WHEN REPRESENTATION CHANGE MISLEADS: EVALUATING ENCODER ADAPTATION IN TIME-SERIES FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Fine-tuning a time-series foundation model reshapes its internal representations, and that reshaping is routinely read as catastrophic forgetting to be prevented. Preventing it presupposes something worth preserving. **Centered kernel alignment (CKA) tells us that the representation changed; a fine-tune-versus-frozen-encoder intervention tells us whether that change was valuable. Across the cells we evaluate, the former is not an actionable substitute for the latter.** Screening **32 cells** (three backbones: Moirai-S/B/L, Chronos-T5, TimesFM-2.5; six datasets; two horizons) against a lookback-96 linear regression *fit on the target data* and scored on windows disjoint from every outcome we report, only **7 of the 31** scored cells show the pre-trained model beating that baseline, and all seven are Moirai on two of the six datasets. On those seven, full fine-tuning *improves* MSE in five. **No cell meets our prespecified criterion for harmful encoder adaptation**: under multiplicity-adjusted paired intervals the 31 cells split **2 freezing-better, 4 adaptation-better, 25 inconclusive**, and both cells where freezing decisively wins are cells our own screen rejects; a pre-registered seed top-up adds a third it *admits*, and the pre-registered unanimity form returns zero throughout. The outcome nonetheless spans -51.8% to $+46.1\%$ against zero-shot, and under an analysis fixed before it was run we find **no evidence** that CKA improves out-of-sample prediction of it beyond backbone identity: across 15 leave-one-cluster-out folds it *lowers* R^2 from $+0.201$ to $+0.176$. In sample CKA **does not provide a reliable, actionable ordering** either: within Moirai, the only backbone with enough cells to ask, $\rho{=}+0.168$ (clustered CI $[-0.469, +0.717]$), and acting on it as prescribed (freeze where drift is largest) is worth -6.97 pp $[-10, +14]$ against always adapting. Two measurement choices decide this: the frozen-encoder comparison must be scored on **held-out** windows (4 of 31 cells otherwise reverse sign, every one of them flattering the frozen encoder, and two of the four are cells that pass the screen), and the baseline must be **fitted**, since an unfitted per-window trend extrapolation that a constant predictor beats puts 16 of 21 Moirai cells on the wrong side of the screen.

1 INTRODUCTION

Fine-tuning a time-series foundation model reshapes its internal representations, and a large literature treats that reshaping as damage to be prevented: by freezing the encoder, by regularising toward the pre-trained weights, by confining updates to low-rank adapters. Every one of those prescriptions presupposes that the pre-trained representation was worth preserving *on the target task*. CKA tells us that the representation *changed*. It takes an intervention (fine-tune the whole model, then fine-tune it again with the encoder frozen, same data, same epochs) to tell us whether that change was *valuable*. **Across the cells we evaluate, the former is not an actionable substitute for the latter.**

One checkpoint, two datasets, opposite lessons. Fine-tune Moirai-Base on ETTh1 and its encoder barely moves (CKA 0.755 against the pre-trained encoder), and the model ends 22.8% *worse* than not fine-tuning at all. Fine-tune the same checkpoint on ETTh2 and the encoder restructures far more, to CKA 0.460, and the model ends 31.4% *better*. Read drift as damage and you would have protected the run that needed no protection and left alone the one that did.

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.

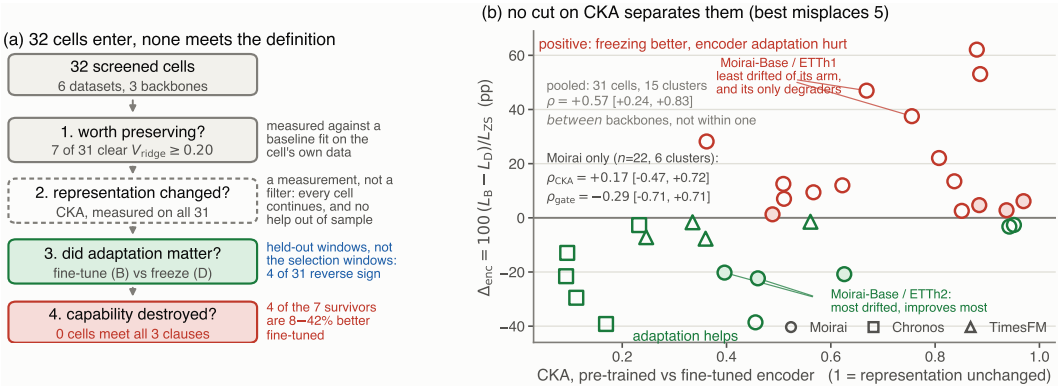


Figure 1: **The whole paper. (a)** The four layers of §3 as a funnel over the 31 cells that carry a selection-split gate: CKA filters none of them, and no cell meets either form of the degradation definition. **(b)** The same cells as CKA against held-out Δ_{enc} (positive = freezing better): **measured drift is not an actionable substitute for the intervention**, and no cut on CKA separates the cells adaptation helped from the cells it hurt. The labelled pairs are the paper's running example: Moirai-Base's two *least* drifted cells are its only two that degrade, and its two most drifted improve the most (§6). Filled markers clear the gate; both ρ are clustered by backbone×dataset, and only Moirai has enough cells to ask within one backbone (Appendix A27).

We do not claim that is surprising. A post-treatment similarity statistic has no reason to identify an interventional contrast, and a reader who finds the negative result obvious is agreeing with us; what we claim is that the substitution is made anyway (by the prescriptions above, and by an earlier draft of this work), and that measuring what it costs takes the intervention nobody runs. So we split the practitioner's question into four layers and answer them in order (Figure 1a): is there a capability worth preserving at all (§4)? did the representation change (CKA, §6)? did letting the encoder adapt help or hurt (§5)? and does the conjunction of the first and third amount to destroying a demonstrated capability (§5.3)? Layer 2 is the cheap measurement and layer 3 the expensive intervention, so the paper's question is whether the first can stand in for the second; a fifth section asks whether the whole screen works *prospectively* (§7).

Design and scope. We screen **32 cells** across three backbones (Moirai Small 13.8M, Base 91M, Large 311M; Chronos-T5 Ansari et al. (2024); TimesFM-2.5 Das et al. (2024), with MOMENT Goswami et al. (2024) screened and excluded), six datasets and two forecast horizons, and pair each with a full fine-tune and a frozen-encoder run at 3–10 seeds (§4). The screen is deliberately weak: it asks only whether the pre-trained model, zero-shot (ZS), beats a lookback-96 linear regression *fit on the practitioner's own target data*: not a tuned supervised forecaster, not an ensemble, a single ridge-OLS map estimated once on the training split and applied unchanged. We write that margin against a baseline b as $V_b = 1 - \text{MSE}_{\text{ZS}}/\text{MSE}_b$ (a *baseline-relative* advantage, never a property of the checkpoint alone), and score it on windows disjoint from every outcome we report. **Every claim below is about the checkpoints, datasets and fine-tuning protocols we evaluated**, and the five readings we do *not* claim are listed under *Scope of the claims* in the Ethics Statement.

The result, in three clauses. The presupposition mostly fails on the benchmarks the field actually uses. **Seven of the 31 scored cells** clear the screen, all of them Moirai and on two of the six datasets: all five of its ETTh2 cells and two of its four on ETTm2, and every failing cell is *below* the baseline rather than short of the threshold. That seven is one rung of eight: **20 of the 31 clear it against at least one admissible rung (14 Moirai, 3 Chronos, 3 TimesFM)**, and that union is the scope every drift analysis below runs on (§4). Where the advantage does exist, letting the encoder adapt usually *helps*: full fine-tuning improves MSE on five of the seven, by 31–42% on the three ETTh2 cells, and the two whose mean shows harm carry only 4/10 and 5/10 seeds agreeing on the sign. And **no cell meets our prespecified criterion for harm**: on multiplicity-adjusted paired intervals only 2 of the 31 favour freezing at these seed counts, and both are cells the screen *rejects* (§5.3).

Contributions.

1. **A negative result on the premise of forgetting mitigation.** Screened against a linear regression fit on the target data, only 7 of 31 cells show a pre-trained advantage to preserve, and all seven are Moirai, five on ETTh2 and two on ETTm2; four of the seven are *improved* by full fine-tuning in every seed, three of them by 31–42% (§4).
2. **The intervention contrast as the estimand, and drift as a failed predictor of it.** We name Δ_{enc} (full fine-tuning minus a frozen encoder) as the quantity this design manipulates, and show that CKA **does not provide a reliable, actionable ordering** of it across an outcome spanning -51.8% to $+46.1\%$: within-backbone $\rho{=}+0.168$ with clustered CI $[-0.469, +0.717]$ *within Moirai*, the only backbone with enough cells to ask (22 cells, 6 clusters), with Chronos and TimesFM read qualitatively beside it rather than pooled with it. A wide interval is a failure to establish an ordering, not evidence against one (§6).
3. **No cell meets our prespecified criterion for harmful encoder adaptation** (2 freezing-better, 4 adaptation-better, 25 inconclusive at a multiplicity-adjusted $q{<}0.05$; zero under the pre-registered unanimity criterion), so the value gate we proposed has no positive instance to predict, so precision is undefined, not poor: TP 0, FP 8 on eight cells whose predictions were frozen in git before those cells were trained, and $\rho{=}-0.293$ against the intervention outcome within Moirai; a direction, not an interval claim (§7).

What is a claim here, and what is a control. Those three are the paper's claims, argued in §§4–7. Everything else is a control or a boundary case, labelled where it appears: influenza-like illness (ILI) and an IoT anomaly detector are external cells that gate-fail, reported as case studies rather than evidence (Appendix A29.3); the mitigation spectrum, strict freeze and the LoRA sweeps ask only whether the frozen-encoder comparison depends on how it is implemented (Appendices A6, A8, A10). No claim above rests on any of them.

2 Background and Related Work

Related work: three lines, and the gap between them. *(i) Forgetting* is studied in continual learning, NLP and fine-tuning dynamics Kirkpatrick et al. (2017); Li & Hoiem (2018); Luo et al. (2023); Kumar et al. (2022); Andreassen et al. (2022), and supplies the prescriptions we test. **This is not a continual-learning study**: no task sequence, no loss on a prior task, one adaptation step, so "forgetting" here is degradation relative to the same checkpoint's zero-shot on the same data, and a catastrophic-forgetting result and ours can hold at once. *(ii) TSFM adaptation* recipes span probabilistic fine-tuning, generative pre-training, LLM reprogramming and cross-task transfer Rasul et al. (2024); Liu et al. (2024); Jin et al. (2024); Zhou et al. (2023), and are evaluated on target-task accuracy. *(iii) Representation analysis* supplies the measurement (representation similarity and probing Kornblith et al. (2019); Hewitt & Manning (2019); Pimentel et al. (2020)), including work probing frozen TSFM internals directly Wiliński et al. (2025), where we instead track how representations *change under fine-tuning*. **None of the three establishes that the pre-trained representation was worth preserving on the target task**, which is the premise (i)'s prescriptions rest on and what this paper measures.

Pre-training value gate. The gate is deliberately cheap and *downstream-only*: the practitioner runs the frozen model zero-shot on the target set and compares against a lookback-96 linear regression *fit on that same dataset* (§3), so neither pre-training data nor pre-training compute is required. We report V_b as a continuous quantity for every cell (Table 3) rather than only its verdict, and the 0.20 operating point is calibrated to ETT short-horizon benchmarks, where linear baselines are famously competitive with deep forecasters Zeng et al. (2023). That competitiveness is the whole reason the gate has teeth: **7 of the 31 scored cells clear it**, all on Moirai and confined to two series, and in the other 24 the pre-trained model has no measured advantage for fine-tuning to destroy.

Task-level consequence: forgetting. We define forgetting as signed relative change in zero-shot-canonical task metric on held-out test: forg.$\% = 100 \cdot (M_{\text{FT}} - M_{\text{ZS}})/|M_{\text{ZS}}|$ (MSE for forecasting, so forg.>0 means the fine-tuned model is worse than the zero-shot one: *task-level forgetting*).

Decoupling harmful forgetting from benign specialisation. Representational change that redirects features toward the target (*benign specialisation*) must be separated from change that discards generally useful structure (*harmful forgetting*). What remains, with an earlier probe-based criterion retracted, is *value-gate conditioning*: change can be harmful only where the structure it discards was

valuable, i.e. in a gate-pass cell; in a gate-fail cell the same change is benign by construction, because no demonstrated advantage existed to lose. Gate-passing plus forgetting>0 is therefore necessary for harm but not sufficient, and whether the encoder's *change* carried the loss is settled by intervention (full fine-tuning against a frozen encoder on the same data for the same epochs), and not by observing the representation.

3 Protocol and Definitions

Moirai for forecasting. Moirai Woo et al. (2024) predicts a probabilistic distribution over a target horizon given context. For ETT forecasting we report MSE on the median of 20 forecast samples at h=96 and h=192 (Appendix A4); the IoT anomaly score of the negative control is defined in Appendix A14.

Training details. AdamW (η=10^{-4}, weight decay 10^{-4}), batch size 16, gradient clipping at 1.0, **20 epochs with no early stopping**: every Moirai cell is read at its final epoch. **22 of the 31** cells train on 1k windows, every row prints its own n, and the sweep covers $n\in\{500, 1\text{k}, 2\text{k}, 5\text{k}, 10\text{k}\}$. **Reproducing any Moirai fine-tune requires the `uni2ts` `PackedStdScaler` patch of Appendix A3**.

Representation diagnostics, and which representation is primary. Two metrics: linear CKA Kornblith et al. (2019) between pre-trained and fine-tuned encoder representations, and weight drift (ℓ_2 distance from pre-trained parameters). **Every CKA in the body is of the final encoder output, mean-pooled over tokens**, on a fixed prefix of the selection split: the comparison a practitioner holding two checkpoints can actually make, and the one variant that needs no choice of layer or token position. The choice is not free: computed token-flattened instead, the same encoders read uniformly high and the per-layer profile compresses too far to order. That variant, the per-layer profiles and a random-initialisation floor are all reported (Appendices A18, A17).

Baseline-relative pre-trained advantage. Nothing can be said about a checkpoint having a *capability worth preserving* except relative to a baseline, so the baseline belongs in the quantity. For a baseline b, $V_b = 1 - \text{MSE}_{\text{ZS}}/\text{MSE}_b$ is positive when the pre-trained model beats b, and thresholding it at 0.20 (a prespecified operating point, not an established boundary) is the *baseline-relative capability screen*, for which we write *gate* in tables, figures and code. V_b **is a property of the pair, not of the checkpoint**: every count we report is a count against a named b, and §4 reports a ladder of eight. *Capability worth preserving* is therefore not a property a checkpoint *has*: a claim of that form would be a claim against every baseline a practitioner might deploy instead, and the ladder is the closest this design comes to making one. Our primary b is one ridge map $\mathbb{R}^{96\times F} \to \mathbb{R}^{h\times F}$ (λ=10^{-4}) fit by least squares on the *training* windows and applied unchanged to the evaluation windows, on the train-normalised scale the runs report (up to 300 windows). Both terms see the same targets, and the pre-trained model sees $96{+}h$ steps of context against the baseline's 96, which if anything makes V_b generous to the pre-trained model.

Three splits, and which quantity each one scores. The *training* split fits both the fine-tuning updates and every baseline; the *selection* split is where any checkpoint is chosen **and where we score** V_b; the *held-out* split is disjoint from both and is where every outcome is scored. So a cell's admission cannot borrow the windows that later judge it.

The estimand, named once. For a cell with S paired seeds,

$$\Delta_{\text{enc}} \;=\; \frac{1}{S}\sum_{s=1}^{S} 100 \cdot \frac{\mathcal{L}_{\text{test}}(\text{B}_s) - \mathcal{L}_{\text{test}}(\text{D}_s)}{\mathcal{L}_{\text{test}}(\text{ZS})}, \tag{1}$$

the difference in forgetting between full fine-tuning (B) and the frozen encoder (D), positive when freezing does better. Both arms share the zero-shot denominator, so Δ_{enc} is the raw contrast $\mathcal{L}_{\text{test}}(\text{B}) - \mathcal{L}_{\text{test}}(\text{D})$ rescaled per cell to make different loss scales comparable, with no sign or ordering changed. It is the estimand because it is the contrast this design *manipulates*: B and D differ in exactly one thing, whether the encoder's weights may move. **Representation drift is not manipulated.** CKA is an observational *post-treatment* measurement (taken after the intervention, on the object the intervention changed), so it enters only as a candidate *predictor* of Δ_{enc}. That asymmetry is why a CKA-to-outcome association would not license a CKA-based decision rule even had we found one.

Definition (degradation of a demonstrated capability), and its interval form. A cell counts as degradation when all three clauses hold: (i) $V_b \geq 0.20$ on selection windows, so there is a measured

pre-trained advantage over b; (ii) full fine-tuning ends worse than not fine-tuning at all, $\text{forg.}_{\text{B}} > 0$; and (iii) the frozen encoder ends better, $\text{forg.}_{\text{D}} < 0$, localising the loss to encoder weight updates rather than to the data or the epoch budget. **Clauses (ii) and (iii) are reported two ways**: as pre-registered, each holds in *every* seed; and (primary from this revision) as a paired interval, the 95% CI on B$-$ZS lying entirely above zero and the CI on D$-$ZS entirely below, propagating the zero-shot reference's own error and adjusted for multiplicity (Appendix A22). Intervals are primary because unanimity measures agreement among seeds rather than the size of an effect or our confidence in it; the registered form stays because retiring a criterion after seeing its outcome is not a choice we get to make. **Both return the same count**, so nothing in this paper turns on which is read.

Conditions. **B** NLL-only fine-tune (baseline: how much does unconstrained fine-tuning drift and forget?); **C** NLL+SupCon (Appendix A6); **D** frozen encoder (control: separates encoder updates from head training and overfitting); **E** LoRA (r=8); **F** L2-SP and **G** EWC. Conditions D and B are the comparison every claim in this paper turns on. **These letters describe Moirai**; the other two backbones differ in objective and head, one non-natively, as the Ethics Statement's *Scope of the claims* says. **We name conditions rather than letter them from here on**: *full fine-tuning* (B), *frozen encoder* (D, more precisely *backbone-frozen*), *strict frozen encoder* (H), *LoRA* (E). We keep the letters for the tables, the clauses of the definition above and Equation 1, so that no result in the body has to be read through a lookup.

What condition D does and does not hold fixed. D freezes the encoder's own weights but leaves the input projection and mask encoding trainable, so it controls *encoder weight updates* and not representations: Δ_{enc} is the effect of adapting one *designated stack*, not of changing what the encoder represents. Its measured CKA is near but never exactly 1 on Moirai (0.889–0.9999) and falls as low as 0.38 on TimesFM, where the encoder stack's weight drift is exactly zero. **H** (strict freeze) also freezes those two modules, is the only Moirai condition at CKA exactly 1.000, and reverses Δ_{enc}'s sign on 4 of the 16 cells we ran it on: every Moirai/ETTm2 cell. Figure 4 draws which modules move under each (Appendix A8).

Dispersion. Every $\pm$ in the body and in every generated table is the standard error of the mean over seeds; where an appendix table or passage uses a standard deviation instead it says so, and says which, where it does.

Two measurement choices that decide this result. *(1) Score the frozen-encoder comparison on windows you did not select on.* Scoring Δ_{enc} on the selection windows instead reverses its sign on 4 of our 31 cells, and all four reverse toward flattering the frozen encoder (§7.1). *(2) Fit the linear baseline.* An earlier version of this work reached the opposite conclusion from a baseline that was an unfitted per-window trend extrapolation (a denominator a constant predictor beats), and correcting the estimator moves 17 of Moirai's 21 cells across the threshold, 16 of them to gate-fail (Appendix A29.2).

4 Experiment 1: Is There a Capability Worth Preserving?

To ask whether fine-tuning damages valuable pre-trained features we first have to find a cell where such features are demonstrably present. This section reports that search, and it comes out negative: across **32 screened cells** (21 Moirai: three capacities, five datasets, two horizons; 5 Chronos-T5, 5 TimesFM-2.5 and 1 Moirai/ILI), **seven** clear the value gate, and all seven are Moirai: five on ETTh2 and two on ETTm2. The gate is scored on selection windows, which exist for 31 of the 32 (TimesFM/Electricity has no paired frozen-encoder run, so it is screened but carries neither a gate nor an intervention row), and every denominator below is 31. Figure 1a is this screen emptying.

Moirai: 7 of 21 (Appendix Table 3). ETTh2 clears at every capacity: Small h=96 ($+0.874$) and h=192 ($+0.932$), Base h=96 ($+0.777$) and h=192 ($+0.925$), Large h=96 ($+0.875$). ETTm2 clears at two of four, Small h=192 ($+0.352$) and Base h=192 ($+0.211$). Every other dataset fails at every capacity, and every failing cell is below the baseline rather than short of the threshold (Table 3). Capacity does not help: Moirai-Base is 6.6$\times$ larger than Moirai-Small and *further* behind the baseline on both ETTh1 and Weather.

Chronos-T5-Small: 0 of 5 and **TimesFM-2.5: 0 of 4** (§6, Appendix Table 7). All nine cells fall in a narrow band about zero, -0.275 to $+0.082$, and none reaches the threshold. The dataset contrast holds on both splits: ETTh2, where Moirai clears the gate at every capacity, is where Chronos is *below*

the fitted baseline (-0.184 on selection, -0.325 on held-out), so the advantage is not a property of the dataset alone.

One rung of eight. The screen's emptiness is not one estimator's artefact. Scoring the same numerator against a **ladder** of eight baselines (on the same selection windows, with the training mean as an admissibility floor), and measuring each cell against the *strongest* admissible rung it faces, **no cell of the 31 clears** 0.20, and none does on either of two further window sets. **20 of the 31 clear it against at least one admissible rung (14 Moirai, 3 Chronos, 3 TimesFM)**: that union, not the seven, is the scope every drift analysis below runs on, so those analyses are not confined to one backbone. Where this paper says seven cells clear the gate it means seven clear *this* rung of the eight; the per-rung counts, the other two window sets, and the four cells a near-constant rung would admit are in Appendix A20. Giving the baseline Moirai's own $96{+}h$ context instead of 96 makes it the stronger denominator on **12 of 21** cells and takes that seven to **six**; on held-out windows the comparison reverses, the count rises from **5 to 7**, and folding the matched column into the ladder leaves the count clearing 0.20 against every admissible rung at **0 of 21 on both splits** (Appendix A20.4).

5 Experiment 2: Does Encoder Adaptation Help or Harm?

What the treatment is. Conditions B and D differ in exactly one thing, whether the encoder stack's weights may move (Equation 1), and where that boundary is drawn is load-bearing: D leaves the input projection and mask encoding trainable, and the strict freeze that closes that gap on 16 of the 31 cells *reverses* Δ_{enc}'s sign on 4 of the 16 cells we ran it on (every Moirai/ETTm2 cell, 2 of which are among the seven survivors below), and 15 cells rest on the ordinary freeze alone (Appendix A8).

5.1 What Fine-Tuning Does to the Seven Survivors

The seven cells Experiment 1 admits are the entire evidence base for a claim about damage to valuable features, and they do not support one (Appendix Table 41).

On four of the seven, full fine-tuning improves the checkpoint in every seed. On Moirai-Base at $h{=}96$ and $h{=}192$ and Moirai-Large at $h{=}96$, all on ETTh2, forgetting is -31.4%, -36.3% and -42.4% against zero-shot, negative in 3/3 seeds each, at CKA 0.460, 0.396 and 0.626, the largest representational change among the seven. Freezing the encoder captures only part of that gain (-9.1%, -16.1%, -21.6%), so encoder adaptation is where most of the improvement comes from. The fourth, Moirai-Base/ETTm2 at $h{=}192$, improves by a smaller 8.4% and is the one survivor where freezing does slightly better (-9.7%), at CKA 0.488.

On the remaining three, the mean harm does not survive inspection. Moirai-Small/ETTh2 at $h{=}192$ reads $\text{forg.}_{\text{B}}{=}{+}1.99{\pm}3.32$ with 5 of 10 seeds positive: a coin flip on the sign, and the closest the matrix comes to degradation. At $h{=}96$ the positive mean (${+}9.71{\pm}10.58$, 4 of 10 seeds) is produced by one seed that diverges under the *frozen* encoder too, so it is an optimisation failure and not an encoder-drift effect; we report means including it throughout (Appendix A29.2). Moirai-Small/ETTm2 at $h{=}192$ runs the other way (a mean 1.10% improvement with 3 of 5 seeds harmful), so its sign is a seed draw in either direction.

5.2 Sample Size, and What Mitigation Is For

Sweeping $n{\in}\{500, \ldots, 10\text{k}\}$ on Moirai-Small/ETTh2 at $h{=}96$ (one of the seven survivors, so a sweep where a pre-trained advantage does exist) separates the two readings directly: drift grows monotonically with n (CKA 0.949 at 500 to 0.518 at 10k) while forgetting is non-monotonic and reverses sign, $+14\%$ at 500, -2.9% by 2k, $-5.3{\pm}2.1\%$ at 10k (7/10 negative). **If drift magnitude predicted harm these columns would track; they do not.** Whatever harm exists is confined to Small at $n{\leq}1$k (Table 4, Figure 3, Appendices A12.1, A2).

What mitigation is for, and what preserving a representation buys. Five strategies spanning unconstrained fine-tuning to full freezing (LoRA, EWC, L2-SP, frozen encoder, strict freeze) are reported in Appendix A6, on Moirai-Small/ETTh2, the one cell where we ran the whole spectrum. Among the four arms trained at $n{=}1{,}000$ there, the ones this cell can compare, the three weight-anchoring arms span CKA 0.939–0.972 and land within 0.003 MSE of one another, so a 0.033 spread in preservation buys no measurable utility; the *most* preserving arm (LoRA, CKA 0.979) is also the

best on the task and the only one to improve on zero-shot (-9.1%), and at h=192 the worst arm on the task is EWC, at the third-highest CKA of the four. **Preserving representational similarity is not the same as preserving useful behaviour**, the dissociation of §6 read across mitigation strategies rather than across cells, but extending LoRA to eight further value-cells leaves full fine-tuning ahead on four of them (Appendix A9).

5.3 No Robust Degradation Instance Survives a Prespecified Definition

Applying the three-clause definition of §3 to all 31 intervention cells returns **zero cells**: **no robust, seed-consistent degradation instance survives a definition fixed in advance**. Relaxing that definition does not manufacture one either: across 8 rungs from the weakest defensible one up to ours, at either gate threshold, **the largest count anywhere is 2**, and it is reached only by the rung that drops the means clause altogether and checks seed signs alone; every rung that keeps the means clause admits at most 1, the 5/10 seed split named below (Appendix A21).

Clause (i) removes 24 of the 31 outright: they have no measured pre-trained advantage, so a worse post-fine-tuning score is benign replacement rather than damage. Of the seven that survive it, four fail clause (ii) in the strongest possible way: full fine-tuning *improves* MSE, by 8–42% with every seed agreeing. That leaves three, and all three fail clause (ii) cell by cell (§5.1); Moirai-Small/ETTh2 at h=192 remains the closest the matrix comes. **Per-cell inference sharpens this rather than rescuing it.** Paired 95% intervals on Δ_{enc}, Benjamini–Hochberg adjusted across the 31 cells, call **2 cells freezing-better, 4 adaptation-better and 25 inconclusive**, and the two where freezing decisively wins are Moirai-Small/ETTh1 at both horizons, which the screen *rejects*, while two of the four where adaptation decisively wins *pass* it, at $+0.925$ and $+0.875$. The interval form of the definition returns zero cells, as the registered form does, and still zero when a pre-registered seed top-up adds a third freezing-better cell, which the screen *admits* (Moirai-Small/ETTm2 h=192, gate $+0.352$; Appendix A22). **What the inconclusive calls cost is itself reportable**: the survivors' own dispersion gives minimum detectable effects of 4.7–14.7 pp, and Moirai-Small/ETTh2 h=96 and Moirai-Base/ETTm2 h=192 would need 182 and 212 paired seeds, so we report those two as *unresolvable under this design* rather than as absent. Aggregated over the 20 value-cells with clustered resampling, Δ_{enc}=$+0.7$ with CI $[-11.1, +14.1]$ and 10 of 20 positive (Appendix A22). Changing the gate's *estimator* does admit cells: six of the ladder's eight rungs admit none, persistence admits three and GBM one. All four rest on a denominator within 5% of the training-mean floor and roughly twice the fitted ridge's error on the same windows, so a *weaker* baseline is what manufactures a degradation cell (Appendix A20).

What we are and are not saying. Not that fine-tuning never damages a pre-trained capability, but that across 32 screened cells we could not construct one instance that survives its own definition. These benchmarks are the wrong place to look: on no cell does the advantage survive every admissible rung of an eight-baseline ladder, and on the seven that clear our ridge gate, fine-tuning on 500–1,000 samples usually improves on it (§8, Appendix A29.1).

6 Experiment 3: Can Measured Drift Replace the Frozen Control?

This is the section the paper's empirical claim is argued in. Neither CKA nor weight drift is manipulated by that design (§3), so both enter here only as candidate *predictors* of Δ_{enc}, and **across the cells we evaluate, the former is not an actionable substitute for the latter.**

Drift and outcome come apart inside a single backbone: within our Moirai matrix, the strongest form of the finding. Beyond the sweep of §5.2: within Moirai-Base the two *least* drifted cells are the only two that degrade (ETTh1 at CKA 0.755 and 0.669, $+22.8\%$ and $+39.5\%$) while its two most drifted improve the most within the arm (ETTh2 at CKA 0.460 and 0.396, -31.4% and -36.3%; the whole eight-cell ordering is Appendix Table 35); and inside Moirai-Large's LoRA arm the learning rate, not the rank, decides the sign at near-identical CKA (0.973 against 0.989; Appendix A10). Eight cells are a *counterexample* to "low CKA means damage" and nothing more, with no reverse relationship claimed from n=8.

Near-identical drift, opposite consequences. Against the fitted baseline none of the ten non-Moirai cells gate-passes, nor does the eleventh we re-ran to check: Chronos on M4-Monthly, which scores $+0.092$ once the gate is computed as defined (Appendix A19). So what the two extra backbones

supply is architectural diversity and the dissociation itself, **not a controlled replication**: we read them qualitatively and never pool them (§8). Four of the five Chronos-T5-Small cells restructure to a degree that would, on the usual reading, be called catastrophic (CKA $\in[0.092, 0.169]$, at or below the 0.215 ± 0.054 an *untrained* encoder of this architecture scores; Appendix A11), yet outcomes across the five span -41.4% **on ETTm2 to** $+20.6\%$ **on ETTh1**, a 62-point spread in which ETTh1 is the *least* drifted and the only one that degrades; the four TimesFM-2.5 cells sit at milder drift and all improve (Appendix Table 7).

Why within a backbone and not pooled. Pooled across all 31 cells CKA does correlate with the intervention (ρ=$+0.567$, clustered CI $[+0.239, +0.828]$), which is not a small number and not one a practitioner can use: backbone and protocol are known before any CKA is computed, so the diagnostic has to order the outcome *given* them rather than pool three backbones and report the gap between two of them (Figure 1b; Appendix A24). Measured that way, within Moirai, the only arm with enough cells to ask, ρ=$+0.168$ with clustered CI $[-0.469, +0.717]$: an interval that admits a moderate positive relationship, and that **does not provide a reliable, actionable ordering** of our cells. Priced as a decision, freezing where drift is largest, as prescribed, is worth -6.97 pp $[-10, +14]$ against always adapting; the orientation that pays out of sample is the inverse (Appendix A28.1). Nor does the second diagnostic §3 defines rescue it: pooled ℓ_2 weight drift gives ρ=$+0.118$ with a CI including zero, and nor do five further geometric measures (Appendices A24, A25). The same dissociation holds *inside* one encoder: the two unfreeze depths we ran at 10 seeds each (the top 3 of 6 layers, and all 6) are indistinguishable on the task at CKA 0.716 against 0.465 (Appendix A18).

We find no evidence that CKA improves prediction beyond backbone identity. With the ladder and deciding statistic fixed in advance (`results/preregister_loco.json`) we asked out of sample whether measuring CKA helps a reader who already knows the backbone, the horizon and n. It does not: leave-one-cluster-out R^2 over the 15 clusters rises from $+0.000$ to $+0.137$ on backbone identity and $+0.201$ with horizon and $\log n$, then *falls* to $+0.176$ once CKA is added (ΔR^2=-0.025; no help in 69% of cluster-bootstrap draws), and the backbone-specific-slope variant is worst of all at $+0.019$. So the predictable part of Δ_{enc} is predictable from protocol identity, though the slope itself, $+43$ pp per unit CKA with a cluster interval of $[-31, +107]$, makes this a null in *prediction* rather than evidence of no relationship (Appendix A28).

7 Experiment 4: Does the Screen Work Prospectively?

7.1 Held-Out Scoring Is a Precondition, Not a Refinement

Scoring Δ_{enc} on the selection windows instead **reverses its sign in 4 of the 31 intervention cells**, and *every one of the four reverses in the same direction*, toward full fine-tuning (Appendix Table 6). Two of the four are among the seven gate-passing cells, so this is not confined to cells nobody would have looked at.

7.2 A Pre-Registered Prospective Test at Zero Prevalence

Everything above is retrospective, so we tested ours forward: predictions for eight unrun cells were committed to git before any outcome run for them existed, and the scoring script decides nothing (Appendix A7). **The outcome the gate was registered to predict never occurs.** None of the eight cells degraded; the gate flagged all eight, giving **TP 0, FP 8, FN 0**: precision 0.00, while **sensitivity and specificity are both unestimable here**. **That makes this a test at zero prevalence, and the prevalence bounds what it can say**: it establishes that the gate's positive predictions are wrong here (at the 0.20 operating point and at every threshold we swept; Appendix A7), and it establishes nothing about the gate's *sensitivity*, which an empty positive class cannot estimate. **A second registered batch, graded on the corrected gate, does estimate the rate batch 1 cannot.** On the 9 of its 10 registered cells the deadline allowed, it fires on 2, none of the 9 degrades, and so **specificity is** 0.78 with sensitivity still unestimable; the deferred cell is gate-failing and had no outcome when it was cut, so running it could only have raised that figure (Appendix A7.1).

What survives. Not a validated detector of preservable capability (eight prospective predictions, eight wrong), but an operational conditioning device: it decides where the harm question is askable, and a cell that fails it cannot supply evidence about damage to pre-trained features. 24 of our 31 fail it.

7.3 A Pre-Registered Positive Control: Near-Orthogonal Drift, and the Capability Survives

A diagnostic cannot be shown to fail at ordering damage if there is no damage to order, so we tried to build the damage. Task A is the strongest capability the screen admits (Moirai-Small/ETTh2, h=192, V_{ridge}=+0.932); task B is a synthetic series, fixed by hash before any run, whose correct forecast is the *negation* of the seasonal-naive prior that is task A's own strongest admissible rung. We fine-tuned on B and measured A over the registered lr $\in \{10^{-4}, 10^{-3}, 10^{-2}\}$ crossed with both conditions at 3 seeds, with all three outcomes and the destruction threshold committed before the first run (Appendix A30).

The attempt buys a sharper dissociation than the benchmark cells can supply. Task B was learned in **21 of 21** runs (minimum $+56.6\%$ against a registered 20%), so this is genuine conflicting learning and not weight thrashing. At the top rung the representation is driven almost to orthogonality (CKA on task A's inputs 0.063, ℓ_2 weight drift 2795, against 0.999 and 7.2 at the gentlest rung), and task A gets *better*, by $23.3 \pm 2.0\%$ in every seed; across all 21 runs the correlation between CKA and retention is ρ=+0.466, the wrong sign. **A representation can be moved most of the way to orthogonality by training measurably in conflict with a capability, and the capability survives.**

The pre-registered outcome that fired is therefore the third: destruction was not achieved. The registered rule was retention$_A \geq 50\%$ in a majority of seeds under full fine-tuning; the maximum over all 21 runs was $+26.7\%$, in one seed, against an evaluator noise floor of 0.48 pp.

8 Limitations

Full detail is in Appendix A2; broader impact is in the Ethics Statement. (1) *A null result is not a proof of absence, and the screen is a low bar.* What would settle it is a cell with a *large* pre-trained margin over a *strong* baseline, and that is the combination these benchmarks do not supply (§4, Appendix A20). The drift analyses run on the 20-cell union spanning all three backbones, but the seven-survivor agreement in §5.3 rests on one backbone and two series (Appendix A2). (2) *The frozen encoder controls encoder weight updates, not representations* (§3); how much of Δ_{enc} rests on where that boundary is drawn is quantified in §5 and Appendix A8. (3) *Seed counts are small:* 3 to 10 per cell, so most per-cell calls are inconclusive by construction, and two of the seven survivors would need 182 and 212 paired seeds to resolve their observed effect (§5.3). (4) *Horizon confounds the cross-backbone comparison.* The non-Moirai arm runs at h=24 and the Moirai arm at $h \in \{96, 192\}$, on different normalisations; we report them separately and never pool them. (5) *A retracted axis.* With the probe diagnostic withdrawn (Appendix A29.4) we say whether restructuring was necessary, not what it changed.

9 Conclusion

We set out to measure how much fine-tuning damages a foundation model's pre-trained capability, and on these benchmarks it is mostly not *measurably* there to damage: no cell beats every admissible rung of an eight-baseline ladder, and no cell meets our prespecified criterion for harmful encoder adaptation: at these seed counts 2 of 31 favour freezing at a multiplicity-adjusted q<0.05, and both are cells the screen rejects. The reason no cheap diagnostic rescues this is structural rather than statistical. CKA tells us that the representation changed; only the intervention (fine-tune, then fine-tune again with the encoder frozen) tells us whether that change was valuable, and **across the cells we evaluate the former is not an actionable substitute for the latter**. The claim we would defend outside this setting is that methodological one: **a post-treatment similarity statistic cannot be assumed to identify the value of a treatment that changed the representation.** That is a falsification and not a discovery, and here we can price it: the best threshold misplaces 5 of the 31 cells – chosen on those same cells, so a lower bound and not a validated classifier.

The advice therefore inverts the literature's. Check that the pre-trained features beat a baseline fitted on your own data before spending effort on LoRA, EWC or L2-SP, whose representation preservation is no guide to their utility (§5.2); if they do, settle the question against a frozen encoder scored on windows you did not select on, in that order, because **a forgetting result measured on cells with no measurable pre-trained advantage is a statement about the benchmark**.

ETHICS STATEMENT

Data. Every experiment uses public benchmark data and involves no human subjects and no personally identifying information. The forecasting datasets (ETTh1, ETTh2, ETTm2, Weather and Electricity) are standard public releases of instrument and utility measurements. The ILI cell uses public aggregate influenza-surveillance counts, not individual health records. The IoT negative control uses N-BaIoT, network traffic captured from a laboratory testbed of consumer devices rather than from a deployed network. We introduce no new data collection.

The main risk in this paper is that its negative result is read too broadly. We report that on these benchmarks we could not construct a cell in which fine-tuning demonstrably damages a pre-trained capability, and that a diagnostic we proposed for predicting such damage has no positive instance to predict. That is a statement about six standard forecasting benchmarks and three backbones, and §8 states what it does not license. It is *not* evidence that fine-tuning is safe in general, and a practitioner who reads it as permission to skip the frozen-encoder control would be drawing the opposite of our recommendation: our prescription is to *run* that control, because we found the observational shortcuts unreliable. Domains where a pre-trained checkpoint holds a large, well-established advantage (which these benchmarks do not, on 24 of the 31 cells we screened) are exactly where the question remains open.

Scope of the claims: one adaptation protocol, not fine-tuning in general. Three facts bound every number in this paper, and we state them together here rather than leaving them to be assembled from three appendices. *(i) The training objective differs by backbone, and one of the three is not native.* Moirai is fine-tuned through its own probabilistic NLL, and TimesFM through its own point head's loss (MSE on the mean channel plus pinball on the nine decile channels), with no head attached to either. The five Chronos-T5-Small cells in the main matrix are instead fine-tuned through an *added* MSE regression head on mean-pooled encoder features, which bypasses the tokenised cross-entropy Chronos was pre-trained with. That mismatch is why Chronos's per-condition forgetting is flagged confounded in the code and excluded wherever a forgetting comparison is made (Appendix A26), and why the M4-Monthly arm, which *is* fine-tuned through the native cross-entropy, is reported separately (Appendix A19) instead of pooled with those five. *(ii) The outcome is one metric on one channel.* Every forecasting result is MSE on a single target channel at a fixed horizon; we test no multivariate-target, multi-task or classification adaptation beyond the IoT negative control. *(iii) One adaptation step, not a task sequence* (§2). "Encoder adaptation" in this paper therefore means what this protocol does, and our results say nothing about a backbone fine-tuned through its own decoder on a task we did not run.

What we do not claim. Five readings of this paper's negative results are ones we explicitly decline. *(i)* That CKA fails to measure representation change: it does measure it, and we use it as a measurement throughout — what we test is whether that measurement substitutes for an intervention. *(ii)* That CKA can never predict downstream utility. We report that it does not reliably order the intervention outcome across the cells we ran, with the one backbone family large enough to condition on carrying the cleanest evidence; a larger matrix, another family or another protocol could show otherwise. *(iii)* That fine-tuning never causes catastrophic forgetting, or that fine-tuning is safe. We could not construct a cell on these benchmarks in which it demonstrably destroys a demonstrated capability, which is a failure to demonstrate and not a demonstration of absence. *(iv)* That freezing, L2-SP, EWC or low-rank adaptation are useless. Our prescription is to *run* the frozen-encoder control rather than to skip it, because it is the observational shortcuts we found unreliable, not the mitigations. *(v)* That foundation models are useless for forecasting, or that these results establish the absence of encoder-adaptation effects in time-series foundation models in general: seven of 31 cells cleared a deliberately weak screen, and the domains where a pre-trained advantage is large and well established are exactly where the question stays open. What we do claim is narrower than any of those: **within the intervention cells we evaluated, and with the cleanest evidence inside the Moirai family, CKA is not an actionable substitute for the intervention, and no cell meets our prespecified criterion for harmful encoder adaptation.** Appendix A1 tabulates every claim we do make with its evidence and its strength.

The value gate is a screen, not a verdict on a checkpoint. A cell that fails the gate is one where the pre-trained model does not beat a linear regression fit on that cell's own target data; it is *not* a cell we claim holds no pre-trained knowledge, and V_b is a property of the (checkpoint, dataset, baseline) triple

rather than of the checkpoint (§3). The screen's role is to decide where the question "was something valuable destroyed?" is even askable. Two things follow that are easy to misread. Failing cells here fail *widely*: every failing Moirai cell is below its baseline rather than short of the 0.20 threshold, so the emptiness of the screen is not an artefact of where the bar sits. And gate-failing cells are still measured and still reported — CKA, Δ_{enc} and the paired intervals are computed on all 31 — because the dissociation between drift and outcome is a claim about the measurement, not about the survivors.

Broader impact. This is a diagnostic study on public forecasting benchmarks, and the effect we intend is to raise the bar for claims about forgetting in time-series foundation models: a paper reporting that fine-tuning damaged pre-trained features should say what baseline established that the features were valuable, and how that baseline was fit. Our own screen is the worked example of why: the earlier version of this work reported eight degradation cells on a baseline a constant predictor beats. The controlled comparison remains the way to answer the question for any particular cell, and §5.3 claims only that the harm cannot be demonstrated on these benchmarks.

Research integrity. This version corrects an earlier version of the same work in two ways that changed its conclusions rather than its presentation, and we flag both rather than absorbing them silently. An earlier version's value gate used an unfitted per-window trend extrapolation as its linear baseline; a constant predictor beats that baseline, and correcting it changes the gate status of 17 of the 21 Moirai cells and removes every degradation instance the paper had; that is, the entire class of cell the diagnostic was built to predict. Separately, we retract a linear-probe diagnostic reported earlier: the statistic it rests on is non-zero only where the probe fails to generalise, and Appendix A15 reports the measurement that establishes this on the same encoders. Neither correction required re-running a model; both are documented with their audit trails in Appendix A29.4 and Appendix A29.2. Where a number in an earlier version could not be traced to a run record, we say so at the site rather than reprinting it.

Compute. The experiments are small by contemporary standards: no run exceeds 10,000 training windows, and the sweep reported here runs on a single GPU or on Apple silicon. We report the protocol deviations forced by the hardware available (Appendix A18) rather than presenting the runs as environment-independent.

REPRODUCIBILITY STATEMENT

The paper is organised so that every quantity can be traced to a stored run record, and we have tried to make the failure modes we hit ourselves cheap for others to avoid.

Where the numbers come from. Every table the source `\inputs` is written by a named emitter that reads the per-run JSON records (none is assembled by hand), and `scripts/rederive_all.sh` runs all of them in one command and then reports which generated files changed. A clean run means the PDF agrees with the run records; any diff is a number that was stale. There is one exception and it is marked where it appears: the two inline tabulars in §A29.4 transcribe a *withdrawn* computation whose probe records are too sparse to re-derive, and they support no claim in this paper. The per-run records carry the fields the tables use (`final_cka`, `forgetting_pct`, `final_val_mse`, `final_weight_drift` and the early-stopping block), together with the seed, epoch budget and sample count, so the dispersion conventions can be recomputed rather than taken on trust.

What "reproducible" means here, exactly, and what it does not. Auditability is not independent reproduction, and we distinguish three tiers rather than letting the stronger word cover the weaker work. *(i)* Every table and figure is re-derivable from a clean clone with no data and no checkpoints, because the run records are in the release; that is what `rederive_all.sh` checks. *(ii)* Re-running the value gate or any fine-tuning cell additionally needs the benchmark CSVs, which are public but not ours to redistribute. The obstacle there is not the download but knowing you fetched the same file: ETT and Weather circulate in variants that differ in column order and date parsing, and a variant with the same name changes every window and so every MSE. `results/data_manifest.json` therefore records the SHA-256, shape and date range of each CSV we used, and `scripts/data_manifest.py -check` verifies a local copy against it. *(iii)* The drift-metric battery's geometry pass needs the fine-tuned encoders themselves, which are excluded for size; its stored output is released, so the table is re-derivable even though the measurement

behind it is not. **No third party has run any of this.** What we offer is re-derivation from committed run records and a bounded two-cell from-scratch check, not independent replication.

What the two-cell check found, and why we report it rather than a tolerance. `scripts/rerun_two_cells.sh` retrains one Moirai cell (Small/ETTh2 h=96, n=500, seed 42, full fine-tuning (B)) and one non-Moirai cell (TimesFM/ETTh1 h=24, n=1,000, seed 42, the same condition) from the raw CSVs and diffs every stored field. **TimesFM reproduces exactly**: all nine fields, including CKA and weight drift, are bit-identical. **Moirai reproduces its MSEs to within a few percent but not its forgetting percentage**: zero-shot MSE moves -0.8%, held-out MSE $+0.8\%$, the selection-window MSE $+3.2\%$, CKA -0.6% (0.957 to 0.952), and weight drift $+25.2\%$, while forg.$_{\mathrm{B}}$ moves from $+6.81$ to $+11.14$, a 63.5% relative change. That last figure is the informative one and it is not a failure of the smaller ones, it is those smaller ones compounded: forg.$_{\mathrm{B}}$ is $(\mathrm{MSE}_{\mathrm{val}} - \mathrm{MSE}_{\mathrm{ZS}})/\mathrm{MSE}_{\mathrm{ZS}}$ on two nearly equal quantities, so a -0.8% move in the denominator against a $+3.2\%$ move in the numerator's larger term is amplified into $+63.5\%$ in the ratio. This is a property of the percent-of-zero-shot scale rather than a fact about this cell: wherever the fine-tuned and zero-shot losses are close, a ratio of their difference to one of them magnifies absolute moves that are themselves small, which makes forg.$_{\mathrm{B}}$ the least stable quantity this paper reports. Δ_{enc} **does not inherit that instability**: both arms share the same zero-shot denominator, so Δ_{enc} is the B-versus-D contrast rescaled per cell (Equation 1), and a shared positive rescaling cannot change its sign, which is what every claim in the paper is stated in terms of. Moirai scores the median of 20 sampled forecasts and TimesFM emits a point forecast, which is why only one of the two is reproducible to the bit. Every sign-bearing field kept its sign, which is what the paper's claims are stated in terms of, but a reader should treat any single-seed Moirai forgetting figure as carrying at least this much run-to-run variation, and that is why the paper reports cells at 3–10 seeds with a dispersion and never a single run.

Two things are required to reproduce the Moirai runs at all. First, the `uni2ts` `PackedStdScaler` performs an in-place write that silently detaches the gradient tape for losses depending on scaled statistics; without the patch of Appendix A3 training can appear to run while not converging. Second, the frozen-encoder condition is not a strict freeze (the input projections and mask encoding remain trainable, which is why its CKA is not 1.000), and the strictly-frozen variant is reported separately (Appendix A8). Comparisons against a "frozen" baseline are not comparable across papers unless which parameters were frozen is stated.

Scoring protocol. All headline comparisons are scored on windows disjoint from the ones used for selection. This is not a refinement: §7.1 shows that scoring on the selection windows instead reverses the sign of the frozen-encoder comparison in 4 of the 31 intervention cells, always in the same direction. Reproducing our numbers therefore requires reproducing the split discipline, and Appendix Table 6 gives the per-arm decomposition needed to check which split a given number was scored on.

Pre-registration. The prospective test of §7.2 was registered before any outcome run for those cells existed: `results/v47_prospective/preregistration.json` for the first batch and `results/v57_prospective3/preregistration_v3.json` for the second, each committed before the first condition-B or condition-D run of its own arm was launched. The scorer, `score_prospective.py`, reads those files and the finished runs and makes no decisions of its own. All are in the release, and each registration's commit precedes every run it scores. **That ordering is checkable in the version-control history, and the supplementary archive does not carry one.** What travels with the archive is each registration's own `written_utc` and the commit hash it was written at; the per-run records store no timestamp, so the archive alone cannot order them against it. We state the limit rather than let "checkable" cover a check the archive does not support: from the archive a reader has the hashes and our word, and the ordering itself is verifiable against the repository released with the paper. The Chronos/M4-Monthly re-run of Appendix A19 carries a second registration (`results/chronos_m4/preregistration.json`), on the same terms and with the same check available. That arm is *not* prospective (the superseded runs' outcome was already known), so what its registration fixes is only what was still fixable: the gate's estimator and split, the seeds, the decision rule, and what each gate outcome would be taken to mean. The gate was computed after that file was committed, which matters because the estimator was itself one of the things being corrected. The N=4 and N=5 rungs of Appendix A18 carry a third (`results/v49_layerunfreeze_10seed/preregistration_ladder.json`), committed before the first of their six runs. It is also not prospective, and what it fixes is the rungs,

the seeds, the condition and flag, the reported statistic, and **what each outcome would mean for the body**: that a non-monotone forgetting profile across depth, or a thin rung landing outside the matched pair's range, would be reported as found with the body's claim narrowed to the matched pair rather than the ladder being dropped. Both of those outcomes occurred, and that is the rule we followed.

The figures check themselves. Both generated figures read their values from `scripts/cell_matrix.py` and *assert* at draw time every count the body quotes from them: the number of gate-passing cells, the number of degradation cells, the number of sign reversals and the sign of each correlation. A run record that moved would fail the figure rather than change it silently, which is the one staleness path the table emitters do not cover. (This sentence used to sit in the caption of Figure 1; it is a statement about the release, and the body has no page to spare for it.)

What we release. The analysis and experiment scripts, the per-run JSON records behind every table, the emitter for every table and the one-command re-derivation script, the benchmark-data manifest, every pre-registration file, and the LaTeX source and style files needed to build this paper. Model checkpoints are excluded for size; the appendices state which runs retained a checkpoint, because two re-measurements we would otherwise have made were blocked by runs whose checkpoints were not kept (Appendix A16).

Use of AI Assistants

An LLM-based coding assistant was used in this work: **a large fraction of commits in this project's version-control history carry an AI co-author trailer**.

Where it was used. Implementation and prose. It wrote the initial draft of most of the analysis and experiment scripts, including the table emitters, the one-command re-derivation sweep, the prose-number checker, the figure scripts, and the anonymity and bundle-inventory gates; and it drafted and edited some of the sections of the paper's text.

Where it was not used. It produced no experimental result. Every number in this paper comes from a run that executed and wrote a JSON record, and the registered checks in `scripts/check_paper_numbers.py` compare the prose against those records, never against anything a model asserted. The research question, the study design, the pre-registered predictions and decision rules, the choice of what to cut and what to report, and every claim the paper makes are ours.

Responsibility, and one caveat. We have reviewed all AI-assisted work in this paper, and we take full responsibility for everything in it, errors introduced during AI-assisted work included. Such errors did occur, and the two kinds we hit repeatedly are worth naming: tables whose numbers had been typed by hand and then drifted from the records they were supposed to summarise, and claims that were reworded in one place while a restatement elsewhere kept the superseded figure. Neither was caught by rereading the paper; each was caught by a mechanical check. That is the reason having a verification apparatus is critical, and it is the part of this methodology we would recommend to anyone writing a paper this way.

References

Anders Andreassen, Yasaman Bahri, Behnam Neyshabur, and Rebecca Roelofs. The evolution of out-of-distribution robustness throughout fine-tuning. *Transactions on Machine Learning Research (TMLR)*, 2022. URL https://arxiv.org/abs/2106.15831.

Abdul Fatir Ansari et al. Chronos: Learning the language of time series. *Transactions on Machine Learning Research (TMLR)*, 2024. URL https://arxiv.org/abs/2403.07815.

Abhimanyu Das et al. A decoder-only foundation model for time-series forecasting. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2310.10688.

Mononito Goswami et al. MOMENT: A family of open time-series foundation models. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2402.03885.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 4129–4138, 2019. doi: 10.18653/v1/N19-1419.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.01728.

Prannay Khosla et al. Supervised contrastive learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. URL https://arxiv.org/abs/2004.11362.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning (ICML)*, pp. 3519–3529, 2019. URL https://arxiv.org/abs/1905.00414.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2202.10054.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018. doi: 10.1109/TPAMI.2017.2773081.

Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: Generative pre-trained transformers are large time series models. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2402.02368.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*, 2023.

Yair Meidan, Michael Bohadana, Yael Mathov, Yisroel Mirsky, Asaf Shabtai, Dominik Breitenbacher, and Yuval Elovici. N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Computing*, 17(3):12–22, 2018. doi: 10.1109/MPRV.2018.03367731.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4609–4622, 2020. doi: 10.18653/v1/2020.acl-main.420.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, Marin Biloš, Sahil Garg, Anderson Schneider, Nicolas Chapados, Alexandre Drouin, Valentina Zantedeschi, Yuriy Nevmyvaka, and Irina Rish. Lag-llama: Towards foundation models for probabilistic time series forecasting. *arXiv preprint arXiv:2310.08278*, 2024. URL https://arxiv.org/abs/2310.08278.

Michał Wiliński, Mononito Goswami, Willa Potosnak, Nina Żukowska, and Artur Dubrawski. Exploring representations and interventions in time series foundation models. In *International Conference on Machine Learning (ICML)*, 2025. URL https://arxiv.org/abs/2409.12915.

Gerald Woo et al. Unified training of universal time series forecasting transformers. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2402.02592.

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. doi: 10.1609/aaai.v37i9.26317.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained LM. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. doi: 10.52202/075280-1877.

APPENDIX

Notation. The appendices write the paper's estimand in its condition form, B−D, because they compare conditions directly (for instance B−D against B−H). It is the same quantity the body calls Δ_{enc}; the two are interchangeable throughout.

A1 WHAT THIS PAPER CLAIMS, AND WHAT IT DOES NOT

Three claims of unequal strength are made in this paper, and they are not interchangeable. **C1** is a null in *prediction*, established out of sample under an analysis registered before it was run. **C2** is a positive claim about what the screen does, at an operating point we chose rather than derived. **C3** is methodological, and it is the only one of the three we would defend outside these benchmarks.

A2 EXTENDED LIMITATIONS AND FUTURE WORK

(1) Single-backbone scope of the dissociation claim. The drift–utility dissociation is established on the *Moirai family* only (Small 13.8M, Base 91M, Large 311M across ETTh1, ETTh2, ETTm2). We attempted three further backbones as second-model controls across three ETT datasets (Table 2).

All nine backbone×dataset cells gate-fail (ZS under-performs the Linear baseline by 14% to 148%), consistent with ETT-family distributions being outside the pre-training coverage of these backbones at these sizes. The Chronos capacity sweep (Chronos-T5-Small/Base/Large on ETTh2) confirms the gap is size-invariant: scaling from Small (210M) to Large (710M) yields no improvement ($-42.6\% \rightarrow -34.3\% \rightarrow -35.2\%$). TimesFM-2.5-200M and MOMENT-1-base gate-fail across all tested datasets. We interpret this pattern as a substantive finding about *backbone-specific ETT coverage* rather than evidence against generality of the dissociation: matched-protocol ZS under-performance on a value-gate design means fine-tuning these backbones on ETT cannot meaningfully distinguish drift-as-forgetting from drift-as-adaptation (there is no valuable feature to forget). A gate-passing second backbone on a different dataset (Traffic, ILI, Exchange, or a model with explicit ETT pre-training coverage) remains the most direct path to a multi-backbone replication and is the highest-priority future-work direction.

(2) Sample-sweep depth and seed count. The central sample-size sweep (Table 4) covers ETTh2 at $n \in \{500, 1\text{k}, 2\text{k}, 5\text{k}, 10\text{k}\}$. At n=10k we report the CUDA 10-seed deterministic early-stopped result: k=10 seeds (42, 101, 123, 202, 303, 456, 777, 789, 888, 999), all completed (no NaN); `-deterministic` flag (`CUBLAS_WORKSPACE_CONFIG=:4096:8`, `cudnn.deterministic=True`, seeded DataLoader, PYTHONHASHSEED). Forg. $-5.3\pm6.6\%$, 7/10 negative; mean best epoch 4.2 ± 1.3. *Dispersions in this item are standard deviations over seeds (ddof=1), matching Table 4 rather than the body's SEM convention (§3), because the four protocols compared below differ in seed count and the comparison is about their spread.* The Ridge $\Delta R^2 = +0.67\pm0.24$ (**10/10** positive) that an earlier version reported alongside these is **retracted** (§A29.4) and is quoted here only as a record of the withdrawn computation; the forgetting and CKA columns are unaffected. The three near-zero positive-forgetting seeds (123: $+1.8\%$, 456: $+4.4\%$, 789: $+0.4\%$) follow the n=5k overshoot pattern (Fig. 3, right panel). For comparison, MPS early-stopped k=10 gave $-5.7\pm10.4\%$ (8/10 neg, `v18_mps_deterministic_n10k`); a second early-stopped run (k=5) gave $-11.7\pm5.3\%$ (all 5 negative, `v17_etth2_n10k_es`); the final-epoch runs gave $+10.6\pm9.8\%$ on k=3 (`v16_etth2_n10k`), which is the sign fragility §5.2 reports. *That last record does not isolate early stopping as the cause, and we do not read it that way*: it differs from the reported reading in four respects at once: its three seeds (101, 456, 789) are a subset of the ten that happens to contain two of the three most positive; it trains for 20 epochs rather than 10; it takes its percentage against a different zero-shot reference (0.1325 against 0.1266); and it reads the final epoch rather than the best. Choosing a different early-stopped baseline reverses which factor looks dominant (seed-matched, the CUDA record reads -2.0% and the MPS one $+4.1\%$), so the pair bounds how far this cell's sign moves across protocols and attributes it to nothing. Moirai-Base is now evaluated at both n=500 and n=10k (10 seeds each); full n-sweep on Moirai-Large is future work.

(3) Probe-head capacity. Ridge and MLP absolute R^2(FT) remains negative on the 96-step-ahead head at all sampled n, which is exactly why the ΔR^2 axis is **retracted** (§A29.4): relative movement

Table 1: The paper's three claims with their evidence and their strength, and then the three readings we do *not* claim. The rows are ordered by how far each travels: C1 holds for the cells we evaluate, C2 for the screen at the threshold we state, C3 for any study that reads a post-treatment similarity statistic as evidence about a treatment's value. The three N rows are not hedges: each is a proposition this paper's own evidence is insufficient to support, and two of them would be supported by results we report if the scope restriction were dropped.

	Claim	Evidence	Strength and scope
C1	CKA magnitude does not identify whether encoder adaptation contributed, in the cells we evaluate.	within Moirai, the only arm with enough cells to ask, ρ=+0.168, clustered CI $[-0.469, +0.717]$, and no cut on CKA separates the cells encoder adaptation helped from the ones it hurt (§6). Out of sample, with the ladder fixed in advance, leave-one-cluster-out R^2 over the 15 clusters *falls* when CKA is added: it *lowers* R^2 from $+0.201$ to $+0.176$ (Appendix A28).	A null in *prediction*, not evidence of no relationship. The in-sample slope is $+43$ pp per unit CKA with a cluster interval of $[-31, +107]$, which admits a moderate positive relationship. Scope: the 31 intervention cells; the within-backbone form is Moirai only.
C2	The screen locates cells where no advantage over a fitted baseline is measurable under a definition we state.	only 7 of 31 cells show a pre-trained advantage over a lookback-96 ridge *fit on the target data* (§4), and no cell beats every admissible rung of the eight-baseline ladder (Appendix A20).	Positive, but at a prespecified operating point of 0.20 we chose and did not derive, and V_b is a property of the model–baseline pair rather than of the checkpoint (§3). A cell the screen fails is one where *this* ladder finds no advantage.
C3	A post-treatment similarity statistic cannot be assumed to identify the value of a treatment that changed the representation.	Structural rather than statistical: the statistic is measured after the treatment and on the object the treatment altered (§3). Our results are consistent with it; they are not what establishes it.	The one claim we would defend outside these benchmarks. It licenses no threshold and predicts no effect size: it says a similarity statistic is the wrong instrument for this question whatever its value.
N1	*Not claimed:* that drift is generally unpredictive across time-series foundation models.	Three backbones cannot establish it, and pooled across backbones the same correlation is in fact clearly positive, which we report rather than set aside (§6).	Would require a backbone sample this paper does not have. C1 is a statement about our cells.
N2	*Not claimed:* that fine-tuning does not harm time-series foundation models.	No cell met a degradation definition fixed in advance, under any protocol we ran; that is a statement about these benchmarks and these protocols (Appendix A21).	A zero count at a stated definition, not a safety guarantee. Loosening the definition produces degradation cells, and we show which (Appendix A21).
N3	*Not claimed:* that mitigation methods are unnecessary.	LoRA, L2-SP and EWC were run as a full spectrum on one cell only, and we report what they did there (§5, Appendices A6, A9).	One cell does not generalise in either direction. Our claim is about the *diagnostic* that would trigger mitigation, not about mitigation.

between two fits that never decode the target is not evidence about the encoder. The probe-design sweep of Appendix A16 shows only that the direction is stable across α, pooling and probe seed *within* that failed-fit regime; it does not rescue the statistic, and no claim in this paper rests on it. Probe capacity is therefore no longer a limitation of a live result but a diagnosis of a withdrawn one.

(4) Sample-sweep on other sizes. Moirai-Large is evaluated at n=500 only; full n-sweep is future work. **(5) IoT scope.** IoT is a negative control (ZS AUC 0.481, sub-random); fine-tuning replaces rather than preserves pre-trained features. We do not claim IoT is a dissociation instance, nor

Table 2: The initial non-Moirai screen, at h=96 with a lookback of 192, one seed (42) and 300 test windows. **This is a different protocol from the paper's non-Moirai screen** (Table 7), which runs h=24 at lookback 96 with three seeds and a paired frozen-encoder condition; the two are therefore not comparable cell by cell, and only the latter feeds the 32-cell count. Chronos-T5-Base/Large and MOMENT appear here and nowhere else. Advantage is V_{ridge} as a percentage. Rows trace to `results/v10_chronos_etth2.json`, `v14_*`, `v18_backbone_gate/*` and `v12_moment_etth2_gate.json`.

Backbone	Dataset	ZS MSE	Linear MSE	Advantage
Chronos-T5-Small	ETTh2	0.304	0.213	−42.6%
Chronos-T5-Small	ETTh1	0.118	0.092	−28.8%
Chronos-T5-Small	ETTm2	0.188	0.110	−70.2%
Chronos-T5-Base	ETTh2	0.286	0.213	−34.3%
Chronos-T5-Large	ETTh2	0.288	0.213	−35.2%
TimesFM-2.5-200M	ETTh2	0.242	0.213	−13.8%
TimesFM-2.5-200M	ETTh1	0.105	0.092	−13.9%
TimesFM-2.5-200M	ETTm2	0.190	0.110	−72.6%
MOMENT-1-base	ETTh2	0.528	0.213*	−148%

*MOMENT was screened at lookback 96, whose baseline is 0.235; its margin was nonetheless computed against the lookback-192 baseline 0.213, and we print the comparator actually used. Against 0.235 the margin is −125%, so the verdict is unchanged either way.

claim generalisation to other IoT distributions (N-BaIoT zero-shot transfer fails; retraining recovers discrimination, consistent with recipe generalisation, not representation transfer).

(6) Three residual unsettled items. These closed the body's limitations list until the page limit took the recap; each is stated where it arises and none is withdrawn. *(a)* The forgetting sign at n=10k is protocol-dependent: item (2) above gives the four protocols and their four signs on the same cell. *(b)* The ILI cell is small, ≈450 windows, which is why Appendix A29.3 calls it a case study and not a pillar. *(c)* Two pre-registered predictions were falsified rather than confirmed: ILI's CKA prediction of 0.90–0.95, against the measured value Appendix A29.3 reports, and the prospective value-gate predictor, which scored precision 0.00 at every threshold in $[0.10, 0.60]$ (§7.2). We report both as failures of our own diagnostic, which is what the pre-registration was for. Values are not repeated here: a recap that retypes them is how the two of them drift apart.

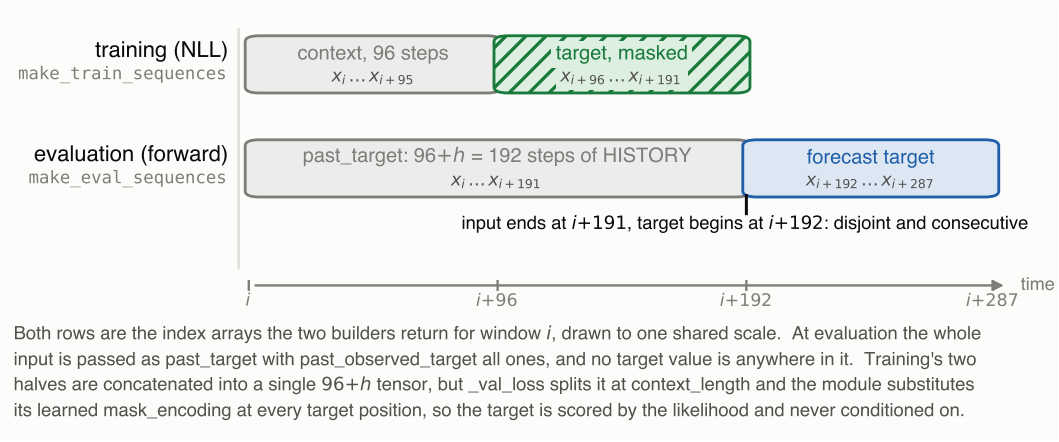
(7) Why similarity is not mechanism, in two directions. A representation can drift far and still carry the task information in a transformed basis, so a low CKA need not mean a capability was lost; and a small drift can destroy one critical direction, so a high CKA need not mean one was kept. Both directions are consistent with the positive control of Appendix A30, where CKA on task A's inputs falls to 0.063 while task A itself *improves*. Our negative claim is about *identifying* damage from similarity alone (§8, item 6), not about CKA's validity as a measure of similarity, which we do not dispute.

A3 UNI2TS PACKEDSTDSCALER PATCH DETAILS

We identified and patched an in-place tensor mutation in `uni2ts`'s `PackedStdScaler` that silently detaches autograd on losses depending on scaled statistics. The issue arises when `PackedStdScaler.transform` performs an in-place `.copy_()` that breaks the gradient tape. Our patch replaces the in-place write with a differentiable `torch.where` operation. *All* Moirai fine-tuning results in this paper use the patched scaler (released with our code); unpatched `uni2ts` runs may converge differently and we have observed silent non-convergence on `uni2ts@main`. We recommend upstreaming this patch to `uni2ts`.

A4 MOIRAI MEDIAN FORECAST SELECTION

Moirai generates probabilistic forecasts as a mixture of distributions (StudentT, NormalFixedScale, NegativeBinomial, LogNormal). The heavy-tailed components (NegBin, LogNormal) produce extreme right-tail outliers when using the *mean* of forecast samples as the point prediction; for ETTh2



Both rows are the index arrays the two builders return for window *i*, drawn to one shared scale. At evaluation the whole input is passed as past_target with past_observed_target all ones, and no target value is anywhere in it. Training's two halves are concatenated into a single $96+h$ tensor, but _val_loss splits it at context_length and the module substitutes its learned mask_encoding at every target position, so the target is scored by the likelihood and never conditioned on.

Figure 2: **Which timesteps Moirai is given, and which it is asked to predict**, at h=96; h=192 is the same layout with h substituted. Both rows are the index arrays that `make_train_sequences` and `make_eval_sequences` return for window i, drawn to one shared scale. At evaluation the model input is the $96+h$ steps *preceding* the forecast window and the target is the h steps strictly after it: the two are disjoint and consecutive, so no target value reaches the model. The figure is generated from those functions' output and asserts that disjointness at draw time.

at h=192, sample means reached 21,431 against a target maximum of 45.5, inflating MSE from 0.30 to 8.58.

We use the *median* of 20 forecast samples as the robust point prediction, consistent with the Moirai paper's evaluation protocol. This eliminates sensitivity to the heavy-tailed mixture components while preserving the distributional forecast's central tendency.

A5 ETTh2 Forecasting Details

Dataset. ETTh2 (Electricity Transformer Temperature, Hourly, dataset 2) contains 17,420 timesteps of 7 features (HUFL, HULL, MUFL, MULL, LUFL, LULL, OT) recorded hourly from electrical transformers. We use the standard chronological split: 12 months training, 4 months validation, 4 months test.

Window construction, and exactly which values Moirai is given. An earlier version of this appendix described the evaluation windows as "extended lookback sequences (context_length + prediction_length timesteps)", which reads as though the forecast target were part of the model input. It is not, and because the training and evaluation window sets differ the arithmetic is worth writing out. Let $x_1, \dots, x_T$ be the series and h the horizon. Both window sets are built by two functions in `scripts/finetune_forecasting.py`, named here so that this paragraph and the code can be checked against each other.

Training (`make_train_sequences`): window i has context $x_i, \dots, x_{i+95}$ and target $x_{i+96}, \dots, x_{i+95+h}$. The two are concatenated into one $96+h$ tensor and passed to Moirai's `_val_loss` with fixed patch size 32, which is the interface's required calling convention: `_val_loss` splits the tensor at `context_length` and marks the latter part with `prediction_mask`, and the module then substitutes its learned `mask_encoding` at every masked position and zeroes the corresponding observed flags. The target values are therefore scored by the likelihood and never conditioned on.

Evaluation (`make_eval_sequences`): window i has input $x_i, \dots, x_{i+95+h}$ (that is, $96+h$ steps of *history*), and target $x_{i+96+h}, \dots, x_{i+95+2h}$. The input's last index is $i+95+h$ and the target's first is $i+96+h$, so the two windows are disjoint and consecutive and no target value appears anywhere in the model input. Only the input is passed, as `past_target` with `past_observed_target` all ones; the point prediction is the median of 20 forecast samples with `patch_size='auto'`. "Extended lookback" thus means a longer *history*, not history-plus-target. Figure 2 draws both layouts

from the index arrays those two functions actually return rather than from this description, and asserts the disjointness at draw time, so the figure cannot drift from the implementation.

The one asymmetry this does introduce is between models: Moirai is given $96+h$ steps of history where the fitted baseline of Appendix A29.2 is given 96. That direction favours the pre-trained model (more context lowers MSE_{ZS}, raises V_b and so admits *more* cells), and Appendix A20.4 measures it rather than arguing it, by giving the baseline the same $96+h$ history.

All data is fed to Moirai in raw scale (Moirai's internal PackedStdScaler handles instance normalization); predictions and targets are then normalized using training-set mean and standard deviation for MSE/MAE computation, matching the standard time-series forecasting evaluation protocol.

Condition C's objective and its contrastive labels. Condition B fine-tunes on NLL only; condition C adds a supervised contrastive term Khosla et al. (2020), $\mathcal{L}(\theta) = \mathcal{L}_{\text{NLL}}(\theta) + \lambda\,\mathcal{L}_{\text{SupCon}}(\mathbf{z}, \mathbf{y};\, \tau)$, with $\lambda = 0.5$ and $\tau = 0.07$. Condition C appears in this paper only as a protocol variant of B and carries no claim; the comparison every claim turns on is B against D. For the contrastive labels we divide training sequences into 8 temporal clusters based on their position in the time series. Sequences within the same cluster are positive pairs; sequences from different clusters are negatives. This encourages the encoder to maintain temporal-regime awareness.

Horizon selection. We evaluate h=96 and h=192, the two horizons at which Moirai-Small on ETTh2 clears the value gate: measured against the fitted lookback-96 baseline the gate is defined with (Appendix A29.2), V_{ridge}=$+0.400$ at h=96 and $+0.464$ at h=192 on held-out windows. Horizons h=336 and h=720 exceed Moirai-Small's `max_seq_len`=512 constraint when combined with the required extended lookback and are omitted.

A6 ETTh2 Drift Diagnosis and Mitigation (Full Tables)

This section contains the full tables referenced in §4. Table 8 shows the mitigation spectrum, conditions B through H on Moirai-Small/ETTh2 at both horizons. An earlier version also printed a three-row B/C/D subset of that table separately, with a caption asserting that Moirai's 28% and 45% zero-shot margins over "Linear" confirmed the pre-trained features were valuable; those margins were measured against the superseded trend baseline (Appendix A29.2), and the subset is dropped rather than recaptioned since every row it carried appears here.

Table 4 gives the sample-size sweep on ETTh2 (h=96, condition B) referenced in §4. The three linear-probe columns it used to carry are retracted, and are transcribed in §A29.4 rather than reprinted here.

The full 31-cell intervention evidence base. Table 5 lists every cell carrying a paired B/D intervention behind the headline claim of §1: 31 of the 32 screened cells, the screen itself being a gate value and needing no intervention run: three backbones, six datasets, three Moirai capacities and two horizons, each with its selection-split gate and the intervention contrast B$-$D (positive = the frozen encoder was better). It is generated by `scripts/cell_matrix.py -latex` directly from the committed run JSON; no value in it is transcribed by hand. The two protocol arms (§4 at h=96/192, §6 at h=24) are listed together for completeness but are never pooled into a single statistic.

Why the four reverse in one direction, and the two mechanisms behind it. The four reversals of §7.1 do not share one per-arm story. Normalising each arm by the zero-shot error on its own split is an identity rather than a model, so the two arms' held-out-minus-selection moves account for the reversal exactly. On **two of the four** the frozen arm collapses out of sample and the selection-optimism reading holds: Chronos/ETTh1 is the largest reversal in the matrix at $+6.8$ on the selection windows against -39.2 held out, a 46.0-point swing (Δ frozen $+53.9$ against Δ adapted $+8.0$ on Chronos/ETTh1, the largest of the four). On the **other two** both arms finish *better* than their own zero-shot out of sample and nothing degrades at all — the adapted arm merely gains more — so half the reversals are a different phenomenon wearing the same sign change. What all four share is not a mechanism but a location: they concentrate where the two splits are least exchangeable: the *untrained* checkpoint's held-out-to-selection error ratio spans 0.98 to 5.07 across these four cells, a quantity that involves no fine-tuning at all. A frozen encoder cannot adapt to that shift and a fine-tuned one partly can, which is exactly the asymmetry that makes same-split scoring flatter freezing, so the direction of

Table 3: Baseline-relative pre-trained advantage $V_{\text{ridge}}=1-\text{MSE}_{\text{ZS}}/\text{MSE}_{\text{Linear}}$ for every Moirai cell, on both splits. The *selection* columns are the paper's primary gate: they are disjoint from the held-out windows every intervention outcome is scored on, so cell inclusion is not a judgement made after seeing the outcome. The *held-out* columns are the retrospective variant, reported as a sensitivity analysis. Bold marks a pass on the *primary* split in both columns of that row, so the bold set is the one the body's counts refer to. **Not a linear probe**: this is a ratio of forecast MSEs computed through Moirai's own forecasting head, and is unaffected by the retraction of the linear-probe ΔR^2 axis. **Estimator, exactly**: MSE_{ZS} is the median of 20 Moirai forecast samples at extended lookback (context + horizon); $\text{MSE}_{\text{Linear}}$ is a single lookback-96 ridge-OLS linear map $\mathbb{R}^{96\times D}\to\mathbb{R}^{h\times D}$ *fit on the cell's training windows* and applied unchanged out of sample; both are computed on the same matched windows within a split (300 selection, 300 held-out), train-normalised. The gate involves no fine-tuning, so a cell has one value at every n: the $n=500$ and $n=10$k (ES) operating points reported in §4 share the entry shown here. Bold = gate-pass at the 0.20 threshold on the primary split. "—" = not run.

		Selection (primary)		Held-out (retrosp.)	
Backbone	Dataset	$h=96$	$h=192$	$h=96$	$h=192$
Moirai-Small					
	ETTh1	-0.177	-0.347	-0.243	-0.331
	ETTh2	$\mathbf{+0.874}$	$\mathbf{+0.932}$	$\mathbf{+0.400}$	$\mathbf{+0.464}$
	ETTm2	-0.164	$\mathbf{+0.352}$	-0.110	$\mathbf{+0.108}$
	Weather	-0.546	-0.506	-0.867	-0.639
	`Electricity7`	-0.650	-0.455	-0.085	-0.047
Moirai-Base					
	ETTh1	-0.312	-0.600	-0.458	-0.501
	ETTh2	$\mathbf{+0.777}$	$\mathbf{+0.925}$	$\mathbf{+0.361}$	$\mathbf{+0.275}$
	ETTm2	-0.257	$\mathbf{+0.211}$	-0.057	$\mathbf{+0.083}$
	Weather	-0.823	-0.633	-1.029	-1.324
Moirai-Large					
	ETTh1	-0.248	—	-0.454	—
	ETTh2	$\mathbf{+0.875}$	—	$\mathbf{+0.219}$	—
	Weather	-0.638	—	-0.621	—

Reading: seven of these 21 cells clear the 0.20 threshold on the selection windows and five on the held-out windows. The selection set contains the held-out set and adds two (ETTm2, ETTm2), so the retrospective variant is the stricter of the two here. The primary survivors span two datasets: ETTh2, ETTm2. $V_{\text{ridge}}<0$ means the linear baseline wins outright. These values supersede the ones printed in earlier versions of this table, which were computed against a per-window trend extrapolation rather than the fitted regression the gate is defined with; the substitution and its consequences are documented in Appendix A29.2. `Electricity7` is Electricity restricted to 7 of its 370 series (six `MT` series plus `OT`), a deliberate protocol deviation that keeps the cell dimensionality-matched to ETT/Weather.

the four reversals is not a coincidence to be noted but a consequence to be expected, and it is why scoring Δ_{enc} on selection windows is not a defensible option rather than merely a less careful one.

The mitigation spectrum, cell by cell. Table 8 reports five strategies spanning unconstrained fine-tuning to full freezing, all on Moirai/ETTh2. The body (§5.2) states only the conclusion; the per-method readings are here. **LoRA** ($r=8$, $\alpha=16$) improves over zero-shot at $h=96$ at CKA 0.979, but is not uniformly safe: on Moirai-Large it fails at the default learning rate ($+25.7\%$) and only a $10\times$ LR reduction rescues it (-8.5%, CKA 0.989), so tune the learning rate before the rank (Appendix A10). **EWC and L2-SP** reduce drift without buying utility: at $n=1{,}000$, $h=96$, each scored against its own zero-shot, they land at $+13.4$ to $+16.0\%$, against the $+11.1\%$ full fine-tuning reaches at $n=500$. Those two figures are not matched on training-set size (no full-fine-tuning arm at $n=1{,}000$ on this cell has more than a single seed), so what we read off them is that anchoring buys no utility, not an ordering between anchoring and full fine-tuning. **Freezing** eliminates drift but cannot adapt, and the strict freeze of Appendix A8 shows that even the residual adaptation condition D permits is doing work. The screen is what makes this spectrum hard to act on: none of these methods has a demonstrated advantage to protect on 27 of our 32 cells, and on three of the remaining five

Table 4: **Sample-size sweep on Moirai-Small/ETTh2** (h=96, condition B). Drift grows monotonically (CKA↓ as n ↑) while forgetting is *non-monotonic* and reverses sign: positive at $n \leq 1{,}000$, negative at n=2,000, positive again at n=5,000 and negative at n=10,000 (7/10 seeds negative, Wilson CI $[0.40, 0.89]$). **If drift magnitude predicted harm the CKA and Forg. columns would track; they do not**: the dissociation of §5.2 measured inside a single cell. **Dispersion here is the sample standard deviation over seeds** (ddof=1), not the standard error the rest of the paper reports (§3): what this table is about is how the spread across seeds moves with n (the n=5,000 crossover *is* a rise in seed variance), and SEM would hide that behind k, which rises at the same place.

Samples	k	MSE	Forg.%	CKA
500	3	.148±.008	+14.0±6.7	.949±.010
1,000	3	.147±.010	+13.4±9.1	.936±.025
2,000	3	.126±.008	−2.9±5.9	.814±.042
5,000	7	.137±.012	+6.6±10.0	.546±.082
10,000[cuda]	10	.120±.007	−5.3±6.6	.518±.199

[cuda] The n=10,000 row is the CUDA arm (A10G): 10 deterministic seeds, 10 epochs *with* early stopping (mean best epoch 4.2±1.3). Every smaller n is MPS, 20 epochs *without* early stopping. That is the protocol dependence §5.2 discusses; Appendix A2 (2) gives the n=10,000 seeds and the matched MPS comparisons, and Appendix A12.1 the trajectories of the 7 n=5,000 seeds. Rows read `final_val_mse`, `forgetting_pct` and `final_cka` from the per-run records in `results/v8_final/n{500,1000,2000,5000}`, `results/v9_n5k_seeds` and `results/v19_cuda_etth2_n10k`. The three withdrawn linear-probe columns this table used to carry are retained as a record in §A29.4.

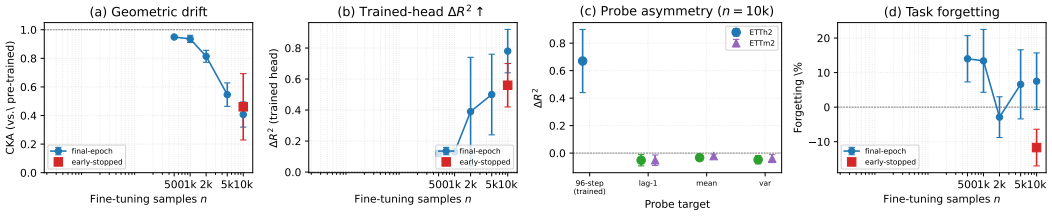


Figure 3: Cost–benefit dynamics on Moirai-Small/ETTh2 (condition B, h=96). (a) CKA decays monotonically with n (final-epoch vs. early-stopped markers); (d) forgetting is non-monotonic and protocol-dependent. Panels (b) and (c) plot the linear-probe ΔR^2 statistic and its sign asymmetry, both **retracted** (§A29.4); they are reproduced unaltered as a record of the withdrawn analysis and support no claim here. The dissociation this figure is cited for is the contrast between (a) and (d): drift monotone in n, outcome not.

the unconstrained condition beats both of the others run there: zero-shot by 31–42% and the frozen encoder by 20–22 points.

Condition coverage differs by cell, and we state it rather than let the widest row imply it. The full six-condition comparison exists only for Moirai-Small/ETTh2, so "best of six" is not a claim available for any other cell. Every intervention cell has B and D. Condition H (strict freeze) adds 16 cells (Appendix A8) and condition E (LoRA) adds 8 (Appendix A9), both on the seeds given in their own tables; L2-SP and EWC exist on Moirai-Small/ETTh2 alone. A claim about a cell is therefore a claim about the conditions that cell ran, and the per-cell seed counts are printed in every table rather than summarised into one figure.

The joint trajectory is plotted in Fig. 3; panels (a) and (d) carry the claim: a monotone drift curve beside a non-monotone outcome curve.

A7 The Pre-Registered Prospective Test, in Full

What was registered, and when. Predictions were written to `results/v47_prospective/preregistration.json` and committed to git at a named revision before any outcome run for those cells existed; the scoring script reads that file and

Table 5: All 31 intervention cells (the 31 of 32 screened cells with a paired B/D run): **selection-split** gate V_{ridge} (the primary gate of §4, scored on windows disjoint from the outcome columns beside it), then B−D on validation and on held-out windows (positive = the frozen encoder was better), and whether the validation and held-out readings disagree in sign. Every row names its own n because n is *not* constant down the column: the ETTh2 spectrum cells ran at 500 while their same-dataset neighbours ran at 1,000, and a row that stayed silent would read as sharing its neighbour's. ILI is the one cell with no cap: it trains on its whole 452-window split, since it is the only series small enough, so its n is a size rather than a subsample. Generated by `scripts/cell_matrix.py -latex`.

Cell	Seeds	Gate V_{ridge}	B−D validation	B−D held-out	Reverses?
Chronos / ETTh1 (h=24, n=8000)	3	-0.116 [f]	$+6.8\pm2.9$	$\mathbf{-39.2\pm13.9}$	yes
Moirai-S / ILI (h=24, n=452)	10	-10.206 [f]	-62.6 ± 1.4	$\mathbf{-38.6\pm1.3}$	no
Chronos / ETTm2 (h=24, n=8000)	3	-0.275 [f]	-24.1 ± 9.8	$\mathbf{-29.5\pm6.4}$	no
Moirai-B / ETTh2 (h=96, n=1000)	3	$+0.777$	$+3.8\pm4.1$	$\mathbf{-22.3\pm2.9}$	yes
Chronos / Weather (h=24, n=8000)	3	-0.203 [f]	-22.4 ± 4.5	$\mathbf{-21.6\pm3.2}$	no
Moirai-L / ETTh2 (h=96, n=500)	3	$+0.875$	$+7.2\pm5.8$	$\mathbf{-20.8\pm1.4}$	yes
Moirai-B / ETTh2 (h=192, n=1000)	3	$+0.925$	-3.9 ± 6.2	$\mathbf{-20.2\pm1.1}$	no
Chronos / ETTh2 (h=24, n=8000)	3	-0.184 [f]	-19.9 ± 3.2	$\mathbf{-12.9\pm0.6}$	no
TimesFM / ETTm2 (h=24, n=1000)	3	$+0.082$ [f]	-12.2 ± 12.4	$\mathbf{-7.7\pm2.6}$	no
TimesFM / ETTh2 (h=24, n=1000)	3	-0.208 [f]	$+0.2\pm2.9$	$\mathbf{-7.1\pm3.1}$	yes
Moirai-S / Electricity7 (h=96, n=1000)	3	-0.650 [f]	-10.4 ± 1.1	$\mathbf{-3.2\pm1.5}$	no
Chronos / Electricity (h=24, n=8000)	3	-0.089 [f]	-10.3 ± 3.2	$\mathbf{-2.7\pm1.9}$	no
Moirai-S / Electricity7 (h=192, n=1000)	3	-0.455 [f]	-8.2 ± 1.9	$\mathbf{-2.6\pm2.1}$	no
TimesFM / Weather (h=24, n=1000)	3	-0.160 [f]	-4.8 ± 0.6	$\mathbf{-1.6\pm1.5}$	no
TimesFM / ETTh1 (h=24, n=1000)	3	-0.101 [f]	-5.4 ± 0.9	$\mathbf{-1.4\pm2.8}$	no
Moirai-B / ETTm2 (h=192, n=1000)	3	$+0.211$	$+3.5\pm1.5$	$\mathbf{+1.3\pm3.4}$	no
Moirai-S / ETTm2 (h=96, n=1000)	5	-0.164 [f]	$+6.5\pm2.9$	$\mathbf{+2.6\pm3.4}$	no
Moirai-S / ETTh2 (h=96, n=500)	10	$+0.874$	$+10.6\pm2.8$	$\mathbf{+2.8\pm3.7}$	no
Moirai-S / ETTm2 (h=192, n=1000)	5	$+0.352$	$+8.6\pm3.3$	$\mathbf{+4.7\pm3.2}$	no
Moirai-S / ETTh2 (h=192, n=500)	10	$+0.932$	$+17.4\pm4.3$	$\mathbf{+6.1\pm3.4}$	no
Moirai-B / ETTm2 (h=96, n=1000)	3	-0.257 [f]	$+3.9\pm3.7$	$\mathbf{+7.0\pm2.4}$	no
Moirai-B / Weather (h=96, n=1000)	3	-0.823 [f]	$+19.3\pm0.8$	$\mathbf{+9.4\pm2.0}$	no
Moirai-L / Weather (h=96, n=1000)	3	-0.638 [f]	$+4.3\pm6.5$	$\mathbf{+12.0\pm7.5}$	no
Moirai-B / Weather (h=192, n=1000)	3	-0.633 [f]	$+30.6\pm12.0$	$\mathbf{+12.5\pm8.7}$	no
Moirai-S / ETTh1 (h=192, n=1000)	5	-0.347 [f]	$+14.8\pm1.9$	$\mathbf{+13.5\pm1.7}$	no
Moirai-S / ETTh1 (h=96, n=1000)	5	-0.177 [f]	$+16.1\pm1.5$	$\mathbf{+22.1\pm2.3}$	no
Moirai-L / ETTh1 (h=96, n=1000)	3	-0.248 [f]	$+13.4\pm2.1$	$\mathbf{+28.2\pm4.9}$	no
Moirai-B / ETTh1 (h=96, n=1000)	3	-0.312 [f]	$+15.8\pm4.6$	$\mathbf{+37.5\pm8.2}$	no
Moirai-B / ETTh1 (h=192, n=1000)	3	-0.600 [f]	$+15.6\pm2.3$	$\mathbf{+47.0\pm13.8}$	no
Moirai-S / Weather (h=96, n=1000)	3	-0.546 [f]	$+55.4\pm4.3$	$\mathbf{+53.1\pm10.2}$	no
Moirai-S / Weather (h=192, n=1000)	3	-0.506 [f]	$+70.2\pm12.3$	$\mathbf{+62.1\pm13.6}$	no

the finished runs and decides nothing. Two rules were registered against each other: the **gate rule**, $\text{gate}_{\text{pre}} \geq 0.20 \Rightarrow$ at risk, which is the paper's own criterion frozen *before* the baseline correction and therefore graded on trend-denominator scores rather than re-derived after seeing the outcomes; and a **dataset rule**, dataset $\in$ {ETTh1, Weather} $\Rightarrow$ degradation, a *post hoc* pattern read off the original Moirai cells and registered as a competitor so the gate is graded against something. Eight cells were then run (Weather, ETTm2, Electricity7, ETTh1 $\times$ Small/Base/Large, 3 seeds each) and scored by the pre-registered outcome definition, unchanged (Table 9).

Result: the outcome the gate was registered to predict never occurs. None of the eight cells degraded. The gate flagged all eight, giving **TP 0, FP 8, FN 0**: precision 0.00, while **sensitivity and specificity are both unestimable**: sensitivity has an empty positive class in its denominator, and specificity is $0/8$ only because every prediction was positive. The competing dataset rule scores no better, for the same reason. **What "prospective" does and does not mean here:** the predictions were frozen before these cells were *trained*, so they are prospective with respect to the intervention, but the datasets and their splits already existed, so this is not a test on newly collected data. Nor

Table 6: **The four sign reversals between selection and held-out windows** (§7.1). Δ adapted and Δ frozen are the change in each arm's forgetting when scoring moves from the selection split to the held-out split; zs_{test}/zs_{val} is the untrained checkpoint's error ratio between the two splits, an exchangeability check that involves no fine-tuning. All four reversals run from "frozen better" to "adapted better". n is per row and not constant (Table 5), and is taken from the published row rather than re-read here. Generated by `scripts/heldout_decomposition.py`.

Cell	B−D val	B−D held-out	Δ adapted (B)	Δ frozen (D)	zs_{test}/zs_{val}
Chronos / ETTh1 (h=24, n=8000)	+6.8	−39.2	+8.0	+53.9	0.98
Moirai-B / ETTh2 (h=96, n=1000)	+3.8	−22.3	+12.2	+38.3	2.31
Moirai-L / ETTh2 (h=96, n=500)	+7.2	−20.8	−52.9	−24.8	5.07
TimesFM / ETTh2 (h=24, n=1000)	+0.2	−7.1	−11.7	−4.3	1.09

Table 7: **Cross-backbone results (h=24)** (§6). Every non-Moirai cell with a paired frozen-encoder run: **selection-split** gate V_{ridge} ([f]=gate-fail at the 0.20 operating point; all nine fail, as does the screened-only TimesFM/Electricity cell, which has no paired run and so no selection-split gate: it scores +0.132 on the held-out windows), CKA after full fine-tuning, and the two outcome readings as % change against zero-shot (*negative is better than zero-shot*). **Protocol note:** this arm is univariate OT, lookback 96, h=24, 200 held-out windows, per-window z-scored; the Moirai arm of §4 is lookback 192, h=96, 300 windows, train-normalised. Gates for both arms come from one source (`results/gate_val_side.json`, via `scripts/cell_matrix.py`); the two arms are never pooled.

Backbone	Dataset	Gate V_{ridge}	CKA	Full FT	Frozen
Chronos-T5-S	ETTm2	−0.275 [f]	0.112	−41.4%	−11.9%
Chronos-T5-S	Weather	−0.203 [f]	0.092	−5.7%	+15.9%
Chronos-T5-S	Electricity	−0.089 [f]	0.232	−2.3%	+0.4%
Chronos-T5-S	ETTh2	−0.184 [f]	0.094	−0.4%	+12.5%
Chronos-T5-S	ETTh1	−0.116 [f]	0.169	+20.6%	+59.8%
TimesFM-2.5	ETTm2	+0.082 [f]	0.360	−51.8%	−44.1%
TimesFM-2.5	ETTh2	−0.208 [f]	0.246	−20.4%	−13.3%
TimesFM-2.5	Weather	−0.160 [f]	0.335	−7.8%	−6.1%
TimesFM-2.5	ETTh1	−0.101 [f]	0.561	−3.0%	−1.6%

is the empty positive class an artefact of the corrected gate: only one of the eight clears 0.20 on either corrected split (base/ETTm2 h=192, +0.21 on selection windows) and it fails clause (ii) in 0 of 3 seeds (Appendix A29.1). **Nor is the 0.20 threshold what fails**: re-scoring the same predictor against the same outcome at thresholds from 0.10 to 0.60, moving nothing else, gives precision 0.00 at every one, with 8, 8, 6, 5, 3 and 2 cells flagged (Table 10); the problem is not the cutoff but that the predicted class is empty.

What a zero-prevalence test licenses. With no positive instance the arm is informative in one direction only. It establishes that the gate's positive predictions are wrong here, at every threshold swept, and that the error is not an artefact of the operating point. It establishes *nothing* about whether the gate would detect degradation if degradation existed: that is a claim about sensitivity, and $0/0$ is not a rate. A reader who takes this arm as a refutation of the gate is reading past the prevalence. The regime the arm cannot reach is one a positive control has to construct deliberately (§7.3, Appendix A30), and which, on these benchmarks, we could not construct.

The gate does not provide a reliable ordering, and there is no actionable threshold. Within Moirai (n=22 cells from six series, the only arm with enough cells to ask), the gate's continuous score correlates *negatively* with held-out Δ_{enc}: Spearman ρ=−0.293, and neither cells nor series exclude zero, so we report the direction and not the interval (Appendix A27). The sign is the wrong one (higher measured pre-trained value goes with encoder adaptation helping *more*, not less, the opposite of what a preservation heuristic needs), but the sharper point is that the magnitude is not

Table 8: **Mitigation spectrum on Moirai-Small/ETTh2.** Rows ordered by representation preservation (CKA at h=96). On *this* cell LoRA preserves representations (CKA$\approx$0.98) while *improving* on zero-shot, and weight anchoring reduces drift without buying utility. **Neither reading generalises, and we checked rather than assumed:** extending condition E to 8 further value-cells (Appendix A9) puts CKA_E anywhere from 0.55 to 0.97 and leaves full fine-tuning ahead of LoRA on 4 of the 8. **Every row is scored against its own zero-shot.** Forgetting% is the mean over seeds of each run's (fine-tuned − zero-shot)/zero-shot, so no row borrows another's denominator; earlier versions of this table divided the anchoring and LoRA rows by a single shared value and read up to 3.4 pp away as a result. MSE, CKA and ℓ_2 drift are means over seeds with population std (ddof=0); the body's SEM convention (§3) does not apply here. **The rows are not matched on training-set size.** Conditions B, C and D use n=500 training samples (zero-shot MSE .1256 at h=96, .1580 at h=192); E, F and G use n=1,000 (zero-shot MSE .1274 and .1596); the two rightmost columns give the seed count and the training-set size separately. We therefore claim no ordering *between* the two groups: there is no full-fine-tuning arm at n=1,000 on this cell with more than a single seed, so "anchoring ties full fine-tuning" is not a comparison these runs support. The dissociation the table is cited for (CKA not predicting utility) is within-group and unaffected. Condition D reads CKA 0.99996 rather than 1: it leaves `in_proj` and `mask_encoding` trainable, so it is a frozen-*encoder* control and not a strict freeze. Only condition H (Appendix A8) is 1.0000 by construction. Earlier versions of this table printed D as 1.000, which rounded that distinction away.

	$h=96$				$h=192$					
Method	MSE	Forg.%	CKA	Drift	MSE	Forg.%	CKA	Drift	seeds	n
B: Full fine-tune	.140$_{\pm.014}$	+11.1	.937$_{\pm.028}$	3.96$_{\pm.56}$	.183$_{\pm.020}$	+15.7	.970$_{\pm.009}$	4.12$_{\pm.32}$	10	500
F: L2-SP (λ=0.01)	.150$_{\pm.006}$	+16.0	.939$_{\pm.023}$	4.98$_{\pm.34}$	.179$_{\pm.009}$	+12.7	.945$_{\pm.007}$	4.84$_{\pm.22}$	3	1,000
C: NLL+SupCon	.138$_{\pm.010}$	+10.1	.949$_{\pm.014}$	3.72$_{\pm.43}$	.178$_{\pm.011}$	+12.9	.964$_{\pm.015}$	3.97$_{\pm.34}$	10	500
F: L2-SP (λ=0.1)	.147$_{\pm.003}$	+13.4	.965$_{\pm.006}$	2.94$_{\pm.08}$	.181$_{\pm.003}$	+13.6	.980$_{\pm.002}$	2.84$_{\pm.02}$	3	1,000
G: EWC (λ=1000)	.149$_{\pm.015}$	+14.9	.972$_{\pm.004}$	3.29$_{\pm.79}$	.188$_{\pm.004}$	+18.3	.978$_{\pm.006}$	4.46$_{\pm.38}$	3	1,000
E: LoRA (r=8)†	**.116**$_{\pm.001}$	**−9.1**	.979$_{\pm.006}$	0.00$_{\pm.00}$	**.157**$_{\pm.002}$	**−1.8**	.986$_{\pm.005}$	0.00$_{\pm.00}$	5	1,000
D: Frozen encoder	.126$_{\pm.006}$	+0.5	.99996$_{\pm.00001}$	1.11$_{\pm.15}$	.155$_{\pm.002}$	−1.7	.99995$_{\pm.00002}$	1.30$_{\pm.18}$	10	500

† LoRA (α=16) adds low-rank updates that do not modify base encoder parameters; ℓ_2 drift is exactly zero by construction.

stable: scoring the same gate on the held-out windows instead gives ρ=−0.518 on the same cells. An ordering that nearly doubles when the gate changes split is not one a practitioner can set a threshold on, and no other quantity we measured orders Δ_enc either (Appendix A26).

Table 9: **The pre-registered prospective test.** Predictions frozen in git before any condition-B or -D run for these cells existed. "Degradation" applies the paper's own definition unchanged: gate-passing $\wedge$ $\text{forg}_\text{B}>0$ in every seed $\wedge$ $\text{forg}_\text{D}<0$ in every seed. gate_pre is the frozen predictor's score, on the pre-correction denominator; gate_val is the corrected selection-split gate of §4.

Cell (new)	gate_pre	gate_val	forg_B	B−D held-out	Degradation?	Predicted
base Weather (h=96)	+0.41	−0.82	−11.7 (0/3)	+9.4±2.0	no	gate ×, data ×
base Weather (h=192)	+0.59	−0.63	−18.1 (0/3)	+12.5±8.7	no	gate ×, data ×
base ETTm2 (h=96)	+0.30	−0.26	−13.1 (0/3)	+7.0±2.4	no	gate ×, data ✓
base ETTm2 (h=192)	+0.26	+0.21	−8.4 (0/3)	+1.3±3.4	no	gate ×, data ✓
small Electricity7 (h=96)	+0.71	−0.65	−16.0 (0/3)	−3.2±1.5	no	gate ×, data ✓
small Electricity7 (h=192)	+0.76	−0.45	−18.6 (0/3)	−2.6±2.1	no	gate ×, data ✓
large ETTh1 (h=96)	+0.35	−0.25	+14.1 (3/3)	+28.2±4.9	no	gate ×, data ×
large Weather (h=96)	+0.47	−0.64	+7.9 (3/3)	+12.0±7.5	no	gate ×, data ×

Table 10: Gate threshold sensitivity on the eight pre-registered cells (§7.2). The predictor and the outcome definition are both as pre-registered; only the threshold moves. Bold is the operating point used in the paper.

Threshold	Flagged	TP	FP	FN	Precision	Recall
0.10	8	0	8	0	0.00	nan
0.20	8	0	8	0	0.00	nan
0.30	6	0	6	0	0.00	nan
0.40	5	0	5	0	0.00	nan
0.50	3	0	3	0	0.00	nan
0.60	2	0	2	0	0.00	nan

A7.1 Batch 3: The Corrected Gate, Registered Before Its Own Outcomes

Why a second batch, and what was registered. The arm above grades a predictor we have since corrected, and it fired on every cell it saw, so it cannot say what happens when the gate *declines* to fire. A second batch was therefore registered in `results/v57_prospective3/preregistration_v3.json`, committed on 2026-09-22 at revision `e5f1d626`, **before any condition-B or condition-D run of any cell in it existed**: **ten** cells: five Moirai (Base/Electricity7 at $h\in\{96, 192\}$, Large/ETTh2 $h=192$, Large/ETTm2 at $h\in\{96, 192\}$) and five TimesFM-2.5 at $h=48$, at 3 seeds in each of conditions B and D, with the *corrected* ridge gate as the predictor and the paired-interval rule of §3 as the registered outcome. The two batches are reported side by side and **never pooled**: they grade different predictors under different outcome rules, and a prospective test may not move its own comparison set, so the BH adjustment runs within this batch alone.

Result: no cell degraded, and this time the gate did not flag everything (Table 11). Across the **9** scored cells (54 outcome runs), **0** degrade under the registered interval rule and **0** under the published unanimity definition, the two agreeing as they do on the retrospective grid. The gate fired on **2** of the 9, giving **TP 0, FP 2, FN 0, TN 7**: precision 0.00 again, sensitivity still unestimable against an empty positive class, but **specificity** 0.78**, which batch 1 could not estimate at all**. That number is the arm's one positive contribution and it is a measure of *restraint*, not of discrimination: at zero prevalence a screen can be credited for the cells it correctly leaves alone, and nothing can be said about the cells it would have caught. The competing dataset rule flags two different cells (TimesFM-2.5 on ETTh1 and on Weather) and scores the same matrix.

The gate's two prospective positives are both wrong, and one of them is wrong decisively. Large/ETTh2 $h=192$ carries a gate of $+0.92$, within 0.008 of Moirai-Small/ETTh2 $h=192$ ($+0.932$), the strongest capability the screen admits anywhere in this paper and the very cell the positive control of §7.3 had to engineer a conflicting task against. On that cell full fine-tuning does not damage the pre-trained capability: it *improves* held-out loss by 33.2% against zero-shot, with a 95% interval of $[-47.2, -19.2]$ that excludes zero. The gate's most confident prospective prediction is therefore not merely unconfirmed but contradicted, in the direction opposite to the one it predicts.

What this arm cannot support, stated in its own terms. Both the B$-$ZS and D$-$ZS intervals exclude zero on only **2** of the 9 cells, and the primary matrix counts an inconclusive cell as not degrading, as registered, so most of the specificity above rests on cells where the data are consistent with a small effect of either sign. Restricted to the 2 decisive cells the matrix is TP 0, FP 1, FN 0, TN 1, which is not a rate anyone should quote. The secondary registered outcome, the three-way call on Δ_{enc} after BH adjustment within the batch, is **inconclusive on all 9**: zero freeze-decisive and zero adapt-decisive cells, so the batch identifies no new cell on which freezing or adapting can be called, and contradicts no call made on the published grid. A batch of nine cells at 3 seeds is a test of the screen's false-positive behaviour and of nothing else.

One registered cell was not run, and the shortfall is disclosed rather than absorbed. Nine of the ten registered cells were scored; the tenth, Moirai-Large/ETTm2 $h=96$, was deferred on cost grounds when the submission window would not hold its measured $\sim$16.5 h of fine-tuning, and the decision is recorded with its timestamp, its reasons and the record count at the moment it was taken in `results/v57_prospective3/deferred_cell_2026-09-24.json`, committed before the deferral took effect. Two facts make it checkable rather than a matter of trust. *No outcome for that cell existed when the decision was made* (0 condition-B and 0 condition-D records), so the choice cannot have been informed by its result; and *the cell is gate-failing* (-0.278), so the only matrix entry it could have supplied is a true negative, which would move specificity from 0.78 to 0.80. Its absence can therefore only weaken the one estimate this arm exists to produce; it cannot manufacture a favourable one. The registration is unchanged and still lists all ten cells, so the shortfall is visible by diffing it against the scored cells.

Two disclosures about the predictor. The five TimesFM cells' registered gate was computed on the seed-42 selection-window draw, while the multi-seed recomputation in `gate_all_cells.timesfm_gates` averages the draws of all three run seeds; the frozen value is the predictor and the recomputation is a sensitivity on it, not a replacement. Those five cells are also the only registered gates the paper's own cache does not reproduce to 10^{-9}, and by design: their numerator was measured by loading the checkpoint, and the one other route to it agrees only to

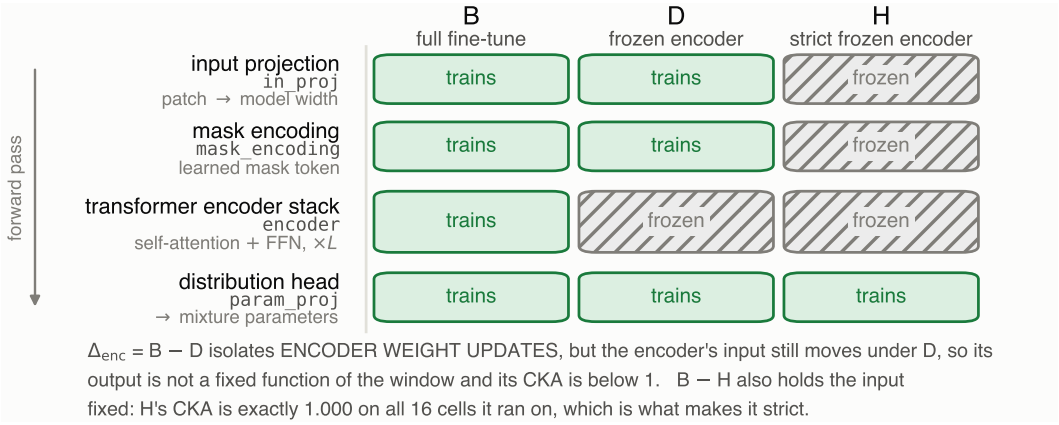


Figure 4: **Where the freeze boundary falls, module by module.** Green boxes are parameters the optimiser may move; hatched boxes are held at their pre-trained values. The grid is transcribed from the `requires_grad` assignments in `scripts/finetune_forecasting.py`. Two of our seven survivors change the sign of Δ_{enc} between D and H (Table 12), so which of these columns a paper means by "we froze the encoder" is not a detail.

$\sim 6\times10^{-7}$ relative, which is real confirmation but would trip a bit-equality check. The five Moirai gates are recomputed by the paper's estimator in a second pass and **all five reproduce the registered value exactly**.

Table 11: **Batch 3 of the pre-registered prospective test: the corrected gate, graded on cells whose outcomes did not exist when it was frozen.** gate_{val} is the corrected selection-split ridge gate of §4, frozen in the registration; "pred." is the registered gate rule at the 0.20 operating point. Δ_{FT} and Δ_{frozen} are B−ZS and D−ZS in percent of zero-shot loss, with 95% intervals propagating the zero-shot reference's own error; "Degr.?" applies the registered interval rule. The Δ_{enc} call is BH-adjusted within this batch only. A tenth registered cell (Moirai-Large/ETTm2 h=96, gate −0.278) was deferred for the submission deadline and is disclosed in Appendix A7.1.

Cell (batch 3)	gate_{val}	pred.	n	Δ_{FT}	Δ_{frozen}	Degr.?	Δ_{enc} call
large ETTh2 (h=192)	+0.92	risk	3	−33.2 [-47.2, -19.2]	−12.9 [-25.2, -0.7]	no	inconclusive
large ETTm2 (h=192)	+0.24	risk	3	−10.3 [-37.5, +16.8]	−4.9 [-22.1, +12.4]	no	inconclusive
TimesFM-2.5 / ETTh1 (h=48)	−0.05	–	3	−2.6 [-5.3, +0.1]	+0.1 [-2.6, +2.8]	no	inconclusive
TimesFM-2.5 / ETTh2 (h=48)	−0.08	–	3	+99.2 [-345.3, +543.7]	−6.4 [-9.1, -3.8]	no	inconclusive
TimesFM-2.5 / ETTm2 (h=48)	−0.01	–	3	−29.4 [-62.1, +3.3]	−16.0 [-20.3, -11.7]	no	inconclusive
TimesFM-2.5 / Weather (h=48)	−0.08	–	3	−9.1 [-15.5, -2.7]	−7.6 [-8.7, -6.4]	no	inconclusive
TimesFM-2.5 / Electricity (h=48)	+0.04	–	3	+12.4 [+6.2, +18.7]	−1.3 [-3.5, +0.8]	no	inconclusive
base Electricity7 (h=96)	−0.59	–	3	−3.7 [-17.2, +9.9]	−18.8 [-19.8, -17.8]	no	inconclusive
base Electricity7 (h=192)	−0.51	–	3	−3.8 [-31.1, +23.5]	−19.2 [-19.9, -18.6]	no	inconclusive

A8 The Strict-Freeze Control

Condition D freezes the encoder's own weights but leaves the input projection (`in_proj`, 247,680 parameters) and `mask_encoding` (384) trainable alongside the distribution head (`param_proj`, 2,956,800). What the encoder *receives* therefore keeps changing, so its output is not a fixed function of the input, which is why D's measured CKA is 0.76–0.90 on ILI and 0.97–0.99 on Moirai-Base rather than 1.000. A reader can reasonably object that wherever "freezing wins", it wins partly by re-fitting the input projection rather than by holding the encoder still.

Condition **H** removes the objection: `in_proj` and `mask_encoding` are frozen too, so only `param_proj` trains and the encoder's output is pinned by construction (measured CKA 1.0000 exactly, as it must be). B−H is then a clean read of what allowing *any* encoder-side change was worth, and B−D versus B−H separates encoder adaptation from input re-fitting. We fixed the reading rule before running: the claim is that the sign and reading of B−D survive strict freezing, so B−H is

reported beside B−D on every cell, and divergence on any cell is reported as that cell's gap being partly input re-fitting rather than being buried.

Result (Table 12). The sixteen cells fall into three groups. Seven were chosen before the gate was corrected: they are the six that met the degradation definition under the superseded trend baseline (Appendix A29.2), plus ILI. Nothing was dropped; two cells the prospective arm later added (Moirai-Large on ETTh1 and on Weather) postdate this control and carry no H run. That selection no longer picks out a privileged class (against the fitted baseline no cell meets the definition), but it remains the set chosen to be *most* favourable to the degradation story, which is the useful property here. Five are the gate-passing survivors of §5.1, run afterwards precisely because the first seven did not reach them: until they existed, the survivors' readings rested on condition D alone. The last four are the Moirai/ETTm2 cells that clear the screen against a rung other than the fitted line (Appendix A23), added so that the control covers value-cells outside ETTh2: all four of them, at three seeds each, so ETTm2 is covered completely rather than partially. Measured CKA is 1.0000 everywhere, confirming the construction.

On twelve of the sixteen the reading survives; on four it reverses, and the four are the ones we added. B−H keeps B−D's sign on the original seven and on the five survivors. There the magnitude changes by -13.1 to $+28.9$ pp, and on the original seven by -5.8 to $+0.5$ pp, so at most 5.8 pp of a gap as large as 62 pp is input-projection re-fitting. On ILI, where adaptation helps, strict freezing makes adaptation look *more* valuable ($-38.6 \rightarrow -43.6$), which is the same direction: holding more of the model still costs more where the encoder was earning its keep. B−D is averaged over the seeds condition H ran, so Moirai-Small/ETTh1's entries here are 3-seed values against the 5-seed values quoted in Table 5.

All four ETTm2 cells diverge, and we said in advance how that would be read. On every one of them Δ_{enc} changes sign between the two controls: Moirai-Small h=96 goes from B−D $+7.3\pm2.9$ to B−H -6.8 ± 2.0, Moirai-Base h=96 from $+7.0\pm2.4$ to -9.2 ± 3.1, Moirai-Small h=192 from $+4.2\pm5.8$ to -2.7 ± 6.1 and Moirai-Base h=192 from $+1.3\pm3.4$ to -2.4 ± 2.9: shifts of -14.2, -16.2, -6.9 and -3.7 pp. The last of those is the weakest of the four and we do not lean on it: both of its intervals straddle zero, so what that cell shows is a shift in the mean and not a resolved reversal. The pattern is carried by the two h=96 cells, where the shift is four to five times the SEM. The consistency of the *direction* across all four is what the reading rests on: four cells, one dataset, four shifts, all negative. The rule fixed before the runs was that divergence means part of that cell's gap is input re-fitting, and that is what these say: on ETTm2, condition D's apparent "freezing wins" was carried *entirely* by the input projection adapting to the target, and once it is held fixed too, allowing the encoder to adapt is the better choice. Two readings follow, and they point the same way as the rest of the paper rather than against it. First, the direction of the correction is towards adaptation on every diverging cell, so nothing here recovers a degradation case; ETTm2 joins ETTh2 in the column where full fine-tuning wins under the strict control. Second, and more important for §5, it is direct evidence that **condition D does not measure encoder adaptation alone**: on four of the sixteen cells where we can separate the two, the loose control's sign is set by the projections rather than by the encoder. That is a limit on B−D as an estimand, and it is why §3 defines Δ_{enc} against the control actually used and reports H beside it rather than substituting one for the other.

On the five survivors the control cuts against freezing, not for it. These are the cells the objection mattered most for, and on the three where full fine-tuning already won, strict freezing *widens* its margin rather than narrowing it: B−D of -22.3, -20.2 and -20.8 becomes B−H of -28.0, -31.8 and -33.9 on Moirai-Base h=96, Base h=192 and Large h=96, a move of -5.6 to -13.1 pp. Part of what condition D was scoring on these cells was therefore the input projection re-fitting to the target data, and removing it makes encoder adaptation look *more* valuable, not less. On Moirai-Small h=192 (the cell that comes closest to degradation), B−H is $+1.2\pm8.5$ against B−D's $+4.1\pm8.6$: freezing still reads as better, by less, with a standard error several times the mean either way. Condition H also improves on the un-tuned model in every seed of all five (forg_{H} from -0.2% to -8.5%), so on the survivors both parts of the degradation definition behave the same way under the strict control as under the loose one.

The one large change is the diverged seed, and it corroborates the earlier reading. Moirai-Small/ETTh2 h=96 moves from B−D $+13.7$ to B−H $+42.6\pm29.5$, the only cell whose magnitude shifts by more than 13 pp. It is seed 42, which §5.1 reports as a training failure rather than an encoder-drift effect: that seed diverges under full fine-tuning ($+98.9\%$) *and* under the frozen encoder

(+91.1%), but it cannot diverge under condition H, where only `param_proj` trains and the run therefore sits at zero-shot (−0.1%). Over the other two seeds B−H is +14.4 against B−D's +16.7. The strict control thus supplies independent support for the seed-42 diagnosis: what fails on that seed is optimisation, and holding the encoder's *inputs* fixed as well as its weights is what prevents it.

Table 12: The strict-freeze control (condition H) beside condition D on sixteen cells: the six that met the degradation definition under the superseded trend baseline, plus ILI, plus the five gate-passing survivors of §5.1, plus all four Moirai/ETTm2 value-cells. **B−H keeps B−D's sign on twelve and reverses it on all four ETTm2 cells**, where condition D's margin was the input projection re-fitting rather than the encoder being held still. B−D is restricted to the seeds condition H ran, and a cell with fewer than three H seeds is excluded and named by the emitter rather than printed thin. n is shown on every row and is not constant down the column; ILI's 452 is its whole training split rather than a cap (Table 5). Generated by `scripts/cell_matrix.py -latex`.

Cell	Seeds	B−D	B−H	forg_H	CKA_H
Moirai-S / Weather (h=192, n=1000)	3	+62.1±13.6	**+56.7±13.5**	−10.6±0.3	1.0000
Moirai-S / Weather (h=96, n=1000)	3	+53.1±10.2	**+50.1±6.6**	−12.8±0.6	1.0000
Moirai-B / ETTh1 (h=192, n=1000)	3	+47.0±13.8	**+47.5±13.7**	−8.0±0.4	1.0000
Moirai-B / ETTh1 (h=96, n=1000)	3	+37.5±8.2	**+32.5±8.2**	−9.7±0.4	1.0000
Moirai-S / ETTh1 (h=96, n=1000)	3	+23.2±3.6	**+20.4±3.6**	−0.6±0.2	1.0000
Moirai-S / ETTh1 (h=192, n=1000)	3	+14.5±1.4	**+8.7±1.5**	+1.2±0.4	1.0000
Moirai-S / ETTh2 (h=96, n=500)	3	+13.7±9.0	**+42.6±29.5**	−0.2±0.1	1.0000
Moirai-S / ETTm2 (h=96, n=1000)	3	+7.3±2.9	**−6.8±2.0**	−2.9±0.4	1.0000
Moirai-B / ETTm2 (h=96, n=1000)	3	+7.0±2.4	**−9.2±3.1**	−3.9±1.8	1.0000
Moirai-S / ETTm2 (h=192, n=1000)	3	+4.2±5.8	**−2.7±6.1**	+1.3±0.3	1.0000
Moirai-S / ETTh2 (h=192, n=500)	3	+4.1±8.6	**+1.2±8.5**	−1.4±0.2	1.0000
Moirai-B / ETTm2 (h=192, n=1000)	3	+1.3±3.4	**−2.4±2.9**	−6.0±0.4	1.0000
Moirai-B / ETTh2 (h=192, n=1000)	3	−20.2±1.1	**−31.8±1.0**	−4.5±2.0	1.0000
Moirai-L / ETTh2 (h=96, n=500)	3	−20.8±1.4	**−33.9±0.1**	−8.5±0.5	1.0000
Moirai-B / ETTh2 (h=96, n=1000)	3	−22.3±2.9	**−28.0±3.3**	−3.5±0.4	1.0000
Moirai-S / ILI (h=24, n=452)	10	−38.6±1.3	**−43.6±0.7**	−7.1±0.5	1.0000

One clause does not survive, and it is not the headline one. "Degradation" has two parts: fine-tuning ends up worse than the un-tuned model, *and* freezing improves on it. The first part is a property of condition B and is untouched by this control. The second is condition D's, and on two cells the strictly-frozen model does *not* beat the un-tuned model: Moirai-Small/ETTh1 h=192 at $\text{forg}_\text{H} = +1.2\pm0.4\%$, positive in all 3 seeds where forg_D is −4.8%, and Moirai-Small/ETTm2 h=192 at +1.3±0.3%, positive in all 3 seeds where forg_D is −5.8%. On those cells the head alone cannot recover the zero-shot forecast; D's improvement came partly from re-fitting the input projection. The other fourteen cells keep both parts (forg_H from −0.2% to −12.8%), so "freezing also beats not fine-tuning" holds strictly on 14 of 16. The first part is unaffected in either direction: where freezing wins it keeps winning under the strict control (B−H positive on all six candidate cells), and where full fine-tuning wins it wins by more (the five survivors above). We report the cells rather than re-defining the term around them.

A9 LoRA on the Value-Cells: the Constrained Middle of the Adaptation Axis

Conditions B and D bracket the adaptation axis: everything trains, or the encoder body does not. Neither tells a practitioner whether a *cheap constrained* adaptation would have beaten the extreme their cell happened to run. Condition **E** (LoRA, r=8, α=16) is that middle point, and until this revision it existed on one cell, which is why every "LoRA is the cheap win" sentence in earlier versions generalised from Moirai-Small/ETTh2. We ran E on the eight Moirai value-cells that lacked it, seeds 42/123/456, each at its own arm's n and epoch budget (Table 13).

What E can and cannot measure, fixed before the runs. LoRA adds trainable low-rank updates to the encoder's attention projections, so the encoder's *function* changes while its base weights do not: ℓ_2 weight drift is exactly 0.0 on all 24 records by construction, and it is omitted from the table rather

Table 13: **Condition E (LoRA, r=8, α=16) on the eight Moirai value-cells that lacked it.** All contrasts are held-out test MSE, normalised by the cell's zero-shot test MSE, in percentage points; $\pm$ is SEM over the seeds in the Seeds column. Sign convention matches B−D and B−H throughout: **positive means full fine-tuning did worse than the comparison arm**. B−D is recomputed on E's seeds, so it is comparable to B−E within a row and may differ from the same cell's row in Table 5, which averages that cell's full seed set. Parenthesised counts are the seeds sharing the mean's sign. ℓ_2 weight drift is exactly 0.0 for every run and is not shown: LoRA leaves the base parameters untouched by construction, which is why CKA, not drift, is the representation measure here. Rows ordered by D−E.

Cell	Seeds	B−D	B−E	D−E	forg_E	CKA_E
Moirai-S / ETTm2 (h=96, n=1000)	3	+7.3±2.9	+13.9±2.1	**+6.5±0.8** (3/3)	−23.6±0.3 (0/3)	0.8438
Moirai-S / ETTm2 (h=192, n=1000)	3	+4.2±5.8	+7.3±6.8	**+3.1±1.0** (3/3)	−8.6±0.9 (0/3)	0.9280
Moirai-B / ETTh2 (h=192, n=1000)	3	−20.2±1.1	−17.7±1.1	**+2.5±0.9** (3/3)	−18.6±3.0 (0/3)	0.6944
Moirai-S / ETTh1 (h=96, n=1000)	3	+23.2±3.6	+23.4±4.9	**+0.2±1.3** (2/3)	−3.6±1.1 (0/3)	0.9093
Moirai-B / ETTm2 (h=96, n=1000)	3	+7.0±2.4	+4.9±2.9	**−2.2±1.8** (1/3)	−18.0±2.7 (0/3)	0.5482
Moirai-L / ETTh2 (h=96, n=500)	3	−20.8±1.4	−27.8±0.7	**−7.0±1.2** (0/3)	−14.6±0.7 (0/3)	0.9674
Moirai-B / ETTm2 (h=192, n=1000)	3	+1.3±3.4	−19.6±7.0	**−20.9±5.3** (0/3)	+11.2±5.3 (3/3)	0.6495
Moirai-B / ETTh2 (h=96, n=1000)	3	−22.3±2.9	−43.3±14.1	**−21.0±11.4** (1/3)	+11.9±10.5 (2/3)	0.9173

than printed as a column of zeros that would invite the reading that the encoder was held still. CKA is the measure that still reads on E. Δ_{enc} remains B−D; E is not a second estimate of it.

The one-cell reading does not survive. CKA_E spans 0.5482 to 0.9674 across these eight cells, against the ≈0.98 the single-cell spectrum reports, so LoRA is not a representation-preserving method in general; it is one on some cells. Full fine-tuning finishes ahead of LoRA (B−E < 0) on **4 of the 8**. LoRA is itself worse than not fine-tuning at all on two cells ($\text{forg}_\text{E} = +11.9\%$, 2 of 3 seeds, on Moirai-Base/ETTh2 h=96; +11.2%, 3 of 3, on Moirai-Base/ETTm2 h=192).

Against the frozen encoder, read as a paired contrast. D−E is paired per seed rather than taken as the difference of two means, which matters: on Moirai-Small/ETTh1 the B−E and B−D means sit 0.2 pp apart with a SEM of 4.9, and a difference of means there would read as "LoRA beat freezing" on a cell that cannot separate them. Paired and requiring a unanimous sign, **LoRA beats freezing on 3 of 8 cells, freezing beats LoRA on 2, and 3 are inseparable**. Moirai-Base/ETTh2 h=192 is the one cell that orders all three points cleanly (full fine-tuning, then LoRA, then frozen), and it is a gate-passing cell, so the ordering is on a cell with something to preserve.

Why this belongs in a paper about drift as a diagnostic. It is a third condition on which the paper's central claim holds. Across these eight cells CKA_E does not order the outcome of constraining adaptation (ρ=−0.167 against B−E; −0.333 against B−D on the same cells): the highest-CKA cell here, Moirai-Large/ETTh2 at 0.9674, is one where the constrained option *loses* to full fine-tuning by 27.8 pp, and the lowest, Moirai-Base/ETTm2 h=96 at 0.5482, is one where it wins. As with B−D, how much the representation moved does not tell you whether constraining the move helped.

Two limits of this arm. The two Moirai-Small/ETTh2 cells already carrying E are excluded here: their E arm ran at n=1,000 against B/D at n=500, a mismatch disclosed in Table 8's caption and not improved by adding a third n. And α is not written into the run records (only r is), so α=16 is traceable to `scripts/run_lora_valuecells.sh` and its committed command line rather than to the per-run JSONs, the same limitation the learning rate carries.

A10 LoRA Rank and Hyperparameter Sensitivity

Table 14 shows LoRA on **Moirai-Small** is robust to rank ($r \in \{4, 8, 16, 32\}$): all ranks achieve CKA≥0.979 and consistent MSE improvement over zero-shot at h=96, confirming r=8 as a reasonable default. On **Moirai-Large** (seed 42, h=96, n=500), rank escalation alone does *not* rescue the failure mode: forgetting stays +19 to +30% across $r \in \{8, 16, 32, 64\}$. Table 15 extends the sweep to α, target-module scope, and LR (all at r=8, seed 42): varying $\alpha \in \{8, 16, 32\}$ and module scope provides no rescue, but reducing LR to 1×10^{-5} does (forg.= $-8.5 \pm 0.7\%$, CKA=0.989±0.003, 5 seeds): the Small-recipe transfers to Large only at a 10× lower learning rate.

Table 14: LoRA rank sensitivity on ETTh2 (condition E, 500 training samples). Moirai-Small: all ranks achieve near-identical MSE and CKA$\geq$0.979. Moirai-Large: all ranks forget $+19$ to $+30\%$, with no rank recovering Moirai-Small's negative forgetting. Mean $\pm$ std across 3 seeds (5 for r=8 Small); Moirai-Large single-seed (s=42) for $r \in \{16, 32, 64\}$ (std not reported; point estimates only), 3 seeds for r=8.

Backbone	Rank	h	MSE	Forg.%	CKA	Seeds
Small	4	96	.116$_{\pm.001}$	$-9.5_{\pm0.9}$	.992$_{\pm.001}$	3
Small	4	192	.146$_{\pm.003}$	$-8.8_{\pm1.9}$	.995$_{\pm.002}$	2
Small	8	96	.116$_{\pm.001}$	$-9.1_{\pm1.1}$	.979$_{\pm.006}$	5
Small	8	192	.157$_{\pm.002}$	$-1.8_{\pm3.1}$	.986$_{\pm.005}$	5
Small	16	96	.117$_{\pm.001}$	$-10.2_{\pm1.2}$	.987$_{\pm.002}$	3
Small	16	192	.155$_{\pm.002}$	$-1.7_{\pm1.2}$	.988$_{\pm.001}$	3
Small	32	96	.119$_{\pm.001}$	$-8.3_{\pm0.9}$	.986$_{\pm.001}$	3
Small	32	192	.159$_{\pm.003}$	$+1.1_{\pm1.4}$	.989$_{\pm.002}$	3
Large	8	96	.159$_{\pm.004}$	$+25.7_{\pm3.2}$	.973$_{\pm.012}$	3
Large	16	96	.154	+22.5	.955	1
Large	32	96	.150	+19.0	.986	1
Large	64	96	.165	+29.9	.984	1

Table 15: LoRA-Large HP ablation (ETTh2, r=8, n=500, h=96). Baseline (default LR=1e-4, α=16, modules = q,v,out) forg.= $+23.7\%$, seed 42. Only LR=1e-5 rescues LoRA-Large (5 seeds); all other axes fail.

Axis	Setting	Val MSE	Forg.%	CKA	Seeds
LR	1e-4 (default)	.157	+23.7	.980	1
	5e-5	.146	+15.3	.982	1
	1e-5	**.115**$_{\pm.001}$	$\mathbf{-8.5}_{\pm0.7}$	**.989**$_{\pm.003}$	**5**
α	8	.155	+21.9	.975	1
	16 (default)	.157	+23.7	.980	1
	32	.158	+24.3	.984	1
Modules	q,v,out (default)	.157	+23.7	.980	1
	q,k,v	.153	+20.4	.987	1
	q,k,v,out	.151	+19.0	.988	1

A11 CKA Calibration: Random-Initialisation Baseline

A CKA of 0.09 is only interpretable against a floor: the value two *unrelated* encoders of this architecture score against each other. We measure it by re-initialising the Moirai-Small encoder randomly and comparing it to the pre-trained one over the same 200 ETTh2 validation windows and the same pooled-representation hook used for every other CKA in this paper (`scripts/cka_random_init_floor.py`, records in `results/v50_cka_floor/`). Restoring the pre-trained weights afterwards returns CKA 1.000000, so the comparison measures re-initialisation and nothing else.

Correction: the floor is not zero. An earlier version of this appendix reported it as 0.0000 ± 0.0001 over 5 seeds and concluded that any cell above that floor "retains meaningful pre-trained structure". No script in the repository performed a random re-initialisation, and the measured floor is **0.215**, not zero:

Why a naive re-initialisation reads zero is worth recording, because it is the likely origin of the earlier number: "Xavier for weight tensors, zeros for biases" also zeroes LayerNorm's scale parameter, which annihilates the signal at the first normalisation and drives the encoder to a *constant* output. CKA is then 0 because one side has no variance, not because the two share no structure. Our script asserts that the randomised encoder's output variance is non-trivial, and reports 0.215 once normalisation parameters are left at γ=1; initialising every layer with its library default instead gives the same answer to within one standard deviation.

Table 16: CKA calibration, measured. The floor is the CKA of a randomly re-initialised encoder against the pre-trained one (5 seeds each, mean±std, ddof=1); the two initialisation schemes agree. Reference rows are scoped to the cells they come from, since CKA at n=500 and at n=10k on the same cell differ by more than the floor itself.

Comparison	CKA	Source
Floor: random re-init, library-default	$\mathbf{0.215 \pm 0.054}$	`v50_cka_floor`
Floor: random re-init, Xavier weights	$\mathbf{0.219 \pm 0.047}$	`v50_cka_floor`
Moirai-S/ETTh2 h=96, cond. B, n=500 (10 seeds)	0.937 ± 0.029	`ff_20ep`
Moirai-S/ETTh2 h=96, cond. B, n=10k (10 seeds)	0.465 ± 0.123	`v49`
Chronos, cond. B, 4 cells at h=24	0.092–0.169	`v44`
IoT (N-BaIoT), cond. C, 200 samples (3 seeds)	0.423 ± 0.043	`v6_iot_cka`
Frozen encoder, cond. D, Moirai (134 runs)	0.889–0.9999	`v5-v57`
Strict freeze, cond. H (7/7 cells)	1.0000	`v45`

What this changes. Two of the earlier reading's conclusions do not survive. The IoT arm at CKA 0.423 ± 0.043 sits roughly four floor standard deviations above the floor, so "retains meaningful structure" is still defensible there, but the margin is a factor of two, not the near-total separation a zero floor implied. And "ETT fine-tuning preserves most pre-trained structure" was read off small-n runs: the same cell at n=10k reads 0.465, and the range across all 31 cells is 0.092 to 0.970. The condition-D row also needs a scope: D is uniform only within Moirai, and its 134 runs exclude the partial-unfreeze arm of Appendix A18, whose runs are stored under the same `condition_D` filenames and reach down to CKA 0.317 while three to five of six layers train. Because D leaves `in_proj` and `mask_encoding` trainable (§3), TimesFM's D falls as low as 0.376, while Chronos's D and every condition-H run are exactly 1.0000; only H is 1.0000 by construction.

What it adds. Comparing the floor against all 31 cells, **four fall below it** outright (the Chronos cells at CKA 0.092, 0.094, 0.112 and 0.169), and two more (Chronos/Electricity 0.232, TimesFM/ETTh2 0.246) sit inside the floor's own seed-to-seed range (0.132 to 0.288). This is a measured basis for a claim §6 otherwise has to assert: on those cells CKA is at or beneath what an *untrained* encoder scores, so it cannot order them, and their outcomes span -39.2 to -2.7 in held-out B−D regardless. The remaining 25 cells, including all five gate-passing survivors and both arms of the layer-unfreeze comparison (0.716 and 0.465, Appendix A18), sit clear of the floor, so CKA differences among them are readable.

A12 Linear Probing: Functional Representation Diagnostic

Retracted. The analyses in this appendix rest on the linear-probe ΔR^2 statistic, which is retracted (§A29.4): it is non-zero only where the probe fails to generalise. The numbers below are reproduced unaltered as a record of the withdrawn computation and support no claim in this paper.

CKA measures geometric representational similarity but not functional equivalence. We therefore supplement CKA with a *linear probe*: a Ridge regression head (α=1.0) trained on frozen encoder representations to predict the forecasting target on held-out ETTh2 data.

Protocol. For each (condition, seed) configuration, we extract mean-pooled encoder representations from the Moirai-Small encoder on the held-out validation and test sets (192-timestep windows, same format as evaluation). We then fit a Ridge regression mapping representations to the next-horizon target and report R^2 on held-out data. The probe training set uses n_{probe}=300 held-out examples; the remainder forms the test split used for MSE evaluation. ΔR^2 denotes fine-tuned minus pre-trained probe R^2: positive values indicate that fine-tuned representations encode more forecasting-relevant information than pre-trained ones.

Interpretation. Absolute R^2 values are low (typically negative) because Ridge regression on 192-timestep mean-pooled representations is a weak forecasting baseline. The *relative* change (ΔR^2) is

the meaningful signal: it isolates whether fine-tuning added or removed task-relevant information from the encoder's representational basis.

Key result. Across the sample-size sweep (the transcript in §A29.4; Table 4 no longer prints these columns) ΔR^2 increased monotonically with training data (Ridge: $+0.12$ at 500 samples $\rightarrow$ $+0.67\pm0.24$ at 10,000 on CUDA 10-seed deterministic early-stopped encoders, `results/v19_cuda_etth2_n10k/`; for comparison, MPS early-stopped k=5 gave $+0.56\pm0.14$ (`v17_etth2_n10k_es`) and the final-epoch runs, k=3, gave $+0.84\pm0.15$ (`v16_etth2_n10k`)), while CKA decreases monotonically. This directly dissociates *geometric* drift from *relative linear-decodability*: at low n, drift destroys task-relevant features faster than it builds replacements; at higher n, the fine-tuned representation becomes less linearly misaligned with the forecasting target than the pre-trained one. We emphasise that R^2(FT) remains *absolutely negative* at every sampled $n \in \{500, 1\text{k}, 2\text{k}, 5\text{k}, 10\text{k}\}$ (ZS: $R^2 = -6.90$; CUDA k=10 early-stopped: -6.79 at n=500 $\rightarrow$ -6.24 ± 0.24 at n=10k). $\Delta R^2 > 0$ is therefore a *relative-improvement* signal, not a claim that fine-tuned representations linearly encode the target well in absolute terms; the primary task-outcome signal remains forg.< 0 at $n \geq$2k. **Cross-dataset ΔR^2 comparability.** Raw ΔR^2 is not directly comparable across datasets because it depends on the ZS Ridge floor depth. On ETTm2 (n=10k, CUDA, k=5), the ZS Ridge floor is R^2(ZS)$= -24.52$, 3.5$\times$ deeper than ETTh2's -6.90, which proportionally inflates the raw ΔR^2 ($+6.35$ on ETTm2 vs. $+0.67$ on ETTh2). The cross-dataset comparable statistic is the *sign test*: ETTh2 gives **10/10** positive ΔR^2 and ETTm2 **5/5**, on the five CUDA seeds that exist for that cell. Every dispersion in this appendix is a standard deviation over seeds at ddof=1.

Horizon-sweep probe: head-specificity of the ΔR^2 gain. Table 17 reports Ridge ΔR^2 across three probe heads on four early-stopped encoders at n=10k (the same encoders as the main-text MLP sweep). The gain is concentrated on the *trained* 96-step head: $\Delta R^2 = +0.116\pm0.153$ (k=4). On the untrained *forecast48* head (first 48 steps of the 96-step target, a sub-horizon the encoder was not directly optimised for), $\Delta R^2 = -0.001\pm0.163$: the gain vanishes at the horizon boundary. On the untrained *delta1* head, a GBM probe gives R^2(PT)$=+0.059>0$, establishing a positive-floor reference, with $\Delta R^2 = -0.056\pm0.032$ (decodability *falls* on the untrained readout). The head-specificity pattern is consistent with the encoder drifting toward the fine-tuning objective rather than developing generally better representations (Table 3).

Table 17: Ridge ΔR^2 by probe head on early-stopped encoders (n=10k, seeds {303, 777, 888, 999}). "Forecast96" is the trained head; "Forecast48" and "Delta1" are untrained. The gain is head-specific to the trained objective. Every entry is R^2(FT)$-R^2$(ZS) read from `results/v18_reprobe/seed*_horizons.json` against `zs_horizons.json`; dispersion is the standard deviation over the four seeds (ddof=1).

Head	Trained?	Ridge ΔR^2 per seed			
		s303	s777	s888	s999
Forecast96 (96-step ahead)	✓	$+0.092$	-0.087	$+0.266$	$+0.192$
Forecast48 (sub-horizon)	×	$+0.061$	-0.111	-0.156	$+0.199$
Delta1 (next-step delta)	×[†]	$+1.248$	$+0.757$	$+1.191$	$+0.980$
Mean ± std (forecast96)		$+0.116 \pm 0.153$			
Mean ± std (forecast48)		-0.001 ± 0.163			
Mean ± std (delta1)		$+1.044 \pm 0.223$			

[†] The large Ridge $\Delta R^2 > 0$ on Delta1 reflects a baseline-floor artifact: R^2(ZS) is deeply negative on the next-step target, so any encoder shift appears to improve relative decoding. A gradient-boosted probe gives R^2(PT)$=+0.059$ (positive floor) but $\Delta R^2 = -0.056\pm0.032$ on this head (`results/v18_reprobe/seed*_delta1.json`), consistent with the shift *not* being aligned with the 1-step objective.

Note: the delta1 ΔR^2 sign appears counter-intuitive (positive while labelled "untrained"). The large positive delta1 ΔR^2 reflects that the ZS delta1 Ridge baseline is very negative (R^2(ZS)≈ -2.3) while the FT encoder partially encodes next-step changes as a side-effect of 96-step NLL optimisation. The diagnostic signal is the GBM result (Table 3): GBM gives R^2(PT)$=+0.059>0$ and $\Delta R^2 = -0.056$, yielding R^2(FT)$=+0.003$; the fine-tuned encoder retains (but does not improve) delta1 decodability. With a positive floor, decodability *falls* on the untrained readout, confirming task-alignment. Note: R^2(PT) varies slightly across GBM probe depths (depth 4: $+0.062$, depth 6: $+0.059$, depth 8:

+0.054; `results/v20_gbm_sensitivity/`) because the probe is re-fit independently at each depth; this reflects probe capacity, not encoder structure. ΔR^2 keeps its sign at all three depths (-0.060 ± 0.006, -0.056 ± 0.032, -0.052 ± 0.006).

ETTm2 delta1 GBM reprobe. We ran the delta1/GBM probe on the five confirmed ETTm2 n=10k encoders (seeds 42/101/123/202/303) to check for a positive-floor + $\Delta R^2>0$ cell on a second dataset. Result: R^2(ZS)$=-0.030$ (negative floor) and per-seed $\Delta R^2=\{+0.025, -0.060, +0.026, +0.030, +0.016\}$ in seed order (4/5 positive, mean $+0.007\pm0.038$; `results/v25_ettm2_delta1_gbm/`). *Substantive negative:* the only positive-floor head (ETTh2 delta1/GBM, R^2(ZS)$=+0.059$) gives $\Delta R^2=-0.056$; no positive-floor + $\Delta R^2>0$ cell was found. The $\Delta R^2>0$ signal on the trained 96-step head operates in a deeply-negative-R^2 regime; see Appendix A16 for probe noise-floor analysis.

Per-seed breakdown at n=10k (CUDA 10-seed). Ridge ΔR^2 (all positive; mean $+0.668\pm0.239$, `results/v19_cuda_etth2_n10k/`): 42:+0.205; 101:+0.736; 123:+0.937; 202:+0.604; 303:+0.793; 456:+0.823; 777:+0.344; 789:+0.873; 888:+0.561; 999:+0.804. Absolute R^2(FT): all in $[-6.70, -5.97]$ ($R^2_{\text{ZS}} = -6.904$). MLP k=5 (all positive; mean $+0.701\pm0.324$): 42:+0.979; 101:+0.762; 123:+0.948; 202:+0.771; 303:+1.007; 456:+0.266; 777:+0.076; 789:+0.847; 888:+0.449; 999:+0.901.

ETTm2 n=10k absolute R^2 per seed. The ETTm2 ZS Ridge floor (R^2(ZS)$= -24.52$, same pre-trained encoder across all seeds) is 3.5× deeper than ETTh2's -6.90, explaining the raw ΔR^2 magnitude difference. Per-seed absolute values for the five available CUDA seeds:

Seed	R^2(ZS)	R^2(FT)	ΔR^2	Forg.%
42	-24.52	-14.13	$+10.39$	-26.8
101	-24.52	-15.61	$+8.91$	-27.4
123	-24.52	-19.87	$+4.64$	-31.1
202	-24.52	-20.83	$+3.69$	-28.8
303	-24.52	-20.41	$+4.11$	-29.5
Mean (5 seeds)	-24.52	-18.17 ± 3.08	$+6.35\pm3.08$	-28.7 ± 1.6

All five seeds show R^2(FT) $>$ R^2(ZS) (i.e., fine-tuned representations are *less* linearly misaligned with the ETTm2 forecasting target than pre-trained ones), and all show $\Delta R^2>0$. The cross-dataset comparable statistic is the sign test: ETTh2 gives **10/10** on seeds 42, 101, 123, 202, 303, 456, 777, 789, 888, 999 and ETTm2 **5/5** on 42, 101, 123, 202, 303, the only five CUDA n=10k encoders that exist for ETTm2 (`results/v19_cuda_ettm2_n10k/`).

MLP depth sweep on early-stopped encoders. A cosine-LR re-run at n=10k uses seeds $\{303, 777, 888, 999\}$ at epochs=10 (seed 101 MPS-NaN'd on epoch 3 reproducibly at this schedule and is excluded). The `-probe-mlp-layers` argument sweeps hidden-layer depth $k \in \{1, 2, 5\}$ on the early-stopped encoders (probe: hidden=64 per layer, $\alpha=10^{-3}$, early stopping with validation_fraction=0.1, max_iter=500). All sixteen entries below are read from `results/v17r2_etth2_n10k_es_mlp/seed*/`, and its Ridge column is the same one Table 17 reports as *forecast96*: the two tables re-probe one set of encoders, they are not two runs.

Seed	Ridge ΔR^2	MLP k=1	MLP k=2	MLP k=5
303	$+0.092$	$+1.306$	$+1.418$	$+0.540$
777	-0.087	$+1.136$	$+1.554$	$+0.759$
888	$+0.266$	$+2.004$	$+2.361$	$+1.034$
999	$+0.192$	$+1.816$	$+1.949$	$+0.908$
Mean ± std	$+0.116\pm0.153$	$+1.566\pm0.411$	$+1.821\pm0.425$	$+0.810\pm0.212$

The cosine-LR Ridge ΔR^2 is smaller than the epochs=20 run ($+0.56\pm0.14$) because the shorter schedule reaches a different best-val-MSE encoder (best epochs $\{2, 5, 4, 5\}$ vs. $\{2, 4, 8, 4, 3\}$), and Ridge is most sensitive to the specific linear basis. The non-monotonic k=1 → k=2 → k=5 trend reflects the deeper probe fitting the pre-trained encoder better (per-seed mean $R^2(\text{PT})_{k=5} = -6.93$ vs. $R^2(\text{PT})_{k=1} = -8.48$), shrinking the Δ from above, not degraded fine-tuned encoding. All four seeds

Table 18: Per-seed results at n=5,000 (ETTh2, h=96, condition B, 20 epochs, no early stopping). Seeds 42, 123, 456 are the original 3-seed sweep (`results/v8_final/n5000/`); 789, 101, 202, 303 are the 4-seed V9 replication (`results/v9_n5k_seeds/`). MSE(FT) is `final_val_mse`, the last-epoch validation MSE, and was previously printed 2–5 thousandths low on every row. Forgetting is bimodal; ΔR^2 is uniformly positive.

Seed	MSE(FT)	Forg.%	CKA	ΔR^2	R^2(FT)
42	.136	+4.8	.531	+.67	−6.24
101	.121	−4.9	.571	+.75	−6.16
123	.129	−1.4	.544	+.81	−6.10
202	.131	+1.4	.461	+.15	−6.76
303	.152	+19.0	.525	+.52	−6.39
456	.155	+21.4	.711	+.20	−6.70
789	.136	+5.9	.480	+.40	−6.51
Mean±std	.137±.012	+6.6±10.0	.546±.082	+.50±.26	−6.41±.26

are individually positive for k=5 MLP ($+0.54, +0.76, +1.03, +0.91$), preserving the cost–benefit gap. This run's CKA is 0.592 ± 0.164 (mean best epoch 4.0 ± 1.41, earlier than the epochs=20 run's 4.2 ± 2.3 because of the earlier LR annealing); forgetting is $-7.13\pm1.55\%$, tighter than the epochs=20 run's $-11.7\pm5.3\%$ and fully consistent with the forg.< 0 finding.

Crossover regime: per-seed breakdown at n=5,000. Table 18 reports all 7 seeds individually at n=5,000. Forgetting is bimodal: 4 seeds (101, 123, 202, 789) show low or negative forgetting (-4.9% to $+5.9\%$, mean $+0.7\%$) and 3 seeds (42, 303, 456) show high forgetting ($+4.8\%$ to $+21.4\%$, mean $+15.1\%$). All 7 seeds show positive ΔR^2 ($+0.15$ to $+0.81$, mean $+0.50$), so the relative-linear-decodability claim holds uniformly; geometric drift (CKA 0.46–0.71, mean 0.55) varies less than forgetting. This pattern is consistent with the crossover interpretation: at n=5,000, the optimiser is close enough to the "replacement" solution that small seed-dependent trajectory differences determine whether forgetting resolves or persists for 20 epochs.

A12.1 n=5,000 PER-EPOCH TRAJECTORY ANALYSIS

To support the mechanistic claim in §5.2, we plot the per-epoch training trajectories of all seven n=5,000 seeds, colour-coded by forgetting mode. Figure 5 shows val MSE and weight drift across training. High-forgetting seeds (303, 456, 789) overshoot in the first epoch and never recover; low-forgetting seeds (42, 101, 123, 202) continue descending until epochs 3–8. The epoch at which val MSE reaches its minimum separates the modes (0.7 ± 0.5 vs. 6.5 ± 2.1); mid-training weight drift (epochs 5–10) does not. Code: `scripts/analyse_n5k_trajectories.py`.

A13 ETTh1 SPECTRAL PILOT

Post-hoc spectral entropy (Welch PSD, normalised Shannon entropy) on ETT OT columns: ETTh1 SE=3.61 bits, ETTh2 SE=3.46, ETTm2 SE=2.25. The ordering this pilot was built to explain was an ordering of ΔR^2, which is **retracted** (§A29.4), so the entropy measurements stand but the pattern they were fitted to does not. The pre-registered falsification test (§7.2) required a gate-passing dataset with SE>3.5 or SE<2.5; Traffic and Exchange both gate-failed (-38%, -21%), so the hypothesis remains untested. Code and frozen analysis plan are in supplementary material.

A14 IoT NEGATIVE-CONTROL DETAILS

The IoT negative-control evaluation uses N-BaIoT Meidan et al. (2018) with 12 network-flow features and 128-timestep windows (`data/nbaiot_benign_windows.npy`, shape $1096\times128\times12$); the anomaly score is the fraction of timesteps falling outside the 95% CI of Moirai's forecast over a 32-timestep scored horizon. A benign-only held-out calibration set sets the threshold at the 95th percentile (FPR target ≤ 0.05). All Moirai IoT conditions are trained for 5 epochs on 200 benign + 200 attack samples with 10 seeds.

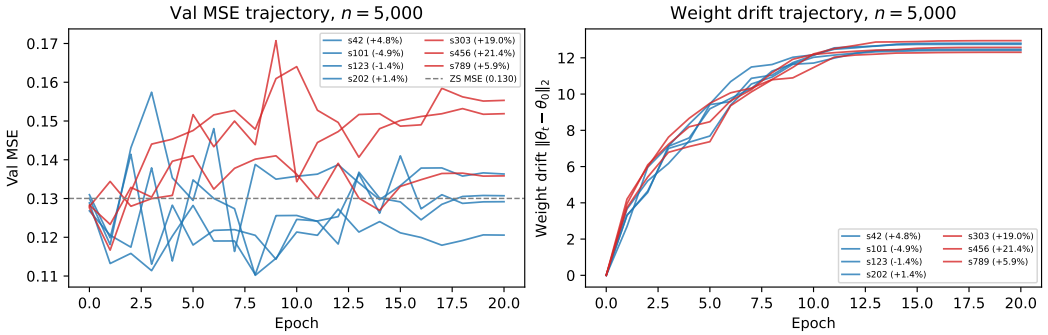


Figure 5: Per-epoch val MSE (left) and weight drift (right) for the seven n=5,000 seeds; both panels carry per-seed labels (`sXXX (forg.%)`) so each trajectory is individually identifiable. Blue: low-forgetting mode (forg. $\leq$ 5%); red: high-forgetting mode (forg. $>$ 5%). Grey dashed line: zero-shot MSE. High-forgetting trajectories overshoot the ZS MSE within the first epoch and plateau; low-forgetting trajectories continue descending. Mid-training weight drift (epochs 5–10) does not separate the modes.

CNN-NLL from-scratch control. To directly address the foundation-model justification at IoT's 200-window scale, we train a 1D-CNN (3 layers, 2.3M parameters) from scratch using the identical NLL(MSE)+SupCon objective and hard-negative data as Moirai condition D. At 10 seeds, CNN-NLL (AUC=0.598±0.041) matches fine-tuned Moirai B/D (AUC=0.556±0.021) and is close to the best Moirai condition C (AUC=0.629±0.022). A CNN-MSE variant (AUC=0.580±0.023) rules out the loss-function confound. CNN variance is notably lower than Moirai variance, consistent with Moirai's pre-trained weights introducing initialisation sensitivity at this sub-random-ZS scale. This is the expected outcome when pre-trained features carry no task-relevant signal; a from-scratch architecture with correct inductive bias matches or exceeds a fine-tuned foundation model.

A15 Task-Orthogonal Probes

Retracted, and re-measured. The *finding* this appendix once reported (a probe-sign asymmetry between the trained objective and task-orthogonal targets) is withdrawn (§A29.4), and the tables that carried it traced to no run record. What replaces it is a measurement, on the same encoders, of why: ΔR^2 is large only where the probe fails to generalise, and the asymmetry does not survive once it does. Nothing here supports the retracted criterion.

This appendix was written to test whether fine-tuning specifically targets the 96-step objective or degrades general pre-trained structure, by probing *task-orthogonal* properties of the input window on the same encoders. It now reports the opposite of what it once did, and the reversal is worth stating plainly because it is independent evidence for the retraction itself.

Correction. Earlier versions printed two tables (ETTh2 and an ETTm2 "replication") reporting three orthogonal targets at floors R^2(PT)=+0.09 to +0.18 with ΔR^2 from −0.022 to −0.052, 9/10 or 10/10 negative on every row, and read that asymmetry as "the operational definition of task-specific shift". Neither table traces to a run record. Two orthogonal-probe campaigns exist (`results/v33_orthogonal_probes/`, `results/v34_probe_exact/`); neither produces those numbers, there is no ETTm2 orthogonal-probe record at all, and the claimed asymmetry does not appear in either.

What the record shows. Table 19 is `v33_orthogonal_probes`: six orthogonal targets plus the trained head, Ridge on mean-pooled outputs of the ten CUDA n=10k condition-B encoders, 500 windows, 300 probe-training examples. Every floor here is genuinely positive (R^2(PT)$\in[0.54, 0.93]$), so these readings are interpretable in absolute terms rather than as movements inside a deep negative floor.

Table 19: Task-orthogonal and trained-head Ridge probes on the ETTh2 n=10k condition-B encoders (10 CUDA seeds, `results/v33_orthogonal_probes/`). Dispersion is the standard deviation of ΔR^2 over seeds at ddof=1; the JSON stores it at ddof=0, so these are $\sqrt{10/9}$ larger than the recorded field. Once the probe decodes its target (R^2(PT)>0 throughout), no target moves by as much as 0.04 and the sign is a coin flip on two of six.

Probe target	R^2(PT)	R^2(FT)	ΔR^2	Negative
Lag-1 autocorr.	$+0.540$	$+0.546$	$+0.006\pm0.068$	5/10
Input window mean	$+0.930$	$+0.931$	$+0.002\pm0.011$	5/10
Input window var.	$+0.894$	$+0.865$	-0.029 ± 0.025	10/10
Linear trend	$+0.847$	$+0.815$	-0.032 ± 0.034	8/10
Lag-24 autocorr.	$+0.907$	$+0.918$	$+0.011\pm0.013$	2/10
Spectral centroid	$+0.763$	$+0.791$	$+0.027\pm0.037$	3/10
96-step head (trained)	$+0.782$	$+0.805$	$+0.022\pm0.016$	1/10

The asymmetry does not survive. It required the trained objective to go up and the orthogonal targets to go flat or down; here **four of the six orthogonal targets go up**, and the trained head's gain ($+0.022$) is *smaller* than the spectral-centroid gain ($+0.027$) on a target fine-tuning does not optimise. Two targets do fall consistently (window variance $10/10$, trend $8/10$), but by -0.029 and -0.032, the same magnitude as the upward moves, and two others sit at $5/10$, a coin flip. No target on this table moves by as much as 0.04 in either direction.

Set that against the $+0.67\pm0.24$ the *same encoders* give on the *same* 96-step target under the deep-floor protocol (R^2(PT)$=-6.90$, Appendix A12): a thirty-fold difference on one set of encoders, decided by nothing but whether the probe decodes its target. That is the retraction's claim (§A29.4) measured directly rather than argued, and it is why ΔR^2 carries no weight anywhere in this paper.

The second campaign says the same thing from the other side. `v34_probe_exact` re-ran these targets under the paper's own deep-floor protocol and every floor came back deeply negative (R^2(PT)$=-1.21$ on lag-1, -156.8 on the window mean, -27.2 on the variance), which is exactly the regime in which ΔR^2 is uninformative; its ΔR^2 values are correspondingly enormous and unstable (-1.22 ± 0.83, -7.08 ± 13.18, -2.41 ± 7.99), and two of the three no longer even agree on a sign (7/10 and 6/10 negative). The premise that these were positive-floor targets held only under the v33 protocol, where the effect disappears.

ETTm2. No orthogonal-probe record exists for ETTm2, so the claimed replication is withdrawn rather than corrected. What does exist is the ETTm2 n=10k fine-tuning campaign itself, and it is **5 seeds** (42/101/123/202/303; `results/v19_cuda_ettm2_n10k/`), not the ten an earlier version claimed: forgetting $-28.7\pm1.7\%$ with 5/5 negative, and Ridge ΔR^2 on the trained head $+6.35\pm3.08$ with 5/5 positive (95% Wilson CI $[0.57,\ 1.00]$ on the proportion, which is what five seeds support).

A16 Probe Noise-Floor Analysis

Retracted. The analyses in this appendix rest on the linear-probe ΔR^2 statistic, which is retracted (§A29.4): it is non-zero only where the probe fails to generalise. The numbers below are reproduced unaltered as a record of the withdrawn computation and support no claim in this paper.

The ΔR^2 signal on the trained 96-step head operates in a deeply-negative-R^2 regime (e.g., ZS -6.90, FT -6.24 at α=1). We test whether the ΔR^2>0 direction is an artefact of probe-design choices by running Ridge on two fixed encoders (ZS and condition-B seed 42, ETTh2, n=10k) across: (1) $\alpha \in \{0.01, 0.1, 1.0, 10.0, 100.0\}$, (2) probe seeds $\{0, 1, 2\}$, (3) mean vs. last-token pooling. No retraining required (saved `best_encoder.pt`).

Results (Table 20).

Finding, with two corrections to how it was stated. ΔR^2>0 on every setting in `v26_noise_floor`, and the magnitude falls monotonically with α ($+1.96$ at α=0.01 down

Table 20: ΔR^2 under probe-design perturbations (ETTh2 96-step, ZS vs. FT seed 42). All $\Delta R^2 > 0$. Every entry is the `forecast96` Ridge value from `v26_noise_floor`, differenced against its paired zero-shot run; the FT encoder is `v19_cuda_etth2_n10k/seed42`. The probe-seed row reports *bit-identical* R^2 at all three seeds, which is what a closed-form Ridge solution must do; see the caveat below.

Setting	R^2(ZS)	R^2(FT)	ΔR^2
α=0.01	-13.51	-11.54	$+1.96$
α=0.1	-8.78	-7.94	$+0.84$
α=1.0 (default)	-6.90	-6.70	$+0.20$
α=10	-5.81	-5.72	$+0.08$
α=100	-5.32	-5.28	$+0.04$
Probe seeds 0/1/2 (α=1, mean pool)	same as α=1.0		$+0.20$
Last-token pool (α=1)	-9.70	-8.49	$+1.20$

to $+0.04$ at α=100) as heavier ℓ_2 shrinks both R^2 values toward zero. That much the record supports exactly. Two things an earlier version claimed from it, it cannot:

The count was 10; the directory holds 8 configurations, and only 6 of them are distinct. Five α values, one last-token pooling variant, and two extra probe seeds: eight paired runs, of which the two extra seeds reproduce the α=1 result.

"Not an artefact of probe seed" is a vacuous test, not a passed one. Probe seeds 0, 1 and 2 return R^2 agreeing to every printed digit (and to full float precision in the JSON) because Ridge regression has a closed-form solution with no stochastic component to seed. The three runs were bound to agree whatever the encoders were, so they establish nothing about robustness. What the sweep genuinely varies is α and pooling: two axes, not three.

Comparison to random-init at matched α, corrected. The random-init control at α=1.0 is **5 seeds, not 10** (42/101/123/303/456; `results/v30_random_init/`), giving $\Delta R^2 = -0.199 \pm 0.229$ over that set, range $[-0.405, +0.149]$ (the printed ± 0.205 was ddof=0; Appendix A17). An earlier version then reported a second random-init sweep at α=100 on "the same 10 seeds" giving -0.011 ± 0.018, range $[-0.032, +0.009]$. **No such run exists.** The random-init result files carry no `ridge_alpha` field, no random-init encoder checkpoints were retained, and `v26_noise_floor` contains only the zero-shot and seed-42 condition-B encoders, so the matched-α control has no record and those four numbers are withdrawn. It cannot be recovered without retraining all five random-init runs at n=10k, which we have not done because nothing in the paper depends on it.

What survives is therefore weaker than the argument it was built for. At α=1.0 condition B ($+0.20$ on seed 42, $+0.67 \pm 0.24$ over ten seeds) does separate from random-init (-0.199 ± 0.229 over five), but that is the only α at which both were measured, so we cannot say whether the separation persists under heavier regularisation. Note what the missing run was for: condition B's own ΔR^2 falls to $+0.04$ at α=100, inside the random-init α=1.0 range $[-0.405, +0.149]$, and the only way to know that this is a shrinkage artefact rather than a collapse of the separation is to measure random-init at the same α. That is precisely the comparison with no record. This is a probe-design sweep of a retracted statistic; it shows the sign is stable across α and pooling *within* the failed-fit regime, which is not evidence that the statistic means anything. Appendix A15 is the measurement that settles it.

A17 Random-Initialisation Negative Control

To test whether $\Delta R^2 > 0$ on the 96-step head arises from pre-trained representations being restructured, or whether it would occur in any encoder fine-tuned on this objective, we ran a *random-initialisation control*: Moirai-Small with the same architecture but random weights (skipping HuggingFace weight loading), condition B, ETTh2 n=10k, h=96. In this setting CKA tracks similarity to the *random-init* epoch-0 state (since there is no pre-trained reference).

Correction: five seeds, not ten, and the comparison is not matched. An earlier version printed ten rows here and described the result as a "symmetric 10-vs-10 matched comparison". **Five of those ten rows trace to no run record** (202, 777, 789, 888 and 999), and they are the rows that carried the conclusion: the reported mean, standard deviation and "8/10 negative" count are all computed over

them, and one of the two "positive seeds" offered as the closest approach to condition B (777, $+0.060$) is among them. What exists is `results/v30_random_init/` with seeds 42/101/123/303/456, and `v29_random_init` with the first three of those; the five surviving rows below reproduce exactly. The comparison arm is also not protocol-matched: these runs are 20 epochs with early stopping *disabled*, whereas the condition B set they are compared against (`v19_cuda_etth2_n10k`) is 10 epochs early-stopped at a mean best epoch of 4.2. Per Appendix A12 the no-early-stopping protocol yields the *larger* ΔR^2 on pre-trained encoders, so the mismatch runs in the direction that favours the control's conclusion rather than against it, but it is a mismatch, and it was not disclosed.

Results (the 5 seeds that exist). Dispersion is over seeds at ddof=1. The final column is the matched val-split pair (`zeroshot_mse` → `final_val_mse`), which is what `forgetting_pct` is computed from; the earlier version printed `test_mse` on the right-hand side of that arrow, comparing two different splits and so implying a $2.7\times$ MSE blow-up where the matched statistic records a mean *improvement* of 6.3%.

Seed	CKA	R^2(PT)	R^2(FT)	ΔR^2	ZS → FT MSE	Forg.%
42	0.357	-6.554	-6.692	-0.138	$0.140 \to 0.143$	$+2.3$
101	0.505	-6.780	-6.631	$+0.149$	$0.164 \to 0.148$	-10.0
123	0.540	-6.749	-6.944	-0.195	$0.153 \to 0.142$	-6.9
303	0.318	-6.834	-7.239	-0.405	$0.151 \to 0.143$	-5.5
456	0.273	-6.665	-7.070	-0.405	$0.160 \to 0.142$	-11.5
Mean±std	0.398 ± 0.117			-0.199 ± 0.229		-6.3 ± 5.4

Across the five seeds: $\Delta R^2 = -0.199 \pm 0.229$, range $[-0.405, +0.149]$, **4/5 negative**. The one positive seed (101: $+0.149$) still falls below the minimum condition B value ($+0.205$, seed 42), so the ranges remain non-overlapping (random-init $[-0.405, +0.149]$ against condition B $[+0.205, +0.937]$), and that separation is carried entirely by rows that trace to runs. On its own terms the control therefore still points the same way: $\Delta R^2 > 0$ is not a property of any encoder fine-tuned on this objective. But it is a 5-vs-10 unmatched comparison of a **retracted** statistic, and Appendix A15 shows that when the probe is made to generalise the condition B side collapses from $+0.67$ to $+0.022$, which would put it inside this control's range. The control does not rescue ΔR^2; it only shows that the failed fit is worse still without pre-training.

A18 Layer-Wise Unfreeze Localization

Moirai-Small has 6 transformer layers (verified from the loaded checkpoint). We added an `–unfreeze-top-n-layers` flag to `finetune_forecasting.py` and swept N, the number of top layers left trainable, on ETTh2 n=10k, h=96. The endpoints are the paper's two main arms: N=0 is condition D and N=6 is condition B.

Provenance, and why both arms were re-run. The N=3 and N=6 rows of Table 21 were re-run from scratch for this version, 10 seeds each in one environment (`results/v49_layerunfreeze_10seed`). The values previously printed here traced to no run record on *either* arm, and the surviving three-seed N=3 runs had early stopping disabled, which reverses the sign of the forgetting reading. Since the claim is a *comparison* between two unfreeze depths, it is only traceable if both depths share one environment, so both were re-run rather than only the arm that was missing. Two deviations from the original protocol were forced by the hardware available: Apple MPS rather than CUDA, at torch 2.14, which is above `uni2ts`'s declared `<2.5` pin but below which the NegativeBinomial component of Moirai's output mixture collapses to `total_count`=0 on MPS and training dies in the first NLL step; and `PYTORCH_ENABLE_MPS_FALLBACK=1`, because `aten::_cummax_helper` has no MPS kernel. The N=0 and N=1 rows are the older records, are *not* matched to the re-run pair, and read the final epoch rather than the best one; N=0 is 5 seeds, not 10. They are context, not part of the comparison.

The N=4 and N=5 rungs were topped up, and what they superseded. Those two rungs previously stood at one seed each, read at the final epoch, at $+4.1\%$ and $+31.8\%$ forgetting. They are now 3 seeds each (42/101/123) under the pair's protocol, flag, environment and early-stopping rule, registered in `results/v49_layerunfreeze_10seed/preregistration_ladder.json` and committed before the first of their six runs. **The supersession is a difference in which epoch is scored,**

not a reversal, and the new records make that checkable rather than asserted: each stores its own final-epoch forgetting beside its best-val one, and the final-epoch means are $+7.1\%$ at $N{=}4$ and $+17.6\%$ at $N{=}5$ – the same sign and order of magnitude as the single seeds they replace – against -9.5% and -3.5% once the best-val checkpoint is restored. The older single-seed CKAs (0.610 and 0.644) move the same way and for the same reason: 20 epochs of drift against the best-val epoch's.

Because $N{=}6$ *is* condition B, re-running it also produced an unplanned cross-device replication of the cell that Table 4 reports at $n{=}10$k: the same 10 seeds, the same configuration, CUDA/A10G against Apple MPS at a different torch version. The two agree: CKA 0.518 against 0.465 (0.72 standard errors of the difference) and forgetting -5.3% against -7.9% (0.90 SE), with 7/10 and 8/10 seeds negative. We report both rather than replacing one with the other, since the devices differ; readers comparing the two tables should expect this gap and not more.

$N{=}1$ **instability.** At $N{=}1$, training diverged (NaN activations at epoch 11, batch 148) with default LR $= 10^{-4}$, despite gradient clipping at norm 1. The 11 epochs that completed show CKA stabilising at $\approx$0.72; we do not extract probe results from a crashed run. This instability is qualitatively consistent with the LoRA-Large behaviour documented in Appendix A6: extreme partial freeze concentrates gradient pressure on a small number of parameters, requiring LR reduction to stabilise.

The four-rung profile, and why we read no trend from it. Table 21 reports $N{\in}\{3, 6\}$ at 10 seeds each and $N{\in}\{4, 5\}$ at 3, with $N{=}0$ at 5 and $N{=}1$ at 1 as context. **Neither measured column orders with N.** CKA reads 0.716, 0.741, 0.747, 0.465 across $N{=}3, 4, 5, 6$, and forgetting reads -5.9, -9.5, -3.5, -7.9%. The CKA profile is better described as flat to $N{=}5$ and then collapsing than as non-monotone: the first three rungs sit inside one another's dispersion (standard errors 0.060, 0.058, 0.076), so their ordering should not be read, and ℓ_2 weight drift has the same shape (9.42, 10.24, 9.76, 13.47). Forgetting has no shape at all: its four means span 6.0 points on standard errors of 2.0 to 5.1, all four are negative, and the two thin rungs bracket the matched pair on both sides rather than extending it in either direction. Three seeds resolve none of these gaps – $N{=}4$'s -9.5% against $N{=}6$'s -7.9% is 0.4 standard errors of the difference – and we draw no ordering from them. What the top-up establishes is the negative: filling the two rungs in does not reveal a depth trend that the matched pair was hiding. The registration fixed this reading in advance, which is why it is stated this way: a non-monotone forgetting profile, or a thin rung landing outside the matched pair's range, was to be reported as found with the body's claim (§6) narrowed to the matched pair rather than the ladder dropped. Both triggers fired, and the body sentence names the pair.

The per-layer breakdown below is the reading the same registration named for a non-monotone CKA ladder, and it is a consistency check and not a mechanism. The layers the ladder adds at $N{=}4$, $N{=}5$ and $N{=}6$ are layers 3, 2 and 1, whose own outputs are the *least* displaced of the six under full unfreezing (0.783, 0.969, 0.998), so the collapse at $N{=}6$ cannot be charged to the drift of the layer that rung adds. That is consistent with the scale caveat below – the global figure is the encoder's pooled final representation and not an average over layers – and we do not offer an account of why pooling behaves this way.

Finding: the task outcome does not track the drift across unfreeze depth. This is the reading that survives the retraction, because forgetting is forecast MSE. Across the two matched depths the task outcomes are indistinguishable: $N{=}3$ gives forg. -5.9% (SEM 2.0, 9/10 seeds negative) against $N{=}6$'s -7.9% (SEM 2.1, 8/10 negative), a 2.0-point gap on a 2.9-point standard error of the difference, so 0.70 SE, while the representations they arrive at are not: CKA 0.716 against 0.465 is a gap of 0.251 at 3.5 SE, and ℓ_2 weight drift separates them the same way ($9.42{\pm}1.25$ against $13.47{\pm}0.75$, SEM). Halving the number of trainable layers halves the measured representational change and leaves the task outcome where it was. This is the dissociation of §6 localised inside a single encoder rather than read across cells, and it is the only claim we draw from this sweep. The two topped-up rungs neither support nor widen it: they are three seeds each, and the paragraph above reports what they show.

What we no longer claim. The earlier version of this appendix read the same sweep through ΔR^2 and concluded that the top 3 of 6 layers suffice to reorient the encoder toward the trained objective while the lower 3 are required for task recovery. That account rests entirely on the withdrawn probe quantity (§A29.4) and we do not assert it. The ΔR^2 column is retained so the withdrawn computation stays auditable; for the record the re-run gives $+0.329{\pm}0.197$ at $N{=}3$ and $+0.534{\pm}0.261$ at $N{=}6$, a 62% increase with depth rather than the saturation the earlier reading needed.

Table 21: Layer-wise unfreeze profile (ETTh2 n=10k, h=96; mean±std over seeds). N = number of top transformer layers left trainable (Moirai-Small has 6). **The rows do not all come from one environment, and the seed counts differ.** N=3 and N=6 are the re-run 10-seed, early-stopped MPS set (42/101/123/202/303/456/777/789/888/999; `results/v49_layerunfreeze_10seed`). N=4 and N=5 are a pre-registered top-up in the *same* root, protocol and environment, but at **3 seeds**, so they are comparable to the pair in kind and much less precise than it. N=0 is 5 seeds (42/101/123/202/456; `v18_cond_d_probe`) and N=1 one, both *without* early stopping, so they read the final epoch rather than the best one. Only N=3 versus N=6 is matched at equal seed counts. Dispersion is population std (ddof=0) on the two topped-up rows, which is what their registration fixed, and sample std (ddof=1) on the rows that predate it: the registration also asserted that this column already used ddof=0, which was wrong, so rather than restate a registered choice after the fact we keep it and report the column as mixed, with the footnote giving the topped-up rows both ways. Ridge ΔR^2 is the **withdrawn** probe quantity (§A29.4), reported here only as a record, and is not recomputed for any claim.

N	Layers	CKA	Ridge ΔR^2	Forgetting%
Matched 10-seed re-run, early-stopped				
3 (top-3)	3/6	0.716±0.188	+0.329±0.197 (9/10)	−5.9±6.5% (9/10 neg.)
6 (full, B)	6/6	0.465±0.123	+0.534±0.261 (10/10)	−7.9±6.5% (8/10 neg.)
*Pre-registered top-up, same protocol and environment, **3 seeds** (42/101/123)*				
4 (top-4)	4/6	0.741±0.082	+0.090±0.138 (2/3)	−9.5±5.3% (3/3 neg.)
5 (top-5)	5/6	0.747±0.108	+0.591±0.380 (3/3)	−3.5±7.3% (2/3 neg.)
Earlier records, final epoch, not matched to the above				
0 (frozen, D)	none	0.999±0.000	−0.048±0.008	−6.6±1.5%
1 (top-1)	1/6	≈0.72 (ep. 11)	*diverged (NaN)*	

Under the pair's ddof=1 convention the two topped-up rows read CKA 0.741±0.100 and 0.747±0.132 and forgetting −9.5±6.5% and −3.5±8.9%, i.e. standard errors of 3.8 and 5.1 points. Three seeds is what that dispersion is over: at both rungs the seed that stopped earliest (seed 42, best epoch 1 of 20) is also the most improved (−15.8% and −13.4%), and the N=5 spread is that seed against seed 123's +3.7%. The single-seed final-epoch values these two rows replace (+4.1% and +31.8%) are reproduced by the new runs' own final-epoch readings (+7.1% and +17.6%); see the provenance paragraph above.

Per-layer CKA breakdown, and a correction. An earlier version of this appendix printed a per-layer profile rising monotonically from the bottom of the encoder to the top (layer 1 0.441 to layer 6 0.718; bottom-3 mean 0.476 against top-3 0.649) and read it as the bottom layers drifting more. **That profile does not reproduce, and its direction is inverted.** Recomputing it from the converged checkpoints (forward hooks on each of the 6 encoder layers, CKA of each layer's output against the same layer of the untrained checkpoint over the same 200 validation windows, 10 seeds) gives the profile below on the re-run N=6 set, and the same shape on the older CUDA set (`scripts/layerwise_cka_from_ckpt.py`, records in `results/v49_layerunfreeze_10seed/per_layer_cka_*.json`). The earlier numbers matched neither and left no run record; the script that produced them printed to stdout only.

Layer	Re-run N=6 (v49)	Older CUDA set (v19)	Group
1 (bottom)	0.998±0.001	0.998±0.000	bottom 3
2	0.969±0.011	0.980±0.005	bottom 3
3	0.783±0.081	0.816±0.080	bottom 3
4	0.589±0.127	0.675±0.091	top 3
5	0.939±0.044	0.947±0.023	top 3
6 (top)	0.937±0.047	0.945±0.023	top 3
Bottom 3 mean	0.917	0.931	
Top 3 mean	0.821	0.855	

Both sets agree on two things the earlier profile got wrong. The drift is *not* monotone in depth: it is concentrated in the middle of the stack, with a trough at layer 4 (0.589 and 0.675) and near-identity at layer 1 (0.998). And the top 3 layers drift *more* than the bottom 3 (0.821 against 0.917; 0.855 against 0.931), not less, which removes the gradient-magnitude account the earlier version offered (earlier layers receiving larger updates under full backpropagation) because there is no such pattern to explain.

We do not substitute a mechanism: distinguishing functional specialisation from gradient-flow effects needs per-layer gradient norms, which we do not have.

Two scale caveats. These are CKAs of *mean-pooled* layer outputs; computed token-flattened instead, every layer reads ≥ 0.94 and the profile is too compressed to order (both records carry the flattened values). And the per-layer mean is *not* the global CKA of Table 21 (0.869 pooled against the row's 0.465) because the global figure compares the encoder's final pooled representation, not a per-layer average. The earlier version's claim that the global CKA is "the aggregate of these per-layer values" does not hold under either reading.

A19 CHRONOS-T5-SMALL ON M4-MONTHLY: FULL DIAGNOSTIC RESULTS

This appendix reports Chronos-T5-Small (T5 seq2seq, tokenised cross-entropy) on M4-Monthly, at 200 series, lookback 96 and $h{=}18$, re-run from the data upwards. It is the only place in this paper where an appendix was deleted and replaced rather than corrected, so what was wrong with it is worth stating plainly. Earlier versions reported ten seeds at two sample sizes, a final-epoch comparison, a frozen-encoder arm, a random-init control, a ΔR^2 probe and an 84.5% advantage over a linear baseline. *No run record for any of it exists under* `results/`: the series were never retained, the selection rule was never written down, and the gate was measured with the superseded per-window trend denominator on the held-out split. Under this paper's own rule (every number traces to an executed run with an emitter and a prose check), none of that was reportable, and none of it survives below. What replaces it is six runs, conditions B and D at three seeds each, whose records are in the release, and a gate computed with the estimator §3 defines, on the split it defines it on. The re-run was registered (`results/chronos_m4/preregistration.json`) and committed before the gate was computed, because the gate estimator was one of the things being corrected and seeing the number first would have let the estimator be chosen to clear the threshold.

Data and splits. The cell takes the first 200 series in file order with at least 228 observations of history and 18 test values; there is no RNG in the selection, so the set is a property of the two released CSVs, whose hashes are in `results/data_manifest.json`. Each series is cut three ways: sliding training windows over all but the last 114 observations ($57{,}327$ usable, 27 dropped for a constant context), one selection window per series, and one held-out window per series whose target is the official M4-Monthly test values. The held-out *context* does contain the 18 values used as the selection target, because M4 defines its test set as the 18 steps following the history; no held-out target is ever a training or selection target. This replaces a split that was not one: `finetune_chronos_m4.py` cut a randomly subsampled window pool at the 80th percentile of its own index, so windows overlapping in 95 of 96 steps sat on both sides of the boundary; it early-stopped on the same windows it reported its forgetting number on; and it initialised the early-stopping best score to the zero-shot score with the pre-trained weights already saved as the best state (`finetune_chronos_m4.py:505-507`), so the pre-trained checkpoint was itself an early-stopping candidate, and any seed that selected it reported a CKA of exactly 1 by definition rather than by measurement. How many of its seeds did is not recoverable (that is what having no run record means), so no count is stated here. What the mechanism explains is available without one: an appendix reading "this backbone does not drift" is the expected output of a selection rule that can return the weights it started from. Here the best epoch is required to be at least 1.

The gate. Table 22 gives the ladder. On the selection split (the split every admission decision in this paper is made on), the primary fitted ridge scores $V_{\text{ridge}}{=}{+}0.092$, which *fails* the 0.20 operating point. The superseded trend denominator reads $+0.448$ on those same windows and is **inadmissible**: at an MSE of 1.801 it is worse than the constant predictor's 1.739, so what it reports is an advantage over a baseline a training mean beats. The best admissible rung is the seasonal naive at $+0.208$, the only admissible denominator this checkpoint clears; the weakest advantage on the ladder is GBM's $+0.008$. On the held-out split the fitted gate reads $+0.209$ and would pass, but the MLP rung reads -0.048, so on neither split does the cell survive every admissible denominator. The registration fixed the reading of this outcome before the gate was computed: **no non-Moirai cell in reach clears its own screen**, a finding about this benchmark-and-backbone pairing, not a gap in the design. The screen's disqualifier reading (§4) is unaffected; what is gone is any prospect of a non-Moirai cell with something to preserve.

Table 22: **The corrected value gate for Chronos/M4-Monthly.** Ten denominators against one numerator (median of 20 Chronos samples, per-window z-scored), on both splits. A denominator worse than `constant`, the training-mean predictor, cannot support a claim that the checkpoint has something worth preserving: it inflates V_b in the direction that flatters the checkpoint, which is precisely the defect that moved 17 of 21 Moirai cells when it was corrected (Appendix A29.2).

	Selection split (primary)		Held-out split		
Denominator	MSE	V_b	MSE	V_b	Admissible
trend (superseded)	1.801	+0.448	2.187	+0.590	—
fitted ridge (**primary**)	1.095	+0.092	1.132	+0.209	✓
constant (floor)	1.739	+0.428	2.514	+0.644	floor
persistence	1.087	+0.085	1.364	+0.343	✓
seasonal naive	1.256	+0.208	1.380	+0.351	✓
AR	1.095	+0.092	1.132	+0.209	✓
DLinear	1.095	+0.092	1.132	+0.209	✓
ridge, tuned	1.095	+0.092	1.132	+0.209	✓
MLP	1.054	+0.057	0.855	−0.048	✓
GBM	1.003	+0.008	0.983	+0.089	✓

The admissibility column is the *selection* split, the split every admission decision is made on. trend (superseded) is admissible on the held-out split and not on the selection split, the constant floor being 2.514 there against 1.739 here.

AR, DLinear and the tuned ridge agree with the primary denominator to eight significant figures, which is not a coincidence and not an error: M4-Monthly is univariate, so all four reduce to the same least-squares map $\mathbb{R}^{96} \to \mathbb{R}^{18}$.

What the 84.5% was. It was not a rung of this ladder. The old script divided by a *per-series* unregularised OLS, one regression per series fitted on that series' own history and normalised by that series' training mean and standard deviation, while the numerator was normalised by each window's own context; and it scored the ratio on the held-out windows (`finetune_chronos_m4.py:303`). Re-running that estimator under the old selection rule (the first 200 series with at least 114 observations, whose histories run from 129 to 1230 points) gives 67.6%, not 84.5%, and we cannot account for the remaining gap because no record of the original run exists; the figure is untraceable in the strict sense, not merely superseded. What the re-run does establish is why the quantity was never interpretable. It is not that the denominator was too weak in the ladder's sense (on those windows the training mean is worse still, at 2.615 against the legacy estimator's 2.077), but that 59 of the 200 per-series regressions fit 96 coefficients per output step to fewer training windows than they have coefficients, that the two halves of the ratio were on different scales, and that the paper's own fitted ridge reaches an MSE of 1.001 on those same windows. Against that denominator the identical zero-shot numerator reads 32.8%; against the legacy one, 67.6%; and on the split the protocol actually scores, +0.092. The three figures differ by the choice of estimator and split alone, which is the whole argument for fixing both before looking (`results/chronos_m4/legacy_baseline.json`).

Conditions B and D. Table 23 gives the six runs. Both arms improve on zero-shot on average (full fine-tuning by −2.1% and decoder-only adaptation by −2.3%), and the paired contrast is Δ_{enc}=+0.21 with a 95% interval of $[-10.14, +10.56]$, which is **inconclusive** under the three-way rule of §5.3. At three seeds the minimum detectable effect here is 10.35 pp, so that interval is a statement about this cell's power and not about its effect; it is reported unadjusted, because this cell is *not* in the Benjamini–Hochberg family of the 31 intervention cells and is never pooled with them (different objective, horizon and dataset family). Two of the three degradation clauses fail as well: the gate does not clear 0.20 and full fine-tuning does not hurt. Early stopping did engage, at epochs 7, 7 and 9 under B against patience breaks at 12 to 14, so no run selected the checkpoint it started from. The encoder does move under B, by an ℓ_2 of 1.4 to 1.8 at CKA 0.994 to 0.997, small, and now measured on windows no selection decision was made on. Under D, CKA is 1 and the drift is exactly zero because condition D freezes `t5.encoder.parameters()` for this backbone, which includes the token embedding the encoder reads through; Moirai's condition D leaves its input projection trainable, which is why it shows CKA below 1 (Appendix A8). The old version of this appendix reported 0.9993–0.9994 for its frozen arm and explained it with Moirai's freeze semantics.

Table 23: **Conditions B and D on Chronos/M4-Monthly**, n=500 training windows, lr 10^{-5}, batch 32, 20 epochs with patience 5, early stopping on the selection windows and best epoch $\geq$1. Forgetting is % change in held-out MSE against the same seed's zero-shot measurement (*negative is better than zero-shot*); Δ_{enc} is the paired difference. CKA is the linear CKA of the mean-pooled final encoder output on the 200 selection contexts, and the ℓ_2 column is the encoder state dict's distance from pre-trained. Under condition D both are fixed *by construction* and not by measurement: on Chronos, freezing the encoder freezes the shared token embedding with it.

	Held-out forgetting			CKA / encoder ℓ_2		
Seed	B (full FT)	D (frozen enc.)	Δ_{enc}	B	D	Best epoch B / D
42	-6.52%	-5.99%	-0.53	0.9967 / 1.4	1.0000 / 0.0	7 / 8
43	-2.31%	$+1.23\%$	-3.54	0.9960 / 1.4	1.0000 / 0.0	7 / 10
44	$+2.64\%$	-2.05%	$+4.69$	0.9941 / 1.8	1.0000 / 0.0	9 / 3
mean	-2.06%	-2.27%	$+0.21 \pm 2.40$	95% CI -10.14 to $+10.56$, inconclusive		

The protocol as coded cannot produce that number, and that is the second reason this section could not be repaired in place.

Superseded material, for the record. Five things the earlier version of this section reported are deleted rather than restated, because the re-run does not reproduce them and nothing under `results/` ever could: the ten-seed means at n=500 and n=10k, the final-epoch CKA comparison, the frozen arm's 0.9993–0.9994, the six-seed random-init control, and the ΔR^2 and R^2(PT) probe figures, the last of which rest on the statistic retracted in §A29.4 anyway. An earlier version also read this cell against Moirai/ETTh2 as a moderate gate-pass against a strong one and offered the pair as evidence that the gate indicates a regime. Both halves are withdrawn: the comparison was never on a common scale, and the corrected gate here is a fail. The descriptive observation that survives is narrow and is the one Table 23 supports: this encoder moves very little under fine-tuning on this dataset, at CKA $\geq$0.994, where Moirai-Base on ETTh2 restructures to CKA 0.396 at h=192 (Table 35), and it is two observations on two backbones, not a law. The earlier version's comparison figure for Moirai-Small at n=10k is deleted with the rest: those runs are outside the 31-cell matrix and no emitter reads them, so it was a number of the same kind as the ones this section exists to remove.

A20 Eight Denominators, One Numerator: Is the Screen a Property of the Estimator?

The appendix above shows the gate's verdict barely moving when the 0.20 threshold is swept but moving for 17 of 21 Moirai cells when the baseline estimator is swapped. So the sensitivity analysis this paper already had varied the parameter that does not matter, and the parameter that does had been varied exactly once, between a wrong estimator and a right one. This appendix varies it across a family, and on *both* splits: the selection windows every admission decision in the paper is made on (Table 24, the 31 cells with a paired frozen-encoder run) and the held-out windows the retrospective variant scored (Table 25, all 32 screened cells). The body's counts are the selection-split ones.

What is held identical. V_b=1$-$MSE$_{\text{ZS}}$/MSE$_{\text{base}}$, and only the denominator depends on the estimator, so the numerator is read from the published gate for that split (`results/gate_val_side.json` or `gate_test_side.json`) and never recomputed: each column differs from the published one in exactly one respect. Before anything is reported, `scripts/gate_baseline_sensitivity.py` recomputes the `fitted` column through the alternative-baseline code path and checks it against both the published denominator *and* the published V_b; all 32 cells agree to a relative tolerance of 10^{-6}, and the script refuses to emit a table if any does not. The two checks are not redundant, and the ILI cell is why: its denominator matched while its V_b did not, because that cell aggregates as a mean over features of per-feature ratios where every other cell is a ratio of means (`gate_all_cells.ili_gate`, matching `finetune_ili`). A denominator check alone would have published one row of this table on a convention no other number in the paper uses.

The estimators. *Persistence* repeats the last observed value for all h steps and is the bottom of the ladder. *Seasonal naive* repeats the last seasonal period and fits nothing; periods are declared per dataset (24 steps for the hourly ETT, Weather and Electricity series, 96 for 15-minute ETTm2, 52 for weekly ILI) rather than inferred, because inferring one from an autocorrelation peak returns 1 on the flatter series and quietly substitutes a persistence baseline. *Per-feature AR* is the paper's map with the cross-channel terms removed, so it isolates how much of the gate rests on multivariate information. *DLinear* is a single linear map on the trend/seasonal decomposition of the lookback, shared across channels. *MLP* is one two-layer network per channel (hidden 128) with the penalty selected per channel on a chronological holdout; the channels that hit the 200-iteration budget are recorded per cell. *Tuned ridge* is the paper's own estimator with λ selected on a chronological 20% holdout of the training windows rather than fixed at 10^{-4}; the selected values span 10^{-6} to 10^{4}, and on all five ETTh2 cells the selection lands at 10^{3} or 10^{4}, so the fixed 10^{-4} the paper uses is badly under-regularised there. *Boosted trees* is one `HistGradientBoostingRegressor` per feature over that feature's lags plus the horizon step as an input feature; one model per (feature, step) would be 672 fits on a 7-feature h=96 cell. Its rows are windows×steps, capped at 200,000 per feature, which on the Moirai cells uses 1,041–2,083 of the 10,165–48,777 available windows. Every cap and every selected penalty is recorded per cell in `results/gate_baselines.json` rather than only in the source.

Four columns coincide on the single-channel arms, and that is arithmetic rather than a copy. On the Chronos and TimesFM cells, which score one target channel, the fitted map, per-feature AR and DLinear are the *same estimator*: with F=1 there are no cross-channel terms for AR to drop, and DLinear's trend/seasonal split is an invertible linear reparameterisation of the lookback, so a shared linear head on it spans exactly the maps a lookback regression spans. Their columns therefore agree to three decimals by construction, and tuned ridge differs from them only through λ. A reader who sees four identical entries on those rows is seeing the ladder collapse where the data has one dimension, not a transcription error.

A20.1 The ladder on the selection windows (the paper's primary)

The fitted map is admissible everywhere; the two rungs that are not are the cheap ones. On the selection windows the fitted map clears the training-mean floor on all 31 cells, per-feature AR, DLinear and tuned ridge on all 31, and the MLP on 30. Persistence falls below the floor on 12 cells, boosted trees on 15 and seasonal naive on 3; for the trees the reason is the same as on the held-out windows, that they cannot extrapolate past their training range. Counting only admissible passes, the eight rungs give **7, 8, 5, 6, 4, 7, 9 and 5 of 31** on this split.

Read as a graded axis, the union spans all three backbones. With V_{best} measured against the strongest admissible rival each cell faces (a minimum over rungs), **no cell of the 31 clears** 0.20 **against every admissible rung**, and **20 clear it against at least one: 14 Moirai, 3 Chronos, 3 TimesFM**. That union is what §6 uses as its analysis scope, and it matters for a specific objection: the seven cells that clear the fitted ridge are all Moirai on two series, so on the fitted rung alone the gate-positive evidence looks confined to one backbone. It is not: 13 of the 20 union cells *fail* the fitted ridge and are admitted by some other admissible rung, including every Chronos and TimesFM cell in the union. The seven that do clear the fitted ridge have V_{best} between -0.545 and $+0.134$, five of them set by seasonal naive and two by DLinear.

On this split the definition does have positive instances, under two weak rungs. With clause (i) taken from each estimator and clauses (ii) and (iii) unchanged, the count is zero under the fitted map, seasonal naive, per-feature AR, DLinear, tuned ridge and the MLP; **three under persistence** (Moirai-Small/ETTh1 h=96 at n=1k, Moirai-Base/ETTh1 h=96, Moirai-Large/ETTh1 h=96, with forg._{B}=+18.5, +22.8 and +14.1 unanimously and forg._{D}=−3.5, −14.7 and −14.1 unanimously); and **one under boosted trees** (Moirai-Small/Weather h=96, forg._{B}=+37.3 in 3/3, forg._{D}=−15.8 in 3/3). We report these rather than absorbing them. Three facts bound how much they carry. All four are ETTh1 or Weather cells, the same datasets as the eight instances this paper retracts. Both admitting denominators sit just above the floor: persistence scores 1.053 against the training mean's 1.104 on ETTh1, and the trees 0.5363 against 0.5397 on Weather, so each is within 5% of a constant and roughly twice the fitted ridge's error (0.576 and 0.266) on the same windows. And every one of the four has *no* demonstrated advantage against a rung that is not near-constant: the fitted ridge puts them at -0.18 to -0.55 and the strongest admissible rung each faces at -0.23 to -0.55. The direction

Table 24: **Gate V_b under eight denominators and a floor, on the *selection* windows.** Same conventions as Table 25; this is the split every admission decision in the body is made on, and the numerator comes from `results/gate_val_side.json`. Generated by `scripts/gate_baseline_sensitivity.py -split val`.

Cell	Fitted ridge (ours)	Training mean	Persist-ence	Seasonal naive	Per-feature AR	DLinear (shared)	Tuned ridge	MLP (128)	Boosted trees
Moirai-S / ETTh2 (h=192)	**+0.932**	**+0.943**	**+0.267**	+0.081	**+0.734**	**+0.331**	**+0.929**	**+0.889**	+0.971†
Moirai-B / ETTh2 (h=192)	**+0.925**	**+0.937**	+0.192	−0.013	**+0.706**	**+0.263**	**+0.922**	**+0.878**	+0.968†
Moirai-L / ETTh2 (h=96)	**+0.875**	**+0.953**	**+0.316**	+0.134	**+0.677**	**+0.243**	**+0.877**	**+0.897**	**+0.953**
Moirai-S / ETTh2 (h=96)	**+0.874**	**+0.953**	**+0.308**	+0.124	**+0.673**	**+0.234**	**+0.876**	**+0.896**	**+0.952**
Moirai-B / ETTh2 (h=96)	**+0.777**	**+0.917**	−0.220	−0.545	**+0.423**	−0.350	**+0.781**	**+0.817**	**+0.916**
Moirai-S / ETTm2 (h=192)	**+0.352**	**+0.716**	+0.092	+0.128	**+0.228**	+0.092	**+0.351**	**+0.497**	+0.723†
Moirai-B / ETTm2 (h=192)	**+0.211**	**+0.654**	−0.105	−0.062	+0.061	−0.106	**+0.210**	**+0.387**	+0.663†
TimesFM-2.5 / ETTm2 (h=24)	+0.082	**+0.217**	+0.041	+0.263†	+0.082	+0.082	−0.029	+0.412†	+0.598†
Chronos-T5-S / Electricity (h=24)	−0.089	+0.161	+0.100	+0.295†	−0.089	−0.089	−0.089	+0.102	−0.069
TimesFM-2.5 / ETTh1 (h=24)	−0.101	**+0.343**	+0.154	**+0.294**	−0.101	−0.101	−0.109	−0.035	−0.050
Chronos-T5-S / ETTh1 (h=24)	−0.116	**+0.358**	+0.157	**+0.322**	−0.116	−0.116	−0.124	−0.006	−0.039
TimesFM-2.5 / Weather (h=24)	−0.160	**+0.489**	**+0.437**	**+0.241**	−0.160	−0.160	−0.161	−0.071	+0.014
Moirai-S / ETTm2 (h=96)	−0.164	**+0.766**	+0.069	−0.252	−0.372	−0.394	−0.175	**+0.614**	+0.794†
Moirai-S / ETTh1 (h=96)	−0.177	**+0.385**	**+0.356**	+0.050	−0.225	−0.176	−0.198	−0.189	+0.501†
Chronos-T5-S / ETTh2 (h=24)	−0.184	**+0.461**	+0.488†	**+0.223**	−0.184	−0.184	−0.153	−0.091	−0.044
Chronos-T5-S / Weather (h=24)	−0.203	**+0.454**	**+0.430**	+0.165	−0.203	−0.203	−0.204	−0.145	−0.058
TimesFM-2.5 / ETTh2 (h=24)	−0.208	**+0.489**	+0.492†	**+0.213**	−0.208	−0.208	−0.182	−0.121	−0.051
Moirai-L / ETTh1 (h=96)	−0.251	**+0.347**	**+0.315**	−0.009	−0.302	−0.250	−0.273	−0.264	+0.470†
Moirai-B / ETTm2 (h=96)	−0.257	**+0.747**	−0.006	−0.352	−0.482	−0.505	−0.269	**+0.583**	+0.778†
Chronos-T5-S / ETTm2 (h=24)	−0.275	**+0.477**	+0.140	+0.527†	−0.275	−0.275	−0.293	−0.019	−0.295
Moirai-B / ETTh1 (h=96)	−0.312	**+0.315**	**+0.282**	−0.058	−0.365	−0.311	−0.335	−0.325	+0.444†
Moirai-S / ETTh1 (h=192)	−0.347	+0.100	+0.214†	−0.028	−0.443	−0.369	−0.347	−0.344	+0.633†
Moirai-S / Electricity7 (h=192)	−0.457	**+0.785**	+0.797†	−0.144	−0.809	−0.748	−0.485	−0.832	+0.868†
Moirai-S / Weather (h=192)	−0.506	**+0.242**	+0.339†	+0.121	−0.340	−0.233	−0.489	−0.284	+0.743†
Moirai-S / Weather (h=96)	−0.546	**+0.239**	+0.363†	+0.127	−0.436	−0.409	−0.537	−0.207	**+0.234**
Moirai-B / ETTh1 (h=192)	−0.600	−0.070	+0.066†	−0.222	−0.714	−0.627	−0.601	−0.597	+0.564†
Moirai-B / Weather (h=192)	−0.632	+0.178	+0.284†	+0.048	−0.453	−0.337	−0.614	−0.392	+0.721†
Moirai-L / Weather (h=96)	−0.639	+0.193	+0.325†	+0.074	−0.523	−0.494	−0.629	−0.279	+0.188
Moirai-S / Electricity7 (h=96)	−0.651	**+0.810**	+0.814†	−0.142	−0.789	−0.732	−0.669	−1.004	**+0.797**
Moirai-B / Weather (h=96)	−0.819	+0.104	+0.250†	−0.027	−0.690	−0.658	−0.808	−0.420	+0.099
Moirai-S / ILI (h=24)	−10.206	−0.518	+0.015†	−12.093	−10.206	−10.206	−11.072	−9.025	−6.171
Clearing 0.20	7/31	24/31	18/31	8/31	6/31	4/31	7/31	10/31	20/31
and admissible	7/31	24/31	8/31	5/31	6/31	4/31	7/31	9/31	5/31
Below the floor	0/31	0/31	12/31	3/31	0/31	0/31	0/31	1/31	15/31
Degradation cells	0/31	5/31	3/31	0/31	0/31	0/31	0/31	0/31	1/31

of the sensitivity is therefore unchanged from the retraction: what manufactures a degradation cell is a *weaker* baseline, not a stronger one revealing damage we missed. Taking that trend to its limit confirms it from the other side: under the training-mean floor itself the count is **five**, the four above plus Moirai-Small/Weather h=192, and all five are cells the retraction above withdraws.

A20.2 The same ladder on the held-out windows (the retrospective variant)

A denominator has to clear a floor before the ratio means anything. If $\mathrm{MSE}_{\mathrm{base}}$ exceeds the training mean's, the gate is dividing by the error of an estimator worse than answering "the average", and a checkpoint can clear the threshold without beating anything, which is precisely what the trend extrapolation did. The fitted map is above the floor on **all 32 cells**, so the correction above did not trade one inadmissible denominator for another. Boosted trees fall below it on 11 cells and seasonal naive on 6. For the trees the mechanism is that they cannot extrapolate beyond their training range, and ETTh2's test split is well outside it: the training-mean predictor itself scores 2.06 there at h=192 against a normalised variance of 1, and the trees score 4.82. Persistence falls below the floor on 17 and the MLP on 2. Counting only admissible passes, the eight rungs give **5, 2, 4, 2, 0, 4, 6 and 7 of 32** in ladder order; the widest is boosted trees, which adds the four ETTm2 cells, and the narrowest is DLinear, which admits **none**.

Table 25: **Gate V_b under eight denominators and a floor, on one numerator, over the *held-out* windows.** The retrospective variant: this is the split an earlier version scored the gate on, retained because the prospective predictor of §7.2 was frozen against it. Bold clears 0.20. A dagger marks a denominator worse than the training-mean predictor on that cell: the ratio is still arithmetic, but it is not evidence of a pre-trained advantage, so daggered cells are excluded from the "and admissible" tally. The training-mean column is the floor, not a candidate estimator. The last row applies the three-clause degradation definition of §3 with clause (i) taken from each estimator and clauses (ii) and (iii) unchanged, over the 31 cells with a paired frozen-encoder run. Generated by `scripts/gate_baseline_sensitivity.py`.

Cell	Fitted ridge (ours)	Training mean	Persist-ence	Seasonal naive	Per-feature AR	DLinear (shared)	Tuned ridge	MLP (128)	Boosted trees
Moirai-S / ETTh2 (h=192)	**+0.464**	**+0.845**	−0.063	−0.267	−0.283	−0.738	**+0.422**	**+0.296**	+0.934†
Moirai-S / ETTh2 (h=96)	**+0.400**	**+0.813**	−0.223	−0.215	−0.843	−0.889	**+0.374**	**+0.406**	**+0.806**
Moirai-B / ETTh2 (h=96)	**+0.361**	**+0.801**	−0.303	−0.295	−0.965	−1.014	**+0.333**	**+0.367**	**+0.793**
Moirai-B / ETTh2 (h=192)	**+0.275**	**+0.791**	−0.438	−0.714	−0.735	−1.350	**+0.218**	+0.048	+0.911†
Moirai-L / ETTh2 (h=96)	**+0.219**	**+0.757**	−0.592	−0.582	−1.400	−1.460	+0.185	**+0.227**	**+0.747**
TimesFM-2.5 / Electricity (h=24)	+0.132	+0.179	+0.254†	+0.301†	+0.132	+0.132	+0.132	+0.186†	+0.146
Moirai-S / ETTm2 (h=192)	+0.108	**+0.830**	+0.101	+0.058	**+0.350**	+0.123	+0.119	**+0.545**	**+0.826**
Moirai-B / ETTm2 (h=192)	+0.083	**+0.825**	+0.076	+0.032	**+0.332**	+0.099	+0.095	**+0.532**	**+0.821**
Chronos-T5-S / Electricity (h=24)	+0.048	+0.100	+0.182†	+0.233†	+0.048	+0.048	+0.048	+0.107†	+0.064
Moirai-S / Electricity7 (h=192)	−0.047	**+0.662**	**+0.625**	−0.035	−0.516	−0.440	−0.071	−0.367	+0.893†
Moirai-B / ETTm2 (h=96)	−0.057	**+0.860**	+0.165	+0.047	+0.014	−0.193	−0.067	+0.193	**+0.855**
TimesFM-2.5 / Weather (h=24)	−0.077	**+0.494**	+0.535†	**+0.209**	−0.077	−0.077	−0.077	−0.002	+0.018
Moirai-S / Electricity7 (h=96)	−0.085	**+0.701**	**+0.649**	+0.093	−0.404	−0.400	−0.102	−0.337	+0.856†
TimesFM-2.5 / ETTh1 (h=24)	−0.101	**+0.424**	+0.124	**+0.291**	−0.101	−0.101	−0.091	+0.005	−0.005
Chronos-T5-S / Weather (h=24)	−0.103	**+0.481**	+0.523†	+0.190	−0.103	−0.103	−0.103	−0.027	−0.006
Moirai-S / ETTm2 (h=96)	−0.110	**+0.853**	+0.124	−0.000	−0.035	−0.252	−0.121	+0.153	**+0.848**
Chronos-T5-S / ETTh1 (h=24)	−0.123	**+0.412**	+0.107	**+0.278**	−0.123	−0.123	−0.112	−0.014	−0.024
Moirai-S / ETTh1 (h=96)	−0.243	**+0.472**	+0.570†	**+0.216**	−0.362	−0.242	−0.297	−0.147	+0.621†
TimesFM-2.5 / ETTh2 (h=24)	−0.287	**+0.475**	+0.527†	+0.021	−0.287	−0.287	−0.238	−0.108	−0.131
Chronos-T5-S / ETTh2 (h=24)	−0.325	**+0.460**	+0.513†	−0.008	−0.325	−0.325	−0.274	−0.141	−0.164
Moirai-S / ETTh1 (h=192)	−0.331	**+0.326**	+0.406†	+0.083	−0.410	−0.351	−0.382	−0.227	+0.758†
Moirai-L / ETTh1 (h=96)	−0.454	**+0.382**	+0.497†	+0.083	−0.594	−0.453	−0.517	−0.342	+0.556†
Moirai-B / ETTh1 (h=96)	−0.458	**+0.380**	+0.496†	+0.080	−0.599	−0.458	−0.522	−0.346	+0.555†
Moirai-B / ETTh1 (h=192)	−0.501	**+0.239**	+0.330†	−0.034	−0.590	−0.524	−0.559	−0.383	+0.727†
Moirai-L / Weather (h=96)	−0.621	**+0.204**	+0.248†	+0.143	−0.531	−0.437	−0.611	−0.611	+0.055
Moirai-S / Weather (h=192)	−0.639	−0.037	+0.276†	+0.163†	−0.775	−0.670	−0.639	−0.648	+0.600†
Moirai-S / Weather (h=96)	−0.867	+0.084	+0.134†	+0.014	−0.763	−0.655	−0.855	−0.855	−0.088
TimesFM-2.5 / ETTm2 (h=24)	−0.945	**+0.576**	+0.136	+0.631†	−0.945	−0.945	−0.901	−0.861	−0.531
Moirai-B / Weather (h=96)	−1.029	+0.004	+0.059†	−0.072	−0.916	−0.799	−1.016	−1.016	−0.183
Chronos-T5-S / ETTm2 (h=24)	−1.281	**+0.503**	−0.013	+0.567†	−1.281	−1.281	−1.229	−1.182	−0.795
Moirai-B / Weather (h=192)	−1.324	−0.470	−0.027†	−0.187†	−1.517	−1.368	−1.323	−1.337	+0.433†
Moirai-S / ILI (h=24)	−1.625	−0.304	+0.027†	−0.786	−1.625	−1.625	−1.638	−1.301	−1.563
Clearing 0.20	5/32	25/32	14/32	8/32	2/32	0/32	4/32	6/32	18/32
and admissible	5/32	25/32	2/32	4/32	2/32	0/32	4/32	6/32	7/32
Below the floor	0/32	0/32	17/32	6/32	0/32	0/32	0/32	2/32	11/32
Degradation cells	0/31	6/31	0/31	1/31	0/31	0/31	0/31	0/31	0/31

The negative headline strengthens. The identity of the five survivors does not. No estimator admits many more cells than the fitted map and two admit far fewer, so "on most cells there is no demonstrated advantage" is not an artefact of one estimator family. But the passing *sets* are nearly disjoint. All five ETTh2 survivors fail against seasonal naive (-0.215 to -0.714), against per-feature AR (-0.283 to -1.400) and against DLinear (-0.738 to -1.460), which is the rung that beats the pre-trained model everywhere. Tuned ridge and the MLP each reproduce four of the five and not the same four: tuned ridge loses Moirai-Large/ETTh2 h=96 at $+0.185$, the MLP loses Moirai-Base/ETTh2 h=192 at $+0.048$. The denominators say why: on ETTh2 h=192 the univariate AR map scores 0.249 where the paper's multivariate map scores 0.595, so the cross-channel terms *hurt* there, and "Moirai beats a target-data regression on ETTh2" is in part a statement about a regression with 672 free inputs. The four cells seasonal naive admits are a different set again: Chronos/ETTh1, TimesFM/ETTh1, TimesFM/Weather and Moirai-Small/ETTh1 h=96. The instability of the passing

set under a change of rung is the same on both splits; it is why the body scopes its drift analyses by the union rather than by any one rung.

Does the zero survive? Under seven of the eight rungs, yes. With clause (i) taken from each estimator, the definition returns **zero cells under the fitted map, persistence, per-feature AR, DLinear, tuned ridge, the MLP and boosted trees, and one under seasonal naive**: Moirai-Small/ETTh1 $h{=}96$, which reads $+0.216$ there against -0.243 against the fitted map and then satisfies both remaining clauses unanimously, forg.$_{\mathrm{B}}{=}+18.5\%$ in 5/5 seeds, forg.$_{\mathrm{D}}{=}-3.5\%$ in 5/5, B$-$D$=+22.1$ at CKA 0.807. We report it rather than absorbing it: against one of the eight rungs this paper's own definition has a positive instance.

Two things about that instance decide how much weight it carries. It clears 0.216 against a threshold of 0.20 and fails at 0.25. And it arises from a *weaker* admissible denominator, not a stronger one: on ETTh1 $h{=}96$ seasonal naive scores 0.657 where the fitted map scores 0.415. The direction of the sensitivity is therefore that loosening the baseline manufactures degradation cells, which is the same mechanism as the retracted eight rather than a stronger baseline revealing damage we missed. The floor column is that trend taken to its limit: under the training-mean denominator the count is **six**, and the six are Moirai-Small/ETTh1 at both horizons, Moirai-Base/ETTh1 at both, Moirai-Large/ETTh1 $h{=}96$ and Moirai-Large/Weather $h{=}96$, the cells the appendix above retracts. That is the retraction confirmed from the other side: the degradation instances the earlier version reported needed a denominator no stronger than a constant.

The ladder read as a graded axis, and why the headline is a minimum. A pass count per rung still treats the gate as binary. The graded reading assigns each cell one number, V_{best}, measured against the *strongest* admissible rival it faces. That is a **minimum** over rungs, not a maximum: a weaker denominator inflates V_b, so the strongest rival is the one that leaves the smallest score. Read that way, **no cell of the 32 clears** 0.20 **against every admissible rung**, and 15 clear it against at least one. The two counts move in opposite directions as the ladder grows (the first can only fall, the second can only rise), so only the first is a claim, and adding a ninth rung could not undo it. The five ETTh2 survivors have V_{best} between -0.738 and -1.460, every one of them against DLinear. Both readings agree with the selection-split ones above (no cell clears every admissible rung on either split), so the graded conclusion does not depend on which split supplies the axis, and the value axis §6 plots is the selection-split one (`results/gate_baselines_val.json`) because its x and y must not share windows. Each file stores V_{best} per cell alongside which rung supplied it.

A20.3 A THIRD WINDOW SET: THE LADDER ON THE TRAIN TAIL

The selection-split gate meets the protocol a reviewer would ask for (the gate's windows are disjoint from the outcome's), but not its residual: the selection windows also drove early stopping and checkpoint choice, so the outcome depends on them indirectly. So we scored the same nine denominators a third time, on the last 20% of each cell's *training* region, held back by index rather than inferred from which windows training happened to skip. Every rung's denominator is fitted on the earlier part of train only and the numerator is a zero-shot pass, so neither arm of the gate has seen these windows. The precise claim is that they are used for neither checkpoint selection nor outcome scoring; it is *not* that they were never seen, because the condition-B and -D runs subsampled their training windows from the whole train region including the tail. That does not touch the gate (whose numerator is the untrained checkpoint), but it is the difference between this arm and a genuinely fresh dataset. The arm covers the 21 Moirai cells (`scripts/gate_thirdsplit.py`, `results/gate_baselines_traintail.json`); Table 26 counts all three window sets over exactly those 21, so its columns differ by windows and not by coverage.

The bound is confirmed on all three window sets; the survivor set is not. No cell clears 0.20 against every admissible rung on any of them: **0, 0 and 0 of 21**. The fitted rung, which is where the paper's seven survivors come from, does not transfer: it admits **7 cells on the selection windows and 3 on the train tail, with only 2 in both** (Moirai-Base/ETTh2 $h{=}192$ and Moirai-Small/ETTh2 $h{=}192$), and the train tail admits one cell the selection windows reject, Moirai-Small/ETTh1 $h{=}192$. The union moves the other way, 14 to 19, because the training-mean floor is so much worse on the tail that more rungs clear admissibility there at all.

What moves is the denominator, not the checkpoint. On Moirai-Base/ETTh2 $h{=}96$ the fitted ridge scores 1.012 on the selection windows and 0.276 on the train tail ($3.7\times$ better), while zero-shot barely

Table 26: **The eight-rung ladder on three window sets.** Cells each rung admits: its denominator is at least as good as the training-mean floor on that cell *and* the cell's V_b clears 0.20. Every column is out of the same 21 Moirai cells (the ones the train-tail arm covers), so the counts are comparable across columns and are not the 31- and 32-cell counts quoted elsewhere. *Selection* is the paper's primary, *held-out* the retrospective variant, and *train tail* the last 20% of the training region, used for neither checkpoint selection nor outcome scoring. The bottom row is the paper's conservative bound, and it is zero on all three.

Rung (clause (i)'s denominator)	Selection	Held-out	Train tail
Fitted ridge (ours)	7	5	3
Training mean (floor)	16	17	20
Persistence	6	2	12
Seasonal naive	0	1	0
Per-feature AR	6	2	0
DLinear (shared)	4	0	0
Tuned ridge	7	4	2
MLP (128)	9	6	3
Boosted trees	5	7	9
Clears 0.20 against *at least one* admissible rung	14	12	19
Clears 0.20 against *every* admissible rung	**0**	**0**	**0**

moves, 0.225 to 0.306. The ridge is fitted on the training region, so its error grows once it is evaluated away from it, and that growth is most of what the ETTh2 survivors' margin measures. Beside its own fit region the ridge is much harder to beat and three cells retain a pre-trained advantage. This is the correction of Appendix A29.2 in a second guise: there the gate moved because the denominator was the wrong estimator, here because it was evaluated on different windows, and in both cases the quantity that moved was the baseline rather than the pre-trained model. A gate-pass count is a statement about a baseline *and* a window set, and this paper's counts should be read as such. What survives all three is the negative claim, which is the one the paper's conclusions rest on.

A20.4 Matching the lookback: is the verdict an artifact of the context asymmetry?

Moirai is evaluated with $96{+}h$ steps of history (Figure 2) while the baseline is given the last 96, and §4 argues that this favours the pre-trained model. An argument is not a measurement, so here is the measurement: the same ridge map, the same windows, the same fitting region, the same normalisation and the same λ, given all $96{+}h$ available steps instead of the last 96. Only the integer passed to the fit changes, so the two columns of Tables 27 and 28 differ in the baseline's information set and in nothing else; the numerator is the same stored MSE_{ZS}, and the `fitted` column reproduces the published denominator on all 21 Moirai cells of each split to a relative tolerance of 10^{-6} before anything is reported.

The effect is small and its sign depends on the split, because a matched map is not automatically a stronger one. On the selection windows (the split every admission decision is made on), the matched map is the stronger of the two on **12 of 21** cells, and the count clearing 0.20 falls from **7 to 6**: **Moirai-Base/ETTm2** $h{=}192$ is the one cell that flips, from $+0.211$ to $+0.125$. On the held-out windows the direction reverses. There the matched map is stronger on only **7 of 21**, the count rises from **5 to 7**, and the two cells that flip do so *into* the survivor set (Moirai-Small and Moirai-Base on ETTm2 at $h{=}192$). The reason is visible in the two MSE columns: the matched map estimates $(96{+}h){\times}F$ inputs from the same training windows, which fits the training region better and generalises to the held-out region worse. So giving the baseline the model's full history moves the count by one or two cells in whichever direction the extra parameters happen to pay off, and no reading in this paper turns on either move.

Why this is reported as its own axis rather than as a ninth rung, and what the separation does not hide. The ladder above varies the estimator *family* at a fixed lookback, and its rungs feed the graded V_{best} that §6 uses as a covariate; a rung that varies the *information set* instead would move

that minimum for reasons belonging to a different question. The separation protects no count, and we checked rather than asserted that: folding the matched column into the graded minimum leaves the number of cells clearing 0.20 against every admissible rung at **0 of 21 on both splits**, changes **no** cell's graded verdict, and makes the matched map the strongest admissible rival on **0** cells on the selection split and **3** on the held-out split.

Table 27: **The gate with the baseline given Moirai's full $96{+}h$ history, on the *selection* windows.** The same fitted ridge in both columns; only how many of the available past steps it may use differs. MSE ratio is $\text{MSE}_{96+h}/\text{MSE}_{96}$, so below 1 means the extra history made the baseline stronger. Bold clears 0.20. Generated by `scripts/matched_lookback_gate.py -split val`.

Cell	V_{ridge} (96)	V_{ridge} ($96{+}h$)	Δ	MSE ratio	Verdict
Moirai-S / ETTh2 (h=192)	$\mathbf{+0.932}$	$\mathbf{+0.903}$	-0.029	0.699	pass (both)
Moirai-B / ETTh2 (h=192)	$\mathbf{+0.925}$	$\mathbf{+0.893}$	-0.032	0.699	pass (both)
Moirai-L / ETTh2 (h=96)	$\mathbf{+0.875}$	$\mathbf{+0.858}$	-0.017	0.879	pass (both)
Moirai-S / ETTh2 (h=96)	$\mathbf{+0.874}$	$\mathbf{+0.856}$	-0.017	0.879	pass (both)
Moirai-B / ETTh2 (h=96)	$\mathbf{+0.777}$	$\mathbf{+0.747}$	-0.031	0.879	pass (both)
Moirai-S / ETTm2 (h=192)	$\mathbf{+0.352}$	$\mathbf{+0.281}$	-0.071	0.901	pass (both)
Moirai-B / ETTm2 (h=192)	$\mathbf{+0.211}$	$+0.125$	-0.086	0.901	pass → **fail**
Moirai-S / ETTm2 (h=96)	-0.164	-0.233	-0.070	0.944	fail (both)
Moirai-S / ETTh1 (h=96)	-0.177	-0.145	$+0.032$	1.028	fail (both)
Moirai-L / ETTh1 (h=96)	-0.251	-0.217	$+0.034$	1.028	fail (both)
Moirai-B / ETTm2 (h=96)	-0.257	-0.332	-0.075	0.944	fail (both)
Moirai-B / ETTh1 (h=96)	-0.312	-0.276	$+0.036$	1.028	fail (both)
Moirai-S / ETTh1 (h=192)	-0.347	-0.368	-0.021	0.984	fail (both)
Moirai-S / Electricity7 (h=192)	-0.457	-0.383	$+0.074$	1.054	fail (both)
Moirai-S / Weather (h=192)	-0.506	-0.453	$+0.053$	1.037	fail (both)
Moirai-S / Weather (h=96)	-0.546	-0.503	$+0.043$	1.029	fail (both)
Moirai-B / ETTh1 (h=192)	-0.600	-0.625	-0.025	0.984	fail (both)
Moirai-B / Weather (h=192)	-0.632	-0.574	$+0.057$	1.037	fail (both)
Moirai-L / Weather (h=96)	-0.639	-0.593	$+0.046$	1.029	fail (both)
Moirai-S / Electricity7 (h=96)	-0.651	-0.904	-0.253	0.867	fail (both)
Moirai-B / Weather (h=96)	-0.819	-0.768	$+0.051$	1.029	fail (both)
Clearing 0.20	7/21	6/21		12/21 stronger	1 lost, 0 gained

A20.5 The numerator is not deterministic either: the evaluator's sampling noise

Every denominator above is a fitted estimator, and the eight rungs ask whether the screen is a property of that choice. The *numerator* deserves the same question. Moirai's forecast is the median of 20 sampled paths, so a cell's zero-shot MSE is a measurement, not a constant: two runs of one cell on the same deterministic windows record two different values (Appendix A4). Each gate value in this paper is therefore one draw, and the width of that draw sets the resolution at which a 0.20 operating point can be read at all.

We can measure it, because the records contain repeats. Across the 26 Moirai cell-scorings (the 21-cell retrospective grid and batch 3's five) there are 227 independent zero-shot measurements, every one of them the same frozen checkpoint on the same windows. Recomputing R^2_{task} from each gives a per-cell spread of at most 0.0244, on Moirai-Base/ETTh2 h=96; substituting the all-replicate mean for the registered denominator moves R^2_{task} by at most 0.0037 on batch 1, 0.0044 on batch 2 and 0.0011 on batch 3. (Batch 2 was abandoned before any outcome run: it contributed six zero-shot measurements, which enter batch 3's registered denominator, and no condition-B or condition-D record anywhere in this paper.)

The comparison that matters is per cell, not global. That largest spread, 0.0244, is bigger than the tightest margin to the operating point anywhere (0.0109, Moirai-Base/ETTm2 h=192 at R^2_{task}=+0.2109), and pairing the two would say the screen sits inside its own noise. It does not: they are different cells, and the cell carrying the 0.0244 sits 0.577 from 0.20, where noise of any size cannot change a call. Sampling noise on a cell can only threaten that cell's own call, so margin and

Table 28: **The same comparison on the *held-out* windows.** The direction reverses: the matched map is stronger on 7 of 21 cells here against 12 of 21 on the selection windows, because the larger design matrix generalises away from its fit region worse. Generated by `scripts/matched_lookback_gate.py -split test`.

Cell	V_{ridge} (96)	V_{ridge} (96+h)	Δ	MSE ratio	Verdict
Moirai-S / ETTh2 (h=192)	$\mathbf{+0.464}$	$\mathbf{+0.624}$	$+0.161$	1.428	pass (both)
Moirai-S / ETTh2 (h=96)	$\mathbf{+0.400}$	$\mathbf{+0.389}$	-0.011	0.982	pass (both)
Moirai-B / ETTh2 (h=96)	$\mathbf{+0.361}$	$\mathbf{+0.349}$	-0.012	0.982	pass (both)
Moirai-B / ETTh2 (h=192)	$\mathbf{+0.275}$	$\mathbf{+0.492}$	$+0.217$	1.428	pass (both)
Moirai-L / ETTh2 (h=96)	$\mathbf{+0.219}$	$\mathbf{+0.205}$	-0.014	0.982	pass (both)
Moirai-S / ETTm2 (h=192)	$+0.108$	$\mathbf{+0.242}$	$+0.134$	1.177	fail → **pass**
Moirai-B / ETTm2 (h=192)	$+0.083$	$\mathbf{+0.221}$	$+0.138$	1.177	fail → **pass**
Moirai-S / Electricity7 (h=192)	-0.047	-0.005	$+0.042$	1.042	fail (both)
Moirai-B / ETTm2 (h=96)	-0.057	$+0.081$	$+0.138$	1.150	fail (both)
Moirai-S / Electricity7 (h=96)	-0.085	-0.150	-0.065	0.944	fail (both)
Moirai-S / ETTm2 (h=96)	-0.110	$+0.035$	$+0.145$	1.150	fail (both)
Moirai-S / ETTh1 (h=96)	-0.243	-0.066	$+0.176$	1.165	fail (both)
Moirai-S / ETTh1 (h=192)	-0.331	-0.179	$+0.152$	1.129	fail (both)
Moirai-L / ETTh1 (h=96)	-0.454	-0.248	$+0.206$	1.165	fail (both)
Moirai-B / ETTh1 (h=96)	-0.458	-0.251	$+0.207$	1.165	fail (both)
Moirai-B / ETTh1 (h=192)	-0.501	-0.330	$+0.171$	1.129	fail (both)
Moirai-L / Weather (h=96)	-0.621	-0.622	-0.001	0.999	fail (both)
Moirai-S / Weather (h=192)	-0.639	-0.522	$+0.117$	1.077	fail (both)
Moirai-S / Weather (h=96)	-0.867	-0.868	-0.001	0.999	fail (both)
Moirai-B / Weather (h=96)	-1.029	-1.030	-0.002	0.999	fail (both)
Moirai-B / Weather (h=192)	-1.324	-1.157	$+0.166$	1.077	fail (both)
Clearing 0.20	5/21	7/21		7/21 stronger	0 lost, 2 gained

noise must be taken together per cell. The minimum of that ratio over all 26 cell-scorings is $9.2\times$, at the tightest cell; and directly, **no cell's** 0.20 **call flips under any single replicate substitution**, which the emitter asserts rather than reports, so a future record that broke it would fail the re-derivation instead of shipping.

Two consequences we accept. First, the screen's resolution is roughly 10^{-2} in R^2_{task}, so Moirai-Base/ETTm2 h=192 is a cell whose admission we would not defend on a remeasurement at a 0.21 threshold, which is why Table 10 sweeps the threshold on the pre-registered cells rather than defending one number. Second, this noise is why the pre-registered roots are read `condition_A` only: averaging the outcome runs' re-measurements into a registered denominator moves the predictor after its outcomes exist, and at $\sim 10^{-3}$ it does so far too quietly to be noticed downstream. An earlier revision of our own code did exactly that, and item (g) of Appendix A29.1 records the repair; had it shipped, 0 **of the** 26 admissions would have differed, which is precisely why nothing downstream caught it.

A21 Is the Zero a Property of the Runs or of the Definition?

The paper's headline is that no cell meets its definition of damaging a demonstrated pre-trained capability, and the first fair objection is that the definition does the work. It requires condition-B forgetting positive in *every* seed and condition-D forgetting negative in *every* seed; unanimity over ten seeds is a demanding bar, and if a simple majority produced five degradation cells the negative result would be a property of the rule rather than of the runs. That objection is answerable from the stored per-seed sign counts with no new runs.

Table 29 walks a ladder of definitions from the weakest defensible one (the two means alone, with seed signs unchecked) up to ours, crossed with both gate thresholds so the sensitivity to *both* free parameters is one table rather than two. Two exclusions apply identically at every rung, exactly as `degradation_cells()` applies them: cells with no zero-shot test reference cannot be scored at all (there are none), and the five Chronos cells are excluded because a head/decoder mismatch makes

their per-condition forgetting non-comparable (limitation iv). Admitting them at the relaxed rungs would make loosening the rule appear to find degradation cells when what it found is a measurement already known to be invalid. Both exclusions are properties of the measurement rather than of the outcome, so applying them uniformly is not a selection on the result. That leaves 26 scorable cells, of which 7 clear a gate of 0.10 and 7 clear 0.20: under the fitted baseline the two thresholds select the same set, so the gate column is a sensitivity check that happens to move nothing rather than a second free parameter.

Table 29: Degradation cells admitted under eight definitions and two gate thresholds. Cells are named, not just counted. Under the corrected gate no cell is admitted at 0.10 alone, so the two columns agree throughout. The excluded confounded cells are Chronos-T5-Small on ETTh1, ETTh2, ETTm2, Weather and Electricity. Emitted by `scripts/degradation_sensitivity.py`.

	Cells admitted		
Definition of a degradation cell	Gate ≥ 0.10	Gate ≥ 0.20	Which cells
Means only (seed signs unchecked)	1	1	S/ETTh2 $h192$
Means, + $\geq$ half the seeds agree	1	1	S/ETTh2 $h192$
Means, + $\geq 70\%$ of seeds agree	0	0	—
Means, + $\geq 80\%$ of seeds agree	0	0	—
Means, + every seed agrees (ours)	0	0	—
Means, + one-sided t on forg$_\text{B}$ at α=0.05	0	0	—
Seed counts only, $\geq$ half agree	2	2	S/ETTh2 $h192$, S/ETTm2 $h192$
Seed counts only, every seed agrees	0	0	—
Gate-passing cells screened	7	7	

Unanimity is not what produces the zero. At the paper's gate threshold the largest count anywhere on the ladder is **two**, and it is produced by a single rung (seed counts only, half the seeds agreeing), which is the one rung that drops the means clause entirely. Of the six rungs that keep it, the *weakest*, the two means with seed signs unchecked, admits one, and requiring 70% of seeds, 80%, unanimity, or a one-sided t-test on forgetting$_\text{B}$ all give zero. Relaxing the gate to 0.10 changes none of these counts, so 5 of the 8 rungs yield zero at both. There is no setting of these two parameters at which the paper's conclusion becomes "several cells degrade".

The one cell that ever appears at a rung that keeps the means clause. Moirai-Small on ETTh2 at h=192 has forgetting$_\text{B}$ $+1.99$ with an SEM of 3.32 over ten seeds, a one-sided p of 0.28 against zero, and 5 of 10 seeds on each side. It is a coin flip, which is what the body already reports it as. Of the other six gate-passing cells, three are strongly *improved* by full fine-tuning rather than marginally harmed (forgetting$_\text{B}$ -31.44 ± 3.65, -36.35 ± 2.80 and -42.42 ± 0.47 SEM, with every seed agreeing), and two more improve on the mean, both ETTm2 at h=192 (-1.10 ± 3.37 and -8.45 ± 3.23). The seventh, Moirai-Small/ETTh2 at h=96, does have a positive mean forgetting$_\text{B}$ of $+9.71\pm10.58$, but fails the means clause on the *other* side, for the reason the next paragraph gives.

Means and seed counts can disagree in direction, and we report both families rather than absorb the disagreement. Moirai-Small/ETTh2 at h=96 has a mean forgetting$_\text{D}$ of $+6.91$ while 9 of its 10 seeds are negative: a single outlier drags the mean across zero. A ladder that silently required the means clause at every rung would hide that, and "majority seed-consistency" is what the objection proposes as a *replacement* for unanimity, not as an extra hurdle on top of it. The last two rows therefore drop the means clause entirely, and the first of them is the only row on the whole ladder that admits two cells, which is how Moirai-Small/ETTm2 at h=192 becomes visible at all.

Which definition the paper uses, and why. We keep unanimity. The reason is not that it yields the fewest cells (at gate 0.20 5 of the 8 rungs also yield zero), but that a degradation claim is a claim that fine-tuning *reliably* destroys something, and a cell whose seeds disagree on the sign of the harm is not evidence of that. The cost of the choice is that it can discard a real but noisy effect, which is exactly what the h=192 cell would be if its true forgetting$_\text{B}$ were small and positive; that cell is therefore reported by name in the body rather than dropped silently.

A22 PAIRED INFERENCE ON ENCODER ADAPTATION, AND WHAT THESE SEED COUNTS CAN RESOLVE

A unanimity count is not inference. This appendix reports Δ_{enc} per cell as a paired estimate with an interval, an effect size, two p-values and a multiplicity-adjusted q; states the three-way call the body is built on; and then says plainly what the matrix's seed counts can and cannot resolve.

The call, and why it has three outcomes. Thirty-one cells are tested, so we adjust: Benjamini–Hochberg across the cells within this contrast gives the q column, and a cell is called decisive only at $q<0.05$. That yields **2 cells where freezing is decisively better, 4 where adapting is decisively better, and 25 inconclusive**. The third category is the substantive addition. Our published criterion recorded a cell with no measurable effect and a cell with a large effect its seeds disagreed about identically, as "not degrading"; under the interval rule the second says *inconclusive*, which is what it is, and the minimum detectable effect below is what separates the two.

Where the decisive cells sit relative to the screen. Both cells where freezing is decisively better, Moirai-Small/ETTh1 at h=96 (Δ_{enc}=+22.1, q=0.011) and h=192 (+13.5, q=0.014), **fail** the screen, at V_{ridge} of -0.177 and -0.347. Two of the four cells where adapting is decisively better, Moirai-Base/ETTh2 h=192 (-20.2, q=0.018) and Moirai-Large/ETTh2 h=96 (-20.8, q=0.022), **pass** it. So on this matrix the screen admits cells where adaptation decisively wins and rejects the only cells where preservation decisively wins. That is a sharper statement against the screen than the unanimity tally could make, and it is stated here because it counts against our own instrument. It also has exactly one exception at the augmented seed counts of Table 32, reported there rather than here because every number in this paragraph is at the published n.

Both degradation criteria are reported, and both return zero. Clause (i) of the definition in §3 is unchanged. Replacing the unanimity clauses (ii) and (iii) with interval clauses (the CI on Δ_{FT}=B−ZS entirely above zero, the CI on Δ_{frz}=D−ZS entirely below) selects **zero cells**, exactly as the pre-registered unanimity version does. The two rules can disagree in principle, in either direction, so the agreement is a result and not a tautology; `scripts/paired_inference.py` computes both and cross-checks the unanimity list against `cell_matrix.degradation_cells`, which is the definition of record, rather than trusting two spellings of one rule to stay in step.

The zero-shot reference's own error is propagated, and only where it belongs. On the 21 Moirai cells the zero-shot denominator is a per-cell mean over condition-A seeds rather than each seed's own measurement, so the B−ZS and D−ZS intervals above are widened by that mean's SEM, se=$\sqrt{\text{se}^2_{\text{within}} + \text{se}^2_{\text{ZS}}}$. Δ_{enc} is *not* widened and must not be: the reference cancels from its numerator and survives only as a common scale factor. That asymmetry is one of the reasons Δ_{enc} is the estimand. The widening is material on 13 cells and is nominally zero on 8, where the reference ran a single condition-A seed; on those the reference's error is *unestimated*, not zero, and we say so rather than let a 0.00 read as a check that passed.

What these seed counts could have detected. Table 31 turns "the interval is wide" into a number for the seven cells the screen admits, which are the only cells about which we make a preservation recommendation. At the multiplicity-adjusted level the call actually uses, **4 of the 7 are unresolvable within our feasibility cap of 20 paired seeds**: they would need 29, 36, 182 and 212. Two of the remaining three are already decisive. So on the cells where a practitioner would consult this screen, the honest summary is that our design settles two of seven and cannot settle four of seven at any budget we could run, which is a limitation of this matrix, not evidence that the effects are absent, and it is the reason §3 reports an interval rather than a sign count.

The exact test's resolution is set by N alone. A sign-flip permutation test on N paired differences has 2^N assignments, so its smallest attainable two-sided p is $2/2^N$: 0.25 at N=3, 0.0625 at N=5, 0.002 at N=10. The paired-N histogram across the 31 cells is {3:24, 5:4, 10:3}, so **only 3 of 31 cells can reach $p<0.05$ without a distributional assumption**, and only one does (ILI, p=0.002). Thirteen cells have a t-interval excluding zero where the exact test cannot reject at any effect size. We report both columns rather than the more favourable one: the t-test's rejections are real inferences under its normality assumption, and the exact column is the price of not making that assumption.

Why the reported interval is the t-interval, not the bootstrap. At N=3 every percentile-bootstrap resample mean lies inside $[\min, \max]$ of the three differences, so a same-sign triple excludes zero

by construction. Such an interval reports the sign of the data, not the uncertainty of the mean. The bootstrap is computed and stored in `results/paired_inference.json`; the table prints the paired t-interval beside the exact p.

The aggregate straddles zero. Averaging cell means and resampling (backbone, dataset) clusters, the 20 value-cells give $\Delta_{\text{enc}}{=}+0.7$ with CI $[-11.1, +14.1]$ and 10 of 20 positive; all 31 give $+2.9$ with CI $[-12.7, +15.1]$ and 16 of 31 positive. Neither excludes zero in either direction. This is the honest aggregate statement of the paper's central contrast: across these benchmarks we cannot establish that allowing encoder adaptation systematically helps *or* hurts, which is a different and weaker claim than either regime law the literature offers.

A pre-registered seed top-up, and the one call it changed. Five of the seven admitted cells are inconclusive, and Table 31 says four of those five cannot be resolved at any budget we can run. What the resolvable ones do with more seeds is therefore the only power question still open on this matrix, and an answer to it is worth something only if the budget was fixed before the seeds ran. `results/power_topup/preregistration_power.json` was committed before any added run existed and fixes the rule (top up a cell iff its call is inconclusive and at least one of its two required n is within the feasibility cap of 20), the three cells that rule selects, their seeds, and the exact n at which to stop. All **26** registered runs completed, and `scripts/emit_power_topup.py` exits non-zero if any one of them is missing, so Table 32 is evidence that the top-up finished rather than our word for it. The published column is not superseded by any of this: every other number in this paper is at the published n, and `cell_matrix.topup_paths()` excludes those 26 records from every published read so that they cannot leak into one.

One cell changes its call, and it is a cell our own registration predicted would not. Moirai-Small/ETTm2 $h{=}192$ goes from inconclusive at $n{=}5$ ($\Delta_{\text{enc}}{=}+4.7{\pm}3.2$, $q{=}0.306$) to *freezing decisively better* at $n{=}12$ ($+5.7{\pm}1.8$, $q{=}0.043$), and the ladder puts the change at $n{=}12$ itself, its last registered seed, with the rung below it at $q{=}0.059$. The registration marked both uncorrected-tier cells *expected to remain inconclusive*, on the ground that they were being run to the uncorrected requirement while a decisive call needed 29 paired seeds. That expectation held for Moirai-Small/ETTh2 $h{=}192$ ($n{=}14$, $q{=}0.059$) and **failed here**. We report the failure and leave the registration file as it was written. **Why it failed is worth more than the fact that it did.** The 29 was computed at α/m, which is Bonferroni; the cell's paired t-test p at $n{=}12$ is 0.0098, the 7th smallest of the 31, so the threshold it actually faces is $7\alpha/m{=}0.011$ — seven times the 0.0016 its budget was sized against. Sizing a budget at α/m bounds the BH threshold from below and so the required n from above, exactly as Table 31's caption says it does, and this cell is what that conservatism looks like when it bites: re-estimated at the augmented n, its own adjusted requirement is 18, not 29.

The adjusted-tier cell moved the other way, which is the same lesson. Moirai-Base/ETTh2 $h{=}96$ was run to $n{=}5$, the budget estimated to make its call decisive, and it is *still* inconclusive: q went from 0.071 to 0.077, because the two added seeds took the mean from -22.3 to -16.4 and the sd from 5.0 to 10.4. Every required n in Table 31 holds the observed sd fixed, and this cell's sd was observed on three seeds. So the top-up returns a cell that resolved on a budget sized not to resolve and a cell that failed to resolve on a budget sized to resolve, and both are the same fact about post-hoc power at these n: the sd is the least certain quantity in the calculation, and it is the one the calculation treats as known.

What the augmented read moves, and what it does not. Across the matrix it changes that one call and no other, and no untopped cell's q moves by as much as 0.019; the three-way split goes from 2 freezing-better, 4 adaptation-better and 25 inconclusive to **3, 4 and 24**. The degradation definition of §3 returns **zero cells** under the augmented read as under the published one, in both the registered unanimity form and the interval form, since Δ_{enc} is not what either clause tests. What does move is the asymmetry stated above, that this matrix's screen admits cells where adaptation decisively wins and rejects the only cells where preservation decisively wins: at the augmented n there is one *admitted* cell where freezing decisively wins, at a gate of $+0.352$. That is the one result in this paper that points the screen's way on the screen's own admitted set, and it came from seeds whose number was fixed before they ran; it is also a single cell at $q{=}0.043$ against a 0.05 threshold whose neighbour misses the same threshold from the other side at 0.059. The body names the exception where it states the split (§5.3) and leaves the arithmetic here, because every number there is at the published n; we do not restate it as support for the screen, which across 31 cells it is not.

Generated by `scripts/paired_inference.py` and, for the top-up, `scripts/emit_power_topup.py`.

Table 30: **Paired seed-level inference on encoder adaptation.** $\Delta_{\text{enc}} = L_{\text{full}} - L_{\text{frozen}}$ per seed, in percent of the cell's zero-shot reference, so a positive value means allowing the encoder to adapt made the cell *worse*. B and D share a seed, an initialisation and a window sample, so this contrast is paired by construction. d is paired Cohen's d. The SEM column was dropped when q was added (nine columns overrun the textwidth and the interval states the same dispersion), so `sem` lives in `results/paired_inference.json`, on the body's convention (§3). Each row names its own n, which is not constant down the column (Table 5). q **is the decision column, adjusted for the 31 tests**: Benjamini–Hochberg within this contrast, with the three-way call (freeze-decisive, adapt-decisive, inconclusive) read off $q<0.05$ and the sign of Δ_{enc}, and bolded where decisive. BH runs on p_t, not p_{exact}, because 24 cells have an exact floor of 0.25 and a step-up procedure there would reject nothing at any effect size. **Two tests, because neither alone is adequate at these seed counts.** p_{exact} assumes nothing, but its resolution is fixed by N: the 24 cells at $N{=}3$ cannot reach $p < 0.05$ at any effect size, their smallest attainable value being 0.25. p_t and the interval do resolve there, and pay for it with a normality assumption on three points that cannot be checked. Both are shown so that a reading resting on that assumption is visible as one; the text states why no bootstrap interval is reported. In the two summary rows the last column is *not* an effect size: it gives the number of cells whose mean Δ_{enc} is positive, out of the cells in that row. † marks cells clearing the gate threshold under at least one admissible baseline of the ladder (Appendix A20). ‡ marks the Chronos cells, whose per-condition forgetting is confounded by the head/decoder mismatch; their Δ_{enc} is unaffected.

Cell	N	Δ_{enc}	95% CI (t)	p_t	q	p_{exact}	d
Chronos / ETTh1 (h=24, n=8000)†‡	3	−39.2	[−98.9, +20.5]	0.106	0.173	0.250	−1.63
Chronos / ETTh2 (h=24, n=8000)†‡	3	−12.9	[−15.5, −10.4]	0.002	**0.016**	0.250	−12.57
Chronos / Weather (h=24, n=8000)†‡	3	−21.6	[−35.5, −7.6]	0.022	0.085	0.250	−3.84
Moirai-B / ETTh2 (h=192, n=1000)†	3	−20.2	[−24.9, −15.5]	0.003	**0.018**	0.250	−10.66
Moirai-B / ETTh2 (h=96, n=1000)†	3	−22.3	[−34.7, −10.0]	0.016	0.071	0.250	−4.50
Moirai-B / ETTh1 (h=96, n=1000)†	3	+37.5	[+2.1, +72.9]	0.045	0.099	0.250	+2.63
Moirai-B / ETTm2 (h=192, n=1000)†	3	+1.3	[−13.4, +16.0]	0.740	0.740	1.000	+0.22
Moirai-B / ETTm2 (h=96, n=1000)†	3	+7.0	[−3.2, +17.3]	0.098	0.173	0.250	+1.70
Moirai-L / ETTh2 (h=96, n=500)†	3	−20.8	[−26.6, −15.0]	0.004	**0.022**	0.250	−8.87
Moirai-L / ETTh1 (h=96, n=1000)†	3	+28.2	[+7.3, +49.1]	0.028	0.098	0.250	+3.35
Moirai-S / ETTh1 (h=96, n=1000)†	5	+22.1	[+15.6, +28.6]	0.001	**0.011**	0.062	+4.21
Moirai-S / ETTh2 (h=192, n=500)†	10	+6.1	[−1.5, +13.7]	0.101	0.173	0.100	+0.58
Moirai-S / ETTh2 (h=96, n=500)†	10	+2.8	[−5.6, +11.2]	0.472	0.522	0.557	+0.24
Moirai-S / ETTm2 (h=192, n=1000)†	5	+4.7	[−4.2, +13.6]	0.217	0.306	0.312	+0.65
Moirai-S / ETTm2 (h=96, n=1000)†	5	+2.6	[−6.9, +12.1]	0.495	0.529	0.562	+0.34
Moirai-S / Electricity7 (h=96, n=1000)†	3	−3.2	[−9.9, +3.4]	0.170	0.251	0.250	−1.22
Moirai-S / Weather (h=96, n=1000)†	3	+53.1	[+9.2, +97.0]	0.035	0.099	0.250	+3.00
TimesFM / ETTh1 (h=24, n=1000)†	3	−1.4	[−13.5, +10.6]	0.658	0.680	0.750	−0.30
TimesFM / ETTh2 (h=24, n=1000)†	3	−7.1	[−20.5, +6.2]	0.148	0.230	0.250	−1.33
TimesFM / Weather (h=24, n=1000)†	3	−1.6	[−8.1, +4.8]	0.387	0.445	0.500	−0.63
Chronos / Electricity (h=24, n=8000)‡	3	−2.7	[−10.7, +5.4]	0.287	0.358	0.500	−0.83
Chronos / ETTm2 (h=24, n=8000)‡	3	−29.5	[−57.1, −1.9]	0.044	0.099	0.250	−2.66
Moirai-B / ETTh1 (h=192, n=1000)	3	+47.0	[−12.4, +106.3]	0.077	0.158	0.250	+1.97
Moirai-B / Weather (h=192, n=1000)	3	+12.5	[−25.1, +50.1]	0.288	0.358	0.500	+0.83
Moirai-B / Weather (h=96, n=1000)	3	+9.4	[+1.0, +17.8]	0.040	0.099	0.250	+2.78
Moirai-L / Weather (h=96, n=1000)	3	+12.0	[−20.3, +44.3]	0.251	0.338	0.250	+0.92
Moirai-S / ETTh1 (h=192, n=1000)	5	+13.5	[+8.8, +18.2]	0.001	**0.014**	0.062	+3.53
Moirai-S / ILI (h=24, n=452)	10	−38.6	[−41.6, −35.6]	0.000	**0.000**	0.002	−9.15
Moirai-S / Electricity7 (h=192, n=1000)	3	−2.6	[−11.7, +6.4]	0.338	0.403	0.500	−0.72
Moirai-S / Weather (h=192, n=1000)	3	+62.1	[+3.6, +120.6]	0.045	0.099	0.250	+2.64
TimesFM / ETTm2 (h=24, n=1000)	3	−7.7	[−18.9, +3.5]	0.097	0.173	0.250	−1.71
Mean of cell means, 95% CI from resampling (backbone, dataset) clusters:							
20 value-cells / 11 clusters		+0.7	[−11.1, +14.1]				10/20
all 31 cells / 15 clusters		+2.9	[−12.7, +15.1]				16/31

Table 31: **What each surviving cell's own seed budget could detect.** The 7 cells clearing $V_{\text{ridge}} \geq 0.20$, with the minimum detectable effect implied by their own dispersion: MDE $= t_{1-\alpha/2,n-1}\,\text{sd}/\sqrt{n}$, the smallest $|\Delta_{\text{enc}}|$ a two-sided paired t would resolve at that n. $n^\star$ is the paired seed count at which the *observed* effect would resolve, holding the observed sd fixed; it is therefore a post-hoc estimate and not a design calculation. **Two columns, because a single test and a decisive call cost different budgets.** $n^\star$ is computed at an uncorrected $\alpha{=}0.05$; $n^\star_{\text{adj}}$ is computed at α/m with $m{=}31$ tests, which bounds the Benjamini–Hochberg threshold the call in Table 30 actually uses from below and so bounds the budget from above. Reporting only $n^\star$ would say Moirai-Base/ETTh2 $h{=}192$ is already resolved at $n{=}3$, which its own $q{=}0.018$ happens to agree with, and would say the same of $h{=}96$, whose q is 0.071. **The 4 starred rows are the point of the table**: they need more than our feasibility cap of 20 paired seeds (≈20 GPU-hours per cell at two fine-tunes per seed), so we report them as *unresolvable under this design*, with the number, rather than as effects that are absent.

Cell	V_{ridge}	n	Δ_{enc}	sd	MDE	$n^\star$	$n^\star_{\text{adj}}$
Moirai-S / ETTh2 ($h{=}192$, $n{=}500$)	$+0.932$	10	$+6.1$	10.63	7.60	14	$36^\star$
Moirai-B / ETTh2 ($h{=}192$, $n{=}1000$)	$+0.925$	3	-20.2	1.90	4.71	2	4
Moirai-L / ETTh2 ($h{=}96$, $n{=}500$)	$+0.875$	3	-20.8	2.35	5.83	3	4
Moirai-S / ETTh2 ($h{=}96$, $n{=}500$)	$+0.874$	10	$+2.8$	11.78	8.43	71	$182^\star$
Moirai-B / ETTh2 ($h{=}96$, $n{=}1000$)	$+0.777$	3	-22.3	4.97	12.33	3	5
Moirai-S / ETTm2 ($h{=}192$, $n{=}1000$)	$+0.352$	5	$+4.7$	7.16	8.89	12	$29^\star$
Moirai-B / ETTm2 ($h{=}192$, $n{=}1000$)	$+0.211$	3	$+1.3$	5.90	14.66	83	$212^\star$

Table 32: **The pre-registered seed top-up, beside the published seed counts.** The 7 cells clearing $V_{\text{ridge}} \geq 0.20$. 3 were topped up under `results/power_topup/preregistration_power.json`, which fixed which cells, which seeds and how many (26 runs) before any of those runs existed; all 26 completed. **No published number is replaced.** Each cell's *published* row is the analysis of Table 30, re-verified against `results/paired_inference.json` at emit time; its *augmented* row is a second computation with the added seeds admitted, on the same estimator and the same Benjamini–Hochberg correction over the same 31 cells. **One call changes**: Moirai-S / ETTm2 ($h{=}192$, $n{=}1000$) moves from inconclusive to freezing-decisive, first at $n{=}12$, with the rung below it at $q{=}0.059$. Across all 31 cells the augmented read changes that call and no other (published 2 freezing-better, 4 adaptation-better, 25 inconclusive; augmented 3, 4, 24). The 2 cells on the *uncorrected* tier were run to a budget below their BH-adjusted requirement and were registered as *expected to remain inconclusive*, which held for 1 of the 2 (Moirai-S / ETTh2 ($h{=}192$, $n{=}500$)) and failed for Moirai-S / ETTm2 ($h{=}192$, $n{=}1000$). A failed registered prediction is reported as one; the registration is not edited. The ladder behind "first at" admits each cell's added seeds in registered order with the other topped-up cells at their final n, so a change is attributable to that cell's own seeds and not to BH moving underneath it. The last column gives each cell's registered budget and the tier it came from (adj. = the BH-adjusted requirement, unc. = the uncorrected one), or the reason it was not topped up.

Cell	V_{ridge}	read	n	Δ_{enc}	q	call	top-up
Moirai-B / ETTh2 ($h{=}192$, $n{=}1000$)	$+0.925$	published	3	$-20.2{\pm}1.1$	0.018	adapt	not run: already decisive
Moirai-B / ETTh2 ($h{=}96$, $n{=}1000$)	$+0.777$	published	3	$-22.3{\pm}2.9$	0.071	incon.	
		augmented	5	$-16.4{\pm}4.7$	0.077	incon.	3→5 (adj.)
Moirai-B / ETTm2 ($h{=}192$, $n{=}1000$)	$+0.211$	published	3	$+1.3{\pm}3.4$	0.740	incon.	not run: needs $n{=}83$
Moirai-L / ETTh2 ($h{=}96$, $n{=}500$)	$+0.875$	published	3	$-20.8{\pm}1.4$	0.022	adapt	not run: already decisive
Moirai-S / ETTh2 ($h{=}192$, $n{=}500$)	$+0.932$	published	10	$+6.1{\pm}3.4$	0.173	incon.	
		augmented	14	$+6.9{\pm}2.5$	0.059	incon.	10→14 (unc.)
Moirai-S / ETTh2 ($h{=}96$, $n{=}500$)	$+0.874$	published	10	$+2.8{\pm}3.7$	0.522	incon.	not run: needs $n{=}71$
Moirai-S / ETTm2 ($h{=}192$, $n{=}1000$)	$+0.352$	published	5	$+4.7{\pm}3.2$	0.306	incon.	
		augmented	12	$+5.7{\pm}1.8$	0.043	freeze	5→12 (unc.)

A23 The Graded Value Axis: Is the Dissociation Confined to Cells With Nothing to Lose?

The screen in the body is binary: a cell either clears V_b=0.20 or it does not. That discards the only interesting covariate, and it leaves open the single strongest objection to this paper's negative result: that drift fails to predict Δ_{enc} only *because* 16 of our 31 cells have no measured advantage to lose, so we are reading noise where there is nothing to read. If that were right, the drift-to-outcome relation would strengthen with pre-trained value, and our finding would be a statement about weak benchmarks rather than about the diagnostic. This appendix estimates that dependence two ways and reports an interval rather than a verdict.

The axis. Value here is the graded score of Appendix A20: V_{best} against the *strongest admissible* rung of the eight-baseline ladder, which is a **minimum** over rungs, since $V_b = 1 - \text{MSE}_{\text{ZS}}/\text{MSE}_{\text{base}}$ and a weaker rival inflates it. It runs from -1.638 (Moirai-Small/ILI) to $+0.058$ (Moirai-Small/ETTh2 h=96) with mean -0.610, and its ceiling below 0.20 is the same fact as "no cell clears the gate against every admissible rung". A cell with no admissible rung at all is *dropped*, never scored as zero value: a missing denominator and a denominator of zero are different facts, and the second would place the cell at the origin of this very axis.

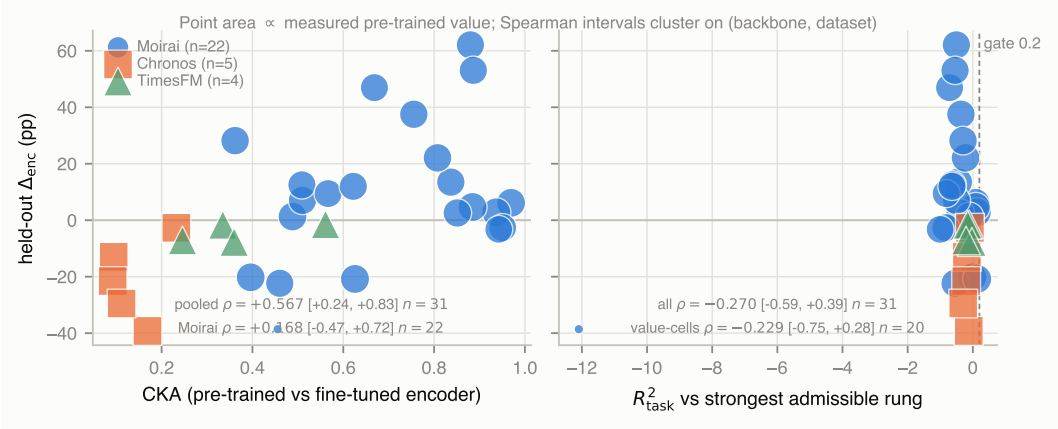


Figure 6: **Neither drift nor measured pre-trained value orders the encoder intervention.** Both panels share y= held-out Δ_{enc} (positive = freezing the encoder did better). Point area is proportional to the graded value score in *both* panels, so the left panel answers the objection visually: if drift predicted the outcome wherever there was something to lose, the large points would trend and the small ones would scatter. Backbone is encoded by colour *and* marker because CKA is not comparable across architectures, which is why the pooled left-panel interval must not be read as a within-backbone one. Drawn by `paper_8/fig_value_axis.py` from `results/value_axis.json`; the script asserts its own counts against that file and fails rather than drawing a figure that disagrees with this text.

The rank answer: the relation is not stronger where there is more to lose. Restricting to the 20 value-cells gives ρ=+0.567 with clustered CI $[+0.191, +0.889]$; the 11 cells with no admissible value give ρ=+0.709 with CI $[-0.077, +0.901]$ (Table 33). The restriction the objection asks for does not sharpen the relation, and the value-cell figure is not evidence that drift orders the outcome: it pools all three backbones, and within Moirai (the only arm with the cells to ask), the same statistic is ρ=+0.168 with an interval spanning zero. We do not test the difference of those two ρs: a difference of rank statistics on 20 and 11 cells from 11 and 7 clusters has no usable sampling distribution, and reporting one would be the kind of number this paper is arguing against.

The slope answer, where the dependence is a single coefficient. On the slope scale the interaction is estimable, and it is b_3=−31.5 pp per unit CKA per unit value with CI $[-209.6, +166.2]$, against a CKA slope at mean value of +39.8 with CI $[-55.1, +179.4]$. Two things to say about it and no more. The point estimate has the *opposite* sign to the objection (drift orders the outcome slightly *less* well where there is more value), and the interval spans zero so widely that only an interaction several times larger than the CKA slope itself would have been visible at 15 clusters. **So this is not a refutation of**

Table 33: **The dependence, estimated two ways.** Rank rows are Spearman ρ with 10,000 bootstrap draws over (backbone, dataset) clusters, seed 0. The slope rows fit all cells at once with backbone identity absorbed as a fixed effect and both regressors centred, so b_1 is the CKA slope *at mean value*; b_3 is the quantity at issue. Emitted by `scripts/value_axis.py -latex`, which delegates the slope fit to `scripts/cka_fixed_effects.py` rather than refitting it.

Relation	n	clusters	ρ or b	95% CI over clusters
CKA vs Δ_{enc}, the value-cells	20	11	$+0.567$	$[+0.191, +0.889]$ excludes 0
CKA vs Δ_{enc}, cells with no admissible value	11	7	$+0.709$	$[-0.077, +0.901]$ includes 0
CKA vs Δ_{enc}, within Moirai	22	6	$+0.168$	$[-0.469, +0.717]$ includes 0
CKA vs Δ_{enc}, pooled (not a licensed reading)	31	15	$+0.567$	$[+0.239, +0.828]$ excludes 0
V_{best} vs Δ_{enc}, the value-cells	20	11	-0.229	$[-0.755, +0.284]$ includes 0
V_{best} vs Δ_{enc}, all cells	31	15	-0.270	$[-0.592, +0.392]$ includes 0
Slope scale, all cells at once: $\Delta_{enc} = b_0 + b_1\text{CKA} +$ *backbone FE* $+ b_2 v + b_3(\text{CKA}\times v)$, $v =$ *value*				
b_1, CKA slope at mean value (pp per unit CKA)	31	15	$+39.8$	$[-55.1, +179.4]$ includes 0
b_3, CKA$\times$value interaction	31	15	-31.5	$[-209.6, +166.2]$ includes 0

the objection; it is the statement that these cells cannot separate it from no dependence at all. That is a weaker claim than we would like to make, and it is the one the data support.

And value does not order the outcome by itself. The graded score against Δ_{enc} gives $\rho{=}-0.270$, CI $[-0.592, +0.392]$ across all 31 cells and $\rho{=}-0.229$, CI $[-0.755, +0.284]$ across the 20 value-cells, the same non-result as CKA, on the covariate that was supposed to rescue it. The right panel of Figure 6 is that row: every point lies left of the 0.20 line, and the cells that improve most under encoder adaptation are not the ones with the least to lose.

Not an artefact of cells with nothing to lose. Drift might fail to order Δ_{enc} only because 11 of the 31 cells beat no rung of the ladder; grading the screen rather than dichotomising it says otherwise: the relation is no stronger on the cells with a measured advantage than on the cells without, and the estimated interaction has the wrong sign for the objection and an interval far too wide to settle it.

A24 Cross-Cell Statistics, and a Pooled Analysis We Do Not License

Across the 31 intervention cells of Table 5 we ask whether either observational diagnostic (CKA, or ℓ_2 weight drift) orders the held-out intervention contrast B$-$D. Two predictors rather than one is deliberate: if *both* fail, the statement is about observational measures of fine-tuning in general and not about CKA specifically. The statistic is Spearman ρ with a 10,000-resample bootstrap CI (seed 0), reported at two resampling levels: over cells, and over the (backbone, dataset) clusters that Appendix A27 argues is the correct level, and emitted by `scripts/clustered_inference.py`. Pooled over all 31 cells:

Predictor	n	Clusters	ρ	95% CI over cells	95% CI over clusters
CKA	31	15	$+0.567$	$[+0.265, +0.759]$ excludes 0	$[+0.239, +0.828]$ excludes 0
ℓ_2 weight drift	25	9	$+0.118$	$[-0.300, +0.518]$ includes 0	$[-0.489, +0.545]$ includes 0

The ℓ_2 row covers 25 of the 31 cells because the five Chronos cells and ILI retain no weight-drift measurement in their run records. No p-values are reported in this appendix or the next: they were previously computed from the cell-level resampling, which mis-states the dependence between cells, and this analysis is descriptive (Appendix A27).

The pooled CKA row is not a licensed analysis, and it is the row that would help us. The three backbones occupy almost disjoint CKA ranges (Moirai 0.36–0.97, Chronos 0.09–0.23, TimesFM 0.25–0.56), *and* differ systematically in outcome: every Chronos and TimesFM cell favours adaptation, and every large positive B$-$D is Moirai's. A rank correlation pooled over 31 cells drawn from three such groups therefore measures which backbone a cell came from at least as much as how far its encoder moved: it is a textbook setting for a group-level artefact of the Simpson's-paradox kind, and CKA magnitude is representation-dependent, which is why this paper compares CKA only within a

backbone (§6). We stress the direction of the incentive. The pooled figure is the *favourable* one (it is the only reading in this appendix under which a diagnostic we introduced orders the intervention with an interval excluding zero), and the body nevertheless leads with the within-backbone figure. We decline to make the positive claim the pooled number would support, because the restriction that forbids it was fixed before these cells were run rather than chosen after seeing them.

Table 34: Is the pooled CKA correlation a backbone artefact? The same statistic (Spearman ρ of CKA against held-out B−D, 10,000-resample bootstrap CI) computed pooled and then stratified by backbone, at both resampling levels. "Cl." is the number of (backbone, dataset) clusters, which is the effective sample size for the clustered interval and is far below n. Values are emitted by `scripts/clustered_inference.py`. The pooled row is reported for audit, not read as evidence; the two small arms carry essentially no information at their cell counts, and for them the two levels coincide because every cell is already its own cluster.

Grouping	n	Cl.	CKA range	ρ	95% CI over cells	95% CI over clusters
Pooled (against our rule)	31	15	0.09–0.97	$+0.567$	$[+0.265, +0.759]$ excl. 0	$[+0.239, +0.828]$ excl. 0
Moirai only	22	6	0.36–0.97	$+0.168$	$[-0.328, +0.598]$ incl. 0	$[-0.469, +0.717]$ incl. 0
Chronos only	5	5	0.09–0.23	$+0.100$	$[-1.000, +1.000]$ incl. 0	$[-1.000, +1.000]$ incl. 0
TimesFM only	4	4	0.25–0.56	$+0.400$	$[-0.778, +1.000]$ incl. 0	$[-0.778, +1.000]$ incl. 0

Within Moirai the interval is wide, and we say so rather than round it to zero. On the 22 Moirai cells $\rho{=}+0.168$ with a cell-level CI $[-0.328, +0.598]$, widening to $[-0.469, +0.717]$ over the six clusters, a width of 0.93 on a scale whose whole range is 2, and 1.19 once the dependence is propagated, which is the honest bound this evidence supports. That interval comfortably admits a moderate positive relationship, and 22 cells cannot rule one out. The claim we license is the one the body makes: *CKA does not order our cells*. It is not that CKA is uninformative in general, and a reader who expects a real within-backbone relationship to exist will find nothing here that contradicts them; what they will not find is a cell ordering they could have acted on.

The sharpest form of the dissociation, inside one capacity. Restricting further to Moirai-Base (eight cells, one backbone, one capacity, so neither architecture nor parameter count varies), the ordering CKA gives is not merely uninformative but *reversed* (Table 35). Sorted by CKA, condition-B forgetting runs monotonically from -36.3% at the most drifted cell to $+39.5\%$ and $+22.8\%$ at the two least drifted, and those two least drifted cells are the only two in the arm that degrade at all. A practitioner reading "CKA is low, the encoder has been restructured, intervene" would have intervened on exactly the six cells that did not need it and left alone the two that did. Eight cells cannot establish the reverse relationship either (this is the same interval as above, restricted), which is the point: the sign of the association is not stable enough within a single backbone and capacity to carry a decision.

Chronos and TimesFM cannot be asked at all. With $n{=}5$ and $n{=}4$ the bootstrap resamples only a handful of distinct rank permutations and the interval spans nearly the entire admissible range, $[-1.000, +1.000]$ and $[-0.778, +1.000]$. Both rows are tabulated for completeness and neither carries information. In particular the $+0.400$ on four TimesFM cells must not be read as agreement with the pooled sign: it is one of very few values the statistic can take on four cells, and its interval covers almost everything.

ℓ_2 weight drift does no better, and it is the second of the two diagnostics we defined. Pooled (already the reading most likely to manufacture a correlation), weight drift gives $\rho{=}+0.118$ with a CI that includes zero at both resampling levels, widening from $[-0.300, +0.518]$ over cells to $[-0.489, +0.545]$ over its nine clusters. Since §3 defines exactly these two representation diagnostics, their joint failure is the more useful statement: it is not that we picked the wrong summary of the encoder's movement, but that neither summary we defined orders the outcome. The two are also not redundant and can move independently: a stack whose weight drift is *exactly* zero can read CKA well below one when the projections feeding it are trainable, which is what condition D does on TimesFM, where ℓ_2 drift over the transformer stack is 0.00 by construction while its measured CKA runs 0.38–0.89 (§3, §6; the analogous Moirai control is Appendix A8). A geometric battery (RBF CKA, SVCCA, orthogonal Procrustes, mean cosine and effective-rank change, on the 59 encoders that retained checkpoints) is Appendix A25. None of those measures orders the outcome either, and

Table 35: **The within-backbone dissociation, one capacity, eight cells.** Every Moirai-Base cell, ordered by CKA (most drifted first). "Full FT" and "Frozen" are percent change in held-out MSE against zero-shot, so negative is better and positive is task-level forgetting; Δ_{enc} is the manipulated contrast between them. Gate values marked [f] fail the 0.20 screen. Reading down the CKA column, the outcome reverses sign: the two least drifted cells are the only two that degrade. Emitted by `scripts/emit_dissociation.py`, which asserts both of those claims against the run records and exits non-zero if they stop holding; these four CKA values and four percentages were previously quoted in body prose alone.

Cell (h)	CKA	Gate	Full FT	Frozen	Δ_{enc}
ETTh2 (192)	0.396	$+0.925$	-36.3%	-16.1%	-20.2
ETTh2 (96)	0.460	$+0.777$	-31.4%	-9.1%	-22.3
ETTm2 (192)	0.488	$+0.211$	-8.4%	-9.7%	$+1.3$
Weather (192)	0.509	-0.633 [f]	-18.1%	-30.6%	$+12.5$
ETTm2 (96)	0.510	-0.257 [f]	-13.1%	-20.1%	$+7.0$
Weather (96)	0.567	-0.823 [f]	-11.7%	-21.1%	$+9.4$
ETTh1 (192)	0.669	-0.600 [f]	$+39.5\%$	-7.5%	$+47.0$
ETTh1 (96)	0.755	-0.312 [f]	$+22.8\%$	-14.7%	$+37.5$

they do not order the *encoders* alike, so the failure is not a property of the summary we happened to pick.

A backbone fixed-effects reading, and its limits. The natural alternative to subsetting is to absorb backbone identity and let CKA enter as a continuous regressor, which answers the objection that we adopted the split that removes the significant relationship. We do fit it (Table 37, `scripts/cka_fixed_effects.py`): the within-backbone slope of held-out B$-$D on CKA is $+39.1$ pp per unit CKA with backbone fixed effects and $+39.9$ once CKA$\times$backbone interactions are added, against $+47.7$ with no backbone term at all. Both adjusted intervals include zero once clusters rather than cells are resampled. So the reduction from $+47.7$ to $+39.1$ shows that absorbing backbone identity does *not* by itself destroy the pooled slope (most of it survives adjustment), while the interval shows that 31 cells drawn from 15 clusters cannot pin the surviving slope down. What we do *not* do is fit variance components: with 22, 5 and 4 cells in the three groups there is no repeated-measures structure to estimate them from, so a mixed model over three groups of that size is not identified and any interval it produced would be an artefact of the specification. Cells here are unequally powered and not independent, so every cross-cell number in this appendix is descriptive.

A25 Is the Reading Specific to CKA? Six Measures on 59 Encoders

The body reads drift through linear CKA, so a reader is entitled to ask whether "drift does not order the outcome" is a fact about representations or a fact about CKA. Appendix A24 gives the answer that covers the whole matrix: ℓ_2 weight drift, a geometrically unrelated quantity defined in §3 alongside CKA, fails on the same 31 cells. This appendix gives the second answer, on geometry, and states up front what it cannot do.

What it cannot do. Only the n=10k runs retained encoder checkpoints. The 22-cell Moirai matrix is n=500/1,000 and kept none, so this battery *cannot* recompute the within-Moirai ρ under a different measure; that would require retraining the matrix, on different hardware, which would perturb every number in the paper. What it can do is test measure-specificity on the 59 encoders that do exist: Moirai-Small at h=96, n=10k, on ETTh1, ETTh2 and ETTm2, under full fine-tuning and under two layer-unfreezing regimes (top 3 of 6, and all 6), in seven (dataset, run-set, regime) groups.

Measured on the representations the paper's CKA column describes. Each run's record stores `final_cka`, computed during training on the first 300 validation evaluation windows with the encoder output mean-pooled. The battery (`scripts/drift_metric_battery.py`) rebuilds exactly those windows (extended lookback 96+96, raw values, the model normalising internally), and recomputes linear CKA as a self-check: it agrees with the stored value to four decimal places on all 59 encoders, so every other column is known to describe the same representations rather than

a similar-looking set. The checkpoint load asserts that `load_state_dict(strict=False)` reports no missing and no unexpected keys, because a namespace mismatch loads nothing silently and returns CKA ≈ 1, which reads as "no drift".

Table 36: **Six measures of representation change on 59 Moirai-Small encoders.** All computed on identical windows and identical pooling. "ρ with CKA" is the rank correlation of each measure with linear CKA over the 59 encoders; the last two columns correlate each measure with forgetting, pooled and then within each of the six groups of at least five encoders. Forgetting here is the *selection-split* figure: these runs stored no zero-shot test MSE, so a held-out contrast cannot be reconstructed from them, and §7.1 shows same-split scoring can invert the sign of the intervention. This arm is therefore a comparison of measures, not a held-out claim. Emitted by `scripts/drift_metric_battery.py`; the pooled column mixes three datasets and is descriptive for the reason Appendix A24 gives about pooling.

	over 59 encoders				ρ with forgetting	
Measure	min	max	mean	ρ with CKA	pooled	per group
Linear CKA	$+0.194$	$+0.915$	$+0.588$	$+1.000$	$+0.350$	$[-0.19, +0.60]$
Linear CKA (token)	$+0.626$	$+0.957$	$+0.751$	$+0.542$	-0.123	$[-0.33, +0.55]$
RBF CKA	$+0.624$	$+0.913$	$+0.734$	$+0.412$	$+0.117$	$[-0.36, +0.90]$
SVCCA	$+0.726$	$+0.939$	$+0.776$	$+0.553$	-0.038	$[-0.70, +0.35]$
Procrustes sim.	$+0.715$	$+0.947$	$+0.818$	$+0.675$	$+0.127$	$[-0.52, +0.80]$
Mean cosine	$+0.660$	$+0.990$	$+0.840$	$+0.073$	-0.287	$[-0.67, +0.60]$
Eff. rank ratio	$+0.969$	$+1.250$	$+1.119$	-0.251	$+0.245$	$[+0.00, +0.65]$

The more invariant the measure, the less drift there is to see. Linear CKA spans 0.194 to 0.915 across these encoders. SVCCA (invariant to any invertible linear map, so it asks whether the same information survives in a possibly rotated and rescaled basis) spans only 0.726 to 0.939, and orthogonal-Procrustes similarity 0.715 to 0.947. The cell that CKA scores at 0.194 is at 0.73 on SVCCA. This is the concrete form of the objection limitation (7) concedes: a large part of what CKA reports as drift is a change of basis under which the subspace, and with it the linearly decodable content, is largely preserved. It is also why we do not claim CKA is invalid (it measures what it measures), only that a paper cannot infer damage from it.

The measures do not order the encoders alike, so "drift" is not one quantity. Rank correlations among them run from $+0.073$ (linear CKA against mean cosine, the least invariant measure in the battery) through $+0.412$ (linear CKA against RBF CKA) to $+0.845$ (RBF CKA against Procrustes). Choosing a similarity measure chooses an ordering of these encoders. We had expected the opposite (a family of near-clones), and report the disagreement because it makes the next paragraph the load-bearing one.

No measure orders the outcome either. Against selection-split forgetting, the pooled rank correlations run from $+0.350$ (linear CKA) to -0.287 (mean cosine), and the pooled figures are the flattering ones: within groups, where dataset identity is held fixed, every similarity measure changes sign across the six groups (linear CKA $[-0.19, +0.60]$, RBF CKA $[-0.36, +0.90]$, SVCCA $[-0.70, +0.35]$, Procrustes $[-0.52, +0.80]$, mean cosine $[-0.67, +0.60]$). Six measures spanning no invariance to full linear-map invariance therefore reproduce the body's reading rather than overturning it: the disagreement among them does not resolve into any one of them ordering the outcome. The single exception is not a similarity measure at all. The effective-rank ratio (how many directions the representation uses after fine-tuning against before) is above 1 on 55 of the 59 encoders (0.969–1.250, so fine-tuning generally *expands* the spectrum) and is the one column with no negative group value ($[+0.00, +0.65]$). At six groups, one backbone and one sample size we record that as a lead for future work, not a diagnostic; it is exactly the kind of pattern this paper's own prospective test (§7.2) was built to keep us honest about.

What this licenses, and what it does not. Read together, the six measures support the generalisation our title implies but the body does not claim: on this arm, *post-fine-tuning representation similarity should not be assumed to identify harmful encoder adaptation*, whichever similarity is chosen. What that reading cannot be is a claim about Δ_{enc}: these 59 encoders are one backbone at one capacity, horizon and sample size, none of them among the 31 cells the intervention is measured on, and their

outcome column is single-arm selection-split forgetting. A measure that failed here could still order a held-out intervention elsewhere, and we did not test one that does.

A26 What Predicts the Intervention, and Why None of It Is a Shortcut

If some quantity measurable from one run ordered B−D, a practitioner could skip the frozen control. We tested every candidate we had, within Moirai only, because it is the only arm with enough cells to ask (22). The statistic is again Spearman ρ against held-out B−D with a 10,000-resample bootstrap CI at both resampling levels, from `scripts/clustered_inference.py`:

Predictor (Moirai only)	n	Cl.	ρ	95% CI over cells	95% CI over clusters
forg_B (condition B only)	22	6	$+0.894$	$[+0.667, +0.981]$ excludes 0	$[+0.594, +0.965]$ excludes 0
forg_D	22	6	$+0.073$	$[-0.324, +0.429]$ includes 0	$[-0.343, +0.412]$ includes 0
Gate V_{ridge}	22	6	-0.293	$[-0.706, +0.188]$ includes 0	$[-0.712, +0.709]$ includes 0
CKA	22	6	$+0.168$	$[-0.328, +0.598]$ includes 0	$[-0.469, +0.717]$ includes 0
ℓ_2 weight drift	21	5	$+0.318$	$[-0.122, +0.673]$ includes 0	$[-0.329, +0.861]$ includes 0

Weight drift covers 21 of the 22 cells because ILI's run records retain no drift measurement. The gate row is the same figure reported in Appendix A29.2 as a consequence of the baseline correction; it appears here beside its competitors. **These 22 cells come from six clusters**, and the gate row is the one place in the paper where that distinction changes a verdict: see Appendix A27.

The forg_B result is an identity, not a discovery. B−D $\equiv \text{forg}_\text{B} - \text{forg}_\text{D}$ holds *exactly*, because both terms are percentage changes against the same zero-shot reference: over all cells the maximum deviation is $\max|(\text{forg}_\text{B} - \text{forg}_\text{D}) - (\text{B}-\text{D})| = 0.000$. The dispersion is then overwhelmingly on the B side, SD 26.2 against forg_D's 8.1, so B−D is close to a rescaled copy of forg_B and $\rho{=}{+}0.894$ is close to arithmetic rather than to a finding. It confers no ability to skip condition D: forg_B exists only once condition B has been run, and at that point running D is one further run on the same cell. This is exactly the kind of correlation that looks like a shortcut and is not, and the near-zero forg_D row is the reason: the rule works only because the frozen arm happens to be nearly constant on this backbone, which is an empirical property of these cells and not a theorem.

The gate's continuous score points the wrong way, but only the sign is supported (Figure 7). On the same 22 cells the gate correlates *negatively* with held-out B−D, $\rho{=}{-}0.293$ on the selection-split gate this paper reports as primary. Neither resampling level supports an interval claim: cells give $[-0.706, +0.188]$ and the six clusters give $[-0.712, +0.709]$. On the retrospective test-side gate the same correlation is $\rho{=}{-}0.518$ with a cell-level CI of $[-0.840, -0.040]$ that *does* exclude zero, and a clustered CI of $[-0.810, +0.500]$ that does not. **We withdrew the "excludes zero" reading on clustering grounds** (Appendix A27); scoring the gate on the split that does not touch the outcome removes it a second time and independently, and the magnitude nearly halves between the two splits, which is itself the point of §7.2, that there is no stable ordering here to exploit. The mechanism is legible in Table 5: the cells with the largest measured pre-trained advantage are the Moirai/ETTh2 cells, and three of those five are the cells where full fine-tuning beats the frozen encoder by the most (B−D -20.2 to -22.3 pp), while the cells with the most negative gates (Weather and ETTh1) are where freezing wins by tens of points, up to $+62.1$ pp. A practitioner following "preserve where the gate fires" would therefore freeze precisely where adapting the encoder pays best. We report this at the strength it has: a within-Moirai point estimate on 22 retrospective cells from six series, with a clustered interval that spans zero. **The negative sign is a description of these cells, not an interval-supported claim, and the magnitude is not estimated to any useful precision.** The mechanism above is legible cell by cell in Table 5 and does not depend on the correlation surviving; that is why we still report the direction.

The other two arms cannot adjudicate. On TimesFM ($n{=}4$) the gate gives $\rho{=}{+}0.800$, CI $[+0.333, +1.000]$, forg_B gives $+1.000$ and CKA $+0.400$, CI $[-0.778, +1.000]$; on Chronos ($n{=}5$) the gate gives $\rho{=}{+}0.500$, CI $[-0.667, +1.000]$, and forg_B gives -0.100. These are reported for completeness and are uninformative at four and five cells: a bootstrap over four cells resamples only a few distinct rank patterns, so even the TimesFM gate interval, which nominally excludes zero, is an artefact of that granularity and we read nothing from it. Chronos's forg_B carries a further defect and

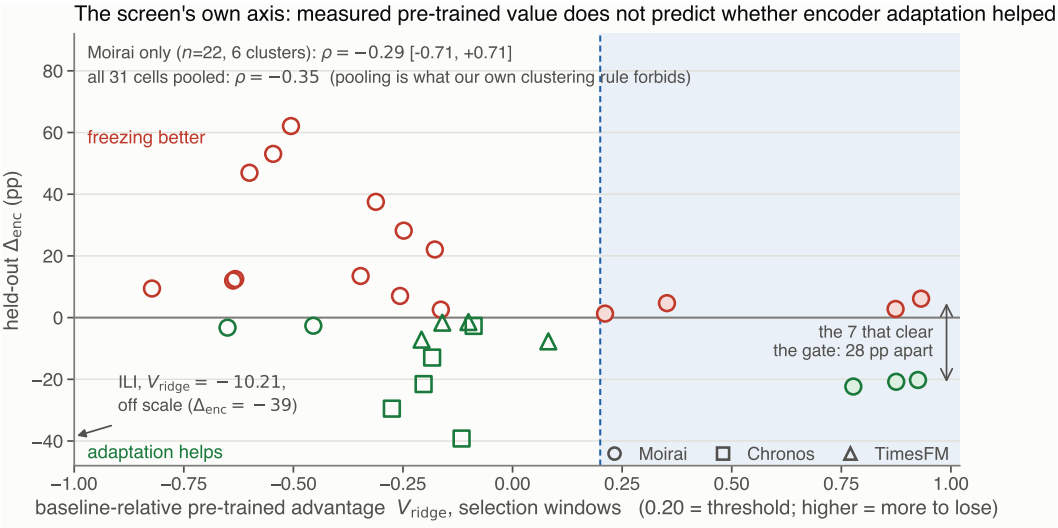


Figure 7: **The screen's own axis.** All 31 intervention cells as the selection-split gate against held-out Δ_{enc} (positive = freezing better). Filled markers clear $V_{\text{ridge}} \geq 0.20$; the shaded strip is the only region a cell could occupy and still satisfy clause (i) of the degradation definition, and most of the cells in it lie *below* zero, where letting the encoder adapt helped. The two axes are scored on disjoint window sets, so a cell's horizontal position is not a re-description of its vertical one. ILI is named on the frame rather than plotted: its gate is -10.21, and an axis wide enough to include it compresses the other 30 cells into a fifth of the panel.

is flagged `forg_confounded` in the code: a head/decoder mismatch means its B-side number is not a clean per-condition forgetting measurement, so it is retained only to show where the forg_B rule breaks and is never quoted as a forgetting value.

Nothing here is a replacement diagnostic. The honest summary of this appendix and of Appendix A24 together is that no observational quantity we measured orders the intervention in a usable direction: CKA and ℓ_2 drift do not order it, the gate orders it *backwards*, and the one predictor that tracks it does so by identity and only after the run that would have told you the answer anyway. On the evidence in Table 5, the only reliable way to know whether encoder adaptation helped a given cell is to run both conditions on that cell.

A27 Cells Are Not Independent: Clustered Intervals for Every Cross-Cell Statistic

Every interval in Appendices A24 and A26 was previously computed by resampling *cells*, which treats each cell as an independent observation. They are not. A cell is a (backbone, size, dataset, horizon) combination, so the 31 cells draw on 3 backbones and 7 series: the Moirai-Small, -Base and -Large runs on ETTh2 at h=96 and h=192 share the series, the chronological split, the normalisation constants and (at fixed size) the pretrained checkpoint. Resampling cells therefore counts shared facts repeatedly and produces intervals that are too narrow, and a p-value computed from that scheme is not conventional inferential evidence. This is a fair objection to how the paper reported these numbers and it is answerable without new runs.

The resampling unit. We take the cluster to be the (backbone, dataset) pair: the level at which the pretraining corpus, the series, the split and the checkpoint family are shared. Clusters are drawn with replacement and all of a drawn cluster's cells enter together (`scripts/cluster_keys.py`, one definition shared by `cell_matrix.py` and `cka_fixed_effects.py` so the two cannot drift apart). A draw spanning fewer than three distinct predictor values admits no rank correlation and is discarded, which is why the draws kept are reported alongside every interval; on the 31-cell and 22-cell analyses the discard rate is under 0.1%, and it is material only on the four- and five-cell arms,

which carry no information anyway. The effective sample size falls sharply: 31 cells are 15 clusters, the 22 Moirai cells are 6, and the 25 cells with a weight-drift record are 9.

What changes, in full. Of the eleven cross-cell correlations reported in this paper, **exactly one changes verdict** when the interval is taken over clusters instead of cells, and it changes against us. Both levels appear side by side in Table 34 and in the two tables of Appendices A24 and A26; the summary is:

- **Unchanged, and these carry the body's claims.** CKA does not order B−D within Moirai at either level (ρ=+0.168; cells $[-0.328, +0.598]$, clusters $[-0.469, +0.717]$). ℓ_2 weight drift does not order it at either level, pooled (ρ=+0.118) or within Moirai (ρ=+0.318). forg_B still orders it with an interval far from zero (ρ=+0.894; clusters $[+0.594, +0.965]$), which is expected, since that relationship is close to an identity (Appendix A26). The unlicensed pooled CKA correlation still excludes zero, and we still decline to read it.
- **Changed: the gate's anti-predictive interval.** On the retrospective test-side gate, ρ=−0.518 between the gate and B−D moves from $[-0.840, -0.040]$ over cells to $[-0.810, +0.500]$ over the six Moirai clusters. **We withdraw the "excludes zero" reading** and report the direction as a description of these 22 cells. On the primary selection-split gate the correlation is ρ=−0.293 and no level excludes zero, so the withdrawal is not what the split flip rests on. The mechanism (the highest-gate cells are the ETTh2 and ETTm2 cells where adaptation pays best) remains visible cell by cell in Table 5 and does not rest on the interval.

Backbone fixed effects. The cluster bootstrap also lets us report the adjustment the previous version of this appendix declined to fit (Table 37).

Table 37: Within-backbone slope of held-out B−D on CKA, in percentage points per unit of CKA, with backbone identity absorbed as a fixed effect rather than by subsetting. Intervals are 10,000 cluster-bootstrap draws over the 15 (backbone, dataset) clusters of 31 cells; draws in which the fixed effects are unidentified or the design is rank-deficient are discarded, and the kept count is reported. Emitted by `scripts/clustered_inference.py` from `scripts/cka_fixed_effects.py`.

Specification	b_1 (pp per unit CKA)	95% CI over clusters	Draws kept
No backbone term	+47.71	—	—
CKA + backbone FE	+39.08	$[-15.37, +98.50]$ includes 0	9895
CKA + backbone FE + CKA×backbone	+39.91	$[-18.95, +109.71]$ includes 0	8522

Most of the unadjusted slope survives adjustment (+47.7 falls only to +39.1), so it is not the case that absorbing backbone identity destroys the pooled relationship, and we do not claim it. What the intervals say is that 15 clusters cannot pin the surviving slope down. **A wide interval is not evidence of absence.** At this cluster count an interval including zero means the design cannot resolve the sign, which is weaker than "CKA carries no information" and is the only thing we assert.

What this design could have resolved. "The design cannot resolve the sign" is a claim with a size, so we report the size. *Post hoc* — this analysis was specified after seeing the interval it explains, and is not part of the pre-registration — we simulate this exact design and ask for the smallest true Spearman correlation at which *our own* clustered interval excludes zero in 80% of realisations: $\text{MDE}_{80} \gtrsim 0.5$. The observed ρ=+0.168 is less than half of it. Only one decimal is quoted: four master seeds place the figure between 0.5 and 0.6 and do not agree on a second digit, which `--seedcheck` reports rather than leaving to be discovered. So the reading this appendix asserts is **un-usability at this sample size, not absence**: six clusters could not have resolved a correlation of the size CKA appears to have, and a reader who believes CKA's true association with Δ_enc is 0.3 should conclude that we could not have seen it, which is a different sentence from ours and is compatible with it.

And more cells of the same kind would not fix it. On normal scores the intracluster correlation of CKA is 0.05 while that of Δ_enc is 0.74: within Moirai, CKA varies almost entirely *within* cluster (across model size and horizon) while the outcome varies almost entirely *between* clusters, across series. **A predictor with no cluster-level signal cannot order an outcome that is almost all cluster-level**, so the width of the interval is a property of where the two quantities' variance sits

and not only of n — and adding Moirai sizes or horizons on the series we already have would add predictor variance in the direction that carries no outcome signal. This is also the second, independent reason this paper reads CKA within a backbone rather than pooled (§6, Appendix A24).

How the number was obtained, and the three ways it cuts against us. The observed CKA is conditioned on rather than simulated, so the real cluster sizes, spread and tie structure are used; only the outcome's latent is drawn, as $rz + \sqrt{1-r^2}\,(\sqrt{t}\,C_{\text{cluster}} + \sqrt{1-t}\,W_{\text{cell}})$ with z the normal scores of the observed CKA ranks. Because Spearman is rank-invariant **no marginal for Δ_{enc} is assumed at all**, and the assumption reduces to a monotone association with a Gaussian residual carrying an exchangeable cluster effect; r is calibrated to ρ by simulation rather than by the bivariate-normal inverse, which is measurably wrong on a fixed 22-point score vector. Three things then work against our own claim and are reported rather than buried. First, the headline residual cluster fraction is the *estimated* one, and it gives **the smallest MDE of the three settings we ran** at every one of the four master seeds, so the figure we quote is the least favourable point of the sensitivity range for our own claim — which is the only answer we can give to "you tuned the simulation" that is checkable rather than asserted, and the emitter asserts it. Second, every analytic cross-check is *higher*: a Fisher-z minimum detectable correlation with the Bonett–Wright factor gives 0.58 at n=22, 0.60 at the Moulton effective n of 20.1 and 0.93 at the six clusters, and inverting the observed half-width gives 0.59. The simulation is authoritative where they disagree, because it simulates the rule that produced our verdict and they describe a test we never ran — but the direction of the disagreement is a measured defect in our own estimator, and we report it: the six-cluster percentile interval is mildly anti-conservative, rejecting a true zero in 7% of realisations against a nominal 5%, and 10–11% in the two sensitivity settings; these are Monte-Carlo estimates on 300 and 100 realisations and reseeding moves them by about a point, so what they establish is that the interval is mildly anti-conservative and not by how much. **So the figure we quote is the optimistic end**, and a correctly sized rule would put the MDE *higher* and this paragraph's claim on firmer ground rather than weaker. Third, an MDE bounds what a *rank* statistic could have seen, and no rank statistic detects a non-monotone relation at any n; the evidence bearing on that case is the pre-specified out-of-sample ladder of Appendix A28 and the decision pricing of Appendix A28.1, neither of which is rank-based. We do not report observed power at the observed ρ: conditioning power on the estimate it is meant to judge is circular, and the emitter records it under `not_a_claim` for that reason.

Two things this appendix does not do. It does not fit variance components: with 22, 5 and 4 cells in the three backbone groups a mixed model is not identified, and any interval it produced would be a property of the specification. And it does not recompute p-values at the cluster level. A p-value would need a null distribution for a rank correlation among dependent observations at an effective sample size near 6, and the paper describes this analysis as descriptive; printing p=0.0136 beside that sentence was a contradiction, so the p-columns are removed rather than re-derived.

A28 Does CKA Add Predictive Information Beyond Backbone Identity? A Pre-Specified Out-of-Sample Test

A wide interval on a within-backbone slope shows that a design cannot resolve a sign; it does not show that the predictor is uninformative, and the previous appendix says so in those words. That leaves the reader with a genuine question we had not answered: *if you already know which backbone you are on, the horizon and the training-set size, does measuring CKA improve your prediction of the value of encoder adaptation?* This appendix answers the decision-relevant version of that question out of sample.

The analysis was fixed before it was run, and the guarantee is weaker than the prospective arm's. The model ladder, the covariate construction, the deciding statistic and the sentence this paper would print under each of three possible outcomes were all written to `results/preregister_loco.json` and committed before any of the models was fitted. That is a real guarantee against specification search, and it is *not* the guarantee of §7.2: there the outcome runs did not yet exist, whereas here the outcome (held-out B−D) was already computed and already published in Table 5. What is frozen is the *analysis*, not a prediction about unseen data, and the registration file says so on its first page. The registered prediction was $\Delta R^2 \leq 0$.

The ladder, and why dataset identity is absent from it. Table 38 fits five nested models to Δ_{enc} over all 31 scored cells, validated by leaving out one (backbone, dataset) cluster at a time. The review

that prompted this analysis asked for dataset fixed effects *and* leave-one-cluster-out validation; those two requests are jointly unsatisfiable, and satisfying them naively would have produced a table that looks right and means nothing. A cluster *is* a (backbone, dataset) pair, so holding one out removes the only cells that identify that dataset's fixed effect, its column is all-zero in the training fold, and the fold silently scores a model nobody specified. Three of the seven series (Electricity, Electricity-7, ILI) are covered by a single backbone each, so no other fold could identify them even in principle. We therefore run two schemes and label each for what it is. The leave-one-cluster-out ladder excludes dataset fixed effects and is genuinely out-of-sample; it is where ΔR^2 is computed. The variant that *carries* dataset fixed effects is reported as an in-sample fit (R^2 rising from $+0.642$ to $+0.711$, an incremental $+0.069$ from CKA, on 7 series and 31 cells with 12 parameters) and is explicitly not called validation.

The result. The registered prediction fires. Knowing the backbone buys $R^2_{\text{LOCO}}{=}+0.137$; adding horizon and $\log n$ takes it to $+0.201$; adding CKA *lowers* it to $+0.176$. $\Delta R^2{=}-0.025$, the median bootstrap draw is -0.083, and adding CKA fails to help in 69% of cluster-bootstrap draws. Two conventions in that sentence are worth stating. First, out-of-sample R^2 is scored against the **training-fold mean**, never the held-out fold's own mean: against its own mean a fold with little internal spread looks catastrophic and one with a lot looks good, so the aggregate would report fold composition rather than model quality. The intercept-only model scores exactly $+0.000$, which is the check that this convention is implemented as described. Second, mean absolute error is reported beside R^2 because the folds are unbalanced (cluster sizes run from 1 to 5 cells), and MAE moves in the same direction ($+0.59$ pp worse with CKA in the model).

One disclosed deviation from the registration, and it is a dispersion measure, not the deciding statistic. The registration said the cluster bootstrap would be reported on ΔR^2 itself. It cannot usefully be: an out-of-sample R^2 is unbounded below, so a draw that duplicates clusters and leaves a one-cell fold can score an arbitrarily large negative value, and the 2.5th percentile of ΔR^2 came out at -3.3, -1.7 and -32.5 on bootstrap seeds 0, 1 and 7 while the point estimate, the median and the share of non-helping draws moved in the third decimal. Quoting that percentile would have put a number in this paper that a reader re-running the script with another seed cannot reproduce; ΔMAE's upper bound is unstable for the same reason. Both therefore carry a median and an interquartile range, which are seed-stable, and the 95% interval is quoted on the one *linear* coefficient in the analysis, CKA's slope in M3, $+43$ pp per unit CKA with a cluster-bootstrap interval of $[-31, +107]$, whose bounds move by under 5 pp across seeds. The registered *deciding statistic*, the sign of ΔR^2, is unchanged.

What this does and does not establish. It does not establish that CKA is uninformative about fine-tuning in general; it establishes that on these 31 cells, CKA carries no incremental out-of-sample information about the value of encoder adaptation once backbone identity, horizon and training-set size are known, and that the part of Δ_{enc} that *is* predictable is predictable from protocol identity rather than from the representation-similarity measurement. That is a sharper statement than a wide slope interval, and it is the sense in which a post-treatment similarity statistic fails to substitute for running the intervention. It is also worth naming what the ladder shows about the alternative: the interaction model M4 collapses to $+0.019$, barely better than the intercept, so "CKA predicts Δ_{enc} but with a backbone-specific slope" is the worst performing account of these data, not a rescue of the pooled correlation.

A28.1 The Same Cells Priced as a Decision, Post-Hoc

Every result above is a statement about an interval: a within-Moirai slope whose sign is not resolved, a ΔR^2 that is negative but whose dispersion cannot be quoted. A reader may fairly reply that none of that is evidence *against* a relationship. So this subsection asks the question in the units a practitioner would act in, where the claim is falsifiable rather than merely unproven: what would following a CKA rule have been *worth*? **This analysis is post-hoc.** Unlike Table 38 nothing here was registered in advance; the policy family, the objective and the folds were all chosen after the outcomes were known, and the emitter (`scripts/cka_decision_value.py`) says so in its first paragraph.

The construction. A practitioner facing one cell picks full fine-tuning or a frozen encoder, and collects Δ_{enc} on every cell they freeze, so the realised saving of a policy against the default of always adapting is the mean over all 31 cells of $\Delta_{\text{enc}} \cdot \mathbf{1}[\text{freeze}]$, in percentage points of MSE. Every

Table 38: **Does CKA add out-of-sample predictive information about Δ_{enc} beyond backbone identity?** Nested models over all 31 scored cells, validated by leave-one-cluster-out across the 15 (backbone, dataset) clusters. R^2_{LOCO} is scored **against the training-fold mean**, never the held-out fold's own mean, so a positive value means the model beats predicting the other clusters' average; MAE is in percentage points of zero-shot loss. **The deciding statistic and this table's interpretation were fixed before the models were fitted** (`results/preregister_loco.json`): $\Delta R^2 = R^2_{\text{LOCO}}(\text{M3}) - R^2_{\text{LOCO}}(\text{M2}) = -0.025$, median bootstrap draw -0.083, IQR $[-0.227, +0.043]$, and adding CKA fails to help in 69% of draws; $\Delta\text{MAE} = +0.59$ pp, IQR $[-0.37, +2.98]$. **Interquartile ranges, not 95% intervals, are quoted for both**: an out-of-sample R^2 is unbounded below and a degenerate bootstrap fold can blow up either statistic, so their extreme percentiles are not reproducible: ΔR^2's 2.5th percentile moved between -1.7 and -32.5 across bootstrap seeds, and ΔMAE's 97.5th between $+11$ and $+19$, while the point estimates, medians and sign share moved in the third decimal. **The 95% interval is carried instead by the one linear coefficient here**, CKA's slope in M3: $+43$ pp per unit CKA, cluster-bootstrap CI $[-31, +107]$, which covers zero, so the slope's sign is not even determined, let alone its size. That interval is quoted to whole percentage points because its bounds move by under 5 pp across bootstrap seeds. The pre-registered deciding statistic (the *sign* of ΔR^2) is unchanged; only the dispersion measure beside it differs from the registration, which says so. Dataset identity is deliberately *absent* from the ladder: a cluster is a (backbone, dataset) pair, so holding one out removes the only cells that identify that dataset's fixed effect and the fold would silently score a model nobody specified: three of the 7 series are covered by a single backbone, so no other fold could identify them even in principle. The in-sample fit *with* dataset fixed effects is reported in the text and is not validation.

Model	Predictors	R^2_{LOCO}	MAE
M0	intercept only	$+0.000$	20.05
M1	backbone FE	$+0.137$	18.77
M2	backbone FE + horizon + $\log n$	$+0.201$	18.78
M3	M2 + CKA	$+0.176$	19.37
M4	M2 + CKA $\times$ backbone	$+0.019$	20.78

single-threshold CKA rule lies in one family, which puts six policies on one scale (Table 39). **The out-of-sample row is the only one a practitioner could have followed**: the cut *and* its orientation are searched on the 14 training clusters and applied unchanged to the held-out cluster, so every cell's decision is taken without seeing that cell. Candidate cuts are interior, so the search cannot quietly return *always freeze* and have it counted as CKA informing anything; both degenerate policies are priced as rows of their own.

The number that does not favour our claim, first. The best-orientation rule is worth $+7.47$ pp out of sample – not zero, and not what a reader would predict from a headline that drift carries no actionable information. Two things bound it. Its interval is $[-4, +16]$ pp and it is worth nothing or less in 25% of cluster-bootstrap draws, so these cells do not *establish* a saving at all: this is a policy whose value is unresolved, exactly as the slope's sign is unresolved, and the honest reading is the same in both cases. And the orientation that earns it is **the inverse of the one the literature prescribes**: it freezes the cells whose representations moved *least*. Freezing where drift is largest – what forgetting mitigation actually recommends – is worth -6.97 pp $[-10, +14]$, and is worth nothing or less in 80% of draws. A practitioner who followed the prescription on these cells would have lost about seven points of MSE per cell relative to not measuring drift at all.

What the other rows add. Always freezing, ignoring CKA entirely, is worth $+2.91$ pp $[-13, +15]$, so the best-orientation rule's advantage over the cheap policy that reads no representation at all is $+4.56$ pp, nothing or less in 20% of draws. Choosing the cut on the cells it is then scored on pays $+8.57$ pp: the $+1.10$ pp gap $[0, +13]$ is the price of a retrospectively chosen threshold, and it is what §9's "chosen on those same cells, so a lower bound and not a validated classifier" costs, measured. Per-cell hindsight is worth $+10.38$ pp $[+1, +19]$. Out of sample the rule freezes 16 of the 31 cells and puts 8 of them on the wrong side of its own decision, against 5 for the in-sample cut – the same count Figure 1b reports for the best threshold, reached here by maximising saving rather than by minimising misplacement, which are different objectives that happen to agree on this data.

Table 39: **What would acting on CKA have saved?** The same 31 cells read as a decision rather than a correlation. A practitioner choosing per cell between full fine-tuning and a frozen encoder collects Δ_{enc} (positive = freezing better) on every cell they freeze, so *saving* = mean over all 31 cells of $\Delta_{\text{enc}} \cdot \mathbf{1}[\text{policy freezes}]$, in percentage points of MSE, against the default of always adapting. **The out-of-sample row is the only one a practitioner could have followed**: the threshold and its orientation are searched on the 14 training clusters and applied unchanged to the held-out cluster, so each cell's decision is made without seeing it. The row below it is the same search run on all cells at once, and the difference between them, $+1.10$ pp $[0, +13]$, prices the optimism in a retrospectively chosen cut. **The headline is the interval, not the point estimate**: the best-orientation rule's $+7.47$ pp comes with $[-4, +16]$ and is ≤ 0 in 25% of draws, so it is a policy whose value these cells cannot establish, not one they refute. **And the orientation that earns it is the inverse of the prescribed one**: freezing where drift is *largest* – what forgetting mitigation actually recommends – is worth -6.97 pp $[-10, +14]$, and the row above it buys its point estimate by freezing the cells that moved *least*, a direction available only because we searched both. That rule still puts 8 of the 31 cells on the wrong side of its own decision out of sample (16 of them wanted freezing), against 5 for the in-sample cut. **This analysis is post-hoc and descriptive**: unlike Table 38, nothing here was pre-registered, and the family searched is a single global CKA threshold in either orientation – a per-backbone or multivariate rule could do better, and 15 clusters cannot rule that out. Candidate cuts are interior (midpoints between consecutive distinct CKA values), so the search cannot return *always freeze* or *always adapt* and have it counted as CKA informing the decision; both are priced as rows of their own. Intervals are 95% percentile cluster bootstrap (10,000/10,000 draws kept, a draw discarded when it leaves fewer than three distinct clusters), re-running the whole leave-one-cluster-out procedure inside each draw with duplicated clusters sharing one fold id. Every statistic is a mean of bounded per-cell quantities, so unlike ΔR^2 in Table 38 these percentiles are reproducible – **to whole percentage points, which is why they are quoted that way**: at two decimals the bounds move across bootstrap seeds (this table's headline lower bound came out -3.97, -3.61, -3.68 and -3.67 on seeds 0, 1, 7 and 13), and `--seedcheck` checks every figure quoted here at its printed precision. The point estimates and the misplacement counts carry no seed at all.

Policy	Saving (pp)	95% cluster CI
Always adapt (fine-tune everything) – *the baseline*	0.00	*by construction*
Always freeze the encoder, ignoring CKA	$+2.91$	$[-13, +15]$
CKA rule *as prescribed*: freeze where drift is largest	-6.97	$[-10, +14]$
CKA threshold, best orientation, chosen out of sample	$+7.47$	$[-4, +16]$
the same rule, threshold chosen on the cells it is scored on	$+8.57$	$[0, +18]$
Oracle: freeze exactly the cells that wanted it	$+10.38$	$[+1, +19]$

What we take from it, and what we do not. We do not claim this refutes a CKA rule: a point estimate of $+7.47$ pp is not a refutation, and one policy family on 15 clusters cannot speak for a per-backbone cut, a multivariate rule or a continuous-shrinkage policy. What it supports is narrower and is the claim this paper makes: on these cells the evidence does not establish a saving stable enough to act on, and the direction in which acting would have paid is the opposite of the direction the mitigation literature points. Both readings survive the reviewer's objection to the null, because neither of them is the assertion that no relationship exists.

A29 Audit Trail and Withdrawn Analyses

Every reading this work has withdrawn is recorded here rather than in the body, which carries one pointer to this appendix. The move drops nothing: the lettered items below index every withdrawal, and the three that need their own account (the gate's denominator, ILI, and the probe axis) are the subsections that follow.

A29.1 What changed from the earlier version

(a) The conclusion changed, not the emphasis. An earlier version of this work reported eight degradation cells, all on ETTh1 or Weather at $h \in \{96, 192\}$. All eight fail clause (i) once the gate is

computed against the linear regression the gate is *defined* as; none of the eight involved re-running a model (Appendix A29.2).

(b) The non-Moirai gates. An earlier version reported "eight of nine non-Moirai cells gate-pass" at $h{=}24$; against the fitted baseline **none of the ten does**, and the earlier count was an artefact of the trend denominator. What the two extra backbones still supply is the dissociation, which does not depend on the gate.

(c) The prospective arm's scorecard. The earlier version of that arm read TP 2, FP 6, precision 0.25, recall 1.00, and we defended the gate as a screen that never missed a degradation cell. With the corrected baseline there are no degradation cells to miss, so the screen has nothing to be right about.

(d) ILI. An earlier public version reported Moirai-Small on ILI as a 57% gate-*pass*; that number is withdrawn, the cell gate-fails at $V_{\text{ridge}}{=}-1.625$, and a separate script defect reported *zero* frozen-encoder improvement in all 10 seeds; corrected, $\Delta_{\text{enc}}{=}-38.6{\pm}1.3$ (§A29.3).

(e) The linear-probe axis. A ΔR^2 probe account of *what* restructured is retracted in full (§A29.4), with the withdrawn numbers and the re-run that refutes them kept in Appendices A12, A18 and A15; the layer-wise *task* readings survive, being forecast MSE, and are in the body.

(f) The cross-backbone dataset ordering. An earlier version observed that Chronos-T5 and TimesFM-2.5 rank the shared datasets identically and read that as evidence that the ranking is a property of the benchmarks rather than of either model. Scoring the gate on the selection windows removes it: the two backbones rank the datasets identically on the held-out windows ($\rho{=}{+}1.00$) and disagree on the selection windows ($\rho{=}{-}0.40$ over the four shared datasets), so the agreement is a property of one split. We withdraw the claim. Both readings rest on nine cells spanning only -0.275 to $+0.082$, which is the more basic reason not to order them: the band is narrower than the split-to-split movement within it. What survives on both splits is the *contrast* the body reports (ETTh2 is where Moirai clears the gate at every capacity and where Chronos sits below the fitted baseline), because that is a sign, not a rank.

(g) A registered denominator was briefly recomputed from outcome runs. The zero-shot MSE in a gate's numerator is a measurement, not a constant (Appendix A20.5), and condition-B/D records carry their own re-measurement of it. Until this revision the glob that collected the pre-registered arms' denominators matched every condition rather than `condition_A`, so those denominators were means over the outcome runs instead of the single reference each registration had frozen, a predictor moved after its outcomes existed, which is the one thing a prospective design must not permit. The roots are now read `condition_A` only, with a structural assertion that fails the run if an outcome file is ever globbed as a registered reference. We record it although it changed *no* published number: the shift was at most 0.0044 in R^2_{task} and would have changed 0 of the 26 admissions, and being unnoticeable downstream is exactly what makes it worth writing down rather than what excuses it.

A29.2 The gate's linear baseline was measured wrongly

Two estimators, one name. The value gate is $V_{\text{Linear}} = 1 - \text{MSE}_{\text{ZS}}/\text{MSE}_{\text{Linear}}$ with threshold 0.20, and §2 defines $\text{MSE}_{\text{Linear}}$ as a lookback-96 linear regression *fit on that same set*: a single ridge-OLS map from $96{\times}D$ inputs to $h{\times}D$ outputs, estimated once on the cell's training windows and applied unchanged out of sample. That estimator is `gate_linear_baseline.fit_linear_map` (applied by `apply_linear_map`). The gate was not computed with it. Through 26 Aug 2026 the denominator came from `gate_linear_baseline.linear_forecast`, which fits a slope and intercept to the last 96 steps *inside each evaluation window* and extrapolates them forward, using no training data at all. These are two different estimators sharing one label, not two implementations of one estimator, and the paper defines the first while reporting the second.

The estimator the gate used is weaker than a constant. Table 40 measures four denominators on the same 300 matched test windows on the train-normalised scale the stored zero-shot MSE is measured on. The load-bearing row is the third: on ETTh1 $h{=}96$, simply predicting the training mean scores 0.9937 against the trend extrapolation's 1.2207. A baseline that a constant predictor beats cannot support the inference the gate exists to license, namely that *there is a valuable pre-trained capability here worth preserving*, which is the sole premise the degradation reading rests on. The defined estimator is stronger by a factor of about three on ETTh1 and 9.4 on Weather. Two independent

checks say the correction runs in the right direction: the fitted ETTh1 $h{=}96$ MSE of 0.4145 is in line with published multivariate Linear/DLinear results Zeng et al. (2023), and Moirai zero-shot losing to a supervised linear model on ETTh1 is a known result, whereas 1.2207 is not a physically plausible score for any linear baseline on this cell. Note also an asymmetry the definition imposes and which we retain: Moirai sees lookback $+\ h$ steps of context while the linear baseline sees 96, so the corrected gate is still a lower bound on the linear model's strength.

Table 40: Baseline strength: train-normalised MSE on 300 matched test windows. Row 1 is the denominator the gate used through 26 Aug 2026; row 4 is the denominator §2 defines. Row 1 is beaten by predicting the training mean (row 3) on *both* cells and is several times weaker than row 4. On the train-normalised scale the constant-mean predictor's MSE is just the mean squared normalised target, so row 3 needs no fitting; a value near 1.0 is what an on-distribution test split should give.

Denominator estimator	ETTh1 $h{=}96$	Weather $h{=}192$
Per-window trend extrapolation (`linear_forecast`)	1.2207	5.0707
Persistence (last observed value held)	1.1984	1.2227
Constant training mean	0.9937	0.8619
Fitted lookback-96 regression (`fit_linear_map`)	**0.4145**	**0.5403**

What it costs, cell by cell. Re-running all four arms with `gate_all_cells.py -baseline fitted` (now the default) moves most of the matrix:

- **Moirai: 5 of 21** cells clear 0.20 on the test split, against 20 of 21 before. Seventeen cells change status: 16 flip pass→fail, and one (Moirai-Large/ETTh2 $h{=}96$) flips fail→pass at $+0.219$.
- **Chronos: 0 of 5**, against 4 of 5 before: ETTh1 -0.123, ETTh2 -0.325, Weather -0.103, ETTm2 -1.281, Electricity $+0.048$.
- **TimesFM: 0 of 5**, against 5 of 5 before: ETTh1 -0.101, ETTh2 -0.287, Weather -0.077, ETTm2 -0.945, Electricity $+0.132$.
- **ILI** (Moirai-Small): $\mathbf{-1.625}$. Appendix A29.3 previously reported this cell's test gate as -0.033, which is its trend-baseline value, and before that a gate-pass of $+57\%$ was withdrawn there; it now carries -1.625. The reading of that appendix is unchanged and strengthened: ILI gate-fails, now by a wide margin.

Across arms, **7 of 31 cells pass on the selection split (the paper's primary), and 5 of 32 on the retrospective test-side variant**. Every survivor under either is Moirai; the selection split adds two ETTm2 cells to the five ETTh2 ones, and no other dataset and no other backbone retains a single gate-pass cell under either. Per-cell corrected Moirai values are in Table 3, and the survivors are given in full in Table 41. Four of the seven survivors *improve* on the released checkpoint under full fine-tuning *in every seed*, three of them by 31–42% and the ETTm2/Base cell by 8.4%, and the two whose mean is harmful have only 4/10 and 5/10 seeds in that direction, so their means are outlier-driven rather than consistent (the Moirai-Small $h{=}96$ cell also has a positive mean condition-D forgetting of $+6.91$ against 9/10 negative seeds, for the same reason). The seventh, Moirai-Small/ETTm2 $h{=}192$, is the mirror image and is why we count unanimity rather than means: its mean is a 1.10% *improvement* but 3 of its 5 seeds are harmful, so on the mean-only reading five of the seven improve and on the every-seed reading four do. We report four throughout, because a cell whose sign depends on which seeds were drawn is not one where "fine-tuning improves the checkpoint" is a safe statement in either direction.

The $h{=}96$ decomposition, since its mean is one seed. Seed 42 of Moirai-Small/ETTh2 $h{=}96$ diverges to $+98.9\%$ under full fine-tuning *and* to $+91.1\%$ under the frozen encoder, which is why §5.1 calls it an optimisation failure rather than an encoder-drift effect; Appendix A8 adds the independent check that it cannot diverge at all under the strict freeze. Dropping it leaves $\text{forg.}_{\text{B}}{=}-0.20{\pm}4.15$ over 9 seeds (3 positive) and $\text{forg.}_{\text{D}}{=}-2.45{\pm}0.16$, so the cell's harmful mean does not survive the removal of a single run. **We nonetheless report the cell means including seed 42 everywhere in this paper**, and record the decomposition here rather than dropping the seed, because a rule that discards runs after seeing their outcome is not one we are willing to apply.

Table 41: The five cells that still clear $V_{\text{ridge}} \geq 0.20$ on held-out windows under the fitted baseline. Forg.% is percent change against zero-shot, so a *negative* entry means fine-tuning improved on the released checkpoint. The bracketed counts are the seeds whose sign matches the degradation direction: positive for B, negative for D. B−D positive means the frozen encoder was better.

Cell	Gate	B Forg.% (+)	D Forg.% (−)	B−D val	B−D held-out
Small/ETTh2 h=192, n=500	+0.464	+1.99 (5/10)	−4.16 (10/10)	+17.4	+6.1
Small/ETTh2 h=96, n=500	+0.400	+9.71 (4/10)	+6.91 (9/10)	+10.6	+2.8
Base/ETTh2 h=96, n=1000	+0.361	−31.44 (0/3)	−9.11 (3/3)	+3.8	−22.3
Base/ETTh2 h=192, n=1000	+0.275	−36.35 (0/3)	−16.12 (3/3)	−3.9	−20.2
Large/ETTh2 h=96, n=500	+0.219	−42.42 (0/3)	−21.62 (3/3)	+7.2	−20.8

No cell now meets the degradation definition. The definition is unchanged (gate ≥ 0.20, *and* condition-B forgetting positive in every seed, *and* condition-D forgetting negative in every seed), and only the gate input moved. With the corrected denominator, `degradation_cells()` returns zero cells. There is one candidate, and the pre-existing unanimity rule excludes it:

```
excluded:  Moirai-small/ETTh2 h192 n500   gate=+0.464
   forg_B=+1.99±3.32SEM (5/10 pos)
   forg_D=−4.16±0.20SEM (10/10 neg)   B-D=+6.15
=> 1 excluded for seed disagreement
```

Half its seeds go the other way on condition B, so it fails the same all-seeds test that the earlier degradation cells were required to pass. We did not weaken or re-tune that rule to keep a cell; applying it as written leaves nothing, and Appendix A21 shows that weakening it, in seven directions and at both gate thresholds, does not recover one either.

Prospective arm and threshold sweep. With no positive instances, the pre-registered prospective test has nothing for the gate to find: on the eight pre-registered cells the gate scores **TP 0, FP 8, precision 0.00**, and recall is undefined rather than 1.00. The earlier reading of this arm (precision 0.25, TP 2, FP 6, recall 1.00) does not survive the correction. The threshold sweep of §7.2 (Table 10) reports precision 0.00 at *every* threshold in $\{0.10, 0.20, 0.30, 0.40, 0.50, 0.60\}$, with $8/8/6/5/3/2$ cells flagged respectively: the failure is not a badly chosen operating point.

A further consequence is newly detectable. Within Moirai (n=22 cells), the gate correlates with held-out B−D at Spearman $\rho = -0.518$: on the evidence in Table 5 the gate points *away* from the intervention outcome it was introduced to predict, in that higher gate goes with adapting the encoder having been the better choice. The interval around that point estimate excludes zero when cells are resampled, $[-0.840, -0.040]$, and includes zero when the six (backbone, dataset) clusters are, $[-0.810, +0.500]$; we report the latter and read only the direction (Appendix A27). The CKA comparison is unchanged by all of this, being gate-independent: $\rho = +0.168$, clustered CI $[-0.469, +0.717]$ within Moirai, and $\rho = +0.567$, clustered CI $[+0.239, +0.828]$ pooled across backbones, which this paper's own within-backbone rule says not to read.

Provenance. The substitution happened between 24 Jul and 26 Aug 2026 and is recorded nowhere: no commit message, no docstring, no code comment marks it, and the paper text was never updated to match. That is also why the paper carried both signs of the same quantity at once. The gate values previously printed in Table 3 (-0.406 on ETTh1, -0.556 on ETTm2, $+0.279/+0.450$ on ETTh2) were read from `results/gate_pass_etth1/metrics.json` and `results/gate_pass_ettm2/metrics.json`, which were produced by the older *fitted* `LinearForecaster` in `scripts/evaluate_moirai_zeroshot_forecasting.py`, while the gate-pass counts quoted around them came from the trend estimator. Nothing was suppressed to make the two agree; they simply were never compared.

The correction is auditable rather than a claim to be taken on trust. `gate_all_cells.py` takes `-baseline {fitted,trend}` with `fitted` as the default, threaded through all four arms; `linear_forecast` is retained and documented as audit-trail-only; and the trend-baseline numbers are preserved verbatim in `results/gate_test_side_trend.json` and `results/gate_val_side_trend.json`, so every superseded figure in this paper's history can be recomputed and compared against its replacement.

One earlier fix should not be confused with this one. The gate's evaluation windows are now built with `ext_lb` = lookback + h, matching the extended lookback the fine-tuning script evaluates with, rather than lookback $\times$ 2; the two coincide at h=96 and diverge at h=192, where the older setting paired the stored zero-shot MSE with a linear baseline measured over a *different* set of target windows. That correction changed no cell's gate status. This one changes 17 of the 21 Moirai cells and removes every degradation instance the paper had.

A29.3 ILI: GATE CHECK AND WITHDRAWN FROZEN CONTROL

ILI: a small-data external case study. Moirai-Small on ILI (weekly influenza, h=24, 10 seeds) restructures heavily and improves, yet **gate-fails decisively** at $V_{\text{ridge}}=-1.625$; at $\approx$450 windows it is a case study rather than a pillar, and its pre-registered CKA prediction was **falsified**. An IoT anomaly-detection negative control behaves the same way (Appendix A14).

The cell is Moirai-Small on national influenza-like illness: 966 weekly observations, 7 features, h=24, lookback=104, split 579/193/194 chronologically.

ILI gate-fails on held-out windows, by a wide margin. Measured against the fitted linear baseline the gate is defined with (Appendix A29.2), and computed from the runs this paper reports (10 seeds), the test-segment gate is $\mathbf{-1.625}$; that is the value carried in Table 5. **Two earlier figures for this cell are withdrawn**: a gate-pass of $+57\%$, and the trend-baseline pair $+0.139$ on validation and -0.033 on test, both of which came from the per-window trend estimator rather than the defined one and are superseded. The per-feature decomposition of that trend value (five of seven features positive but only two clearing 0.20, and NUM. OF PROVIDERS at -0.86 because a near-straight test series is fitted almost exactly by extrapolating its own trend) explained the superseded number and is dropped with it. Against the fitted baseline the validation split gives -10.206, but that mean is not the quantity to read: it is dominated by NUM. OF PROVIDERS, whose near-straight validation segment the ridge map fits to MSE 0.034 and which therefore alone contributes -44.0. The median per-feature reading is -3.40 on validation and -1.51 on test, and **0 of 7 features clear** 0.20 **on either split**, so the verdict is robust even where the mean is not. ILI is therefore not a degradation cell and no claim in §6 rests on its gate; what changes is that it can no longer be cited as a moderate-gate cell, which makes the falsified prediction below correspondingly weaker evidence. A heavily restructuring encoder on a cell with *no* demonstrated zero-shot advantage is much less surprising than one on a moderate-gate cell.

A withdrawn frozen control. On ILI the encoder restructures heavily (CKA 0.456 ± 0.008) and full fine-tuning improves by $-80.0\pm0.6\%$ on validation and $-50.77\pm0.49\%$ held out (10/10 negative on both, 0/10 positive). We previously reported B$-$D$=-80.7$ pp here, that is, that encoder adaptation supplied *all* of the gain. **That is withdrawn.** The frozen control it was measured against had never been trained: the script assigned the zero-shot MSE as the fine-tuned MSE and wrote a forgetting of exactly 0.0, so the "0.0% in all 10 seeds" was one constant, not ten measurements. Re-run so that condition D actually trains (encoder frozen, the 3.2M parameters outside it updated), the gap is -62.6 ± 1.4 on validation and -38.6 ± 1.3 held out (10/10 negative on both): the frozen encoder recovers a large share of the gain rather than none of it, improving by $-12.18\pm1.28\%$ held out on its own (10/10 negative). D reaches CKA 0.76–0.90, not 1.000, because the projections feeding the encoder remain trainable (Appendix A8). Condition B is essentially unaffected: it gives -80.0 ± 0.6 against the previously published -80.7 ± 0.3, and CKA 0.456 ± 0.008 against the published 0.478. The forgetting figures agree within about one standard error; the CKA means differ by 2.8 standard errors, though 0.478 sits inside the per-seed range we measure (0.409–0.505), so we describe the two B arms as consistent in outcome rather than identical. Either way the entire B$-$D discrepancy was the untrained control, not condition B. Table 42 gives the per-seed values behind both conditions; the point of printing all ten is that the withdrawn reading was a constant where these are ten distinct measurements.

Interpretation. The pre-registered prediction had two components, and neither now supports the hypothesis. The CKA component (0.90–0.95, intermediate behaviour) is **falsified**: CKA is in the restructuring regime (0.456), not intermediate. The probe-asymmetry component ($\geq$7/10 seeds $\Delta R^2>0$) was reported as confirmed, but rests on the statistic we retract (§A29.4) and is therefore **withdrawn** rather than confirmed. The prediction as a whole is falsified, not partially so. What it establishes is narrow: gate strength does not linearly predict drift magnitude. We can no longer read a restructuring-to-stability transition off this cell at all, because on held-out windows ILI does not

Table 42: ILI per-seed results (Moirai-Small, h=24, lookback=104), both conditions, regenerated for this revision. The ΔR^2 column of the earlier version of this table is dropped with the retracted probe axis; condition D is shown because it previously did not exist as a measurement.

	B (full fine-tuning)		D (frozen encoder)	
Seed	CKA	Forg.% (val / held-out)	CKA	Forg.% (val / held-out)
42	0.458	−78.6 / −51.5	0.804	−16.2 / −14.6
101	0.465	−81.4 / −50.1	0.898	−13.5 / −6.7
123	0.476	−80.0 / −48.9	0.756	−27.6 / −19.5
202	0.457	−81.4 / −52.1	0.791	−19.7 / −13.6
303	0.409	−78.1 / −49.0	0.823	−18.3 / −11.5
456	0.463	−75.5 / −50.0	0.851	−15.4 / −13.2
777	0.437	−81.6 / −52.5	0.822	−18.0 / −12.4
789	0.449	−80.3 / −49.5	0.865	−12.7 / −4.9
888	0.505	−82.0 / −50.8	0.848	−16.1 / −12.8
999	0.437	−80.8 / −53.4	0.807	−16.1 / −12.7
Mean	0.456	−80.0 / −50.8	0.827	−17.4 / −12.2

clear the gate. ILI's low CKA reflects the small dataset (966 timesteps, ≈450 training windows per feature) which permits aggressive restructuring within 20 epochs.

A29.4 Retraction: the probe-sign asymmetry axis

An earlier version of this work reported a second diagnostic axis alongside CKA drift: a *probe-sign asymmetry*, in which fine-tuning raised linear decodability of the trained objective ($\Delta R^2>0$) while leaving task-orthogonal probes flat or lower. We reported it as replicating across every gate-passing cell. **We retract it.** It does not survive a protocol with every choice fixed in advance, and nothing in this paper's remaining claims depends on it.

What fails. The statistic reproduces numerically: the headline $\Delta R^2{=}+0.67{\pm}0.24$ on Moirai-Small/ETTh2 at n=10k is recoverable to four significant figures, but only in a regime where the probe does not generalise. Under a temporal split the pre-trained probe scores $R^2(\text{PT})$ between -2.7 and -164.1: it is not decoding the target at all, so ΔR^2 measures relative movement between two failed fits. Re-running the identical probes, encoders and windows under a randomised split, where all seven probes generalise ($R^2(\text{PT}) \in [0.678, 0.994]$), no $|\Delta R^2|$ exceeds 0.010.

Split	R^2(PT) range	$\max\lvert\Delta R^2\rvert$
Temporal (as originally reported)	-164.1 to -2.7	21.4
Randomised, same windows	$+0.678$ to $+0.994$	**0.010**

The effect is therefore an artifact of probe failure, not evidence about the encoder. It also fails our own published criterion for discarding a probe: we stated that a probe with negative R^2 should not be read as evidence, and then read one.

What this costs, and what it does not. Every claim resting on probe asymmetry is withdrawn: the "dissociation" bullet in §1, criterion (i) of §2, and the per-cell ΔR^2 readings in §6 and the appendix tables, which we retain only as a record of the retracted computation. Table 4 carried three such columns. They no longer appear there, and we transcribe them here instead, so that the withdrawal is legible against what was printed rather than quietly absorbed into a regenerated table. Unlike every other table in this paper these values are **not** re-derivable by `scripts/rederive_all.sh`: the probe records behind them are scattered across result directories at one to three seeds per n, and we are not going to write an emitter whose output is numbers we are telling the reader not to use. They are a transcript, they carry no dispersion where k=1, and they support no claim.

Retracted. Transcribed from the withdrawn version of Table 4; not re-derived.

Samples	R^2(FT)	ΔR^2 (Ridge)	ΔR^2 (MLP)
500	-6.79 (k=2)	$+.12$ (k=2)	$-.26$ (k=1)
1,000	-6.77 (k=1)	$+.13$ (k=1)	$+.30$ (k=1)
2,000	-6.51 (k=3)	$+.39_{\pm.35}$	$+.73_{\pm.75}$
5,000	-6.41 (k=7)	$+.50_{\pm.26}$	n/a
10,000	n/a	$+.67_{\pm.24}$	n/a

The zero-shot floor is R^2_{ZS}=-6.90, so R^2(FT) is negative in every row: the probe never decodes the target, which is precisely the regime this retraction is about, and the ΔR^2 column is then movement between two failed fits. The MSE, forgetting and CKA columns that remain in Table 4 are unaffected and are the only ones to read.

The paper's other findings share no machinery with the probes: the value gate is a downstream MSE comparison, CKA drift is computed from encoder representations directly, and the frozen-encoder comparison is read in forecast MSE. The value gate was nonetheless found to have a defect of its own, independently of this retraction and by a different route (its linear denominator was not the estimator the gate is defined as), and that correction is documented separately in Appendix A29.2. The two are unrelated: neither the probe re-analysis nor the gate re-computation used any result from the other.

Why we report this. A negative result about one's own tooling is cheap to bury and expensive to discover downstream. The re-analysis code and both split protocols are released with the paper.

A30 The Pre-Registered Positive Control, in Full

What was registered, and when. `results/positive_control/preregistration.json` was committed at a named revision before any run of this arm existed, and it fixed: task A, chosen from the run records as $\arg\max V_{\text{ridge}}$ over the seven gate-passing cells (Moirai-Small/ETTh2, h=192, V_{ridge}=$+0.932$); task B by SHA-256, with its generator (`scripts/make_conflicting_series.py`); the grid; the primary readout; *which checkpoint is primary*; *where CKA is measured*; two falsifiable validity conditions; and all three outcome sentences.

Task B, and why its construction matters. $x_j(t) = a_j\,(-1)^{\lfloor t/P \rfloor}\, s_j(t \bmod P) + \sigma_j \varepsilon$, with s_j a three-harmonic unit-variance periodic shape and P=192=h. The correct forecast is therefore the *negation* of the previous period, so the seasonal-naive prior (which is task A's own strongest admissible baseline rung) is maximally wrong on task B: seasonal-naive loses to its own negation by $200.9\times$. This is what makes learning B a genuine conflict with A rather than the addition of an unrelated skill, and it is the reason a null here is informative.

Where CKA is measured, and why it differs from the main matrix. On **task A's** inputs, not task B's. In the main matrix the two coincide, because a cell trains and is evaluated on one series; here they cannot, and measuring drift on the series being trained *on* would answer a different question than "did the representation supporting A change". This was the design choice we expected to be probed, so it is registered and stated rather than buried.

Both checkpoints, and the noise floor that makes them readable. The early-stopped state is primary, as registered; the final-epoch state is reported beside it as the maximal-destruction bound, because the flag selects on task B's validation loss and so can return a checkpoint from before damage to A has finished developing. **The reason the registration gave for that choice (comparability with the main matrix, "which early-stops the same way") was wrong**: the matrix passes no `--early-stopping` flag and is read at its final epoch (§3). We leave the registered choice and the registration's wording as they are and report both states, so the error is in a stated rationale and not in a result; the arm's own runs do restore the best selection-split epoch, which is what the primary column is. That epoch was often the last one, so the two states are frequently *identical*, and then the gap between their two retention numbers is not redundancy but a direct read of the evaluator's own sampling noise, since Moirai's forecast is the median of 20 samples. Over the 17 identical-state pairs that gap is at most 0.48 pp and 0.20 pp on average. **Every retention number in this arm should be read against that floor**, which is why the emitter computes it across all such pairs rather than asserting a tolerance.

The validity conditions, both of which hold. (i) Task B must actually be learned; destruction without learning is weight thrashing, not forgetting. It was, in **21 of 21** runs, by at least $+56.6\%$ against a registered 20% threshold. (ii) The frozen encoder should preserve task A; if it does not, the damage is not in the encoder. It does: retention stays inside $+5\%$ at all three completed rungs ($+1.3 \pm 0.4$, $+4.2 \pm 2.2$, $+3.4 \pm 9.4$), against a destruction threshold of $+50\%$.

The outcome, quoted from the registration. Destruction was not achieved: the registered rule was retention$_A \geq 50\%$ in a majority of seeds at some rung under condition B, and no rung meets it: the per-rung maxima are -7.1, $+26.7$, $+4.9$ and -20.1. The registered sentence for this outcome reads: "the positive control failed to construct… deliberately adversarial fine-tuning did not destroy the capability – itself a finding about TSFM robustness, and a reason the main matrix's null is unsurprising." Across all 21 runs Spearman ρ between CKA on task A's inputs and retention is $+0.466$ (p=0.03, n=21): a positive coefficient is the wrong sign for a preservation heuristic, so the ordering fails here as it does retrospectively.

The declared extension, and the three runs that diverged. The stopping rule permitted one extension to lr=10^{-1} if destruction was not achieved at 10^{-2}, and then a stop. Condition B completed all three seeds and supplies the ladder's sharpest row. Condition D diverged in 3 of 3: with the encoder frozen every gradient lands on the decoder, and its likelihood head leaves the domain its distribution requires: two seeds on the negative binomial's `total_count`, one on a categorical's `logits`. **We do not re-run them.** Guarding the evaluation call and retrying would be re-running the declared extension after seeing its outcome, which is the thing a pre-registration exists to prevent; the divergence is reported as the result for that rung. It is also not load-bearing: condition D's role is to localise damage in the encoder, and there is no damage anywhere in the ladder to localise.

What this arm cannot show, as registered: that the capability at stake was valuable against a strong baseline (task A beats seasonal-naive by only $+0.081$, and does not clear the full admissible ladder, so this tests *detection* of a retention loss rather than the value of what is lost); that anything here generalises beyond one backbone, one series and one synthetic task B; or that a sign-inverting target task resembles a deployment scenario. It also cannot show that a retention loss is impossible on these benchmarks, only that we could not construct one, having tried four learning-rate rungs spanning three orders of magnitude with a task built to conflict.

Table 43: **The pre-registered positive control: 21 runs, no destruction.** Task A is Moirai-Small/ETTh2 h=192; task B is the anti-persistent synthetic series, fixed by SHA-256 before any run. "task B learned" is the MSE reduction on B's own held-out windows (registered validity threshold 20%). "retention on task A" is $100(\text{MSE}_A(\theta_{\text{FT}}) - \text{MSE}_A(\theta_{\text{ZS}}))/\text{MSE}_A(\theta_{\text{ZS}})$, so *positive is worse* and the registered destruction threshold is $+50\%$ in a majority of seeds under full fine-tuning. CKA is measured on **task A's** inputs. $\pm$ is the SEM over 3 seeds. † marks the single declared ladder extension. Read every retention number against the 0.48 pp evaluator noise floor derived in the text.

Diverged, not omitted: 3 run(s) at lr $= 0.1$, frozen encoder did not complete; training diverged and the run is reported as a divergence rather than a number (see text).

		task B	retention on task A (%)		CKA on	ℓ_2
lr	encoder	learned (%)	early-stopped	final epoch	task A	drift
0.0001	full fine-tune	$+99.0_{\pm 0.0}$	$-10.8_{\pm 1.9}$	$-10.7_{\pm 1.8}$	$0.796_{\pm 0.034}$	$7.63_{\pm 0.02}$
0.0001	frozen encoder	$+57.3_{\pm 0.6}$	$+1.3_{\pm 0.4}$	$+1.2_{\pm 0.3}$	$0.999_{\pm 0.000}$	$7.19_{\pm 0.06}$
0.001	full fine-tune	$+99.3_{\pm 0.0}$	$+8.6_{\pm 13.4}$	$+8.8_{\pm 13.6}$	$0.234_{\pm 0.032}$	$42.37_{\pm 0.85}$
0.001	frozen encoder	$+79.6_{\pm 0.7}$	$+4.2_{\pm 2.2}$	$+4.3_{\pm 2.4}$	$0.975_{\pm 0.003}$	$25.87_{\pm 0.49}$
0.01	full fine-tune	$+95.8_{\pm 1.2}$	$-13.4_{\pm 9.7}$	$-13.2_{\pm 9.6}$	$0.279_{\pm 0.101}$	$456.22_{\pm 16.10}$
0.01	frozen encoder	$+85.3_{\pm 1.0}$	$+3.4_{\pm 9.4}$	$+3.7_{\pm 9.3}$	$0.900_{\pm 0.055}$	$78.09_{\pm 3.38}$
$0.1^\dagger$	full fine-tune	$+84.1_{\pm 1.9}$	$-23.3_{\pm 2.0}$	$-23.3_{\pm 2.0}$	$0.063_{\pm 0.016}$	$2794.83_{\pm 40.24}$