1542 \pm 0.0013}$	$\mathbf{0.1579 \pm 0.0024}$	$\mathbf{0.1624 \pm 0.0048}$	$\mathbf{0.1597 \pm 0.0071}$
ETTm1 (40)	Tail holdout	0.3405 ± 0.0006	0.3256 ± 0.0051	0.3172 ± 0.0025	0.3448 ± 0.0078
	DRTV	$\mathbf{0.2784 \pm 0.0004}$	$\mathbf{0.2700 \pm 0.0025}$	$\mathbf{0.2658 \pm 0.0015}$	$\mathbf{0.3082 \pm 0.0132}$
ETTh1 (10)	Tail holdout	0.3954 ± 0.0013	0.4022 ± 0.0055	0.4049 ± 0.0041	0.4100 ± 0.0038
	DRTV	$\mathbf{0.3151 \pm 0.0008}$	$\mathbf{0.3188 \pm 0.0067}$	$\mathbf{0.3199 \pm 0.0036}$	$\mathbf{0.3486 \pm 0.0191}$
ETTh2 (10)	Tail holdout	0.3287 ± 0.0087	0.3068 ± 0.0041	0.3661 ± 0.0112	0.3263 ± 0.0166
	DRTV	$\mathbf{0.2355 \pm 0.0028}$	$\mathbf{0.2364 \pm 0.0032}$	$\mathbf{0.2402 \pm 0.0020}$	$\mathbf{0.2598 \pm 0.0180}$
PEMS datasets					
PEMS03 (20)	Tail holdout	$\mathbf{10094.3 \pm 5.7}$	3645.5 ± 8.6	$\mathbf{1470.2 \pm 22.3}$	2345.6 ± 39.7
	DRTV	10101.3 ± 19.4	$\mathbf{3571.9 \pm 37.5}$	1499.0 ± 19.0	$\mathbf{2319.5 \pm 53.0}$
PEMS04 (15)	Tail holdout	$\mathbf{13512.0 \pm 2.8}$	$\mathbf{7053.2 \pm 33.0}$	1389.6 ± 12.3	$\mathbf{3955.7 \pm 25.9}$
	DRTV	13562.3 ± 17.0	7157.1 ± 88.9	$\mathbf{1371.9 \pm 24.2}$	3978.4 ± 117.5
PEMS07 (20)	Tail holdout	$\mathbf{19989.1 \pm 2.6}$	$\mathbf{8123.7 \pm 19.9}$	1945.8 ± 22.3	$\mathbf{4310.5 \pm 22.5}$
	DRTV	20022.4 ± 20.6	8218.7 ± 91.6	$\mathbf{1764.8 \pm 12.4}$	4313.3 ± 79.0
PEMS08 (15)	Tail holdout	$\mathbf{10853.0 \pm 3.7}$	$\mathbf{4057.6 \pm 44.5}$	969.9 ± 5.0	2374.7 ± 31.2
	DRTV	10878.8 ± 14.6	4105.6 ± 27.1	$\mathbf{891.1 \pm 39.5}$	$\mathbf{2341.3 \pm 102.1}$
Other datasets					
Electricity (10)	Tail holdout	0.1966 ± 0.0001	0.1644 ± 0.0003	0.1428 ± 0.0007	$\mathbf{0.1492 \pm 0.0005}$
	DRTV	$\mathbf{0.1960 \pm 0.0002}$	$\mathbf{0.1636 \pm 0.0003}$	$\mathbf{0.1391 \pm 0.0003}$	0.1495 ± 0.0007
Weather (20)	Tail holdout	0.1963 ± 0.0016	0.1705 ± 0.0008	0.1651 ± 0.0005	0.1712 ± 0.0036
	DRTV	$\mathbf{0.1862 \pm 0.0010}$	$\mathbf{0.1662 \pm 0.0011}$	$\mathbf{0.1588 \pm 0.0006}$	$\mathbf{0.1658 \pm 0.0022}$
Traffic (10)	Tail holdout	0.6752 ± 0.0004	0.4581 ± 0.0007	0.5240 ± 0.0032	0.4146 ± 0.0011
	DRTV	$\mathbf{0.5273 \pm 0.0006}$	$\mathbf{0.3556 \pm 0.0012}$	$\mathbf{0.3653 \pm 0.0012}$	$\mathbf{0.3281 \pm 0.0005}$
W/T/L	DRTV vs. tail holdout	7/0/4	8/0/3	10/0/1	8/0/3
Median improvement (%)	DRTV vs. tail holdout	+5.13%	+2.51%	+9.30%	+3.14%

m is the nominal validation size. DLinear, PatchTST, STID, and iTransformer are the frozen forecasting models; the tail holdout and DRTV are validation-position methods. All displayed cells have $n = 5$; incomplete cells are shown as dashes and are not filled from another size or method. W/T/L compares dataset-level five-seed means, with a win indicating lower DRTV error. Median improvement is $(\text{Tail} - \text{DRTV})/\text{Tail} \times 100\%$ over datasets with both complete cells. Raw errors are compared only within a dataset.

D Nominal-Size Analysis

Figure 7 shows model-specific effects for all 11 datasets, including the three shown as four-model means in Figure 5. The dataset panels form two columns and six rows (one dataset in the left column of the final row); within each panel, the four MAE model columns sit directly beside the four MSE columns, with model abbreviations repeated below each panel.

In the seed-1 nominal-size scan, DRTV reaches the RQ3 tail-holdout reference MAE in 41 of 44 model–dataset cases; 37 do so at a smaller nominal size while retaining more training windows. For MAE, DRTV has lower four-point variance, coefficient of variation, and total variation in 31, 31, and 30 of 44 cases, respectively; the corresponding MSE counts are 35, 35, and 34. These descriptive comparisons do not establish equivalence, non-inferiority, or cross-seed stability.

Table 3: MAE/MSE source values for Figure 3. Values are arithmetic mean ± sample standard deviation over four forecasting models and five training seeds ($n = 20$ per method–dataset cell). Values are rounded to three decimals here; the source CSV retains full precision. Lower error is better.

	MAE		MSE	
Dataset (m)	Tail holdout	DRTV	Tail holdout	DRTV
ETTm2 (40)	0.280 ± 0.019	0.254 ± 0.006	0.194 ± 0.018	0.159 ± 0.005
ETTm1 (40)	0.370 ± 0.006	0.343 ± 0.009	0.332 ± 0.012	0.281 ± 0.018
ETTh1 (10)	0.414 ± 0.007	0.379 ± 0.013	0.403 ± 0.007	0.326 ± 0.017
ETTh2 (10)	0.379 ± 0.022	0.322 ± 0.008	0.332 ± 0.024	0.243 ± 0.013
Electricity (10)	0.253 ± 0.016	0.252 ± 0.016	0.163 ± 0.021	0.162 ± 0.022
Weather (20)	0.233 ± 0.019	0.230 ± 0.014	0.176 ± 0.013	0.169 ± 0.011
PEMS03 (20)	41.299 ± 19.737	41.108 ± 19.983	4388.903 ± 3471.688	4372.917 ± 3476.835
PEMS04 (15)	51.694 ± 23.293	52.089 ± 23.297	6477.634 ± 4647.089	6517.408 ± 4673.005
PEMS07 (20)	58.928 ± 30.942	58.593 ± 31.220	8592.264 ± 7119.623	8579.800 ± 7176.791
PEMS08 (15)	41.995 ± 22.013	42.114 ± 22.350	4563.778 ± 3890.625	4554.192 ± 3924.537
Traffic (10)	0.323 ± 0.052	0.309 ± 0.048	0.518 ± 0.101	0.394 ± 0.080

Table 4: **Model-specific overall MAE relative to paired Random.** Entries are mean ± sample SD of same-seed ratios (%) over five paired seeds. Lower is better; the lowest unrounded mean in each dataset–model row is bold. Random is 100 by construction. Each model uses the frozen dataset-specific nominal validation size. Each dataset spans four model rows.

Dataset	Model	Random	DRTV	Uniform	Tail holdout
Electricity	DLinear	100.000 ± 0.000	99.684 ± 0.832	100.235 ± 0.775	**99.511 ± 0.966**
	PatchTST	100.000 ± 0.000	**99.621 ± 0.645**	99.909 ± 0.528	99.706 ± 0.747
	STID	100.000 ± 0.000	**99.930 ± 0.403**	100.052 ± 0.725	101.107 ± 0.561
	iTransformer	100.000 ± 0.000	99.529 ± 1.010	101.786 ± 1.440	**99.191 ± 0.937**
ETTh1	DLinear	100.000 ± 0.000	**93.882 ± 1.117**	100.217 ± 2.627	103.532 ± 1.467
	PatchTST	100.000 ± 0.000	**93.941 ± 1.760**	97.904 ± 3.786	103.802 ± 2.241
	STID	100.000 ± 0.000	**94.329 ± 2.316**	96.907 ± 2.452	104.773 ± 3.074
	iTransformer	100.000 ± 0.000	**94.609 ± 3.747**	101.972 ± 2.361	100.015 ± 1.915
ETTh2	DLinear	100.000 ± 0.000	**90.585 ± 4.353**	96.672 ± 6.532	108.165 ± 4.919
	PatchTST	100.000 ± 0.000	**89.960 ± 5.742**	95.209 ± 6.740	101.587 ± 6.783
	STID	100.000 ± 0.000	**90.061 ± 8.417**	96.072 ± 10.507	114.089 ± 11.002
	iTransformer	100.000 ± 0.000	**87.844 ± 7.306**	96.447 ± 6.272	98.159 ± 6.797
ETTm1	DLinear	100.000 ± 0.000	**98.706 ± 0.386**	101.054 ± 0.517	107.807 ± 0.996
	PatchTST	100.000 ± 0.000	**98.022 ± 0.798**	100.490 ± 0.585	107.495 ± 1.430
	STID	100.000 ± 0.000	**99.387 ± 1.034**	99.775 ± 0.895	106.714 ± 1.131
	iTransformer	100.000 ± 0.000	**99.711 ± 1.738**	100.045 ± 0.976	105.391 ± 1.725
ETTm2	DLinear	100.000 ± 0.000	97.082 ± 1.334	**96.909 ± 2.993**	104.638 ± 3.534
	PatchTST	100.000 ± 0.000	**97.397 ± 3.871**	97.609 ± 4.328	104.101 ± 3.770
	STID	100.000 ± 0.000	98.647 ± 6.432	**97.387 ± 6.635**	117.781 ± 6.861
	iTransformer	100.000 ± 0.000	**97.261 ± 4.220**	98.092 ± 3.355	103.875 ± 2.709
PEMS03	DLinear	100.000 ± 0.000	100.037 ± 0.912	100.510 ± 0.849	**99.643 ± 0.777**
	PatchTST	100.000 ± 0.000	99.662 ± 0.570	**99.556 ± 1.168**	100.863 ± 0.497
	STID	100.000 ± 0.000	100.423 ± 1.242	**98.626 ± 2.103**	101.594 ± 1.351
	iTransformer	**100.000 ± 0.000**	100.903 ± 0.694	100.277 ± 0.999	101.996 ± 0.985
PEMS04	DLinear	100.000 ± 0.000	99.837 ± 0.914	100.123 ± 0.812	**99.563 ± 0.706**
	PatchTST	100.000 ± 0.000	100.451 ± 1.270	100.326 ± 1.124	**99.387 ± 0.636**
	STID	**100.000 ± 0.000**	101.733 ± 1.284	101.973 ± 0.535	100.924 ± 0.937
	iTransformer	100.000 ± 0.000	102.278 ± 2.408	**99.999 ± 0.548**	100.928 ± 0.172
PEMS07	DLinear	100.000 ± 0.000	99.841 ± 0.535	**99.752 ± 0.499**	100.035 ± 0.494
	PatchTST	100.000 ± 0.000	100.712 ± 0.729	**99.676 ± 0.952**	99.909 ± 0.779
	STID	**100.000 ± 0.000**	100.687 ± 0.589	101.156 ± 0.574	104.571 ± 0.429
	iTransformer	**100.000 ± 0.000**	100.174 ± 1.200	100.401 ± 1.450	101.642 ± 1.067
PEMS08	DLinear	100.000 ± 0.000	100.473 ± 0.283	**99.532 ± 0.316**	99.953 ± 0.332
	PatchTST	**100.000 ± 0.000**	102.283 ± 0.634	101.235 ± 0.538	100.756 ± 0.789
	STID	**100.000 ± 0.000**	101.773 ± 3.742	118.789 ± 4.464	105.775 ± 2.173
	iTransformer	**100.000 ± 0.000**	105.408 ± 4.801	104.543 ± 1.506	104.669 ± 1.770
Traffic	DLinear	100.000 ± 0.000	**99.878 ± 0.616**	99.982 ± 0.895	104.544 ± 0.729
	PatchTST	100.000 ± 0.000	**99.003 ± 1.888**	99.865 ± 1.782	102.467 ± 1.715
	STID	100.000 ± 0.000	**99.923 ± 0.733**	100.021 ± 0.707	105.954 ± 0.978
	iTransformer	**100.000 ± 0.000**	100.055 ± 0.219	100.908 ± 0.372	103.806 ± 0.236
Weather	DLinear	100.000 ± 0.000	**94.494 ± 4.092**	95.663 ± 5.592	98.180 ± 5.000
	PatchTST	100.000 ± 0.000	96.950 ± 7.674	96.762 ± 7.500	**96.070 ± 7.581**
	STID	100.000 ± 0.000	95.241 ± 6.033	**94.528 ± 6.422**	95.526 ± 5.964
	iTransformer	100.000 ± 0.000	**94.649 ± 10.527**	95.525 ± 9.451	95.868 ± 8.140

Table 5: **Model-specific overall MSE relative to paired Random.** Entries are mean ± sample SD of same-seed ratios (%) over five paired seeds. Lower is better; the lowest unrounded mean in each dataset–model row is bold. Random is 100 by construction. Each model uses the frozen dataset-specific nominal validation size. Each dataset spans four model rows.

Dataset	Model	Random	DRTV	Uniform	Tail holdout
Electricity	DLinear	100.000 ± 0.000	**99.145 ± 1.219**	100.187 ± 1.252	99.461 ± 1.238
	PatchTST	100.000 ± 0.000	**99.012 ± 1.039**	99.951 ± 1.232	99.511 ± 1.213
	STID	100.000 ± 0.000	**99.478 ± 0.868**	100.007 ± 1.286	102.115 ± 0.953
	iTransformer	100.000 ± 0.000	99.086 ± 1.640	102.342 ± 1.758	**98.890 ± 1.306**
ETTh1	DLinear	100.000 ± 0.000	**87.739 ± 2.432**	95.650 ± 2.961	110.068 ± 3.096
	PatchTST	100.000 ± 0.000	**87.382 ± 3.786**	94.660 ± 6.101	110.285 ± 5.514
	STID	100.000 ± 0.000	**88.244 ± 3.848**	91.601 ± 4.380	111.742 ± 5.468
	iTransformer	100.000 ± 0.000	**88.105 ± 7.776**	103.327 ± 4.732	103.487 ± 3.427
ETTh2	DLinear	100.000 ± 0.000	**82.052 ± 9.688**	90.402 ± 12.473	114.459 ± 13.135
	PatchTST	100.000 ± 0.000	**81.519 ± 11.922**	89.571 ± 13.216	105.813 ± 15.671
	STID	100.000 ± 0.000	**79.858 ± 14.336**	89.217 ± 17.953	121.856 ± 22.893
	iTransformer	100.000 ± 0.000	**76.630 ± 14.020**	92.635 ± 12.515	95.788 ± 13.074
ETTm1	DLinear	100.000 ± 0.000	**95.976 ± 0.882**	101.433 ± 0.860	117.388 ± 1.152
	PatchTST	100.000 ± 0.000	**96.271 ± 0.681**	100.882 ± 0.829	116.081 ± 1.547
	STID	100.000 ± 0.000	**97.439 ± 1.349**	99.329 ± 1.629	116.280 ± 1.400
	iTransformer	100.000 ± 0.000	100.168 ± 5.771	**98.443 ± 2.286**	112.133 ± 6.017
ETTm2	DLinear	100.000 ± 0.000	**94.158 ± 5.529**	95.286 ± 6.466	112.259 ± 7.646
	PatchTST	100.000 ± 0.000	**94.724 ± 7.739**	95.030 ± 9.608	110.157 ± 9.027
	STID	100.000 ± 0.000	97.252 ± 11.408	**96.304 ± 12.340**	134.073 ± 14.839
	iTransformer	100.000 ± 0.000	**93.809 ± 10.207**	95.082 ± 7.187	108.618 ± 6.234
PEMS03	DLinear	100.000 ± 0.000	99.425 ± 0.979	99.992 ± 0.984	**99.355 ± 0.855**
	PatchTST	100.000 ± 0.000	99.410 ± 1.072	**99.203 ± 1.669**	101.461 ± 0.753
	STID	100.000 ± 0.000	102.095 ± 2.518	**96.784 ± 3.900**	100.146 ± 3.091
	iTransformer	100.000 ± 0.000	100.633 ± 2.294	**98.315 ± 2.741**	101.760 ± 1.234
PEMS04	DLinear	100.000 ± 0.000	99.985 ± 0.361	99.840 ± 0.359	**99.614 ± 0.348**
	PatchTST	100.000 ± 0.000	100.379 ± 1.682	100.471 ± 1.566	**98.922 ± 1.014**
	STID	100.000 ± 0.000	**99.023 ± 2.687**	105.283 ± 2.193	100.308 ± 2.512
	iTransformer	**100.000 ± 0.000**	102.659 ± 3.308	100.818 ± 1.406	102.066 ± 0.793
PEMS07	DLinear	100.000 ± 0.000	99.924 ± 0.187	99.927 ± 0.221	**99.757 ± 0.132**
	PatchTST	100.000 ± 0.000	100.985 ± 1.014	**99.523 ± 0.978**	99.820 ± 0.696
	STID	**100.000 ± 0.000**	101.194 ± 1.216	101.378 ± 1.249	111.571 ± 1.353
	iTransformer	**100.000 ± 0.000**	101.361 ± 1.415	100.013 ± 1.601	101.305 ± 1.333
PEMS08	DLinear	100.000 ± 0.000	100.079 ± 0.083	100.210 ± 0.243	**99.842 ± 0.185**
	PatchTST	**100.000 ± 0.000**	103.707 ± 1.189	101.650 ± 0.984	102.489 ± 0.944
	STID	**100.000 ± 0.000**	101.636 ± 5.297	140.976 ± 8.709	110.642 ± 4.120
	iTransformer	**100.000 ± 0.000**	105.889 ± 6.744	105.657 ± 2.104	107.339 ± 3.040
Traffic	DLinear	100.000 ± 0.000	**99.082 ± 1.771**	100.074 ± 1.896	126.867 ± 2.350
	PatchTST	100.000 ± 0.000	**98.486 ± 2.827**	100.084 ± 2.602	126.883 ± 3.466
	STID	100.000 ± 0.000	**99.191 ± 2.213**	99.441 ± 2.103	142.293 ± 3.733
	iTransformer	100.000 ± 0.000	**99.391 ± 1.373**	100.745 ± 1.698	125.606 ± 1.847
Weather	DLinear	100.000 ± 0.000	**94.738 ± 6.760**	95.542 ± 7.458	99.894 ± 7.663
	PatchTST	100.000 ± 0.000	96.324 ± 9.835	**95.740 ± 10.176**	98.857 ± 10.588
	STID	100.000 ± 0.000	93.245 ± 8.758	**92.211 ± 8.674**	96.915 ± 8.942
	iTransformer	100.000 ± 0.000	**90.729 ± 10.130**	95.546 ± 10.427	93.506 ± 8.859

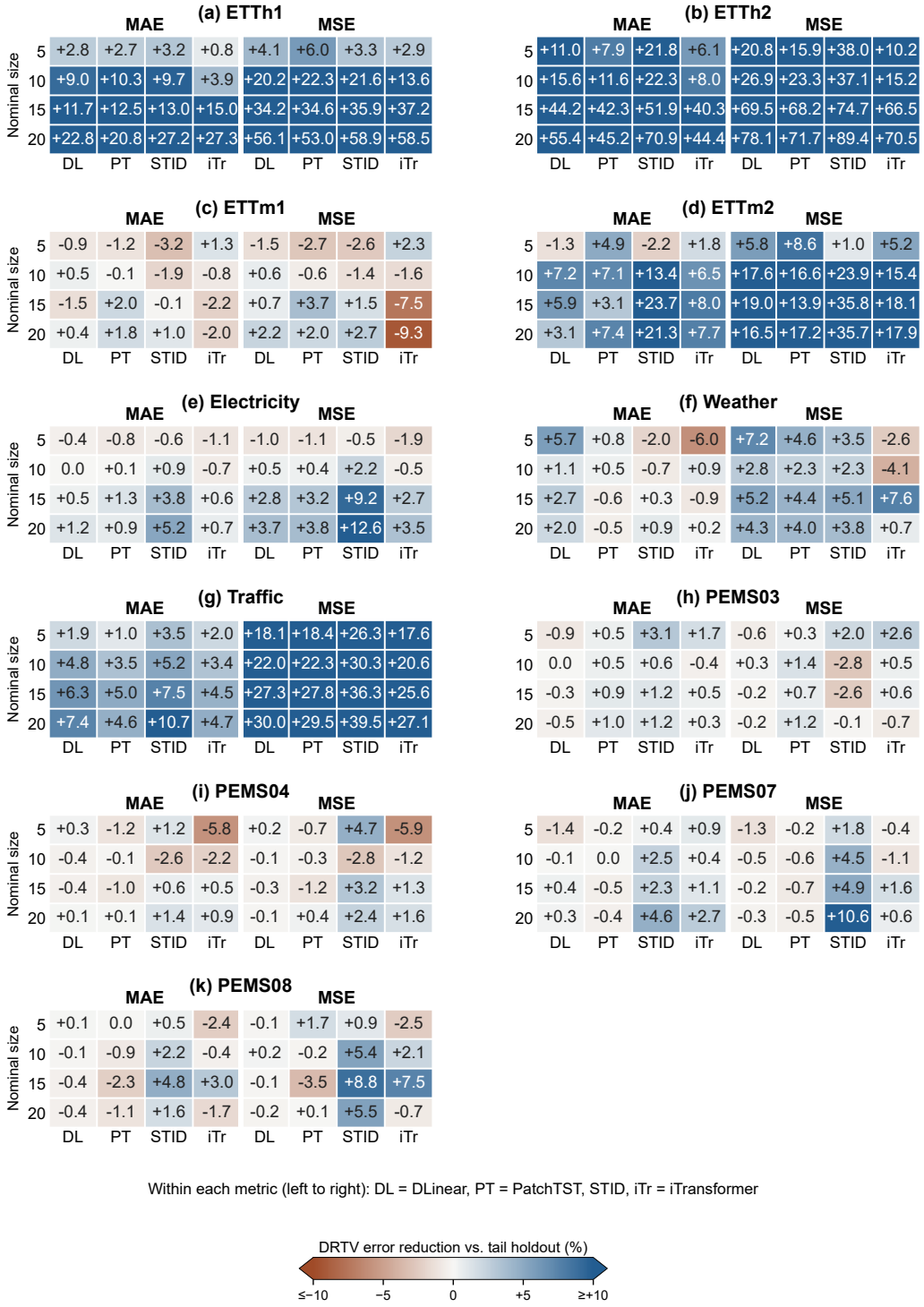


Figure 7: **Model-level DRTV effects vary across the 11 datasets.** Each cell is $100(e_{\text{tail}} - e_{\text{DRTV}})/e_{\text{tail}}$ for the same model, metric, nominal size, and seed 1. Positive blue values favor DRTV; negative brown values favor tail holdout. Within each metric, columns are DLinear (DL), PatchTST (PT), STID, and iTransformer (iTr), left to right; rows are nominal sizes 5, 10, 15, and 20. Annotations are rounded to one decimal place. The shared color scale saturates beyond $\pm 10\%$ without truncating the annotated values. These are descriptive single-seed comparisons at equal nominal size, not equal realized validation-window counts.