What Drives Online Adaptation of Frozen Forecasters? A Controlled Study of Drought-Index Nowcasting

Anonymous authors
Paper under double-blind review

Abstract

Drought estimates depend on both recent weather and earlier dry or wet conditions. An online adapter can use these observations to improve a pretrained forecast, but that improvement does not reveal how much the forecast itself contributes. We study this question through retrospective drought-index nowcasting. Our evaluation compares three ways to use the same local observations: predict the index directly, correct a frozen forecast, or learn how much weight to give that forecast. A separate comparison holds the local features and update rules fixed, so forecast weighting can be examined without changing feature selection. Direct prediction explains much of the gain from fixed residual correction, while learning the forecast's weight improves further in both comparisons. We distinguish decisions made at each monthly step from decisions about where to apply adaptation. Historical errors help prioritize cases under a limited adaptation budget, but validation favors adapting all evaluated cases; reversion rarely changes predictions in the natural replay. These findings identify when a frozen forecast adds value beyond local learning in this retrospective setting.[1]

1 Introduction

A month with little rainfall can follow a wet season or extend an existing dry spell. Updating a drought estimate therefore requires both the recent weather and the conditions that preceded it. The standardized precipitation evapotranspiration index (SPEI) captures this history through water balance accumulated over different periods (Vicente-Serrano et al., 2010). Short windows emphasize recent changes; longer windows retain the influence of earlier dry and wet periods.

Time-series foundation models offer forecasts based on patterns learned across time series, with no need to retrain their weights for each task (Ansari et al., 2024; Das et al., 2024; Woo et al., 2024). Online adapters can revise these forecasts as local observations arrive (Lee et al., 2025; Dai et al., 2026). For drought estimation, those observations include weather that is directly related to the index. Improvement may come from this local information or from its combination with the pretrained forecast. The central question is: *what does the pretrained forecast add beyond what an online predictor can learn from the same local observations?*

Figure 1 shows why improvement alone is insufficient. Both panels have identical targets, frozen forecasts, and adapted forecasts. Local data explain the improvement in (a); the frozen forecast contributes complementary variation in (b). A predictor using the same local inputs without the forecast distinguishes these cases.

A second question is how the frozen forecast should enter the estimate. Residual correction fixes its weight at one; a learned combination adjusts that weight (Bates & Granger, 1969; Wolpert, 1992; Timmermann, 2006). We hold local features and update rules fixed to separate forecast weighting from feature search and predictor selection.

[1]Code and reproduction instructions: `https://anonymous.4open.science/r/A-Controlled-Study-of-Drought-Index-Nowcasting-F362`.

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.

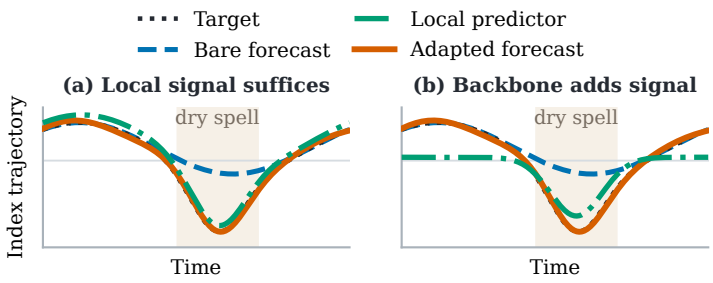


Figure 1: **The same improvement can reflect different uses of a pretrained forecast.** The target and both forecast trajectories are identical in the schematic panels. Local information suffices in (a). In (b), the local predictor captures the dry spell and the backbone supplies complementary background variation.

Historical evidence also guides deployment. Selecting a prediction or returning to the original forecast at each step differs from ranking cases under an adaptation budget or deciding whether to adapt at all. We evaluate these decisions separately.

We study these questions through monthly nowcasting of the retrospective SPEI-GD product (Liu, 2023), covering twelve regions, three accumulation scales, and twelve frozen forecasting configurations. The replay provides past index values and completed-month weather and climate inputs. RAIL (Residual-Adaptive Intervention Layer) is the reference system: it learns corrections, selects predictors from past errors, and can return to the original forecast. Our main comparisons predict before revealing the target and updating the adapter. The study makes three contributions:

- **A reference for what local learning can explain.** We compare adapters with the same input types and introduce online predictors that use no pretrained forecast. Further controls vary input access, delay weather and climate data, and remove calendar-month means to examine input timing and seasonal structure (sections 5.2 and 5.3).
- **Controlled tests of what the pretrained forecast adds.** Building on that reference, we compare direct prediction, residual correction, and learned forecast combinations. A separate experiment holds local features and updates fixed, with no selection or gates, to examine forecast weighting independently of those design choices (sections 5.3 and P).
- **An assessment of how historical evidence guides adaptation.** We compare historical scores for allocating an adaptation budget and evaluate prediction selection and reversion. A separate threshold test uses the predictability of past forecast errors to decide whether to adapt. We also bound excess loss over each consecutive run of adapted predictions for finite-window reversion, assuming bounded targets and clipped predictions (sections 4.3, 5.6 and 5.7).

2 Related Work

Post-processing and forecast combination. Model-output statistics and statistical calibration learn from past forecast errors (Glahn & Lowry, 1972; Wilks, 2011; Vannitsem et al., 2018). Forecast pooling, Bayesian model averaging, and stacking study how to weight multiple predictions (Bates & Granger, 1969; Raftery et al., 2005; Timmermann, 2006; Wolpert, 1992). These traditions motivate treating the frozen forecast's weight as an explicit experimental variable.

Online learning and deployment decisions. Prequential evaluation predicts before each target is revealed (Dawid, 1984). Expert-advice methods combine candidates using past losses (Freund & Schapire, 1997; Cesa-Bianchi & Lugosi, 2006); streaming learning studies changing predictor quality (Gama et al., 2014). We distinguish per-step prediction selection and reversion from case-level prioritization and adaptation decisions.

Adaptation around frozen foundation models. Test-time methods such as TTT and Tent update model parameters using test inputs (Sun et al., 2020; Wang et al., 2021); our backbone is fixed and only its external wrapper learns from observed targets. The δ-Adapter also keeps the forecaster fixed, learning bounded input edits or output residual corrections (Liang et al., 2026). Closer black-box online precedents are ELF, which combines a frozen forecast with an online forecaster (Lee et al., 2025), and ORCA, which learns a context-conditioned residual and routes between base and adapted predictions using past errors (Dai et al., 2026). We compare reimplementations with a history-only RAIL control on matched dates, then expand RAIL's inputs and recompute its FIT choices.

Forecasting models. The frozen forecasts come from several time-series foundation model families (Ansari et al., 2024; 2025; Das et al., 2024; Woo et al., 2024; Liu et al., 2025; Ekambaram et al., 2024; Cohen et al., 2024; Khwaja et al., 2026; Auer et al., 2025). Locally fitted deep forecasters provide further references (Zeng et al., 2023; Nie et al., 2023; Liu et al., 2024b; Wu et al., 2023). Varying backbones under a common protocol tests how the findings depend on the forecasting architecture.

The attribution question is how local inputs, forecast weighting, and online selection contribute to adaptation gains. Matched-input controls and a fixed-design Direct–Residual–Stack comparison distinguish the frozen forecast's added value from complete-system accuracy.

3 Problem Setting

We estimate monthly samples of the retrospective SPEI-GD product (Liu, 2023; Liu et al., 2024a) from index history and monthly weather observations. SPEI-GD is constructed from daily ERA5 precipitation and Singer's potential evapotranspiration. Our predictors instead use coarser monthly NCEP/NCAR fields and climate indices. They therefore learn a mapping to the released index from a different observation set; they do not receive the daily inputs needed to recompute that product.

Within each region, $e \in \{1, \dots, E\}$ indexes sampled land grid points (called entities below), and t indexes calendar months. The target $y_{e,t}$ is the product's value on the first day of month t at an accumulation scale of $s \in \{30, 90, 180\}$ days; we suppress s in the notation. A frozen time-series foundation model (TSFM) supplies a base forecast $b_{e,t}$ for $y_{e,t+1}$. The main task has a one-month index step ($H = 1$). A *scenario* is one region and scale. An *evaluation case* adds one frozen backbone to that scenario; it is the unit of our model comparisons.

Issue and update order. In the historical replay, at origin t, target history through $y_{e,t}$ and completed month-t environmental covariates $z_{e,t}$ are available, together with the frozen forecasts. Let $\mathbf{y}_t$, $\mathbf{z}_t$, and $\mathbf{b}_t$ collect these quantities across the E entities. The wrapper π_t determines all predictions without using any $y_{e,t+1}$:

$$\mathcal{I}_t = (\mathbf{y}_{\le t}, \mathbf{z}_{\le t}, \mathbf{b}_t) \xrightarrow{\pi_t} \tilde{\mathbf{y}}_t \xrightarrow{\text{reveal}} \mathbf{y}_{t+1} \xrightarrow{\text{update}} \pi_{t+1}. \tag{1}$$

Here $\tilde{y}_{e,t}$ estimates $y_{e,t+1}$, whose date is the first day of month $t{+}1$. Online state may pool entities, but every entity's forecast at an origin precedes that origin's target reveal. The backbone weights remain fixed; the external wrapper may update its own state only after the reveal.

Completed month-t weather contributes to the accumulation window of $y_{e,t+1}$, making the effective lead short. Adjacent 180-day targets also share roughly five sixths of their windows. Thus the $H = 1$ experiment measures sequential nowcasting of a fixed historical index. Every method uses the same archive-standardized target. The reveal in eq. (1) is a historical replay step; actual input and target release dates would determine the operational update schedule (section B).

Feature choices and aggregation settings for RAIL and the matched controls use an initial fitting segment (FIT); the separate diagnostic threshold uses validation data (section J). Test evaluation is one continuous forward pass with updates only from revealed targets. Within each case, mean squared error (MSE) pools valid entity-origin errors; headline summaries then average across cases. A case's relative gain is $100[1 - \mathrm{MSE}(\mathrm{wrapper})/\mathrm{MSE}(\mathrm{bare})]$; reported gains average these percentages. Each wrapper is compared with its own bare backbone, which also serves as the final fallback for RAIL and the gated stack.

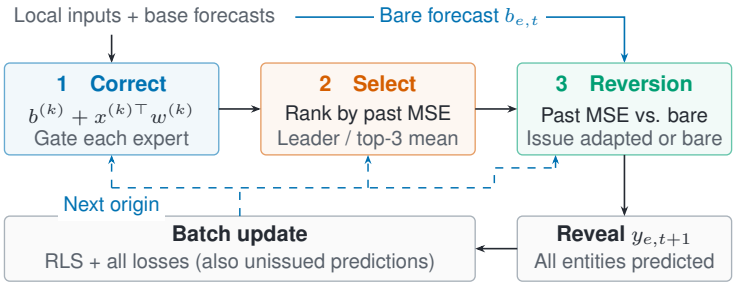


Figure 2: **RAIL at one prediction step.** Each expert can return to its own base. Past errors select an aggregate, and the final gate can return to the wrapped backbone. All entities predict before the target reveal; dashed paths update state for the next step.

4 Reference System and Controlled Comparisons

Our contribution is a controlled evaluation of information access, forecast weighting, and output selection. We first define the comparisons, then specify RAIL as their residual-correction reference system.

4.1 Three Ways to Use Local Observations

Let $x_{e,t}$ contain features of target history, weather, climate indices, and the calendar. With frozen forecast $b_{e,t}$, the three forms are

$$\underbrace{x_{e,t}^\top \beta_t}_{\text{Direct}}, \qquad \underbrace{b_{e,t} + x_{e,t}^\top \beta_t}_{\text{fixed residual}}, \qquad \underbrace{\alpha_t b_{e,t} + x_{e,t}^\top \beta_t}_{\text{Stack}}. \tag{2}$$

Direct measures what local observations can explain. Fixed residual correction retains the forecast with weight one. Stack learns that weight along with the local coefficients. Here $x_{e,t}$ excludes the frozen forecast; allowing the residual predictor to use $b_{e,t}$ would also let it change that weight. The fixed-design experiment holds local features and updates constant to isolate this coefficient constraint (section P).

Complete-system comparisons also include feature search and candidate pools. Direct-1 uses one regressor; Direct pool uses a pool. RAIL-Single uses one base forecast; RAIL-Pool also admits an ensemble base. Both pool residual experts. An optional identity candidate emits the bare forecast. Applicable FIT choices are recomputed for each system (sections 5.3 and L).

4.2 Reference Implementation: RAIL

RAIL follows the cycle in fig. 2. A candidate regressor, called an expert, uses feature subset S_k and predicts

$$\hat{y}_{e,t}^{(k)} = b_{e,t}^{(k)} + x_{e,t}^{(k)\top} w_t^{(k)}, \qquad x_{e,t}^{(k)} = \phi_{S_k}(y_{e,\le t}, z_{e,\le t}). \tag{3}$$

For eq. (3), recursive least squares (RLS) (Haykin, 2014) learns the shared $w_t^{(k)}$ from observed residuals $y_{e,t+1} - b_{e,t}^{(k)}$. RLS provides a shared sequential linear learner for all three forms in eq. (2), keeping the update mechanism comparable. All entities predict before a batch update. RAIL-Single uses the wrapped backbone as every expert's base; RAIL-Pool also offers the per-entity median of eleven frozen univariate backbones.

Expert gates retain their own bases during warm-up or when correction has not improved past loss. Selection compares gated outputs by past MSE, using either the leader or the top-three mean. This provides a simple choice between one candidate and a small ensemble; FIT chooses between them. The final *reversion gate* issues the aggregate only when its past MSE beats the wrapped backbone's. All losses are recorded after reveal, including unissued forecasts; decisions at t use losses only through $t-1$.

Feature subsets vary the local information used by each expert. FIT-inner orders features through regularization paths, forward selection, and bootstrap stability (Efron et al., 2004; Meinshausen & Bühlmann, 2010); prefixes give candidates of different sizes. FIT-late selects the loss-history window (including cumulative history), leader versus top-three aggregation, and expert-gate variant. These are reference-system choices held fixed during validation and test; RLS and loss histories continue updating. The fixed-design comparison removes feature search, selection, and gates to isolate forecast weighting. Section A and algorithm 1 give the full specification.

4.3 What the reversion gate bounds

The gate bounds excess loss within a consecutive adapted run; it does not guarantee improvement at each step. Let $\ell_t(p)$ be MSE over entities at origin t. Record the aggregate's excess loss at every origin, including when it is not issued: $d_t = \ell_t(\text{aggregate}) - \ell_t(\text{bare})$. Assume $|y| \le Y$ and clip forecasts to $[-5, 5]$. Losses then lie in $[0, B]$, where $B = (5 + Y)^2$, and $|d_t| \le B$. With a complete finite window W, the gate is $g_t = \mathbf{1}\{\sum_{s=t-W}^{t-1} d_s < 0\}$, with issued excess $u_t = g_t d_t$.

Proposition 1 (Excess loss within a deployed run)**.** *For any maximal deployed run $[a, b]$ with $g_t = 1$, $\sum_{t=a}^{b} u_t \le WB$. Following a persistent break with $d_t \ge \delta > 0$, the gate reverts within at most W origins.*

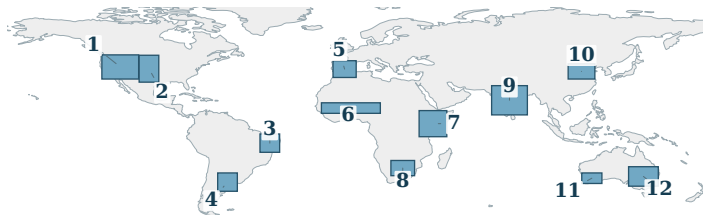
The proof groups errors into negative-sum windows (section I). The bound restarts with each run and requires Y to bound future targets; large cumulative-history windows make it uninformative.

5 Experiments

5.1 Setup

All main comparisons follow the nowcasting timeline in section 3.

The twelve regions in fig. 3 and three accumulation scales form 36 scenarios. Each region contains $E = 150$ sampled land grid points with $T = 480$ monthly targets and covariates. Targets are first-of-month samples from the daily $0.25°$ SPEI-GD archive (1982–2021) (Liu, 2023; Liu et al., 2024a); weather comes from NCEP/NCAR reanalysis (Kalnay et al., 1996), and climate indices include Niño-1+2/3.4/4 and the Dipole Mode Index (Trenberth, 1997; Saji et al., 1999). All covariates are completed month-t means. Sections A and B detail construction and availability; section D describes an exploratory longer-horizon pilot.



1 US West	5 Iberia-Maghreb	9 India
2 US Plains	6 Sahel	10 N China
3 NE Brazil	7 Horn of Africa	11 SW Australia
4 Pampas	8 S Africa	12 E Australia

Figure 3: **Benchmark regions.** Each region has 150 sampled land grid points at three SPEI accumulation scales.

Twelve frozen forecasting configurations serve as backbones: Chronos-T5 (small, base) (Ansari et al., 2024), Chronos-Bolt (tiny, small, base) and Chronos-2 (Ansari et al., 2025), TimesFM-2.5 (Das et al., 2024), Moirai-2 (Liu et al., 2025), TiRex (Auer et al., 2025), Toto 2.0 (Khwaja et al., 2026), and TTM (Ekambaram et al., 2024), plus one covariate-input Chronos-2 variant (Chronos-2r; FIT-selected inputs). Each backbone is evaluated on the 36 scenarios, giving $12 \times 36 = 432$ cases. Each wrapper

is scored against *its own* bare backbone; causal rolling backbone forecasts are cached for wrapper replay.

Every case uses FIT $[0, 288)$, validation $[288, 328)$, and test $[336, 479)$; the gap $[328, 336)$ updates state but is not scored. RAIL and matched controls select meta-parameters on FIT; deep baselines use validation early stopping, and the diagnostic threshold uses validation losses. A single forward test pass gives MSE and MAE pooled over entity–origin pairs within each case; headline means weight cases equally. Per-case tests use a paired entity/temporal cluster-block bootstrap (Lahiri, 2003) (block length 8; 4,000 resamples), with a spatial-block check in section 5.8.

Baselines for each scenario include persistence and seasonal naive; five classic combination rules over the 11-backbone univariate pool; locally trained ridge and histogram gradient boosting (HGB) on the core 52-feature pool; and four locally trained deep forecasters (DLinear, PatchTST, iTransformer, TimesNet) with 150 entities as channels and validation early stopping.

5.2 Additional information versus adapter design

We reimplemented ELF (Lee et al., 2025) and ORCA (Dai et al., 2026) from released code on matched prediction dates (section A and table 13). With only target history, RAIL-Single's modest gain is not statistically distinguishable from ORCA's; ELF has higher error than the bare forecast. Adding weather, climate, and calendar inputs and recomputing FIT choices yields RAIL-Single's larger advantage. The following controls test how much those inputs explain without the frozen forecast.

5.3 Matched controls: what does the backbone contribute?

We evaluate the Direct, residual, and Stack forms in eq. (2). The complete systems use the same split and forward test pass, with applicable FIT choices recomputed. The final contrast holds local features and updates fixed to isolate forecast weighting (sections L and P). Table 1 reports the results. Paired uncertainty estimates appear in table 16; they refer to absolute MSE differences.

Table 1: **Online controls and simple references** (432 cases). Gain averages per-case percentage reductions against each case's bare TSFM, so it need not equal the reduction in mean MSE (bare mean 0.241). Seasonal persistence uses another base; its own-base gain is reported in section L. Climatology is a FIT-only calendar-month mean scored on valid targets; online controls also require valid inputs.

Policy	MSE	Gain vs. bare TSFM
FIT climatology (fixed)	0.468	-173.1%
Seasonal-persistence base + same machinery	0.577	-221.1%
Direct-1: one online RLS, no backbone	0.157	$+28.2\%$
Direct pool: no backbone, pool + selection	0.153	$+30.1\%$
RAIL-Single (fixed additive residual)	0.161	$+27.7\%$
RAIL-Pool	0.153	$+32.5\%$
Stack: backbone as free coefficient	0.141	$+37.0\%$
Stack + live reversion gate	0.141	$+37.0\%$
+ explicit identity candidate	0.141	$+37.0\%$

How much can local information explain?

A single Direct regressor reaches a similar mean error to RAIL-Single; the two pooled systems are also close. Neither paired comparison detects a difference in mean MSE. Local inputs and online fitting therefore account for much of the average improvement obtained by fixed residual correction. This pattern varies with backbone quality: RAIL-Single beats Direct pool whenever the bare backbone itself is better than Direct pool, while Direct pool usually wins against the weaker TTM and T5-small backbones. The value of retaining a forecast thus depends on what it already contributes beyond the local inputs.

Do simpler bases explain the gain?

Correction improves seasonal persistence over its own base, but remains less accurate than RAIL-Single in every case. Calendar-month climatology also performs poorly. The gains therefore exceed a recurring calendar level, and relative improvement over a weak base does not establish competitive final accuracy.

Does learning the backbone weight help?

Stack learns the forecast's weight and improves on both Direct pool and RAIL-Single. Its mean MSE is 8.1% lower than Direct pool's, showing that the frozen forecast can supply useful information beyond local learning. Both paired confidence intervals favor Stack. Adding reversion and the bare-forecast candidate leaves its average error essentially unchanged. These are complete-system comparisons with reselected features and settings; the next test isolates forecast weighting from those choices.

Does the coefficient result survive a fixed design?

The stack retains its advantage when local features, FIT scaling, and online updates are held fixed, with no pools, gates, or meta-selection (table 23 and section P). Learning the forecast's weight therefore helps beyond the effects of feature search and selection in the complete systems.

The input controls clarify what supports local learning. Delaying weather and climate data by one month increases Direct's error across every scenario, showing the importance of recent environmental information under the current feedback schedule. Direct also beats the calendar-mean reference after calendar means are removed from the target and its history. Its predictive value extends beyond that seasonal level, although this comparison does not separate weather information from target history.

5.4 Accuracy across scales and local baselines

Table 2 compares test MSE at each accumulation scale. Within a region–scale scenario, wrapped-system results average the twelve backbone configurations; the local baselines have one result per scenario. The scale columns then average twelve regions. Full scenario-level results appear in tables 5 to 7.

Table 2: **Test MSE by SPEI accumulation scale**, averaged over twelve regions. FM-Avg averages twelve bare backbones on the RAIL-Pool scoring masks; RAIL, Direct pool, and Stack average their corresponding wrapped configurations. Mean-Ens and FIT-selected Best-FM use the 11-backbone univariate pool; the other baselines are local. **Bold** marks the column minimum.

Method	30d	90d	180d	All
Persistence	0.993	0.438	0.284	0.571
S-Naive	0.845	0.674	0.791	0.770
DLinear	0.557	0.196	0.136	0.296
PatchTST	0.625	0.169	0.117	0.304
iTransformer	0.603	0.207	0.138	0.316
TimesNet	0.646	0.391	0.417	0.485
Mean-Ens	0.430	0.154	0.091	0.225
Best-FM	0.435	0.144	0.091	0.223
Ridge	0.308	0.117	0.090	0.172
HGB	0.300	0.112	0.085	0.166
FM-Avg	0.446	0.170	0.106	0.241
Direct pool	0.273	0.104	0.082	0.153
RAIL-Single	0.286	0.116	0.083	0.161
RAIL-Pool	0.277	0.108	0.075	0.153
Stack + gate + identity	**0.251**	**0.096**	**0.074**	**0.141**

The gated stack with an identity candidate has the lowest mean error at every scale and leads on most scenarios; Direct pool stays close to RAIL-Pool. HGB is the strongest locally trained baseline, and the deep models lead on no scenarios. These comparisons concern local training on short regional series with entities treated as channels.

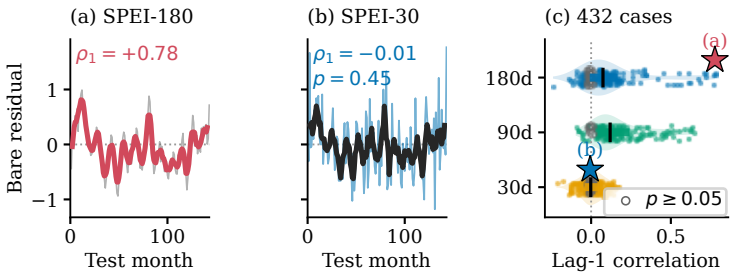


Figure 4: **Residual memory varies across accumulation scales.** Entity-mean test residuals of the same TTM backbone on E. Australia have long memory at SPEI-180 (a) and near-zero lag-1 memory at SPEI-30 (b). Panel (c) shows lag-1 correlations across the 432 cases; open circles mark $p \geq 0.05$. The panels describe scale-dependent variation in the evaluated residuals.

5.5 Backbone heterogeneity and drought events

RAIL-Pool improves on its own bare backbone in every evaluated case (table 10 and fig. 8). Simply replacing each backbone by the ensemble median harms some cases, so access to an ensemble alone does not ensure improvement. Relative gains are largest for weaker backbones and remain substantial for stronger ones. The spatial and temporal resampling check also supports improvements across most cases after correction for multiple comparisons (section A).

RAIL-Pool reduces error at all evaluated drought thresholds, increasingly for severe events. At the severe threshold, recall rises but precision falls: lower index error does not improve every detection metric. Section M reports these descriptive results, which pool repeated predictions across backbones.

5.6 Ablations: candidates, selection, and gates

Ensemble-base experts and expanded own-base features each improve gains; combining them yields the strongest residual system (table 12). These branches change candidate access and FIT choices together, so their gains need not add independently.

Selection and reversion. Table 14 changes selection or gating while keeping the expert updates fixed. The full pool outperforms a single FIT-selected expert, a contrast that changes both candidate access and selection. Replacing the selector with fixed-rate exponential weighting lowers gain. After validation tuning, however, exponential weighting has similar gain to the reference, with no statistically detected gap (section O). Its use of later tuning data makes this a sensitivity check; the evidence does not establish a general ranking of selection rules.

Removing the final gate, or all gates, barely changes average gain; the ungated pool also improves on every bare backbone. The observed gains therefore do not depend on gate interventions. Under an artificial change in residual feedback, gating reduces excess loss but reacts slowly, with a median delay of 15 months among cases that revert (section E). This test keeps cached backbone forecasts and some history features fixed. It illustrates the gate's delayed response under that specific degradation process.

5.7 Can historical errors guide where to adapt?

We distinguish budgeted prioritization from deciding whether to adapt. Scale-dependent residual memory (fig. 4) motivates the analysis. Those test patterns are descriptive; the residual-predictability index uses only FIT, while comparison scores may use validation.

Ranking future gains. We measure how well a ridge model predicts past backbone errors from local features. Its held-out improvement over zero correction defines the *residual-predictability index*; fitting and scoring use separate parts of FIT (section N).

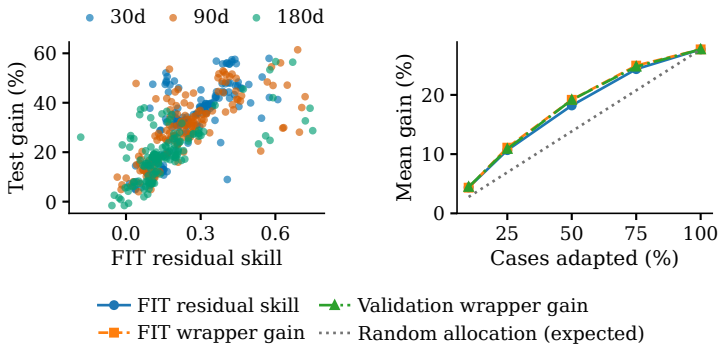


Figure 5: **Historical skill and future correction gain.** Left: FIT-only ridge residual skill versus RAIL-Single test gain across 432 cases. Right: the mean test relative gain contributed across all cases when adapting only a fixed fraction, ranked using historical scores. Non-selected cases retain their bare forecasts. Budget fractions are fixed independently of test outcomes; gray denotes expected gain from random allocation.

Higher residual skill is associated with larger test gains (fig. 5), including after adjustment for scale and bare error and for RAIL-Pool. Low-score cases can still benefit, especially with ensemble access: the score ranks gains but does not establish harm.

Other historical scores also rank future gains (table 20). Past wrapper gain correlates more strongly with test gain, although its FIT score reuses selection data and may be optimistic. The validation score uses later history. At the tested half-budget, residual skill and validation wrapper gain both improve on random allocation (fig. 5).

Choosing whether to adapt. Validation chooses adaptation everywhere, also when each region is held out (section J). Harmful RAIL-Single cases have small excess error, while low-score cases still gain: withholding adaptation loses more than it saves. Budget allocation and whether to adapt are distinct decisions.

5.8 Gate behavior and scope of the evidence

RAIL-Pool is issued at almost every natural test origin. Most configurations use finite windows covered by proposition 1; its loose bound limits excess loss during one consecutive adapted run, assuming a future target range (section H.1). It does not explain the observed gains: reversion rarely intervenes and clipping never binds.

Evidence is limited to one index product. Full-archive standardization and entity screening have unmeasured effects. Replay uses completed-month inputs and current target feedback; the delayed-input test changes only covariate timing. FIT-only target construction and screening, actual publication delays, and independent products remain necessary tests.

6 Conclusion

Local observations explain much of the gain from residual correction in retrospective drought-index nowcasting. Learning the frozen forecast's weight improves further under fixed local features and updates; Direct prediction reveals its added value. Historical scores prioritize limited adaptation budgets, but validation favors adapting all cases and reversion rarely intervenes. Actual release dates, delayed feedback, and independent drought products remain necessary tests of operational relevance.

Acknowledgements of AI assistance

AI assistants supported manuscript drafting, code review, analysis-script preparation, and submission checks, including bibliography searches and internal-consistency checks. The authors remain responsible for the accuracy of the methods, results, and claims.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=gerNCVqqtR.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025. doi: 10.48550/ARXIV.2510.15821. URL https://arxiv.org/abs/2510.15821.

Andreas Auer, Patrick Podest, Daniel Klotz, Sebastian Böck, Günter Klambauer, and Sepp Hochreiter. Tirex: Zero-shot forecasting across long and short horizons with enhanced in-context learning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 57529–57580. Curran Associates, Inc., 2025. doi: 10.52202/085713-1929. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/5356603f9c47399adfd372f77a677057-Paper-Conference.pdf.

J. M. Bates and C. W. J. Granger. The combination of forecasts. *Journal of the Operational Research Society*, 20(4):451–468, December 1969. ISSN 1476-9360. doi: 10.1057/jors.1969.103. URL http://dx.doi.org/10.1057/jors.1969.103.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Ben Cohen, Emaad Khwaja, Kan Wang, Charles Masson, Elise Ramé, Youssef Doubli, and Othmane Abou-Amal. Toto: Time series optimized transformer for observability. *arXiv preprint arXiv:2407.07874*, 2024. doi: 10.48550/ARXIV.2407.07874. URL https://arxiv.org/abs/2407.07874.

Xilin Dai, Yiding Liu, Hongjie Xia, Yifan Hu, Zewei Dong, Jiang-Ming Yang, and Qiang Xu. Learning the context of errors: Black-box online adaptation of time series foundation models. *arXiv preprint arXiv:2606.14222*, 2026. doi: 10.48550/arXiv.2606.14222. URL https://arxiv.org/abs/2606.14222.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10148–10167. PMLR, 2024. URL https://proceedings.mlr.press/v235/das24c.html.

A Philip Dawid. Present position and potential developments: Some personal views: Statistical theory: The prequential approach. *Journal of the Royal Statistical Society: Series A*, 147(2): 278–292, 1984. doi: 10.2307/2981683. URL https://academic.oup.com/jrsssa/article/147/2/278/7106293.

Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2):407–451, 2004.

Vijay Ekambaram, Arindam Jati, Pankaj Dayama, Sumanta Mukherjee, Nam H. Nguyen, Wesley M. Gifford, Chandra Reddy, and Jayant Kalagnanam. Tiny time mixers (ttms): Fast pre-trained models for enhanced zero/few-shot forecasting of multivariate time series. In *Advances in Neural Information Processing Systems*, volume 37, pp. 74147–74181. Curran Associates, Inc., 2024. doi: 10.52202/079017-2359. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/874a4d89f2d04b4bcf9a2c19545cf040-Paper-Conference.pdf.

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997. ISSN 0022-0000. doi: 10.1006/jcss.1997.1504. URL http://dx.doi.org/10.1006/jcss.1997.1504.

João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):1–37, 2014.

Harry R Glahn and Dale A Lowry. The use of model output statistics (MOS) in objective weather forecasting. *Journal of Applied Meteorology*, 11(8):1203–1211, 1972.

Simon S Haykin. *Adaptive Filter Theory*. Pearson, 5 edition, 2014.

Eugenia Kalnay, Masao Kanamitsu, Robert Kistler, William Collins, Dennis Deaven, Lev Gandin, Mark Iredell, Suranjana Saha, Glenn White, John Woollen, et al. The NCEP/NCAR 40-year reanalysis project. *Bulletin of the American Meteorological Society*, 77(3):437–472, 1996.

Emaad Khwaja, Chris Lettieri, Gerald Woo, Eden Belouadah, Marc Cenac, Guillaume Jarry, Enguerrand Paquin, Xunyi Zhao, Viktoriya Zhukov, Othmane Abou-Amal, Chenghao Liu, Ameet Talwalkar, and David Asker. Toto 2.0: Time series forecasting enters the scaling era. *arXiv preprint arXiv:2605.20119*, 2026. doi: 10.48550/arXiv.2605.20119. URL https://arxiv.org/abs/2605.20119.

Soumendra Nath Lahiri. *Resampling Methods for Dependent Data*. Springer, 2003.

Thomas L Lee, William Toner, Rajkarn Singh, Artjom Joosen, and Martin Asenov. Lightweight online adaption for time series foundation model forecasts. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 33736–33764. PMLR, 2025.

Daojun Liang, Qi Li, Yinglong Wang, Jing Chen, Hu Zhang, Xiaoxiao Cui, Qizheng Wang, and Shuo Li. The forecast after the forecast: A post-processing shift in time series. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/6f7d90b1198fec96defd80b5ebd5bc81-Abstract-Conference.html.

Chenghao Liu, Taha Aksu, Juncheng Liu, Xu Liu, Hanshu Yan, Quang Pham, Silvio Savarese, Doyen Sahoo, Caiming Xiong, and Junnan Li. Moirai 2.0: When less is more for time series forecasting. *arXiv preprint arXiv:2511.11698*, 2025. URL https://arxiv.org/abs/2511.11698.

Xuebang Liu. Spei-gd: The first global multi-scale daily spei dataset for 1983-2020. Zenodo, 2023. URL https://zenodo.org/records/8060268.

Xuebang Liu, Shuying Yu, Zhiwei Yang, Jianquan Dong, and Jian Peng. The first global multi-timescale daily spei dataset from 1982 to 2021. *Scientific Data*, 11(1):223, February 2024a. ISSN 2052-4463. doi: 10.1038/s41597-024-03047-z. URL https://www.nature.com/articles/s41597-024-03047-z.

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*, 2024b. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/2ea18fdc667e0ef2ad82b2b4d65147ad-Abstract-Conference.html.

Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B*, 72(4):417–473, 2010.

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. URL https://openreview.net/forum?id=Jbdc0vTOcoL.

Adrian E Raftery, Tilmann Gneiting, Fadoua Balabdaoui, and Michael Polakowski. Using Bayesian model averaging to calibrate forecast ensembles. *Monthly Weather Review*, 133(5):1155–1174, 2005.

N H Saji, B N Goswami, P N Vinayachandran, and Toshio Yamagata. A dipole mode in the tropical Indian Ocean. *Nature*, 401(6751):360–363, 1999.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248. PMLR, 2020. URL https://proceedings.mlr.press/v119/sun20b.html.

Allan Timmermann. Forecast combinations. In *Handbook of Economic Forecasting*, volume 1, pp. 135–196. Elsevier, 2006.

Kevin E Trenberth. The definition of El Niño. *Bulletin of the American Meteorological Society*, 78 (12):2771–2778, 1997.

Stéphane Vannitsem, Daniel S Wilks, and Jakob W Messner (eds.). *Statistical Post-processing of Ensemble Forecasts*. Elsevier, 2018. doi: 10.1016/C2016-0-03244-8. URL https://www.sciencedirect.com/book/9780128123720/statistical-postprocessing-of-ensemble-forecasts.

Sergio M. Vicente-Serrano, Santiago Beguería, and Juan I. López-Moreno. A multiscalar drought index sensitive to global warming: The standardized precipitation evapotranspiration index. *Journal of Climate*, 23(7):1696–1718, April 2010. ISSN 0894-8755. doi: 10.1175/2009jcli2909.1. URL http://dx.doi.org/10.1175/2009JCLI2909.1.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

Daniel S Wilks. *Statistical Methods in the Atmospheric Sciences*. Academic Press, 3 edition, 2011.

David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53140–53164. PMLR, 2024. URL https://proceedings.mlr.press/v235/woo24a.html.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023. URL https://ojs.aaai.org/index.php/AAAI/article/view/26317.

A Implementation Details

This appendix provides the implementation and data timing details, complete results, gate analyses and proofs, and additional controls. Sections L and P specify the two levels of comparison: complete systems with their own FIT choices, and regressors with fixed local features and updates.

Data construction. The SPEI-GD archive (Liu, 2023; Liu et al., 2024a) provides daily SPEI fields at $0.25°$. We sample the first day of each calendar month at the 30-, 90-, and 180-day accumulation scales. Thus $y_{e,t}$ is the first-of-month value for month t, and origin t estimates $y_{e,t+1}$ after month-t inputs are complete. The product's standardization uses its full archive, so the target definition incorporates information beyond FIT. Every method uses this same retrospective target; the effect on comparative performance remains unmeasured.

SPEI values with $|y| > 8$ are treated as missing. Leading spin-up months with more than 50% missing values in a region are dropped. We retain land entities with at most 2% missing values and standard deviation at least 0.55, then subsample 150 per region with seed 7. These screens use the released series rather than FIT alone and tend to exclude nearly constant series in hyper-arid areas.

Local covariates are monthly NCEP/NCAR reanalysis fields. Twelve of the thirteen fields use a Gaussian grid of approximately $1.9°$; surface pressure uses a $2.5°$ grid. Each field is assigned to the target grid points by nearest neighbor. Several target points can therefore share the same coarse weather input. SPEI-GD uses daily ERA5 precipitation and Singer's potential evapotranspiration instead (Liu et al., 2024a); our monthly inputs differ in source and resolution. Monthly Niño indices and the Dipole Mode Index (DMI) represent ENSO and the Indian Ocean Dipole. Covariates are standardized using FIT $[0, 288)$ moments. Targets remain in SPEI units, so drought thresholds -1.0, -1.5, and -2.0 apply directly. Scoring excludes entity–origin pairs with missing targets, base forecasts, or features. Stored bare-reference MSEs use valid target/forecast pairs. In the joint MSE/MAE tables, FM-Avg is additionally restricted to finite RAIL-Pool outputs; its values can therefore differ slightly from the stored control references in table 7. The fixed-design comparison uses common scored masks (section P).

Timing protocol. Table 3 summarizes what each segment may do. Two rules deserve emphasis. (i) *Within-origin batching:* at each origin all $E{=}150$ entity forecasts are determined from pre-update predictions and loss histories. Expert predictions are stored before their states consume that origin's targets; subsequent updates cannot change the issued forecasts. RLS updates use a single vectorized entity batch, so no entity's score uses a sibling's current target. (ii) *Continuous state:* online state (RLS weights, gates, loss histories) updates on every origin from $t{=}37$ onward and is never reset at segment boundaries; the excluded gap $[328, 336)$ still updates state but is never scored, separating validation and test evaluation.

Table 3: **Timing protocol.** Origin t forecasts target y_{t+1}. "State" = RLS weights, gates, loss histories; "fit" = feature-ordering computation; "select" = meta-configuration choice; "score" = counted in reported metrics.

Origins	Role	State updates	May fit/select	Scored
$[2, 199)$	FIT-inner	from origin 37	feature orderings	no
$[200, 288)$	FIT-late	yes	meta-selection (all per-case choices)	no
$[288, 328)$	validation	yes	diagnostic threshold	yes (val)
$[328, 336)$	gap	yes	no	no
$[336, 479)$	test	yes	no	yes (test)

Loss histories and tie-breaking. Every candidate carries a trailing prequential loss over the meta-selected window: each expert's loss history records the per-origin mean squared error of its *gated* prediction (which equals its base whenever the expert's own gate is closed); the reversion gate's two loss histories record the *counterfactual* losses of the selected aggregate and of the bare backbone, appended at every origin whether the aggregate was deployed or not. All loss history updates occur after selection and deployment for that origin. Selection argmin resolves ties toward the earlier pool entry; candidates with empty loss histories are ranked last. Warm-up: expert RLS updates begin at origin $t \geq 37$ and an expert's gate may open only after $2|S_k|$ observations (16 for the per-entity bias expert).

Feature pool. The 52-feature pool contains an intercept; three target levels (y_t, y_{t-1}, y_{t-2}); current and one-step-lagged values of 13 local fields (precipitation rate, latent and sensible heat flux, maximum and minimum temperature, air temperature, specific humidity, downward shortwave radiation, zonal and meridional wind, snow water equivalent, skin temperature, and pressure); current and lagged Niño-1+2/3.4/4 and DMI indices; current and lagged two- and three-step rolling sums of precipitation, a water-balance proxy, and the target; and two calendar harmonics. Latitude, longitude, region code, entity ID, and time index are not regressors in this pool. The 66-feature pool adds season$\times$covariate products ($\sin\theta \cdot x$, $\cos\theta \cdot x$ for six covariate bases and $y_t\times$ precipitation/temperature); the 87-feature pool adds the 21 pairwise products of the six highest-ranked features.

Orderings and expert sets. On the FIT-internal segment $[2, 199)$, three orderings are computed per base: the LARS regularization path, greedy forward selection by incremental R^2 (collinearity-aware, stop at relative RSS gain $< 10^{-5}$, at most 24), and entity-bootstrap stability selection (20 resamples). The deployed 52-pool sets take LARS prefixes $\{4, 8, 12, 16, 20, 24, 32, 40, 52\}$, greedy prefixes $\{4, 8, 12, 16, 20, 24\}$, and stability prefixes $\{4, 8, 12, 16, 20, 24, 32\}$, plus the full pool ($\leq 23$ experts; prefixes are truncated when an ordering is shorter); the 66- and 87-pool sets add LARS $\{4, 8, 12, 16, 24, 32\}$, greedy $\{8, 16, 24\}$, stability $\{16, 24, 32\}$, and the full pool (≤ 13 each), and the ensemble base adds the same 13-set structure on the 52-pool. RAIL-Single thus runs up to 49 experts and RAIL-Pool up to 62 per case, the exact count varying only where an ordering falls below a prefix size.

RLS details. Updates use λ=1 (no forgetting), initialization $w_0 = 0$, $P_0 = I$; observations are the masked rows of all 150 entities at each origin. The expert gate opens when the expert has accumulated at least $2|S_k|$ observations and its cumulative corrected loss is below the base's. FIT can instead select a variant that compares losses over a trailing window.

Meta-selection outcomes. Across the 432 cases the FIT-late winner concentrates on evidence windows $\{16{:}68,\ 32{:}84,\ 64{:}137,\ 128{:}76\}$, with the cumulative window (10^9, effectively full history) winning the remaining 67; the same window length governs the candidate loss histories and the reversion comparison, so proposition 1's bound applies to 84% of deployments. Top-3 aggregation wins in 333 cases and the single leader in 99; the trailing-window expert-gate variant wins in 141.

Baselines. Ridge and HGB use the same 52-feature pool with FIT training (ridge penalty selected on a FIT tail; HGB early stopping with an internal validation fraction). Deep baselines (DLinear, PatchTST, iTransformer, TimesNet) use the Time-Series-Library implementations with 150 entities as channels, lookback 96, prediction length 1, d_{model}=128, two encoder layers, Adam with learning rate 5×10^{-4}, batch 16, up to 50 epochs with early stopping on the validation segment. Combination baselines select their hyperparameters (window, rate) on the FIT-late segment. All scoring uses the deployed protocol's test origins and entity-masked pooled errors.

Backbone checkpoints. The frozen checkpoint IDs are listed below. Chronos-2r uses the Chronos-2 checkpoint with FIT-selected past covariates. Context is capped at 2,048 steps for Chronos and 512 for the other backbones. Probabilistic outputs use the checkpoint's mean or median point forecast, as specified in the inference code.

Configuration	Checkpoint ID
Chronos-T5-small	`amazon/chronos-t5-small`
Chronos-T5-base	`amazon/chronos-t5-base`
Chronos-Bolt-tiny	`amazon/chronos-bolt-tiny`
Chronos-Bolt-small	`amazon/chronos-bolt-small`
Chronos-Bolt-base	`amazon/chronos-bolt-base`
Chronos-2 / Chronos-2r	`amazon/chronos-2`
TimesFM-2.5	`google/timesfm-2.5-200m-pytorch`
Moirai-2	`Salesforce/moirai-2.0-R-small`
TiRex	`NX-AI/TiRex`
Toto	`Datadog/Toto-2.0-4m`
TTM	`ibm-granite/granite-timeseries-ttm-r2`

Dependence-aware inference. The primary per-case test uses 4,000 resamples of entities and moving 8-month temporal blocks. As a sensitivity check, we partition each region into twelve geographic blocks (latitude tertiles by within-tertile longitude quartiles), resample these blocks, and jointly resample 12-month temporal blocks. The spatial sensitivity check uses 1,000 bootstrap

Algorithm 1 Deployed policy at origin t (one case, all E entities)

Require: base forecasts b_t (and $\bar{b}_t$ for RAIL-Pool); past loss histories $\{\ell_k\}$ of gated candidates; aggregate and bare loss histories ℓ^{agg}, ℓ^{bare}; online states $\{(w_t^{(k)}, P_t^{(k)})\}$ (RLS weights and covariances)

1: candidates: $\tilde{y}_t^{(k)} \leftarrow$ gated output of expert k ▷ its own gate may return the base
2: select: aggregate $\tilde{a}_t \leftarrow$ leader / top-three mean of $\{\tilde{y}_t^{(k)}\}$ by trailing ℓ_k through $t-1$
3: gate: $g_t \leftarrow \mathbf{1}\{\,\mathrm{mean}(\ell^{\mathrm{agg}}) < \mathrm{mean}(\ell^{\mathrm{bare}})\}$ ▷ past window D_{t-1}
4: **issue** $\tilde{y}_t \leftarrow g_t\,\tilde{a}_t + (1-g_t)\,b_t$ ▷ forecast *for* y_{t+1}
5: reveal y_{t+1} after all forecasts are issued: append every candidate's loss to its loss history, $\tilde{a}_t$'s loss to ℓ^{agg}, and b_t's loss to ℓ^{bare} (both counterfactual, deployed or not); batch-update all RLS weights and expert gates over entities ▷ no update reads y_{t+1} before step 4

replicates per case to obtain one-sided p-values, then adjusts its 432 p-values with the Benjamini–Hochberg procedure at level 0.05. Headline CIs resample the twelve regions while retaining all scales and backbones within each selected region. The residual-predictability analysis likewise uses region-block bootstrap intervals and region-block permutations. RAIL-Pool improves in 429 cases at nominal $p < 0.05$ in the primary test, and in 424 cases after the spatial/temporal check and Benjamini–Hochberg correction. These comparisons share regional data.

Reproducibility. Wrapper comparisons use deterministic replays of cached backbone forecasts with fixed seeds. The primary replay agrees numerically with the stored baseline (absolute MSE difference $< 10^{-9}$ in 427 of 432 cases). Five cases use a documented historical cache convention; their replay values serve as the reference. The selection and gate ablations of table 14 are computed in a single replay pass per case: an expert's recursive updates and trailing losses do not depend on which prediction was issued, so gate-off and reweighted variants share the same expert trajectory and differ only in the issued forecast. The one-pass outputs were verified bitwise (zero maximum difference) against independently replayed per-variant runs on representative cases spanning leader/top-k selection, both gate variants, and cases with frequent reversion. The anonymous code release includes reproduction instructions; data, caches, and per-case result files are generated locally.

External adapters. table 13 uses reimplementations of ELF (Lee et al., 2025) and ORCA (Dai et al., 2026) that follow the authors' released code and published hyperparameters. Only scale parameters are adapted: ELF's context 520 and 200-step refit exceed the 480-observation series and are rescaled to context 120 with refit every 12 origins, while its Fourier forecaster, Woodbury updates, and fast/slow/merge weighters keep their published coefficients. ORCA keeps its DLinear-style adapter (two blocks, width 128, kernel 7), Boltzmann routing at temperature 0.1, prior loss, EMA 0.2, FIFO buffer of 3000, and the published refill/update cadence. Both adapters see only the backbone's point forecasts and the entity series, no covariates, no candidate pool, matching their own protocols.

Target-only ablation. The target-only rows of table 13 restrict the residual feature pool to target history: the current and two lagged values, the two- and three-step rolling sums evaluated at t and $t-1$, and the intercept (8 of the 52 features), restricting wrapper inputs to target history and the backbone's point forecast, as in the external adapters. The methods retain their own history lengths and feature representations. A 10-feature sensitivity adds the deterministic calendar harmonics $(\mathrm{msin}, \mathrm{mcos})$. Two protocols are reported: the *deployed-config* row keeps the full-feature run's per-case meta-configuration (removing covariates while retaining the full-input configuration), and the *re-fitted* row re-makes every FIT-stage choice under the restricted pool. The three feature orderings are recomputed on the FIT-inner segment, the expert pool rebuilt from both the filtered stored orderings and the recomputed orderings plus the full restricted pool, and the per-case meta-configuration re-selected on FIT-late over the same grid as the deployed variant.

Figure 2 summarizes the prediction and update cycle.

B Input Availability Audit

Table 4 distinguishes the timestamps used in the replay from operational availability. The target is stamped on the first day of month $t + 1$, but its estimate uses completed month-t covariates. Their publication times were not reconstructed, and the retrospective SPEI-GD archive does not supply the target release schedule used by this replay. The experiment therefore measures sequential estimation of the historical index. Establishing an operational lead requires the actual input and target release dates.

Table 4: **Data availability at issue.** The estimate for $y_{e,t+1}$ (SPEI at the first day of month $t+1$) is issued once month-t inputs are complete. Input history ends at month t; the replay does not reconstruct publication or revision dates.

Item	Coverage used	Availability / note
SPEI-GD target $y_{e,t}$	first day of month t; accumulation window ends there	Retrospective product: distribution parameters fitted on the full archive, so absolute index calibration has product-level lookahead; the replay reveals $y_{e,t+1}$ only after issuing all predictions. All methods use this standardized target.
NCEP/NCAR R1 fields	completed month-t means (lag-0 features)	Completed month-t means are assumed available at issue; publication latency is not modeled. No partial-month values enter. Most fields have approximately 1.9° resolution; surface pressure has 2.5° resolution. Nearest-neighbor assignment to 0.25° target points preserves this coarse weather information.
Niño-1+2/3.4/4, DMI	values through month t	Completed monthly indices are assumed available; the replay does not model their release or revision schedules.

Entity selection uses missingness $\leq 2\%$ and standard deviation ≥ 0.55 computed on the released entity set, with random subsampling to at most $E{=}150$ per region; these dataset-level screens are applied before the online replay.

C Full Results by Scenario

Tables 5 and 6 give separate MSE and MAE tables for the baseline columns of table 2 on all 36 scenarios (twelve regions $\times$ three scales; the matched controls are summarized in table 15). Including Direct pool and the gated stack in the comparison, the stack leads at every scale and on most scenarios. RAIL-Pool is the strongest fixed-residual system, with Direct pool close behind. The full tables show where these averages conceal regional differences.

Table 5: **Test MSE on all 36 scenarios.** Methods follow table 2: Pers. = persistence, S-N = seasonal naive, DLin. = DLinear, Patch = PatchTST, iTr. = iTransformer, TNet = TimesNet, Mean/Best = Mean-Ens/Best-FM, FM = FM-Avg, R-S/R-P = RAIL-Single/RAIL-Pool. Each block lists twelve regions at one accumulation scale. Bold marks the lowest MSE within this baseline subset; Direct and Stack appear in table 7.

Region	Pers.	S-N	DLin.	Patch	iTr.	TNet	Mean	Best	Ridge	HGB	FM	R-S	R-P
SPEI-30													
East Australia	.9579	1.0565	.5790	.6416	.6203	.6640	.5367	.5476	.2425	.2899	.5461	.2418	**.2410**
Horn of Africa	1.2499	.9269	1.1145	1.3867	1.3680	1.4247	.4110	.4119	.5884	.4344	.4420	.3844	**.3596**
Iberia–Maghreb	.7459	.6160	.3027	.3260	.3140	.3492	.2930	.3006	.1796	.1891	.3062	.1777	**.1692**
India	.8428	.4263	.2392	.2424	.2303	.2322	.2196	.2213	.2158	**.1698**	.2407	.1937	.1832
NE Brazil	.9005	.7909	.9466	1.0823	1.0169	1.2704	.3928	.4016	.2667	**.2567**	.4106	.2783	.2787
North China	1.5048	1.1581	.6995	.7289	.7323	.7291	.6466	.6487	.4607	**.3877**	.6753	.4558	.4529
Pampas	1.4476	1.5480	.7876	.8811	.8800	.8249	.7645	.7536	.5363	.6440	.7761	.5208	**.5000**
Sahel	.5663	.2120	.1181	.1295	.1191	.1262	.1096	.1119	.1447	.1316	.1199	.1087	**.1015**
Southern Africa	1.2956	1.2648	.7371	.7107	.6977	.8842	.6844	.7135	.4540	.4706	.7097	.4491	**.4382**
SW Australia	.5470	.4521	.2341	.2542	.2588	.2563	.2254	.2269	.1223	.1312	.2350	.1250	**.1166**
US Great Plains	1.1930	1.2028	.6677	.8355	.7286	.7119	.6317	.6375	**.3325**	.3555	.6411	.3416	.3359
US West	.6596	.4875	.2579	.2839	.2754	.2836	.2419	.2436	.1564	**.1356**	.2526	.1539	.1496
Scale mean	.9926	.8452	.5570	.6252	.6035	.6464	.4298	.4349	.3083	.2997	.4463	.2859	**.2772**
SPEI-90													
East Australia	.3775	.7943	.2191	.2216	.2972	.5781	.1756	.1726	**.0874**	.1071	.1903	.0935	.0885
Horn of Africa	.6428	1.0617	.2238	.1873	.2542	.4921	.1846	.1742	.2349	.1801	.2136	.1793	**.1632**
Iberia–Maghreb	.3791	.4546	.1132	.1090	.1372	.2422	.0941	.0887	**.0597**	.0674	.1061	.0671	.0605
India	.4829	.3020	.1068	.0984	.1038	.1858	.0896	.0888	.0904	**.0637**	.1037	.0856	.0784
NE Brazil	.3775	.5870	.1902	.1813	.1977	.3474	.1547	.1478	.1197	**.1068**	.1701	.1243	.1152
North China	.6060	.8311	.2925	.2638	.2998	.5110	.2575	.2427	.1948	**.1741**	.2815	.1997	.1959
Pampas	.4383	1.3244	.4654	.2900	.3556	.7149	.2710	.2421	.1882	.2252	.2901	.1956	**.1817**
Sahel	.3790	.1276	.0495	.0461	.0535	.0756	.0352	.0343	.0486	.0381	.0428	.0374	**.0329**
Southern Africa	.5492	1.0510	.2898	.2415	.2917	.6219	.2493	.2260	.1860	**.1777**	.2775	.1840	.1792
SW Australia	.2887	.2602	.0828	.0766	.1035	.1531	.0563	.0618	.0332	.0395	.0647	.0385	**.0315**
US Great Plains	.4006	1.0034	.2360	.2248	.2653	.6027	.2123	.1922	**.1162**	.1276	.2288	.1338	.1278
US West	.3353	.2861	.0803	.0887	.1201	.1685	.0639	.0615	.0457	**.0421**	.0728	.0505	.0464
Scale mean	.4381	.6736	.1958	.1691	.2066	.3911	.1537	.1444	.1171	.1124	.1702	.1158	**.1084**
SPEI-180													
East Australia	.2551	.8215	.1443	.1253	.1749	.5669	.0959	.0947	.0700	.0749	.1114	.0697	**.0640**
Horn of Africa	.2941	1.9076	.1652	.1927	.1845	.6804	.1449	.1455	.1927	.1582	.1685	.1477	**.1316**
Iberia–Maghreb	.3029	.5009	.0744	.0794	.0750	.2391	.0539	.0528	.0471	.0524	.0660	.0519	**.0437**
India	.3245	.4005	.0896	.0694	.0781	.2112	.0623	.0642	.0725	.0596	.0754	.0669	**.0581**
NE Brazil	.2609	.6392	.1293	.1045	.1394	.3911	.0969	.0986	.0978	.0859	.1111	.0908	**.0833**
North China	.3194	.8330	.1766	.2102	.1808	.6207	.1519	.1506	.1485	.1404	.1703	.1348	**.1301**
Pampas	.2351	1.3174	.3627	.1723	.1980	.6218	.1298	.1282	.1280	.1380	.1452	.1162	**.1067**
Sahel	.3192	.1710	.0483	.0349	.0484	.0871	.0244	.0247	.0398	.0316	.0331	.0300	**.0230**
Southern Africa	.3178	1.2285	.1906	.1820	.2267	.6546	.1600	.1536	.1387	.1343	.1849	.1320	**.1259**
SW Australia	.2471	.2663	.0592	.0488	.0674	.1427	.0296	.0328	.0255	.0304	.0386	.0289	**.0220**
US Great Plains	.2422	1.1128	.1342	.1277	.2320	.6081	.1097	.1059	.0854	.0885	.1260	.0869	**.0790**
US West	.2851	.2937	.0601	.0597	.0512	.1768	.0360	.0363	.0358	.0299	.0455	.0354	**.0289**
Scale mean	.2836	.7911	.1362	.1172	.1380	.4167	.0913	.0907	.0901	.0853	.1063	.0826	**.0747**
Overall mean	.5714	.7699	.2963	.3039	.3160	.4847	.2249	.2233	.1718	.1658	.2409	.1614	**.1534**
Best count	0	0	0	0	0	0	0	0	4	9	0	0	23
Mean rank	11.8	12.4	8.7	8.6	9.4	11.1	5.0	5.1	3.9	3.3	7.2	3.0	1.5

Table 6: **Test MAE on all 36 scenarios.** Methods follow table 2: Pers. = persistence, S-N = seasonal naive, DLin. = DLinear, Patch = PatchTST, iTr. = iTransformer, TNet = TimesNet, Mean/Best = Mean-Ens/Best-FM, FM = FM-Avg, R-S/R-P = RAIL-Single/RAIL-Pool. Each block lists twelve regions at one accumulation scale. Bold marks the lowest MAE within this baseline subset; Direct and Stack appear in table 7.

Region	Pers.	S-N	DLin.	Patch	iTr.	TNet	Mean	Best	Ridge	HGB	FM	R-S	R-P
SPEI-30													
East Australia	.7290	.7060	.5570	.5855	.5788	.5740	.5185	.5270	.3575	.3920	.5262	.3520	**.3493**
Horn of Africa	.8404	.6844	.8758	.9972	.9722	.9801	.4736	.4723	.6078	.5030	.4911	.4645	**.4510**
Iberia–Maghreb	.6824	.5873	.4216	.4360	.4349	.4673	.4142	.4204	.3209	.3336	.4241	.3228	**.3146**
India	.6564	.4419	.3490	.3516	.3423	.3426	.3291	.3324	.3340	**.2949**	.3440	.3078	.3016
NE Brazil	.6868	.6132	.8093	.8676	.8414	.9250	.4360	.4417	.3846	**.3640**	.4479	.3857	.3915
North China	.9031	.7764	.6176	.6345	.6319	.6346	.5807	.5849	.4839	**.4305**	.5939	.4727	.4725
Pampas	.9188	.9398	.6879	.7174	.7160	.7063	.6722	.6695	.5721	.6220	.6778	.5459	**.5352**
Sahel	.6020	.3327	.2532	.2646	.2583	.2656	.2399	.2413	.2895	.2709	.2521	.2453	**.2369**
Southern Africa	.8618	.7646	.6477	.6188	.6210	.6866	.5947	.6074	.5099	.4942	.6086	.4828	**.4792**
SW Australia	.5856	.4777	.3609	.3871	.3915	.3702	.3455	.3461	.2640	.2710	.3538	.2620	**.2535**
US Great Plains	.8065	.8120	.6208	.7244	.6601	.6411	.5905	.5955	.4307	.4316	.5954	.4252	**.4224**
US West	.6258	.5025	.3825	.4140	.3952	.3939	.3613	.3616	.2923	**.2659**	.3696	.2847	.2802
Scale mean	.7415	.6365	.5486	.5832	.5703	.5823	.4630	.4667	.4039	.3895	.4737	.3793	**.3740**
SPEI-90													
East Australia	.4950	.6241	.3513	.3389	.4032	.5599	.2995	.2966	.2161	.2337	.3122	.2241	**.2150**
Horn of Africa	.6232	.7527	.3552	.3201	.3813	.5317	.3145	.3024	.3740	.3201	.3381	.3135	**.2977**
Iberia–Maghreb	.5072	.5127	.2566	.2539	.2875	.3676	.2293	.2240	**.1831**	.1955	.2454	.1967	.1859
India	.5351	.3880	.2400	.2301	.2349	.3177	.2137	.2149	.2178	**.1799**	.2303	.2106	.2016
NE Brazil	.4654	.5518	.3007	.2977	.3142	.4345	.2571	.2541	.2447	**.2283**	.2760	.2518	.2403
North China	.5888	.6702	.4007	.3823	.4019	.5326	.3716	.3644	.3224	**.2956**	.3890	.3266	.3242
Pampas	.5151	.8675	.5575	.4163	.4689	.6518	.3979	.3789	.3391	.3730	.4123	.3408	**.3281**
Sahel	.4959	.2631	.1679	.1607	.1736	.2085	.1405	.1397	.1721	.1477	.1549	.1468	**.1375**
Southern Africa	.5782	.7132	.4059	.3692	.4027	.5818	.3669	.3585	.3337	**.3132**	.3876	.3196	.3160
SW Australia	.4609	.3760	.2208	.2167	.2525	.3001	.1780	.1859	.1391	.1549	.1913	.1491	**.1349**
US Great Plains	.5094	.7730	.3812	.3741	.4053	.6006	.3524	.3344	**.2594**	.2673	.3647	.2769	.2704
US West	.4874	.3926	.2142	.2233	.2636	.3100	.1887	.1847	.1581	**.1512**	.2013	.1682	.1602
Scale mean	.5218	.5737	.3210	.2986	.3325	.4497	.2758	.2699	.2466	.2384	.2919	.2437	**.2343**
SPEI-180													
East Australia	.4229	.6547	.2862	.2588	.3214	.5670	.2219	.2184	.1942	.2005	.2409	.1950	**.1834**
Horn of Africa	.4209	1.0703	.2938	.3263	.3237	.6374	.2631	.2675	.3329	.2943	.2886	.2794	**.2609**
Iberia–Maghreb	.4662	.5496	.2112	.2191	.2118	.3797	.1747	.1719	.1618	.1726	.1934	.1723	**.1572**
India	.4692	.4794	.2271	.1980	.2097	.3573	.1832	.1838	.2020	**.1771**	.2022	.1926	.1773
NE Brazil	.4022	.6015	.2574	.2255	.2739	.4734	.2115	.2132	.2210	.2042	.2319	.2148	**.2027**
North China	.4430	.7072	.3254	.3607	.3238	.6191	.2969	.2938	.2991	.2856	.3139	.2802	**.2749**
Pampas	.3917	.8868	.5015	.3316	.3539	.6226	.2801	.2756	.2810	.2911	.2957	.2646	**.2528**
Sahel	.4465	.3077	.1674	.1392	.1667	.2261	.1170	.1145	.1510	.1304	.1346	.1288	**.1120**
Southern Africa	.4457	.8310	.3336	.3276	.3617	.6212	.2948	.2895	.2819	.2809	.3178	.2754	**.2676**
SW Australia	.4250	.3925	.1907	.1757	.2062	.2984	.1322	.1401	.1238	.1359	.1502	.1311	**.1151**
US Great Plains	.4109	.8339	.2908	.2868	.3902	.6193	.2555	.2488	.2248	.2286	.2722	.2267	**.2148**
US West	.4497	.4067	.1896	.1877	.1741	.3275	.1438	.1442	.1415	.1301	.1609	.1425	**.1281**
Scale mean	.4328	.6434	.2729	.2531	.2764	.4791	.2146	.2134	.2179	.2109	.2335	.2086	**.1956**
Overall mean	.5654	.6179	.3808	.3783	.3931	.5037	.3178	.3167	.2895	.2796	.3331	.2772	**.2680**
Best count	0	0	0	0	0	0	0	0	2	10	0	0	24
Mean rank	11.9	12.3	8.9	8.8	9.5	11.0	4.8	4.8	4.2	3.4	6.9	3.2	1.5

Table 7: Complete core-system test MSE on all 36 scenarios. Values average twelve backbone configurations; Direct-1 and Direct pool are repeated comparators. Bold marks the lowest MSE among these six systems. The gated identity stack is denoted Stack-GI. R-S and R-P denote the two residual systems.

Scenario	Bare	Direct-1	Direct	RAIL-S	RAIL-P	Stack-GI
East Australia / 30	0.5461	0.2229	0.2194	0.2418	0.2410	**0.2160**
Horn of Africa / 30	0.4421	0.5513	0.5264	0.3844	**0.3596**	0.3666
Iberia Maghreb / 30	0.3062	0.1691	0.1602	0.1777	0.1692	**0.1581**
India / 30	0.2407	0.1825	0.1775	0.1937	0.1832	**0.1573**
NE Brazil / 30	0.4106	0.2510	0.2406	0.2783	0.2787	**0.2329**
North China / 30	0.6758	0.4128	0.3899	0.4558	0.4529	**0.3857**
Pampas / 30	0.7761	0.4918	0.4976	0.5208	0.5000	**0.4777**
Sahel / 30	0.1199	0.1205	0.1143	0.1087	0.1015	**0.0976**
Southern Africa / 30	0.7097	0.4240	0.4034	0.4491	0.4382	**0.3912**
SW Australia / 30	0.2350	0.1121	0.1146	0.1250	0.1166	**0.1110**
US Great Plains / 30	0.6411	0.3062	0.2962	0.3416	0.3359	**0.2947**
US West / 30	0.2525	0.1414	0.1348	0.1539	0.1496	**0.1289**
East Australia / 90	0.1903	0.0801	0.0778	0.0935	0.0885	**0.0763**
Horn of Africa / 90	0.2136	0.2135	0.1989	0.1793	0.1632	**0.1603**
Iberia Maghreb / 90	0.1061	0.0572	0.0558	0.0671	0.0605	**0.0549**
India / 90	0.1037	0.0774	0.0744	0.0856	0.0784	**0.0668**
NE Brazil / 90	0.1701	0.1144	0.1083	0.1243	0.1152	**0.0983**
North China / 90	0.2820	0.1750	0.1710	0.1997	0.1959	**0.1640**
Pampas / 90	0.2901	0.1750	0.1720	0.1956	0.1817	**0.1702**
Sahel / 90	0.0428	0.0412	0.0389	0.0374	0.0329	**0.0317**
Southern Africa / 90	0.2775	0.1661	0.1686	0.1840	0.1792	**0.1582**
SW Australia / 90	0.0647	0.0296	0.0299	0.0385	0.0315	**0.0294**
US Great Plains / 90	0.2288	0.1104	0.1085	0.1338	0.1278	**0.1071**
US West / 90	0.0728	0.0419	0.0399	0.0505	0.0464	**0.0383**
East Australia / 180	0.1114	0.0635	0.0636	0.0697	0.0640	**0.0607**
Horn of Africa / 180	0.1685	0.1811	0.1795	0.1477	**0.1316**	0.1444
Iberia Maghreb / 180	0.0660	0.0447	0.0440	0.0519	0.0437	**0.0427**
India / 180	0.0754	0.0656	0.0645	0.0669	0.0581	**0.0562**
NE Brazil / 180	0.1111	0.0949	0.0967	0.0908	0.0833	**0.0822**
North China / 180	0.1703	0.1352	0.1343	0.1348	0.1301	**0.1221**
Pampas / 180	0.1452	0.1173	0.1155	0.1162	**0.1067**	0.1077
Sahel / 180	0.0331	0.0342	0.0327	0.0300	**0.0230**	0.0259
Southern Africa / 180	0.1849	0.1237	0.1230	0.1320	0.1259	**0.1164**
SW Australia / 180	0.0386	0.0244	0.0238	0.0289	**0.0220**	0.0230
US Great Plains / 180	0.1260	0.0807	0.0805	0.0869	0.0790	**0.0775**
US West / 180	0.0455	0.0318	0.0302	0.0354	0.0289	**0.0279**
Mean	0.2409	0.1574	0.1530	0.1614	0.1534	**0.1406**

D Exploratory Longer-Horizon Pilot

The main results use $H = 1$. An exploratory decomposition of the retrospective target found substantial seasonal structure, with a per-entity seasonal fit explaining up to 86% of variance in some regions. This decomposition motivates the calendar controls and the standardization sensitivity discussed in the main experiments.

A separate three-scenario pilot at longer horizons uses climatology decomposition, target-phase features, damped persistence, online RLS, and per-origin/per-entity gates. It scores current fitted parameters retrospectively on recent non-overlapping blocks after fitting. Reported gains range from $+5.8\%$ to $+18.4\%$ in selected scenarios, with a -4.4% degradation case. These are fitted-error results, so they do not establish gains under the main paper's predict-before-update protocol. A prequential multi-horizon evaluation remains necessary: each prediction must be scored before its target enters fitting.

E A Degradation Stress Test for the Reversion Gate

We evaluate the gate's response to a sustained degradation regime through a controlled residual-sign reversal. For 36 test origins ($[384, 420)$), each case uses the transformed target $y'_{e,t+1} = 2b_{e,t} - y_{e,t+1}$. This reverses the bare forecast's residual while preserving its squared error: the systematic bias the correctors were fitted to reverses, and unconditional correction can substantially increase error until the online weights relearn. The unchanged bare squared error provides the reference for the pooled policy contrasts.

We replay the deployed configuration on the stressed stream for all 432 cases. The replay changes realized targets used for RLS updates and loss histories. The 87-feature pooled predictor keeps target-history features from the original series, while the static 52-feature comparator uses the perturbed history. Cached backbone forecasts are also unchanged. The pooled gate variants share the same features, so their contrasts measure gating under this change in residual feedback. Comparisons with the static predictor also change the history supplied to the model. The original weather and cached backbone trajectories are retained. Origins are monthly, so the response lags below are in months.

Gating reduces cumulative excess loss during the intervention for both tested selection rules (fig. 6 and table 8). It also leaves the gated policies ahead over the full stressed test period. After the intervention, all evaluated policies have negative mean excess loss, indicating recovery relative to the bare forecast.

The response is slow. Among cases that revert, the median delay is 15 months, and reversion lasts for about half of the intervention. Most cases that never revert use cumulative history, which gives the temporary degradation little weight. The loss window therefore matters for response speed as well as prediction selection.

Table 8: **Residual-feedback stress summary.** In-window cumulative excess loss is relative to the bare forecast, summarized across cases. Full-test gain averages per-case relative MSE reductions.

Policy	Median excess	Mean excess	Full-test gain
Gated pool	$+2.3$	$+5.4$	$+4.3\%$
Pool without final gate	$+3.1$	$+7.2$	$+3.0\%$
Pool without any gates	$+3.6$	$+7.9$	$+2.1\%$
EWA without reversion	$+2.9$	$+6.8$	$+3.5\%$
EWA with reversion	$+1.7$	$+4.7$	$+5.2\%$

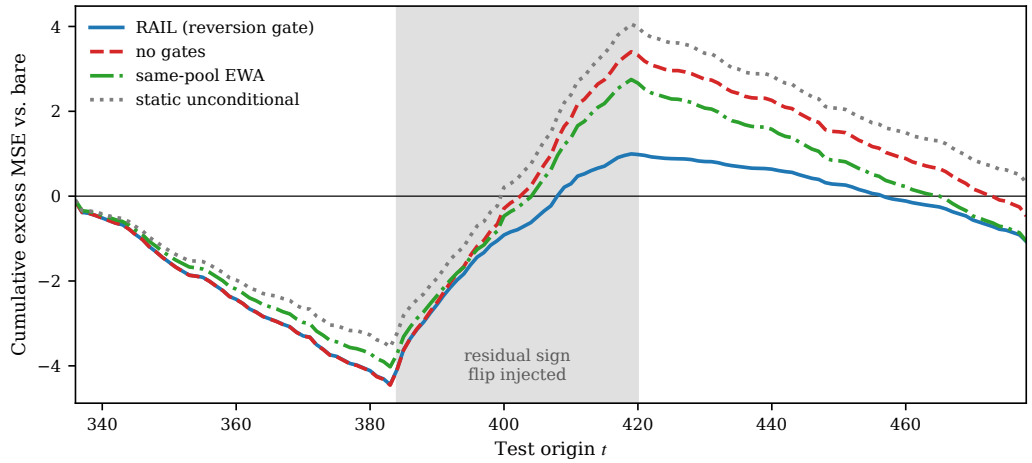


Figure 6: **Degradation stress test.** Cumulative excess squared error over the bare model, averaged across all 432 cases, with a residual sign flip injected over origins $[384, 420)$ (shaded). Ungated policies incur larger average in-window excess loss. The gate reduces this loss with a delayed response; 71 cases never revert. Cached backbone forecasts and pooled target-history features remain unchanged; the static comparator uses perturbed history.

F Case-Study Trace

Figure 7 shows how expert selection and reversion interact over one test stream. It illustrates the policy's decisions rather than an independent estimate of average benefit.

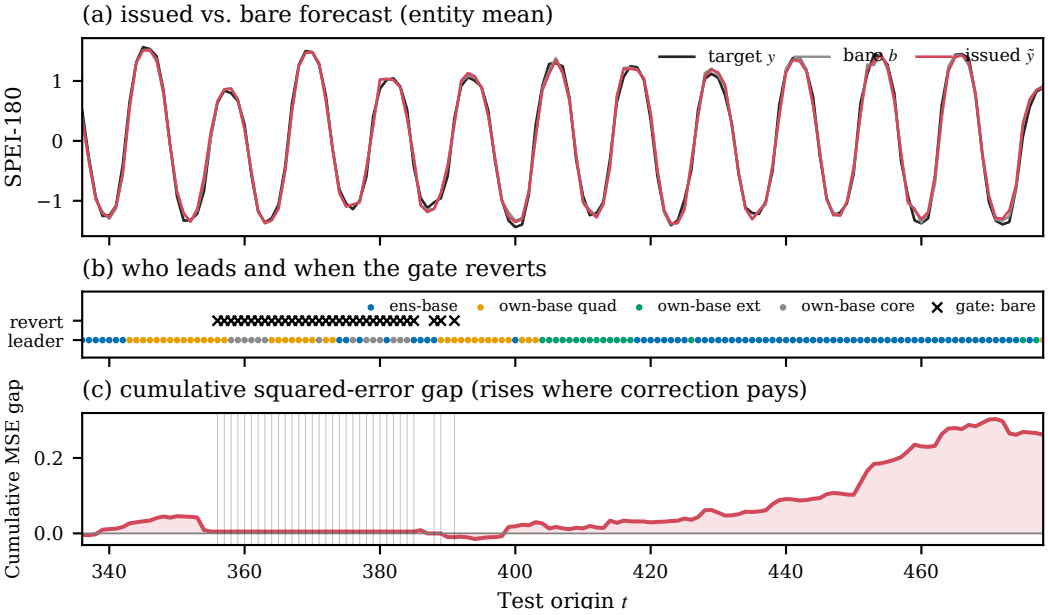


Figure 7: **RAIL in action on one case** (TimesFM-2.5 on Sahel SPEI-180). (a) Entity-mean target, bare, and issued forecasts over test origins. (b) The leading expert family at each origin (dots) and the origins where the reversion gate reverted to the bare forecast (crosses). (c) Cumulative squared-error gap $\sum(\mathrm{se}_{\mathrm{bare}} - \mathrm{se}_{\mathrm{RAIL}})$: flat through the reverted stretch, rising once the aggregate is redeployed. Leadership is contested across all four expert families, and the gate's intervention is visible as a sustained reversion block. The horizontal axis uses the same absolute origin index as the stress test.

G Positioning and Per-Backbone Tables

Table 9 summarizes which objects each adaptation setting updates.

Table 9: **Objects updated in representative adaptation settings.** The rows describe the forms discussed here; other implementations may add candidate selection or fallback rules. RAIL's final fallback is available without an explicit bare-forecast candidate.

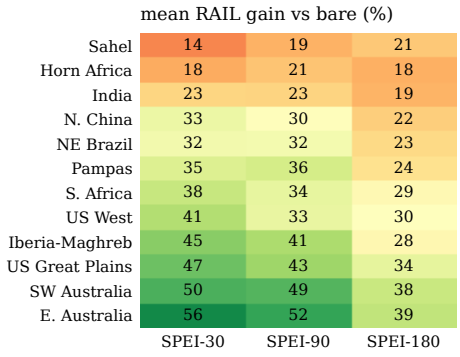
Setting	What is updated	Role of the original forecast
Parameter fine-tuning	Forecast-model weights	Starting model is modified
TTT / Tent	Model parameters using test inputs	Starting model is modified
Statistical post-processing	Calibration or correction map	Fixed predictor input
Online forecast combination	Candidate weights or selection	Can be included as a candidate
RAIL	Residual predictors, selection, and gates	Fixed base and final fallback
Free stack with gate and identity	Forecast coefficient, local regression, and selection	Learned input, explicit candidate, and fallback

Table 10: **Every backbone improves.** Mean test MSE by backbone (36 scenarios each), bare versus RAIL-Pool. Gain is the mean of the 36 per-case relative MSE reductions (the headline metric).

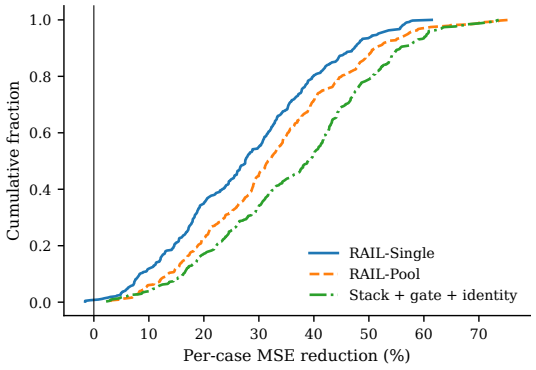
Backbone	Bare	RAIL-Pool	Gain
Chronos-T5-small	.2606	.1574	+36.0%
Chronos-T5-base	.2480	.1570	+31.6%
Chronos-Bolt-tiny	.2468	.1539	+37.5%
Chronos-Bolt-small	.2370	.1541	+32.9%
Chronos-Bolt-base	.2367	.1544	+31.7%
Chronos-2	.2292	.1535	+28.5%
Chronos-2r	.1935	.1456	+20.0%
TimesFM-2.5	.2230	.1493	+27.5%
Moirai-2	.2404	.1567	+31.3%
TiRex	.2308	.1526	+28.5%
Toto	.2336	.1548	+28.8%
TTM	.3117	.1520	+55.2%
average	.2409	.1534	+32.5%

H Variation in Adaptation Gains

The regional heatmap summarizes where RAIL-Pool gains are larger; the distributions compare how consistently each system improves on its own bare backbone.



(a) Mean RAIL-Pool gain by region and accumulation scale (12 backbones per entry).



(b) Gain distributions for the residual systems and the free stack with both fallback routes.

Figure 8: **Variation across the 432 cases.** (a) RAIL-Pool gain by region and scale. (b) Empirical distributions of system-level gain; the no-gates RAIL-Pool control also has no negative-gain cases.

H.1 Observed gate activity and bound size

The gate passes RAIL-Pool at 99.1% of natural test origins; 365 of 432 configurations use finite windows. Clipping does not bind. At the observed maximum $|y| = 7.2$, $B \approx 148$, and $W = 64$ gives a per-run bound of approximately 9.5×10^3 cumulative excess MSE. This calculation illustrates how loose the bound is at the observed range. A prospective guarantee requires a bound on future targets.

I Proofs

Proof of proposition 1. The gate's decision at origin t precedes the reveal of its target y_{t+1}, so its statistic sums only past origins: $D_{t-1} = \sum_{s=t-W}^{t-1} d_s$, where d_s is the aggregate's counterfactual excess loss for the forecast issued at origin s, recorded every origin whether deployed or not. Within a deployed run the issued forecast is the aggregate, so the realized excess equals $u_t = d_t$; while reverted, $u_t = 0$.

Let $[a, b]$ be a maximal deployed run of length $L = b-a+1$. If $L \le W$ then $\sum_{t=a}^{b} u_t \le LB \le WB$ trivially. For $L > W$, write $L - W = mW + r$ with $0 \le r < W$ and split the run at $b - W$: the last W origins contribute at most WB, and the prefix $[a, b-W]$ is handled as follows. Every W-block $[t, t+W-1] \subseteq [a, b-W]$ is exactly the past window behind the deployed origin $t+W \in [a, b]$, so it sums to $D_{t+W-1} < 0$. The first m consecutive W-blocks of the prefix therefore contribute a negative sum. If $r \ge 1$, the r leftover prefix terms $[b-W-r+1, b-W]$ together with the first $W-r$ trailing terms form the past window D_{b-r} behind the deployed origin $b-r+1 \in [a, b]$, so they too sum negatively and only the last r trailing terms survive, contributing $\le rB < WB$. If $r = 0$ the prefix is entirely covered by negative blocks and only the W trailing terms remain. Either way $\sum_{t=a}^{b} u_t \le WB$, as claimed.

For the detection bound, if per-origin excess stays $\ge \delta > 0$ after a break at τ, then $D_{t-1} \ge W\delta > 0$ once the window is entirely post-break, forcing reversion within W origins at most. The bound holds per run: after a reversion the gate may open again, and each new deployed run pays its own bound, so cumulative harm over a period containing several runs is the sum of per-run bounds. □

Proposition 2 (Cumulative-loss bound with an identity candidate)**.** *Consider cumulative-history exponential weighting with $K \ge 2$ experts, where the pool contains a fixed identity expert that always emits the bare forecast and candidates are combined by exponential weighting with $\eta = \sqrt{8 \ln K/(TB^2)}$ on losses in $[0, B]$. The standard bound gives $L_T \le \min_k L_T(k) + B\sqrt{T \ln K/2} \le L_T(\text{bare}) + O(B\sqrt{T \ln K})$: the mixture's cumulative excess loss over the bare forecast is bounded by a term that grows sublinearly in T.*

Proof of proposition 2. For losses in $[0, B]$, exponential weighting over $K \geq 2$ experts with $\eta = \sqrt{8 \ln K/(TB^2)}$ bounds the weighted mean loss by $\min_k L_T(k) + B\sqrt{T \ln K/2}$ (Cesa-Bianchi & Lugosi, 2006). Squared loss is convex, so the mixture forecast's loss is no greater than that weighted mean. A fixed identity expert in the pool has cumulative loss exactly $L_T(\text{bare})$, giving the second inequality. This bound uses cumulative losses and the stated learning rate; it does not apply to the sliding-window EWA sensitivity rows. The deployed sliding-window leader uses the gate analyzed in proposition 1. □

J Choosing Whether to Adapt

Section 5.7 reports that the FIT-only residual-predictability index S_{res} ranks realized gains. Here we evaluate it as a deployment rule: for a threshold τ, a case deploys the corrector iff $S_{\text{res}} \geq \tau$ and otherwise keeps its bare backbone. The threshold is chosen without test data in two ways: (a) *val-tuned*, minimizing the induced policy's pooled MSE over validation origins $[288, 328)$; and (b) *leave-one-region-out* (LORO), where τ is tuned on the other eleven regions' validation losses and applied to the held-out region's test cases.

Both systems choose adaptation everywhere. For RAIL-Single, this choice also holds in every leave-one-region-out fold. The threshold policy therefore has the same test error as always adapting. The observed downside is small, while low-score cases still benefit on average. Withholding correction from those cases loses more aggregate gain than it avoids in error. Table 11 records the numerical summaries underlying this interpretation.

Table 11: **Threshold-policy summaries.** Low residual skill means $S_{\text{res}} < 0.02$. Gains average relative MSE reductions over cases; the harmful-case excess refers to RAIL-Single.

Summary	RAIL-Single	RAIL-Pool
Validation-optimal threshold	$-\infty$	$-\infty$
Threshold / always-adapt MSE	0.161	0.153
Always-bare MSE	0.241	0.241
Mean gain at low residual skill	+9.8%	+17.2%
Cases worse than bare	3	0
Mean excess MSE among harmful cases	3×10^{-4}	—

K Comparison of Adaptation Gains

Figure 9 compares mean gains across the adaptation systems.

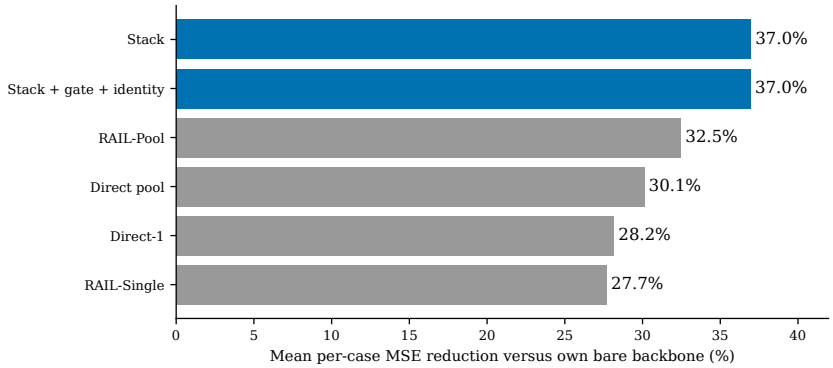


Figure 9: **Comparison of adaptation gains.** Mean per-case gain over each case's own bare backbone (432 cases). The core systems are sorted by gain, with the free-coefficient variants included explicitly. The free stack leads the evaluated systems; Direct and Direct-1 show how much improvement is obtainable without a frozen base. The oracle rows in table 12 give upper bounds for their specified candidate sets.

Table 12: **Candidate-pool branches.** Mean gain over each case's bare backbone (432 cases). The two middle rows are *alternative* single-branch extensions; RAIL-Pool combines both. CIs resample the twelve regions, preserving all scales and backbones within region. The oracle rows are upper bounds *within their named candidate sets* (bare backbones; RAIL-Pool plus backbone swap). The Stacking control uses a separate candidate family.

Configuration	Gain	95% CI
One-base reference (v0)	$+24.3\%$	$[18.3, 30.3]$
v0 + ensemble-base experts only	$+31.1\%$	n/a
RAIL-Single (feature/meta branch)	$+27.7\%$	$[21.6, 33.7]$
RAIL-Pool (both branches)	$+32.5\%$	$[27.1, 37.9]$
references		
Bare ensemble (median of 11)	$+9.5\%$	n/a
Online follow-the-leader (FTL) over the 12 bare backbones	-26.5%	n/a
Oracle: swap to test-best bare backbone	$+20.0\%$	n/a
Oracle: max(own corrected, bare ensemble)	$+29.3\%$	n/a
Oracle: max(RAIL-Pool, test-best backbone)	$+32.6\%$	n/a

Table 13: **Nearest black-box adapters under the same protocol.** Mean per-case gain over each case's own bare backbone (432 cases), region-clustered 95% CI, worst case, and negative-gain case count. Baselines follow the authors' released code with rescaled protocol parameters. The target-only rows restrict RAIL-Single to the adapters' information budget (target history only): the first keeps the deployed per-case configuration (a fixed-configuration input ablation); the second recomputes FIT-stage choices under the restricted input pool.

Method	Mean	95% CI	min	$n < 0$
ELF (Fourier forecaster + weighter)	-7.3%	$[-10.2, -4.9]$	-41.7%	371
ORCA (residual adapter + router)	$+3.4\%$	$[+2.2, +4.5]$	-3.1%	109
RAIL-Single, target-only, deployed config	$+3.8\%$	$[+3.0, +4.5]$	-5.7%	71
RAIL-Single, target-only, re-fitted on FIT	$+3.5\%$	$[+3.0, +4.1]$	-4.0%	97
RAIL-Single (same single backbone)	$+27.7\%$	$[+21.6, +33.7]$	-1.6%	3
RAIL-Pool	$+32.5\%$	$[+27.1, +37.9]$	$+2.9\%$	0

Table 14: **Selection and gate ablations.** Shared expert updates and FIT-frozen feature orders and windows. Aggregation and gate rows retain the corrected pool; the static rows restrict it to a single own-base expert. Mean per-case gain over each case's bare backbone (432 cases), with region-clustered 95% CI, tenth percentile, worst case, and the count of negative-gain cases. The static rows deploy the single FIT-winning own-base expert; EWA uses a fixed rate 16 under the deployed evidence window. The leader versus EWA gap measures selector replacement within this configuration.

Policy	Mean	95% CI	p10	min	$n < 0$
RAIL-Pool (deployed)	$+32.5\%$	$[+27.1, +37.9]$	$+13.6\%$	$+2.9\%$	0
no reversion gate	$+32.6\%$	$[+27.3, +38.0]$	$+13.8\%$	$+4.1\%$	0
no gates at all	$+32.7\%$	$[+27.6, +38.0]$	$+14.7\%$	$+3.9\%$	0
single FIT-winner expert, unconditional	$+24.0\%$	$[+17.9, +30.1]$	$+7.0\%$	-3.0%	4
single FIT-winner expert, both gates	$+23.7\%$	$[+17.4, +29.9]$	$+5.6\%$	-2.7%	9
same pool, exponential weighting	$+28.7\%$	$[+23.3, +34.3]$	$+11.5\%$	$+4.1\%$	0
same pool, EWA + reversion gate	$+28.7\%$	$[+23.2, +34.2]$	$+11.0\%$	$+4.1\%$	0

L Matched Online Controls

Table 15 varies access to the backbone, its coefficient, and the fallback rule under the same split and forward test pass. Feature orderings, expert construction, and FIT choices are recomputed where applicable, so these are comparisons of complete systems.

Direct-1 uses one online RLS on all 66 features, with no backbone or expert selection and a fixed evidence window $W = 64$. Direct pool retains the expert pool and selector with base forecast zero. The Seasonal control replaces the backbone with $b_{e,t} = y_{e,t-11}$. Stack includes the backbone as a regression feature and learns its coefficient. The final two controls add a reversion gate against the bare backbone, then an explicit identity candidate. Every control is scored against the same case's bare backbone; paired CIs against RAIL-Single resample regions.

Direct-1, Direct pool, and the initial Stack retain gates against a zero predictor. Available gate-activity audits show no aggregate vetoes in 432 Direct-1, 371 Direct-pool, and 368 Stack cases. This audit coverage is partial; the performance results cover all 432 cases. The gated-stack variants use the actual bare backbone as their final fallback.

Table 15: **Matched online controls** under the same causal split and forward test pass (432 cases). Mean test MSE, mean gain over the case's bare backbone, and a region-clustered 95% CI for the paired MSE difference against RAIL-Single (negative favors the control). "zero" marks an active gate whose comparator is the zero predictor. All rows report the completed grid.

Control	Base	Pool+sel.	gate	MSE	Gain vs bare	95% CI for ΔMSE vs RAIL-S
RAIL-Single (deployed)	backbone	yes	yes	0.161	+27.7%	N/A
Direct-1	none	no	zero	0.157	+28.2%	$[-0.015, +0.013]$
Direct pool	none	yes	zero	0.153	+30.1%	$[-0.019, +0.007]$
Seasonal-persist.	y_{t-11}	yes	yes	0.577	−221.1%	$[+0.286, +0.545]$
Stack (learned b_t coefficient)	none	yes	zero	0.141	+37.0%	$[-0.026, -0.016]$
+ live reversion gate	none	yes	yes	0.141	+37.0%	$[-0.026, -0.016]$
+ explicit identity candidate	none	yes	yes	0.141	+37.0%	$[-0.026, -0.016]$

The value of retaining a backbone varies across configurations. RAIL-Single is more competitive against Direct pool for Chronos-2r and TimesFM than for TTM and T5-small. Larger bare-backbone errors are associated with a larger disadvantage for fixed residual correction (Spearman $\rho = 0.56$). The seasonal control likewise improves on its own base while remaining worse than RAIL-Single in every case. Its own-base gain is 24.8%; table 15 shows why that relative improvement does not imply a competitive final prediction.

Stack improves on both local prediction and fixed residual correction, with paired intervals favoring it in the complete-system and fixed-design comparisons (table 16). The fixed-design experiment separates forecast weighting from feature reselection and other FIT choices.

Adding reversion or the identity candidate leaves Stack's mean error essentially unchanged. Most case-level errors are identical, and the remaining changes are small. These additions provide routes back to the bare forecast without explaining the stack's accuracy advantage. The finite-window gate has the per-run bound in proposition 1 for either a residual aggregate or a stack.

Table 16: **Paired error contrasts.** Confidence intervals (CIs) resample regions. Differences are first model minus second model in absolute MSE units; negative favors the first. Fixed-design contrasts hold local features and updates fixed.

Design	Contrast	95% CI for MSE difference
Complete system	Direct-1 − RAIL-Single	$[-0.015, +0.013]$
Complete system	Direct pool − RAIL-Pool	$[-0.017, +0.012]$
Complete system	Stack − Direct pool	$[-0.025, -0.005]$
Complete system	Stack − RAIL-Single	$[-0.026, -0.016]$
Fixed design	Stack − fixed residual	$[-0.0242, -0.0138]$
Fixed design	Stack − Direct	$[-0.0277, -0.0045]$
Input delay	Delayed Direct − Direct	$[+0.0659, +0.1177]$

For the target-only adapter comparison, the paired interval for RAIL-Single minus ORCA gain is $[-1.4, +1.5]$ percentage points. Unlike the MSE differences above, a positive gain difference favors the first model. This interval includes no difference.

M Drought Event Metrics

The event-conditioned metrics of section 5 are recomputed for RAIL-Single under the identical replay, conditioning scored entity-origins on realized SPEI $\leq -1.0, -1.5, -2.0$ over all 432 cases (approximately 772,200 scenario–entity–origin observations, repeated across twelve wrapped backbones for 9.27M scored predictions). Tables 17 and 18 report the RAIL-Pool and RAIL-Single results. Both variants improve dry-tail MSE. Detection metrics are computed over all valid origins so false positives on non-drought targets remain in their denominators; event-conditioned MSE uses only true events.

Table 17: **Event-conditioned MSE gain, RAIL-Pool.** Gains are relative to the bare backbone and pool scored predictions across cases.

SPEI threshold	≤ -1.0	≤ -1.5	≤ -2.0
MSE gain	$+42.9\%$	$+50.1\%$	$+52.7\%$

At the severe threshold, RAIL-Pool increases recall while precision falls from 0.65 to 0.56. The improvement in detecting events therefore comes with more false alarms.

Table 18: **Event-conditioned metrics, RAIL-Single (432 cases, pooled).** MSE gain over the bare backbone, and threshold detection at each drought level. Arrows indicate bare $\rightarrow$ RAIL-Single; n counts event predictions, including repetitions across the twelve backbones.

Threshold	n	MSE gain	F1	Precision	Recall
SPEI ≤ -1.0	2.14M	$+41.1\%$	$0.77 \rightarrow 0.83$	$0.83 \rightarrow 0.84$	$0.72 \rightarrow 0.81$
SPEI ≤ -1.5	751k	$+49.8\%$	$0.50 \rightarrow 0.65$	$0.72 \rightarrow 0.67$	$0.38 \rightarrow 0.63$
SPEI ≤ -2.0	128k	$+53.5\%$	$0.27 \rightarrow 0.45$	$0.65 \rightarrow 0.50$	$0.17 \rightarrow 0.41$

Table 19 breaks down RAIL-Single event gains by accumulation scale.

Table 19: **Event MSE gain by accumulation scale, RAIL-Single.** The largest gains occur at the shortest accumulation scale, SPEI-30.

Threshold	SPEI-30	SPEI-90	SPEI-180
SPEI ≤ -1.0	$+44.8\%$	$+38.6\%$	$+27.8\%$
SPEI ≤ -1.5	$+52.5\%$	$+49.1\%$	$+34.8\%$
SPEI ≤ -2.0	$+55.8\%$	$+52.2\%$	$+38.5\%$

N Additional historical-score comparisons

For each case, a ridge model predicts the bare residual from the 52-feature pool. Its penalty is selected on FIT $[2, 250)$; predictions on the separate tail $[250, 287)$ define $S_{\text{res}} = 1 - \sum(\hat{r} - r)^2 / \sum r^2$. The denominator is the loss of zero correction, not centered residual variance. The score is available before deployment. Table 20 compares historical rankings; table 21 gives distribution, adjustment, and budget summaries. Historical residual skill remains associated with future gain after adjusting for scale and backbone error, while the wrapper's own past gain supplies a stronger ranking in this benchmark.

Table 20: Historical scores versus RAIL-Single test relative gain (432 cases). Spearman correlations with 95% region-bootstrap intervals (2,000 resamples). FIT scores end before origin 288; validation scores additionally use origins 288–327. The FIT wrapper score is selected on FIT-late and can be optimistic.

Historical score	ρ	95% CI
FIT ridge residual skill	0.78	$[0.60, 0.90]$
FIT residual lag-1 correlation	-0.21	$[-0.31, -0.10]$
FIT-late bare MSE	0.51	$[0.32, 0.72]$
FIT-late selected wrapper gain	0.90	$[0.84, 0.94]$
Validation bare MSE	0.56	$[0.39, 0.74]$
Validation wrapper gain	0.88	$[0.82, 0.92]$
Validation Direct-pool gain	0.84	$[0.74, 0.89]$
Validation Stack-GI gain	0.83	$[0.74, 0.88]$

Table 21: **Additional historical-score summaries.** Correlations are Spearman coefficients. Budget gains average over all cases, including those retaining their bare forecast.

Summary	Value
Residual-skill median (10th–90th percentile)	0.20 (0.06–0.44)
Skill versus RAIL-Single gain, adjusted for scale and bare MSE	$\rho = 0.71$
Skill versus RAIL-Pool gain	$\rho = 0.76$
RAIL-Single mean gain where $S_{\text{res}} \geq 0.10$	$+31.2\%$
Half-budget gain: residual-skill ranking	18.2%
Half-budget gain: validation-wrapper ranking	19.2%
Half-budget gain: random allocation (expected)	13.8%

O Tuned Aggregation Sensitivity

The fixed-rate EWA row in table 14 is a configuration ablation. A separate cache contains all 36 scenarios for nine backbone configurations: the three Chronos-Bolt sizes, Chronos-2, Chronos-2r, Moirai-2, the two Chronos-T5 sizes, and TimesFM. We report those 324 cases, selected by complete cache availability across scenarios. On the identical candidate prediction streams, EWA rates $\{2, 8, 16, 64, 256\}$ and evidence windows $\{16, 64, 128, \infty\}$ are selected by validation MSE on $[288, 328)$. The reference RAIL selector remains FIT-selected. The sensitivity comparison therefore gives EWA a later tuning history.

Validation-tuned EWA and the FIT-selected reference have similar gains, with both paired intervals including no difference (table 22). The contrast with the fixed-rate ablation shows that the conclusion depends on tuning. Matching the search budgets and tuning periods would isolate the aggregation-rule comparison.

Table 22: **Tuned EWA sensitivity.** Mean relative gain on the 324-case subset. Differences are in percentage points against the reference; CIs use 4,000 region-bootstrap resamples.

Policy	Gain	Difference	95% CI
FIT-selected reference	30.78%	—	—
Validation-tuned EWA	31.03%	$+0.25$	$[-0.15, +0.68]$
Validation-tuned EWA with reversion	31.00%	$+0.22$	$[-0.17, +0.65]$

P Fixed-Design Contrasts and Input Sensitivity

To separate coefficient parameterization from pool and feature-search choices, we fit three online regressors on the same 66-feature design: Direct predicts $x_t^\top \beta_t$, fixed residual predicts $b_t + x_t^\top \beta_t$,

and the free stack predicts $\alpha_t b_t + x_t^\top \beta_t$. The stack adds one input, the cached backbone forecast. All use pooled entity batches, unit initial ridge precision, no forgetting, no expert pool, no gates, and no selected meta-parameters. At origins 37–478 they predict before updating; validation and test use the primary protocol's origins. All outputs are clipped to $[-5, 5]$ for scoring.

Columns are standardized on FIT origins 37–287. Non-intercept columns with FIT standard deviation below 10^{-8} are set to zero; the intercept remains. This rule was added after a numerical audit, using only FIT statistics. It prevents fields constant during FIT, such as snow water equivalent, from being amplified by the legacy 10^{-9} divisor when they later change. All three regressors share this design; the original RAIL grid retains its existing standardization.

For input sensitivity, all local weather fields and global climate indices are shifted one month before feature construction, while target history and calendar remain current. Their lagged and rolling features shift consistently. This tests covariate availability for Direct while keeping the target feedback schedule fixed. For seasonal sensitivity, a per-entity calendar-month mean estimated on the first 288 target months is removed from the target and its history, and Direct is rerun with the same feature construction. The zero-anomaly predictor is the FIT calendar mean expressed in anomaly coordinates. This transformation removes the FIT calendar level while retaining the archive's SPEI standardization parameters and the model's seasonal interaction features.

Table 23: Fixed-design online contrasts and input sensitivity. The three parameterization contrasts share a FIT-standardized 66-feature design, with FIT-constant non-intercept columns zeroed. Sensitivity rows rebuild and FIT-standardize features after the stated transformation. All use no pool, gates, or meta-selection. Stack appends the frozen forecast. Errors use identical valid entity-origins within each scenario and average twelve backbone configurations.

Model / sensitivity	30d	90d	180d	All
Direct	0.282	0.107	0.083	0.157
Fixed residual	0.290	0.116	0.082	0.163
Free stack	0.259	0.099	0.075	0.144
Direct, weather/indices delayed one month	0.496	0.148	0.104	0.249
Direct, FIT calendar anomalies	0.276	0.101	0.077	0.151
Zero anomaly (FIT calendar mean)	0.482	0.410	0.512	0.468

The free stack outperforms fixed residual regression and Direct in almost every case; both paired intervals favor it (table 16). Delaying covariates increases Direct's error in every scenario. Direct also beats the calendar-mean reference in every scenario after removing that mean from the target and its history. The latter contrast evaluates the full predictor beyond a seasonal level; isolating the weather contribution would require a target-only anomaly control. CIs use 4,000 resamples of the twelve regions, and compared methods share the same scoring masks.