CONSENSUS FORGETS FIRST: ONLINE ADAPTATION OF FROZEN PREDICTORS

Anonymous authors
Paper under double-blind review

ABSTRACT

When a pool of frozen models predicts the same target under distribution shift, the right adaptation memory is unknown and itself non-stationary. The Gram matrix of the pool's predictions, however, has a spectrum known before deployment, which acts as a prior on memory. With one covariance-inflation scalar, the dominant direction, the consensus level, carries the shortest memory while contrasts between models keep long memory. We take the ratios between memory lengths from the spectrum and hedge over their common scale by aggregating recursive least-squares experts online. After standardizing by the training scale, the layer has no tuned parameter. We prove deterministic oracle inequalities: safety against the best raw predictor, and path-length tracking against the best time-varying combination. On French electricity load (2019–2023) and NYC taxi ridership, the layer improves on every frozen base and on aggregation of the raw pool in every regime, and on validation-tuned adaptive filters overall. Without the inflation scalar, the same hedge loses to a validation-tuned Kalman filter.

1 INTRODUCTION

Operational forecasters can fail abruptly when the data-generating process shifts faster than their update mechanism. The COVID-19 lockdown sharply disrupted French national electricity load within days. Models calibrated on years of pre-pandemic data incurred large short-run errors, while retraining required post-shift labels not yet available and could not correct immediately. This pattern recurs across deployed forecasting settings, including short-term energy load, retail demand, and user behaviour. The appropriate adaptation memory is both unknown in advance and itself non-stationary: a horizon well tuned to one regime can be badly miscalibrated for the next.

We consider the common deployment setting in which several models, trained offline on years of data and then frozen, predict the same scalar target (Chen & Guestrin, 2016; Ke et al., 2017; Hastie & Tibshirani, 1986; Gorishniy et al., 2021; Hollmann et al., 2023; Qu et al., 2025), and an online layer on top of them is the only part that moves. The layer has to decide how much of the past to remember. The forgetting factor of a recursive filter (Hayes, 1996) or the loss window of a forecast combination (Bates & Granger, 1969; Cesa-Bianchi & Lugosi, 2006; Gaillard et al., 2014) sets one speed for the whole coefficient vector. Does the pool itself tell us which directions must forget fast and which can remember?

It does. The pool's training Gram matrix has a spectrum that is fixed before deployment, and its dominant direction is the consensus level; a single covariance-inflation scalar turns this spectrum into ordered memory lengths. The consensus re-calibrates within days of a break, while the contrasts between models keep long memory. No estimate of the shift enters this allocation (Section 3).

This structure motivates MELO (**M**emory-hedged **E**xponentially weighted **L**east-squares **O**nline aggregation), a causal, tuning-free adaptation layer that wraps any frozen base-predictor pool (Section 3). It requires no regime labels, change-point annotations, or base-model retraining.

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.

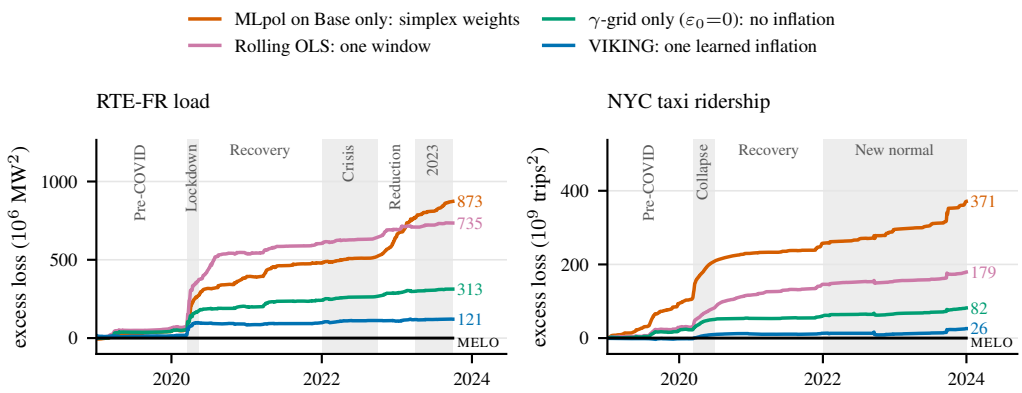


Figure 1: **The gains accrue at the breaks and never revert.** Cumulative excess squared loss of each reference over MELO, the zero line; shading alternates between regimes. The legend states how each reference adapts, and MELO hedges over the inflation; the final values follow the legend order on both streams. A step marks a break that the reference absorbs more slowly, a flat stretch a regime where the two are on par, and a falling stretch a regime the reference wins; the only one is VIKING before COVID on the taxi stream.

Findings.

- The spectral prior of a prediction pool turns a single scalar into ordered, tuning-free memory lengths; without the scalar the pool loses to a validation-tuned Kalman filter on both streams.
- We prove deterministic oracle inequalities for the inflated adaptation recursion: safety against the best raw predictor, and path-length tracking against the best time-varying affine combination.
- On French electricity load 2019–2023 and NYC yellow-taxi ridership, with no regime information and no parameter selected on validation data, the layer improves on every frozen base and on aggregation of the raw pool in every regime, and on validation-tuned adaptive filters overall.

2 Related work

Forecast combination and online aggregation. Classical forecast combination averages heterogeneous forecasters with fixed or slowly-adapted weights (Bates & Granger, 1969; Timmermann, 2006); stacking learns combination coefficients by cross-validated regression (Wolpert, 1992; Breiman, 1996). Both are offline. Prediction with expert advice (Cesa-Bianchi & Lugosi, 2006) gives an online-learning alternative with regret guarantees, from the best fixed expert to shifting experts (Herbster & Warmuth, 1998; Bousquet & Warmuth, 2002) and time-varying comparators in dynamic regret (Zinkevich, 2003; Zhang et al., 2018). Second-order methods such as MLpol (Gaillard et al., 2014) adapt to the realised loss sequence; we use MLpol throughout. Past-posterior mixing (Bousquet & Warmuth, 2002; Cesa-Bianchi et al., 2012) hedges memory implicitly. MELO makes memory hedging explicit: it maintains a grid of EWLS adaptation horizons as competing experts, each updating a recursive state with no refitting on rolling windows.

Adaptive filtering and test-time adaptation. Forgotten RLS tracks time-varying linear coefficients (Kalaba & Tesfatsion, 1989) with one forgetting factor for all directions. Directional-forgetting schemes address the resulting estimator windup (Kulhavý & Kárný, 1984; Cao & Schwartz, 2000); in online learning, ARCOR and last-step regression control the covariance spectrum (Vaits & Crammer, 2011; Moroshko et al., 2015). These approaches estimate the needed directions from data. A Kalman filter with isotropic state noise, such as VIKING (de Vilmarest & Wintenberger, 2024), inflates the covariance as MELO does and learns one noise level online; on a prediction pool the directions are fixed by a spectrum known before deployment, and MELO hedges over the amount of inflation. Section 5 compares against a rolling window and against VIKING. A parallel line of work wraps frozen forecasters with lightweight post-hoc modules at test time (Liang et al., 2026; Lee et al.,

2025; Ansari et al., 2026), operating on single backbones or using static combination, without online adaptation under a time-varying comparator. Unlike jointly-trained mixture-of-experts gating (Jacobs et al., 1991), every base prediction in MELO enters the pool unchanged and the combination carries deterministic regret guarantees.

3 THE ADAPTATION LAYER AND ITS SPECTRAL PRIOR

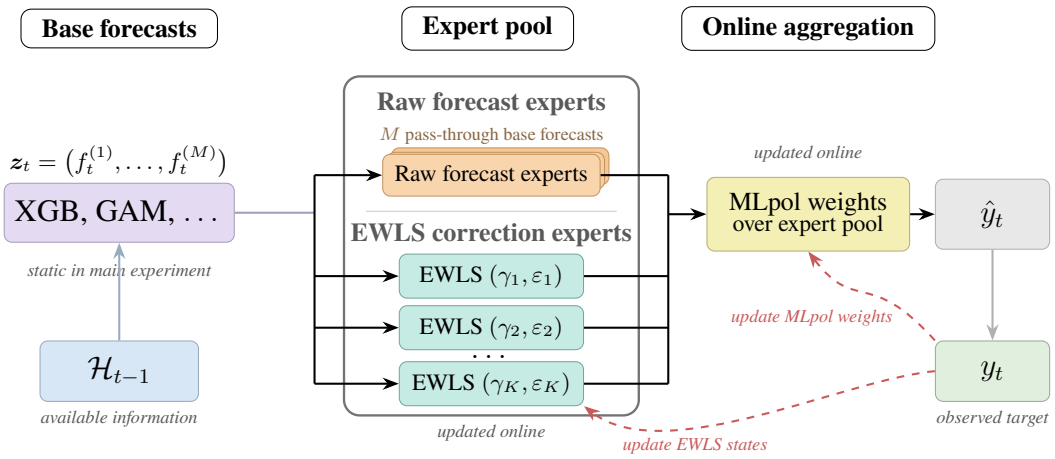


Figure 2: Overview. Base predictions $\boldsymbol{z}_t$ enter the pool directly and through K EWLS experts; MLpol aggregates the $M + K$ candidates into $\hat{y}_t$. After y_t is revealed, EWLS states and MLpol weights update online.

We study sequential one-step-ahead prediction. Throughout, *causal* means non-anticipating in time: the prediction at time t may use only inputs and labels revealed before t, and not future outcomes; we use *prediction* and *forecast* interchangeably. Let $\mathcal{H}_{t-1}$ denote all information available to the learner before y_t is revealed. By convention, $\mathcal{H}_{t-1}$ includes all past covariates, predictions, and revealed outcomes, as well as current-period inputs (calendar variables, lagged outcomes, exogenous covariates) already known at prediction time.

At step t, the learner uses $\mathcal{H}_{t-1}$ to form a prediction $\hat{y}_t$, then observes y_t, which enters $\mathcal{H}_t$ for all later steps. For the prediction sequence $\hat{y} = (\hat{y}_1, \ldots, \hat{y}_T)$, we aim to minimise the cumulative squared loss:

$$L_T(\hat{y}) = \sum_{t=1}^{T} \ell_t(\hat{y}_t) = \sum_{t=1}^{T} (\hat{y}_t - y_t)^2. \tag{1}$$

Let $z_{t,m} = f_t^{(m)}(\mathcal{H}_{t-1})$ be the prediction produced by base predictor $m \in [M]$, and collect the base predictions as $\boldsymbol{z}_t \equiv (z_{t,1}, \ldots, z_{t,M})^\top \in \mathbb{R}^M$. Put $\tilde{\boldsymbol{z}}_t = (\boldsymbol{z}_t^\top, 1)^\top \in \mathbb{R}^{M+1}$. Each $f_t^{(m)}$ may be frozen ($f_t^{(m)} \equiv f^{(m)}$) or causally retrained on expanding data in $\mathcal{H}_{t-1}$.

The base predictors are treated as black-box causal prediction streams, index m referring to the same predictor throughout. In the main experiments the base models are frozen, isolating the layer's adaptation from any benefit of base-model retraining. Throughout, $\boldsymbol{z}_t$ plays the role of a learned representation of the target: all M coordinates estimate y_t with different inductive biases, and the final prediction is an affine function of $\boldsymbol{z}_t$.

The layer runs on the standardized stream: with s^2 the mean squared target on the training window, it receives $(y_t/s,\ \boldsymbol{z}_t/s)$ and returns s times its prediction. Without the constant coordinate of $\tilde{\boldsymbol{z}}_t$ the slope dynamics are nearly invariant to this choice; standardizing fixes the scale of that coordinate, and every remaining scalar in the design is a pure number, shared across datasets (Appendix B.7).

The spectral prior and EWLS experts. The coordinates of $\tilde{\boldsymbol{z}}_t$ are predictions of the same target, so on the standardized stream the training Gram matrix $T^{-1} \sum_t \tilde{\boldsymbol{z}}_t \tilde{\boldsymbol{z}}_t^\top$ has total mass $\widehat{B}_z^2 \approx M + 1$

on every dataset, and its spectrum orders the coefficient space into distinct scales that are fixed before the test stream begins. The top eigendirection is the consensus level, because every column estimates the same quantity. On the two streams of Section 5 its $\boldsymbol{z}$-block has cosine 1.000 and 0.9999 with the uniform vector; the second eigendirection, smaller by a factor 183 and 92, is the offset, dominated by the intercept coordinate, which carries the additive part of the affine correction as opposed to the multiplicative gain on the level; the model contrasts follow, from a factor $1{,}645$ and 273 down.

Each EWLS expert combines the base predictions with time-varying coefficients. For a forgetting factor $\gamma \in (0, 1)$ and a covariance-inflation scalar $\varepsilon > 0$, the coefficient path solves

$$\{\boldsymbol{w}_{j|t-1}^{(\gamma,\varepsilon)}\}_{1\le j\le t-1} = \underset{\{\boldsymbol{w}_j\}}{\arg\min} \Big\{ \sum_{j=1}^{t-1} \gamma^{(t-1)-j} \big(y_j - \boldsymbol{w}_j^\top \tilde{\boldsymbol{z}}_j\big)^2 + \frac{1}{\varepsilon} \sum_{j=1}^{t-2} \gamma^{(t-1)-j} \big\|\boldsymbol{w}_{j+1} - \boldsymbol{w}_j\big\|_2^2 \Big\}, \quad (2)$$

and the expert predicts with the terminal coefficient: $\breve{y}_t^{(\gamma,\varepsilon)} = \tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t^{(\gamma,\varepsilon)}$ with $\boldsymbol{w}_t^{(\gamma,\varepsilon)} := \boldsymbol{w}_{t-1|t-1}^{(\gamma,\varepsilon)}$. The second sum penalises the roughness of the coefficient path: as $\varepsilon \to 0$ it forces $\boldsymbol{w}_1 = \cdots = \boldsymbol{w}_{t-1}$, recovering a constant-coefficient (static) EWLS fit, while larger ε permits a faster-moving path. The covariance $\boldsymbol{P}_t$ tracks the estimation precision along each coefficient direction; the $+\varepsilon \boldsymbol{I}$ term inflates it at every step, injecting fresh uncertainty that keeps the estimator responsive. The terminal coefficients satisfy a forgetting recursive least-squares (RLS) recursion (Kalaba & Tesfatsion, 1989; Gunnarsson, 1996), and are computed exactly and recursively without re-solving (2) (Algorithm 1, Appendix B.4). The carried covariance updates as

$$\boldsymbol{P}_t \;\leftarrow\; \gamma^{-1}\Big(\boldsymbol{P}_{t-1} - \frac{\boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top \boldsymbol{P}_{t-1}}{\gamma + \tilde{\boldsymbol{z}}_t^\top \boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t}\Big) + \varepsilon\, \boldsymbol{I}. \quad (3)$$

The gain vector and coefficient update are

$$\boldsymbol{K}_t = \frac{\boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t}{\gamma + \tilde{\boldsymbol{z}}_t^\top \boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t}, \qquad \boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \boldsymbol{K}_t\,(y_t - \tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t). \quad (4)$$

The correction at each step is proportional to the prediction error, scaled by the gain $\boldsymbol{K}_t$: directions where $\boldsymbol{P}_{t-1}$ has large eigenvalues receive larger corrections. The constant 1 appended to $\boldsymbol{z}$ is the offset channel. Within short windows it is nearly collinear with the consensus level, and the consensus gain does its work; over long windows the seasonal movement of the level separates additive offset from multiplicative gain, and dropping the column costs 2.3% and 3.9% on the two streams of Section 5 (Appendix B.7). The forgetting factor controls the adaptation speed via nominal memory length $h(\gamma) = 1/(1-\gamma)$: smaller $h(\gamma)$ reacts faster but is noisier.

Memory lengths via covariance inflation. Write the eigendecomposition $\boldsymbol{P}_{t-1} = \sum_i p_i \boldsymbol{u}_i \boldsymbol{u}_i^\top$. From (4), the gain decomposes as

$$\boldsymbol{K}_t = \frac{1}{s_t} \sum_i p_i\, (\boldsymbol{u}_i^\top \tilde{\boldsymbol{z}}_t)\, \boldsymbol{u}_i, \qquad s_t = \gamma + \tilde{\boldsymbol{z}}_t^\top \boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t,$$

so the coefficient adjustment along eigendirection $\boldsymbol{u}_i$ is proportional to p_i. The $\varepsilon \boldsymbol{I}$ term in (3) floors every eigenvalue p_i at ε. A direction receiving per-step squared input λ_i settles well above this floor, at $p_i \approx \sqrt{\varepsilon/\lambda_i}$ when $\varepsilon\lambda_i \ll 1$, and its memory is about $1/\sqrt{\varepsilon\lambda_i}$ steps (Lemma A.5 with λ_i for B_z^2, as for the steady-state gain of a local-level Kalman filter (Harvey, 1990)). At the training spectrum the offset and the first contrast therefore keep about 14 and 41 times the memory of the level on the electricity stream, and 10 and 17 times on the taxi stream. The level re-scales the pool consensus within days of a break, the offset follows, and the relative weights between models keep the longest memory. The forgetting factor γ discounts every direction at one rate and caps every memory at $h(\gamma)$; the spectrum fixes the ratios, and c shifts all memory lengths. We couple the two scalars by $\varepsilon_k = \varepsilon_0(1-\gamma_k)$ inside each sub-pool, which makes the inflation cost in the oracle inequality the same at every memory length (Remark A.6); ε_0 itself is hedged over a second grid (see the grid design below). Nothing in this allocation is estimated from the test stream: the layer computes only $\widehat{B}_z^2$ from the training predictions, and the grids are fixed. Section 6 measures how little room this leaves to any tuned alternative.

MLpol aggregation. The appropriate (γ, ε) pair is unknown and may itself change over time. Instead of selecting it, MELO maintains K EWLS experts over a grid of pairs $(\gamma_k, \varepsilon_k)$ and hedges over them with the MLpol aggregator. At time t, the pool of $N = M + K$ candidate predictions concatenates the raw base predictors and the EWLS experts:

$$\tilde{\boldsymbol{y}}_t \equiv (\tilde{y}_{t,1}, \cdots, \tilde{y}_{t,N})^\top = \big(z_{t,1}, \cdots, z_{t,M}, \check{y}_t^{(\gamma_1,\varepsilon_1)}, \cdots, \check{y}_t^{(\gamma_K,\varepsilon_K)}\big)^\top.$$

The final predictor is $\hat{y}_t = \boldsymbol{p}_t^\top \tilde{\boldsymbol{y}}_t$, with weights produced by MLpol (Gaillard et al., 2014), the polynomially weighted average with multiple adaptive learning rates. Writing $r_{s,j} = 2(\hat{y}_s - y_s)(\hat{y}_s - \tilde{y}_{s,j})$ for the gradient-trick excess loss of the aggregate over expert j and $R_{t,j} = \sum_{s\le t} r_{s,j}$,

$$p_{t,j} = \frac{\eta_{t-1,j}\,[R_{t-1,j}]_+}{\sum_{i=1}^N \eta_{t-1,i}\,[R_{t-1,i}]_+}, \qquad \eta_{t,j} = \frac{1}{1 + \sum_{s\le t} r_{s,j}^2 + b_t}, \quad b_t = \max_{s\le t,i} r_{s,i}^2, \tag{5}$$

where $b_0 = 0$, $\eta_{0,j} = 1$, and $[x]_+ = \max\{x, 0\}$; uniform weights apply when all positive parts vanish. An expert gains weight when it would have moved the aggregate prediction toward the observed target. Raw base predictors are included directly in the pool, so the aggregate can fall back to them when online adaptation is unnecessary. Each candidate prediction is an affine function of $\tilde{\boldsymbol{z}}_t$, so is the aggregate.

The forgetting factors sit on a geometric grid in nominal memory length, $h_k = 1/(1-\gamma_k)$, over $[h_{\min}, h_{\max}]$: only $O(\log(h_{\max}/h_{\min}))$ points are needed for some grid point to approximate the oracle memory length up to a constant factor. The deployed grids are $h \in [20, 5000]$ with 15 points plus the $\gamma = 1$ endpoint on the electricity stream, and $h \in [20, 2000]$ with 12 points plus the endpoint on the taxi stream. The inflation is hedged on a second, dimensionless grid: for $c \in \mathcal{C} = \{1, 10^2, \ldots, 10^{10}\}$, one sub-pool runs the coupled schedule at $\varepsilon_0 = c/\widehat{B}_z^2$, where $\widehat{B}_z^2$ is the mean squared design norm on the training window; on the standardized stream $\widehat{B}_z^2 \approx M + 1$, so the grid covers the same memory lengths of the consensus level on every dataset. With the six values of c, the pools hold $K = 96$ and $K = 78$ EWLS experts next to the $M = 5$ raw bases. A validation study backs the hedge (Appendix B.5). The hedge beats every single value of c on both test periods. Selecting c on the pre-test year works on one stream and fails on the other: on the taxi stream the 2018 ranking is inverted, and deploying its choice costs 2.8% out of sample.

Implementation. EWLS states follow the standard RLS recursion with causal initialization (Algorithm 1, Appendix B.1). A persistent ridge $\lambda = 10^{-12}$ floors the precision at near-zero eigenvalues (Remark A.10), preventing covariance windup from collinear pool members (Appendix F). The layer carries no tuned hyperparameters: every scalar is either hedged or fixed; Section 4 makes the grid guarantees precise. The rank-one EWLS update costs $O(d^2)$ per expert ($d = M + 1$). The current dense solve for persistent ridge makes the implemented cost $O(Kd^3 + N)$ per step, still independent of T and small at $d = 6$.

4 Guarantees for the adaptation layer

We state the guarantee that motivates the design. The formal statements, constants, clipping conventions, and proofs for the recursion with covariance inflation are in Appendix A. Write $L_T(a_{1:T}) = \sum_{t=1}^T (a_t - y_t)^2$ for any prediction sequence. For a time-varying affine combination path $\boldsymbol{u}_{1:T}$, define

$$L_T(\boldsymbol{u}_{1:T}) = \sum_{t=1}^T \big(\tilde{\boldsymbol{z}}_t^\top \boldsymbol{u}_t - y_t\big)^2, \qquad P_T(\boldsymbol{u}_{1:T}) = \sum_{t=2}^T \|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|_2,$$

so P_T measures the total movement of the comparator and accommodates both gradual drift and abrupt changes; let $\mathcal{U}_T(R) = \{\boldsymbol{u}_{1:T} : \|\boldsymbol{u}_t\|_2 \le R \text{ for all } t\}$. We assume bounded data, $\|\tilde{\boldsymbol{z}}_t\|_2 \le B_z$ and $|y_t| \le B_y$, and bounded EWLS iterates along the realised sequence, $\|\boldsymbol{w}_t^{(\gamma)}\|_2 \le R$; the latter is an explicit stability condition on the realised recursion, not a consequence of forgetting alone. Let $\hat{y}_{1:T}$ be the MLpol aggregate over the deployed pool: the M raw base forecasts $f^{(j)}$ and the $K = |\Gamma|\,|\mathcal{C}|$ EWLS experts on the (γ, c) grid, expert (γ, c) running the inflation $\varepsilon_{\gamma,c} = c\,(1-\gamma)/\widehat{B}_z^2$.

Theorem 1 (MELO oracle inequality; informal, see Corollary A.13)**.** *Under the assumptions above and the bounded-loss MLpol convention of Appendix A, suppressing logarithmic and initialisation terms, the aggregate satisfies, simultaneously,*

$$L_T(\hat{y}_{1:T}) \;\le\; \min_{1\le j\le M} L_T(f^{(j)}_{1:T}) \;+\; \Delta_{N,T}, \qquad \text{(safety)}$$

$$L_T(\hat{y}_{1:T}) \;\le\; L_T(\boldsymbol{u}_{1:T}) + \frac{C_1\, dT}{h_\gamma} + C_2\left(h_\gamma \wedge h_\varepsilon\right) P_T(\boldsymbol{u}_{1:T}) + C_3\, c\, T \;+\; \Delta_{N,T}, \qquad \text{(tracking)}$$

the latter for every comparator path $\boldsymbol{u}_{1:T} \in \mathcal{U}_T(R)$ and every grid pair $(\gamma, c) \in \Gamma \times \mathcal{C}$. Here $h_\gamma = 1/(1-\gamma)$; $h_\varepsilon = (\varepsilon_{\gamma,c} B_z^2)^{-1}$ is the scale at which the precision cap of pair (γ, c) binds, and under the plug-in coupling $h_\varepsilon = (\widehat{B}_z^2/B_z^2)\, h_\gamma/c$; $\Delta_{N,T}$ is the aggregation overhead discussed below; C_1, C_2, C_3 hide only boundedness and initialisation constants.

The aggregate competes with two benchmarks simultaneously. The (safety) bound keeps it competitive with the best individual raw predictor up to the sublinear aggregation overhead. The (tracking) bound captures distribution shift through the best time-varying affine combination, and displays the stability–adaptivity trade-off: short memory reduces the term with P_T but inflates the term dT/h. When the two terms balance,

$$L_T(\hat{y}_{1:T}) \;\le\; L_T(\boldsymbol{u}_{1:T}) + C\sqrt{dT\, P_T(\boldsymbol{u}_{1:T})} + C_3\, c\, T + \Delta_{N,T},$$

where the second term is the classic path-length dynamic-regret rate, achieved here by a tuning-free causal procedure, and $C_3\, c\, T$ is the price of the inflation. The latter is linear in T and proportional to c, because the inflation leaks a fixed amount of precision at every step (Appendix A); in exchange it shortens the memory h_ε in the term with P_T. The bound holds for every grid pair, so the aggregate inherits the best trade-off on the grid; $c = 1$ caps the term at $C_3 T$, and it vanishes at $\gamma = 1$, where $\varepsilon = 0$. A geometric grid ensures that some h_γ is within a constant factor of the oracle scale (Corollary A.15); the c-grid hedges the inflation–tracking trade-off (Remarks A.6–A.8). The aggregation overhead $\Delta_{N,T}$ is the MLpol second-order bound for the implemented adaptive rates (Lemma A.12), at most $O_B(\sqrt{NT\ln(1+T)})$ for fixed clipping radius B. On both streams this overhead is not visible: the causal hedge matches or beats the best fixed cell of its own family (Section 6). The tracking bounds use the stated initialization and omit the persistent numerical ridge of Section 3; Remark A.10 states its role. The aggregation bound covers the implemented adaptive-rate update.

5 Two operational streams under distribution shift

Can one tuning-free layer improve frozen forecasters under structurally different regime shifts? We test on two operational one-step-ahead streams: French national electricity load through the COVID-19 lockdown and the 2022 energy crisis, and NYC yellow-taxi daily ridership through a slow ride-hailing decline followed by the COVID collapse. In both cases the base pools are trained once, frozen before the test period, and wrapped by the same adaptation layer. For the bases and the layer, the only tuning is the base models' hyperparameter search on pre-test data; the layer itself has no tuned parameter. The two adaptive references below select their nuisance parameters on the 2018 validation leg (Appendix B.6). Figure 1 shows the result on both streams: each reference's cumulative excess loss over MELO steps up at every break.

5.1 Only MELO wins every regime on electricity load

We evaluate one-step-ahead forecasting of French national electricity load (Devaine et al., 2013; Obst et al., 2021; De Vilmarest & Goude, 2022). Five base forecasters (Lag-1, Ridge, GAM, XGBoost, LightGBM) are trained once on 2012–2018 and frozen; the adaptation layer runs the deployed configuration of Section 3 with nothing selected on validation data. Online evaluation spans 2019–2023, covering six regimes including the COVID-19 lockdown and the 2022 European energy crisis (Appendix B.2).

Table 1 shows the main message on this stream: MELO improves on base-only aggregation in every regime, with the largest gains where base models degrade most. No base is strong everywhere, so any fixed choice among them is wrong in some regime; the aggregate with the adaptation layer is best in

Table 1: Per-regime and overall RMSE (MW) on RTE-FR Load, 2019–2023. **Bold**: best per column. The three MLpol rows use the same rule with the gradient trick (Section 3). Rolling OLS refits an affine combination daily on the last W stream days; VIKING is the variational Kalman filter of de Vilmarest & Wintenberger (2024); both select their nuisance parameters on the 2018 validation leg (Appendix B.6). The $\varepsilon_0{=}0$ row hedges the same γ-grid without covariance inflation (Section 6). Paired moving-block bootstrap 95% intervals for the overall difference against MELO are $[+6, +97]$ MW for VIKING and $[+142, +404]$ MW for the rolling window (Appendix B.6).

Method *Regime length (days)*	Pre-COVID *441*	Lockdown *56*	Recovery *599*	Crisis *273*	Reduction *182*	2023 *183*	Overall *1734*
Lag-1 (base)	3918.6	2626.3	3420.3	3193.1	3531.4	2849.9	3455.5
Ridge (base)	1999.0	3116.8	1938.3	1930.6	2527.0	1973.3	2072.6
GAM (base)	946.6	3201.3	1179.4	1545.0	1488.8	1139.5	1336.0
XGBoost (base)	842.8	3972.8	1341.4	1220.9	4031.4	2127.9	1931.4
LightGBM (base)	821.9	4410.9	1460.2	1315.5	4221.4	2498.3	2082.9
MLpol on Base only	747.0	2244.2	882.3	775.1	1419.4	948.2	987.1
Rolling OLS ($W{=}120$)	782.7	2539.7	902.1	765.0	1005.4	696.7	945.9
VIKING	698.1	1572.4	658.0	698.8	832.8	596.3	735.2
γ-grid only ($\varepsilon_0{=}0$)	751.2	1801.0	737.2	729.9	914.7	643.2	807.1
MELO	**662.9**	**1063.0**	**655.7**	**660.5**	**815.8**	**580.9**	**686.2**

all six. VIKING comes close in the calm regimes and falls behind in the lockdown: its isotropic state noise gives it the ordered memory lengths, but its single learned inflation barely moves at the break (Appendix B.6). The γ-grid family without inflation has one memory for every direction and loses to VIKING. Section 6 traces this mechanism; all overall differences against MELO are significant under a paired moving-block bootstrap (Appendix C.3).

Foundation-model references (TabPFN and TabICL refit daily on an expanding in-context window) are compared on the 2019–2021 sub-window in Appendix C.2. The strongest reference reaches lower overall RMSE, 628.9 against 697.2 MW for MELO, at three orders of magnitude higher per-step cost. The framework absorbs such references: adding the reference forecast to the pool lowers the RMSE below the reference itself, significantly only for TabICL (Appendix I).

5.2 On taxi ridership the layer improves on the raw pool in every regime

The second stream is daily NYC yellow-taxi trips, 2019–2023, with two forms of shift: a slow ride-hailing decline that moved ridership below the training range before 2020, and the COVID collapse that removed about 95% of daily trips. The protocol mirrors RTE-FR, with five bases frozen on 2011–2017 (seasonal naive, calendar-and-weather OLS, ridge, XGBoost, LightGBM) and nothing selected on validation data. The two targets differ in units and scale, but standardization places both designs on a common scale ($\widehat{B}_z^2 \approx 6$). The same dimensionless inflation grid can therefore be used on both streams (Appendix D).

Table 2: Per-regime and overall RMSE (daily trips) on the NYC yellow-taxi test period. Rolling OLS refits an affine combination daily on the last W eval-stream days, W selected on 2018; VIKING selects its nuisance parameters on 2018, and its overall difference against MELO has paired bootstrap interval $[+100, +1319]$ trips (Appendix B.6). The $\varepsilon_0{=}0$ row hedges the same γ-grid without inflation. Full table with all base rows and the hindsight rolling window: Appendix D.

Method *Regime length (days)*	Pre-COVID *436*	Collapse *111*	Recovery *549*	New normal *730*	Overall *1826*
Best single base (seasonal-naive)	29903	35548	10730	18106	21354
MLpol on Base only	20729	31183	11181	15508	17188
Rolling OLS ($W{=}365$)	15653	23625	12337	11419	13796
VIKING	**13092**	13716	6574	10118	10309
γ-grid only ($\varepsilon_0{=}0$)	15120	18266	7345	10710	11702
MELO	13231	**9284**	**6096**	**9213**	**9599**

MELO is the only method whose error in the collapse is below its own pre-COVID error, as it should be for a method that re-scales quickly once most trips are gone. Its pre-COVID margin over the

raw pool comes from the slow decline alone, before any pandemic effect (Table 2). VIKING is slightly better before COVID and falls behind in the collapse, where its learned inflation again barely moves. All overall differences against MELO are significant under a paired moving-block bootstrap (Appendices D and B.6).

6 WHY IT WORKS

Section 5 established the gains. We now trace where they come from. All numbers are computed on the RTE-FR 2019–2023 stream with the five-member pool and the deployed pipeline unless stated otherwise.

The break is absorbed at the shortest memory lengths. A controlled pair isolates the channels (Figure 3a,b, with the offset in Appendix H): one EWLS expert at fixed $\gamma = 1$, inflation toggled between $\varepsilon = 0$ and $\varepsilon = 1/\widehat{B}_z^2$, its coefficient path decomposed into the consensus gain $\sum_m w_m$ (the motion along the dominant eigendirection of Section 3), the offset, and the contrasts. The inflation alone opens the fast channels: the consensus gain and the offset absorb the break at the two shortest memory lengths, while the contrasts, an order of magnitude quieter, keep the relative weights between models on their long memory.

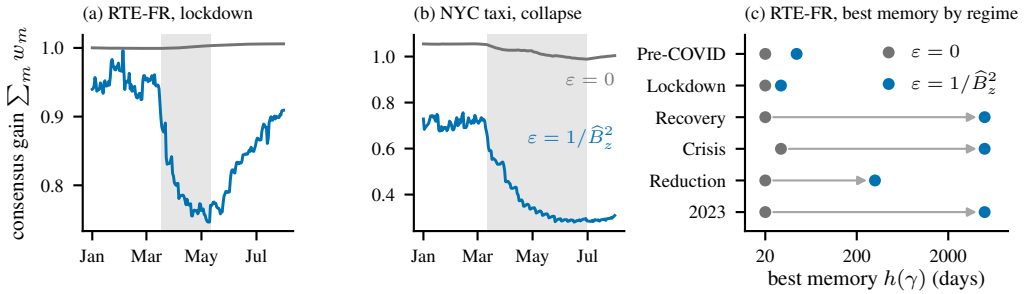


Figure 3: **The inflation opens a fast consensus channel.** (a, b) Consensus gain $\sum_m w_m$ of one EWLS expert at $\gamma = 1$ with $\varepsilon = 0$ and $\varepsilon = 1/\widehat{B}_z^2$; shading marks the break. (c) Memory $h(\gamma)$ with the lowest RMSE in each regime for standalone experts at fixed γ, with the same two inflations; the worst memory costs up to 127% in RMSE without the inflation and at most 7% with it (Figure 5).

One memory speed cannot serve two directions. Figure 3c sweeps standalone EWLS experts at fixed forgetting scales under the same toggle. Without inflation the sweep contradicts the naive reading that calm regimes want long memory. With one speed for every direction, the level drift present in every regime dictates the choice, and the estimator gives up the contrast memory to track the level. Once the fast channels carry the level and the offset, what remains is the memory trade-off in the contrast directions, which the γ-grid hedges, and the calm regimes recover their long-memory optimum. The value of keeping the contrasts stable is visible only once the level is decoupled.

The inflation matters on both streams. The $\varepsilon_0 = 0$ rows of Tables 1 and 2 hedge the same γ-grid without inflation, which is all a multi-scale forgotten-RLS family can do. On both streams that family loses to VIKING, which learns a single inflation online, and with the inflation hedged the same pool beats VIKING; the gap between the $\varepsilon_0 = 0$ row and MELO is widest at the breaks. Appendix H shows the aggregator exploiting the regime-dependent preferences of Figure 3 online: at lockdown onset weight reallocates sharply into the EWLS experts with nominal memory $h < 100$.

Hedging exhausts the family. A natural alternative to hedging is selection: evaluate the cells of the (γ, c) family on a trailing window and play the winner. On both streams the causal hedge beats every tested window selector and matches the per-regime best-cell splice with clairvoyant break timing: the paired bootstrap intervals for the overall difference between the splice and the hedge are $[-11, +14]$ MW on RTE-FR and $[-526, +111]$ trips on NYC taxi (Table 3; NYC taxi in Appendix G). Selection loses because the window length W is the memory choice again: a short window tracks the break and

Table 3: Hedging vs. causal window selection vs. hindsight oracle over the deployed pool (101 cells), RTE-FR 2019–2023, RMSE in MW. **Bold**: best per column. Select-W plays, each day, the cell with the lowest mean squared error over the last W days. All six windows and the NYC replication: Appendix G.

Method	Pre-COVID	Lockdown	Recovery	Crisis	Reduction	2023	Overall
Best fixed cell (hindsight)	693.8	1209.0	652.3	682.1	815.6	593.5	704.9
Select W=30	702.6	988.2	694.3	719.0	851.3	632.5	723.4
Select W=365	742.3	1116.7	674.5	684.8	815.1	591.6	719.8
Oracle splice (hindsight)	693.8	**978.6**	**648.4**	682.1	**809.6**	588.0	690.7
MELO	**662.9**	1063.0	655.7	**660.5**	815.8	**580.9**	**686.2**

pays in the calm regimes, a long window does the opposite, and no single W is right throughout. The hedge can match the splice because its weights keep moving inside a regime and mix cells whose errors differ, and a mixture of cells is not in any selector's action set.

Table 4: Wrapping each base alone (RTE-FR 2019–2023, deployed pipeline). RMSE in MW; Δ is the reduction from raw to wrapped.

	Overall			Lockdown		
Base	raw	wrapped	Δ	raw	wrapped	Δ
Lag-1	3455.5	3338.3	$+3.4\%$	2626.3	2658.0	-1.2%
Ridge	2072.6	1735.8	$+16.3\%$	3116.8	1678.6	$+46.1\%$
GAM	1336.0	911.1	$+31.8\%$	3201.3	1374.2	$+57.1\%$
XGBoost	1931.4	736.6	$+61.9\%$	3972.8	1032.4	$+74.0\%$
LightGBM	2082.9	742.9	$+64.3\%$	4410.9	1110.8	$+74.8\%$

The layer repairs only calibration. Table 4 wraps each base alone in the same layer, which changes only the calibration of a base, the affine map from its prediction to the target; the discrimination, the information the base extracts from its inputs, must come from the pool. The gains order the pool by what each base carries. Lag-1 gains almost nothing: it re-anchors on yesterday's load by construction, so it carries almost no persistent miscalibration for the layer to remove, and no exogenous structure either. The frozen feature-based models gain the most, especially in lockdown: their feature-to-load relations broke at the shift, and that is the low-dimensional, persistent miscalibration the affine layer repairs. On the taxi stream the raw bases degrade in the same order, from the calendar-and-weather OLS that carries the broken relation to the self-anchoring seasonal-naive predictor (Appendix D). Wrapping a daily-refit tabular foundation model shows the same ordering on the 2019–2021 sub-window (Appendix C.2). A foundation model refit every day on an expanding context is already adaptive, and what the layer adds depends on how much of the drift the model can see. Given no drift feature, TabPFN and TabICL gain about 27% from the wrap; given a time index and a lockdown-stringency covariate, the gain falls to 1.4% or less, and the strongest reference, TabPFN on the differenced target, is worsened when wrapped alone. Wrapping an already adaptive base is therefore not uniformly beneficial: the layer repairs persistent miscalibration, and a base that repairs its own leaves nothing for it. Appendix E runs the converse probes: added members carrying unspanned structure help, while weak or deliberately overfit members cost essentially nothing.

7 Conclusion

The spectral prior of a frozen prediction pool provides ordered memory lengths. MELO hedges over their scale, and its aggregate satisfies deterministic oracle inequalities. With no tuned hyperparameters, the causal hedge matches its family's hindsight per-regime splice on both streams within bootstrap noise, and trailing-window selection over the same pool loses on both streams.

The gains have a clear boundary. The layer repairs persistent, low-dimensional miscalibration of a frozen pool. It cannot supply discrimination the pool lacks. A base that re-anchors on the recent past, or a model refit daily with the drift visible in its inputs, leaves little to repair, and the wrap gain on such bases is near zero. Where the pool carries broken exogenous relations, as the feature-based models do after a break, the gain is largest. The two streams here are scalar and daily; multivariate targets, structured losses, and actively retrained pools are the next tests.

AI USE STATEMENT

In this work, we used generative AI tools for code debugging, literature search, and editing of the text. All AI-assisted work was reviewed and verified by the authors. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All base-model training, hyperparameter selection, and online evaluation protocols are described in Appendices B–B.5. The algorithm is fully specified in Algorithm 1. Theoretical results are stated with explicit constants and proved in Appendix A. The source code, the shipped base predictions, and a script that regenerates every table and figure are included in the supplementary material.

REFERENCES

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631, 2019. URL https://doi.org/10.1145/3292500.3330701.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasber Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torber, Andrew Gordon Wilson, Yuyang Wang, and Jan Gasthaus. Test-time efficient pretrained model portfolios for time series forecasting. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2510.06419.

John M. Bates and Clive W. J. Granger. The combination of forecasts. *Journal of the Operational Research Society*, 20(4):451–468, 1969. doi: 10.1057/jors.1969.103. URL https://doi.org/10.1057/jors.1969.103.

Olivier Bousquet and Manfred K. Warmuth. Tracking a small set of experts by mixing past posteriors. *Journal of Machine Learning Research*, 3:363–396, 2002. URL https://jmlr.org/papers/volume3/bousquet02b/bousquet02b.pdf.

Leo Breiman. Stacked regressions. *Machine Learning*, 24(1):49–64, 1996. URL https://doi.org/10.1023/A:1018046112532.

Liyu Cao and Howard M. Schwartz. A directional forgetting algorithm based on the decomposition of the information matrix. *Automatica*, 36(11):1725–1731, 2000. URL https://www.sciencedirect.com/science/article/abs/pii/S0005109800000935.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006. URL https://doi.org/10.1017/CBO9780511546921.

Nicolò Cesa-Bianchi, Pierre Gaillard, Gábor Lugosi, and Gilles Stoltz. Mirror descent meets fixed share (and feels no regret). In *Advances in Neural Information Processing Systems 25 (NeurIPS)*, 2012. URL https://proceedings.neurips.cc/paper/2012/hash/8e6b42f1644ecb1327dc03ab345e618b-Abstract.html.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *KDD*, 2016. URL https://doi.org/10.1145/2939672.2939785.

Joseph De Vilmarest and Yannig Goude. State-space models for online post-covid electricity load forecasting competition. *IEEE Open Access Journal of Power and Energy*, 9:192–201, 2022. URL https://ieeexplore.ieee.org/document/9677626.

Joseph de Vilmarest and Olivier Wintenberger. Viking: Variational Bayesian variance tracking. *Statistical Inference for Stochastic Processes*, 27(3):839–860, 2024. doi: 10.1007/s11203-024-09312-7. URL https://doi.org/10.1007/s11203-024-09312-7.

Marie Devaine, Pierre Gaillard, Yannig Goude, and Gilles Stoltz. Forecasting electricity consumption by aggregating specialized experts. *Machine Learning*, 90(2):231–260, 2013. doi: 10.1007/s10994-012-5314-7. URL https://doi.org/10.1007/s10994-012-5314-7.

Pierre Gaillard, Gilles Stoltz, and Tim van Erven. A second-order bound with excess losses. *COLT*, 2014. URL https://proceedings.mlr.press/v35/gaillard14.html.

Pierre Gaillard, Yannig Goude, and Raphaël Nedellec. Additive models and robust aggregation for GEFCom2014 probabilistic electric load and electricity price forecasting. *International Journal of Forecasting*, 32(3):1038–1050, 2016. doi: 10.1016/j.ijforecast.2015.12.001. URL https://doi.org/10.1016/j.ijforecast.2015.12.001.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://arxiv.org/abs/2106.11959.

Yannig Goude, Raphaël Nedellec, and Nicolas Kong. Local short and middle term electricity load forecasting with semi-parametric additive models. *IEEE Transactions on Smart Grid*, 5(1):440–446, 2014. doi: 10.1109/TSG.2013.2278425. URL https://doi.org/10.1109/TSG.2013.2278425.

Svante Gunnarsson. Combining tracking and regularization in recursive least squares identification. In *Proceedings of the 35th IEEE Conference on Decision and Control*, volume 3, pp. 2551–2552, 1996. doi: 10.1109/CDC.1996.573481. URL https://doi.org/10.1109/CDC.1996.573481.

Andrew C. Harvey. *Forecasting, structural time series models and the Kalman filter*. Cambridge university press, 1990. URL https://doi.org/10.1017/CBO9781107049994.

Trevor Hastie and Robert Tibshirani. Generalized additive models. *Statistical Science*, 1(3):297–318, 1986. doi: 10.1214/ss/1177013604. URL https://doi.org/10.1214/ss/1177013604.

Monson H Hayes. *Statistical digital signal processing and modeling*. John Wiley & Sons, 1996. URL https://www.wiley.com/en-us/Statistical+Digital+Signal+Processing+and+Modeling-p-9780471594314.

Mark Herbster and Manfred K Warmuth. Tracking the best expert. *Machine learning*, 32(2):151–178, 1998. URL https://link.springer.com/article/10.1023/A:1007424614876.

Noah Hollmann et al. Tabpfn: A transformer that solves small tabular classification problems in a second. In *ICLR*, 2023. URL https://arxiv.org/abs/2207.01848.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. URL https://doi.org/10.1162/neco.1991.3.1.79.

Robert Kalaba and Leigh Tesfatsion. Time-varying linear regression via flexible least squares. *Computers & Mathematics with Applications*, 17(8–9):1215–1245, 1989. doi: 10.1016/0898-1221(89)90091-6. URL https://doi.org/10.1016/0898-1221(89)90091-6.

Guolin Ke et al. Lightgbm: A highly efficient gradient boosting decision tree. In *NeurIPS*, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html.

Rudolf Kulhavý and Miroslav Kárný. Tracking of slowly varying parameters by directional forgetting. In *Proceedings of the 9th IFAC World Congress*, pp. 687–692, 1984. URL https://www.sciencedirect.com/science/article/pii/S1474667017610516.

Haraldur Tomas Lee, Théo Dumont, Tim Januschowski, and Jan Gasthaus. Lightweight online adaption for time series foundation model forecasts. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 33736–33764, 2025. URL https://arxiv.org/abs/2502.12920.

Daojun Liang, Qi Li, Yinglong Wang, Jing Chen, Hu Zhang, Xiaoxiao Cui, Qizheng Wang, and Shuo Li. The forecast after the forecast: A post-processing shift in time series. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2601.20280.

Edward Moroshko, Nina Vaits, and Koby Crammer. Second-order non-stationary online learning for regression. *Journal of Machine Learning Research*, 16:1481–1517, 2015. URL https://jmlr.org/papers/v16/moroshko15a.html.

David Obst, Joseph de Vilmarest, and Yannig Goude. Adaptive methods for short-term electricity load forecasting during COVID-19 lockdown in France. *IEEE Transactions on Power Systems*, 36(5):4754–4763, 2021. doi: 10.1109/TPWRS.2021.3067551. URL https://doi.org/10.1109/TPWRS.2021.3067551.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=0VvD1PmNzM.

Allan Timmermann. Forecast combinations. In Graham Elliott, Clive W. J. Granger, and Allan Timmermann (eds.), *Handbook of Economic Forecasting, Volume 1*, chapter 4, pp. 135–196. Elsevier, 2006. URL https://doi.org/10.1016/S1574-0706(05)01004-9.

Nina Vaits and Koby Crammer. Re-adapting the regularization of weights for non-stationary regression. In *Algorithmic Learning Theory (ALT)*, pp. 114–128, 2011. URL https://link.springer.com/chapter/10.1007/978-3-642-24412-4_12.

David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992. URL https://doi.org/10.1016/S0893-6080(05)80023-1.

Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pp. 1323–1333, 2018. URL https://arxiv.org/abs/1810.10815.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML)*, pp. 928–936, 2003. URL https://dl.acm.org/doi/10.5555/3041838.3041955.

A PROOFS FOR THE PATHWISE ONLINE GUARANTEES

This appendix proves the deterministic guarantees stated in Section 4: the dynamic-regret bound for a single forgotten-RLS adaptation expert, first for the idealised recursion and then for the implemented covariance-inflated one (Appendix A.6), the MLpol oracle inequality over the enlarged Base+EWLS pool, and the geometric-grid corollary. The analysis is sequence-wise: apart from the bounded-iterate condition, or the residual and distance bounds that it implies, we impose no distributional, stationarity, or persistence-of-excitation assumptions on the realised sequence.

A.1 SETUP AND NOTATION

We use the same notation as in the main text. At step t, the EWLS adaptation expert observes the augmented base-prediction vector $\tilde{\boldsymbol{z}}_t = (\boldsymbol{z}_t^\top, 1)^\top \in \mathbb{R}^d$, where $d = M + 1$, predicts $\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t$, then observes y_t and incurs squared loss

$$\ell_t(\boldsymbol{w}) = \big(\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w} - y_t\big)^2.$$

Here $\boldsymbol{w}_t$ denotes the pre-update coefficient used to predict at step t. The proof below uses no property of the raw covariates or of the base models beyond the boundedness of $\tilde{\boldsymbol{z}}_t$.

A.2 MAP TO THE INFORMAL STATEMENTS IN SECTION 4

The main text states Theorem 1 in informal form, suppressing constants and lower-order terms for readability. The formal statements for the deployed, covariance-inflated recursion are Theorem A.8 (single-scale tracking, which supplies the per-pair tracking bound) and Corollary A.9 (MELO oracle inequality), proved in Appendix A.6 and restated over the (γ, c) grid as Corollary A.13; their $\varepsilon = 0$ specialisations are Theorem A.3 and Theorem A.14, proved in Appendices A.5 and A.10. The geometric-grid statement of Section 4 is formalised as Corollary A.15 and extended to the deployed pool in Remark A.8. The informal constant C_3 of Theorem 1 corresponds to the inflation leakage $D_\varepsilon^2 \mathcal{E}_\varepsilon(T)$ of (38), which satisfies $\varepsilon h \le \varepsilon_0$ under the schedule (51). We restate the remaining correspondence here so that the notation in the formal statements is unambiguous.

The function Ψ_h. The single-scale tracking penalty appearing implicitly in the informal theorems is, with all constants made explicit,

$$\Psi_h(T,P) \;=\; \underbrace{\delta R^2}_{\text{initialisation}} \;+\; \underbrace{D^2 d \log\!\Big(1 + \tfrac{B_z^2 h}{\delta}\Big)}_{\text{log term}} \;+\; \underbrace{\frac{2D^2 dT}{h}}_{\text{finite-memory term}} \;+\; \underbrace{4R(\delta + B_z^2 h)\, P}_{\text{tracking term}}, \qquad (6)$$

where $D = B_y + B_z R$ and $d = M + 1$. The "log and initialisation terms" referenced in the informal Theorem 1 are the first two summands; the constants C_1, C_2 in the informal statement correspond to

$$C_1 = 2D^2, \qquad C_2 = 8RB_z^2 \quad \text{(under the mild regime } \delta \le B_z^2 h\text{)}.$$

The unsimplified C_2 is $4R(\delta + B_z^2 h)/h$, which reduces to the above once $\delta \le B_z^2 h$; see (17).

Equivalent affine view of the aggregate. The informal Theorem 1 compares $\hat{y}_t$ to time-varying affine combinations of the base-prediction vector. The justification is that every candidate in the MELO pool is itself affine in $\tilde{\boldsymbol{z}}_t$: each raw base expert is a coordinate of $\boldsymbol{z}_t$, and each EWLS expert is $\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t^{(\gamma)}$. Hence the MLpol aggregate satisfies

$$\hat{y}_t \;=\; \tilde{\boldsymbol{z}}_t^\top \bar{\boldsymbol{w}}_t, \qquad \bar{\boldsymbol{w}}_t \;=\; \sum_{j=1}^{N} p_{t,j}\, \boldsymbol{w}_{t,j},$$

for coefficients $\bar{\boldsymbol{w}}_t$ determined causally by past losses and EWLS states, where $\boldsymbol{w}_{t,j}$ is the affine representation of expert j (a canonical basis vector for raw base experts, $\boldsymbol{w}_t^{(\gamma_k)}$ for EWLS experts). The comparator class $\mathcal{U}_T(R)$ in Theorem A.3 is therefore the natural benchmark: it is the same affine class that $\hat{y}_t$ itself ranges over.

Clipping convention for the named base forecasts. In the formal oracle inequality (44), $L_T(f_j)$ is the loss of the clipped raw prediction sequence, since the theory clips every expert before it enters MLpol. When $|y_t| \le B$, Lemma A.10 shows that clipping cannot increase squared loss, so the bound also holds against the unclipped raw losses reported in Section 5.

Forgotten-RLS recursion. Fix a forgetting factor $\gamma \in (0,1)$ and define the nominal scale $h = (1-\gamma)^{-1}$. Let $\boldsymbol{A}_0 = \delta \boldsymbol{I}$ with $\delta > 0$, $\boldsymbol{b}_0 = \boldsymbol{0}$, and $\boldsymbol{w}_1 = \boldsymbol{0}$. The forgotten-RLS recursion is

$$\boldsymbol{A}_t = \gamma \boldsymbol{A}_{t-1} + \tilde{\boldsymbol{z}}_t \tilde{\boldsymbol{z}}_t^\top, \qquad \boldsymbol{b}_t = \gamma \boldsymbol{b}_{t-1} + y_t \tilde{\boldsymbol{z}}_t, \qquad \boldsymbol{w}_{t+1} = \boldsymbol{A}_t^{-1} \boldsymbol{b}_t. \tag{7}$$

Equivalently,

$$\boldsymbol{A}_t = \gamma^t \delta \boldsymbol{I} + \sum_{s=1}^{t} \gamma^{t-s} \tilde{\boldsymbol{z}}_s \tilde{\boldsymbol{z}}_s^\top.$$

Since $\boldsymbol{A}_t \succ 0$, the update is well-defined. Moreover, because $\boldsymbol{A}_{t-1} \boldsymbol{w}_t = \boldsymbol{b}_{t-1}$, a direct calculation gives

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \frac{1}{2} \boldsymbol{A}_t^{-1} \boldsymbol{g}_t, \qquad \boldsymbol{g}_t := \nabla \ell_t(\boldsymbol{w}_t) = 2(\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t) \tilde{\boldsymbol{z}}_t. \tag{8}$$

Boundedness assumptions. We use the following deterministic boundedness conditions:

(B1) $\|\tilde{\boldsymbol{z}}_t\|_2 \le B_z$ for all t.

(B2) $|y_t| \le B_y$ for all t.

(B3) The comparator path satisfies $\|\boldsymbol{u}_t\|_2 \le R$ for all t.

(B4) The unprojected forgotten-RLS iterates are stable along the realised sequence: $\|\boldsymbol{w}_t\|_2 \le R$ for all t.

Forgetting alone does not imply (B4). Without a conditioning or persistence-of-excitation condition, $\boldsymbol{A}_t$ may be ill-conditioned in directions the realised sequence rarely excites, and $\boldsymbol{w}_t = \boldsymbol{A}_{t-1}^{-1} \boldsymbol{b}_{t-1}$ need not stay bounded. We state the pathwise tracking bound on this stability event and make no stochastic excitation assumption. Remark A.2 gives weaker bounds that suffice, and Appendix B.9 checks the condition along the realised RTE-FR trajectory.

Set

$$D := B_y + B_z R.$$

Under (B1), (B2), and (B4), the realised RLS residuals satisfy

$$|\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t| \le D.$$

Under (B1), (B2), and (B3), any comparator $\boldsymbol{u}_t$ also satisfies

$$|\tilde{\boldsymbol{z}}_t^\top \boldsymbol{u}_t - y_t| \le D.$$

For a comparator path $\boldsymbol{u}_{1:T} = (\boldsymbol{u}_1, \dots, \boldsymbol{u}_T)$, define

$$L_T(\boldsymbol{u}_{1:T}) = \sum_{t=1}^{T} (\tilde{\boldsymbol{z}}_t^\top \boldsymbol{u}_t - y_t)^2, \qquad P_T(\boldsymbol{u}_{1:T}) = \sum_{t=2}^{T} \|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|_2. \tag{9}$$

We write

$$\check{y}_t^{(\gamma)} = \tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t, \qquad L_T(\check{y}^{(\gamma)}) = \sum_{t=1}^{T} \big(\check{y}_t^{(\gamma)} - y_t\big)^2$$

for the prediction sequence and cumulative loss of the forgotten-RLS expert with nominal scale $h = (1-\gamma)^{-1}$.

A.3 A ONE-STEP POTENTIAL IDENTITY

Lemma A.1 (One-step identity for forgotten RLS)**.** *For every $\boldsymbol{u} \in \mathbb{R}^d$ and every $t \ge 1$,*

$$\ell_t(\boldsymbol{w}_t) - \ell_t(\boldsymbol{u}) = \gamma \|\boldsymbol{w}_t - \boldsymbol{u}\|_{\boldsymbol{A}_{t-1}}^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{u}\|_{\boldsymbol{A}_t}^2 + \frac{1}{4} \boldsymbol{g}_t^\top \boldsymbol{A}_t^{-1} \boldsymbol{g}_t. \tag{10}$$

Proof. For squared loss, expanding around $\boldsymbol{w}_t$ gives the exact identity

$$\ell_t(\boldsymbol{w}_t) - \ell_t(\boldsymbol{u}) = \boldsymbol{g}_t^\top(\boldsymbol{w}_t - \boldsymbol{u}) - \big(\tilde{\boldsymbol{z}}_t^\top(\boldsymbol{w}_t - \boldsymbol{u})\big)^2. \tag{11}$$

From (8),

$$\boldsymbol{g}_t = 2\boldsymbol{A}_t(\boldsymbol{w}_t - \boldsymbol{w}_{t+1}).$$

Therefore

$$\begin{aligned}\boldsymbol{g}_t^\top(\boldsymbol{w}_t - \boldsymbol{u}) &= 2(\boldsymbol{w}_t - \boldsymbol{w}_{t+1})^\top \boldsymbol{A}_t(\boldsymbol{w}_t - \boldsymbol{u}) \\ &= \|\boldsymbol{w}_t - \boldsymbol{u}\|_{\boldsymbol{A}_t}^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{u}\|_{\boldsymbol{A}_t}^2 + \|\boldsymbol{w}_t - \boldsymbol{w}_{t+1}\|_{\boldsymbol{A}_t}^2 \\ &= \|\boldsymbol{w}_t - \boldsymbol{u}\|_{\boldsymbol{A}_t}^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{u}\|_{\boldsymbol{A}_t}^2 + \frac{1}{4}\boldsymbol{g}_t^\top \boldsymbol{A}_t^{-1}\boldsymbol{g}_t. \end{aligned} \tag{12}$$

The recursion $\boldsymbol{A}_t = \gamma \boldsymbol{A}_{t-1} + \tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top$ implies

$$\|\boldsymbol{w}_t - \boldsymbol{u}\|_{\boldsymbol{A}_t}^2 = \gamma\|\boldsymbol{w}_t - \boldsymbol{u}\|_{\boldsymbol{A}_{t-1}}^2 + \big(\tilde{\boldsymbol{z}}_t^\top(\boldsymbol{w}_t - \boldsymbol{u})\big)^2.$$

Substituting this expression and (12) into (11) cancels the curvature term $\big(\tilde{\boldsymbol{z}}_t^\top(\boldsymbol{w}_t - \boldsymbol{u})\big)^2$, yielding (10). □

A.4 A FORGETTING-ADAPTED LOG-DETERMINANT INEQUALITY

The classical online-Newton log-determinant argument relies on monotonicity of $\boldsymbol{A}_t = \boldsymbol{A}_{t-1} + \tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top$. With forgetting, $\boldsymbol{A}_t = \gamma\boldsymbol{A}_{t-1} + \tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top$ is no longer monotone in the PSD order. The next lemma is the corresponding log-determinant bound for forgotten RLS.

Lemma A.2 (Forgetting-adapted log-determinant inequality)**.** *Let $\boldsymbol{A}_t = \gamma\boldsymbol{A}_{t-1} + \tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top$ with $\boldsymbol{A}_0 = \delta\boldsymbol{I}$, $\delta > 0$, $\gamma = 1 - 1/h$, and $h \geq 2$. Under (B1),*

$$\sum_{t=1}^{T} \tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t^{-1}\tilde{\boldsymbol{z}}_t \leq d\log\left(1 + \frac{B_z^2 h}{\delta}\right) + \frac{2dT}{h}. \tag{13}$$

Proof. Define

$$q_t := \tilde{\boldsymbol{z}}_t^\top(\gamma\boldsymbol{A}_{t-1})^{-1}\tilde{\boldsymbol{z}}_t = \gamma^{-1}\tilde{\boldsymbol{z}}_t^\top\boldsymbol{A}_{t-1}^{-1}\tilde{\boldsymbol{z}}_t \geq 0.$$

By the Sherman–Morrison formula,

$$\boldsymbol{A}_t^{-1} = (\gamma\boldsymbol{A}_{t-1})^{-1} - \frac{(\gamma\boldsymbol{A}_{t-1})^{-1}\tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top(\gamma\boldsymbol{A}_{t-1})^{-1}}{1 + \tilde{\boldsymbol{z}}_t^\top(\gamma\boldsymbol{A}_{t-1})^{-1}\tilde{\boldsymbol{z}}_t}.$$

Contracting with $\tilde{\boldsymbol{z}}_t$ gives

$$\tilde{\boldsymbol{z}}_t^\top\boldsymbol{A}_t^{-1}\tilde{\boldsymbol{z}}_t = \frac{q_t}{1 + q_t}. \tag{14}$$

The matrix-determinant lemma gives

$$\det\boldsymbol{A}_t = \det(\gamma\boldsymbol{A}_{t-1})\big(1 + q_t\big) = \gamma^d\det(\boldsymbol{A}_{t-1})(1 + q_t).$$

Thus

$$\log(1 + q_t) = \log\det\boldsymbol{A}_t - \log\det\boldsymbol{A}_{t-1} + d\log(1/\gamma).$$

Summing over t and using $\log(1 + q) \geq q/(1 + q)$ for $q \geq 0$,

$$\begin{aligned}\sum_{t=1}^{T}\tilde{\boldsymbol{z}}_t^\top\boldsymbol{A}_t^{-1}\tilde{\boldsymbol{z}}_t &= \sum_{t=1}^{T}\frac{q_t}{1 + q_t} \leq \sum_{t=1}^{T}\log(1 + q_t) \\ &= \log\frac{\det\boldsymbol{A}_T}{\det\boldsymbol{A}_0} + dT\log(1/\gamma). \end{aligned} \tag{15}$$

Next,

$$\boldsymbol{A}_T = \gamma^T\delta\boldsymbol{I} + \sum_{s=1}^{T}\gamma^{T-s}\tilde{\boldsymbol{z}}_s\tilde{\boldsymbol{z}}_s^\top \preceq \left(\delta + B_z^2\sum_{j=0}^{T-1}\gamma^j\right)\boldsymbol{I} \preceq (\delta + B_z^2 h)\boldsymbol{I}.$$

Therefore

$$\log \frac{\det \boldsymbol{A}_T}{\det \boldsymbol{A}_0} \le d \log\left(1 + \frac{B_z^2 h}{\delta}\right).$$

Finally, since $h \ge 2$, $\gamma = 1 - 1/h \ge 1/2$, and

$$\log(1/\gamma) = -\log(1 - 1/h) \le \frac{2}{h}.$$

Substituting into (15) proves (13). □

Remark A.1 (Origin of the T/h term)**.** *When $\gamma = 1$, the second term in* (15) *vanishes and one recovers the classical log-determinant control. Forgetting contributes $dT \log(1/\gamma) \approx dT/h$. This is the explicit cost of discarding past curvature at a finite memory scale.*

A.5 Dynamic regret of a single forgotten-RLS expert

Theorem A.3 (Dynamic regret of forgotten RLS)**.** *Assume (B1)–(B4), let $h = (1-\gamma)^{-1} \ge 2$, and let $\boldsymbol{u}_{1:T}$ be any comparator path satisfying (B3). Then*

$$L_T(\check{y}^{(h)}) - L_T(\boldsymbol{u}_{1:T}) \le \delta R^2 + D^2 d \log\left(1 + \frac{B_z^2 h}{\delta}\right) + \frac{2D^2 dT}{h} + 4R(\delta + B_z^2 h) P_T(\boldsymbol{u}_{1:T}). \quad (16)$$

In particular, if $\delta \le B_z^2 h$, then

$$L_T(\check{y}^{(h)}) - L_T(\boldsymbol{u}_{1:T}) \le \delta R^2 + D^2 d \log\left(1 + \frac{B_z^2 h}{\delta}\right) + \frac{2D^2 dT}{h} + 8RB_z^2 h\, P_T(\boldsymbol{u}_{1:T}). \quad (17)$$

Ignoring logarithmic and initialization terms, the h-dependent part is

$$O\left(\frac{dT}{h} + hP_T(\boldsymbol{u}_{1:T})\right).$$

Proof. Apply Lemma A.1 with $\boldsymbol{u} = \boldsymbol{u}_t$ and sum over t:

$$\begin{aligned} L_T(\check{y}^{(h)}) - L_T(\boldsymbol{u}_{1:T}) &= \sum_{t=1}^{T} \big(\ell_t(\boldsymbol{w}_t) - \ell_t(\boldsymbol{u}_t)\big) \\ &\le \underbrace{\sum_{t=1}^{T} \Big[\gamma \|\boldsymbol{w}_t - \boldsymbol{u}_t\|^2_{\boldsymbol{A}_{t-1}} - \|\boldsymbol{w}_{t+1} - \boldsymbol{u}_t\|^2_{\boldsymbol{A}_t}\Big]}_{S_1} + \underbrace{\frac{1}{4}\sum_{t=1}^{T} \boldsymbol{g}_t^\top \boldsymbol{A}_t^{-1} \boldsymbol{g}_t}_{S_2}. \end{aligned} \quad (18)$$

Bounding S_2. By (B1), (B2), and (B4),

$$|\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t| \le D, \qquad \boldsymbol{g}_t = 2(\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t)\tilde{\boldsymbol{z}}_t.$$

Hence

$$\boldsymbol{g}_t^\top \boldsymbol{A}_t^{-1} \boldsymbol{g}_t \le 4D^2 \tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t^{-1} \tilde{\boldsymbol{z}}_t.$$

Using Lemma A.2,

$$S_2 \le D^2 \sum_{t=1}^{T} \tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t^{-1} \tilde{\boldsymbol{z}}_t \le D^2 d \log\left(1 + \frac{B_z^2 h}{\delta}\right) + \frac{2D^2 dT}{h}. \quad (19)$$

Bounding S_1. Shift the second sum in S_1:

$$\begin{aligned} S_1 = \gamma \|\boldsymbol{w}_1 - \boldsymbol{u}_1\|^2_{\boldsymbol{A}_0} - \|\boldsymbol{w}_{T+1} - \boldsymbol{u}_T\|^2_{\boldsymbol{A}_T} \\ + \sum_{t=2}^{T} \Big[\gamma \|\boldsymbol{w}_t - \boldsymbol{u}_t\|^2_{\boldsymbol{A}_{t-1}} - \|\boldsymbol{w}_t - \boldsymbol{u}_{t-1}\|^2_{\boldsymbol{A}_{t-1}}\Big]. \end{aligned} \quad (20)$$

The terminal term is non-positive and can be dropped. Since $\boldsymbol{w}_1 = \mathbf{0}$, $\boldsymbol{A}_0 = \delta \boldsymbol{I}$, $\gamma \le 1$, and $\|\boldsymbol{u}_1\| \le R$,

$$\gamma \|\boldsymbol{w}_1 - \boldsymbol{u}_1\|^2_{\boldsymbol{A}_0} \le \delta R^2.$$

For the bracketed term, let

$$\boldsymbol{x}_t := \boldsymbol{w}_t - \boldsymbol{u}_t, \qquad \boldsymbol{\Delta}_t := \boldsymbol{u}_t - \boldsymbol{u}_{t-1}.$$

Then $\boldsymbol{w}_t - \boldsymbol{u}_{t-1} = \boldsymbol{x}_t + \boldsymbol{\Delta}_t$, and

$$\begin{aligned}\gamma \|\boldsymbol{x}_t\|^2_{\boldsymbol{A}_{t-1}} - \|\boldsymbol{x}_t + \boldsymbol{\Delta}_t\|^2_{\boldsymbol{A}_{t-1}} &= -(1-\gamma)\|\boldsymbol{x}_t\|^2_{\boldsymbol{A}_{t-1}} - 2\boldsymbol{\Delta}_t^\top \boldsymbol{A}_{t-1}\boldsymbol{x}_t - \|\boldsymbol{\Delta}_t\|^2_{\boldsymbol{A}_{t-1}} \\ &\le 2|\boldsymbol{\Delta}_t^\top \boldsymbol{A}_{t-1}\boldsymbol{x}_t|. \end{aligned} \tag{21}$$

Moreover,

$$\boldsymbol{A}_{t-1} = \gamma^{t-1}\delta \boldsymbol{I} + \sum_{s=1}^{t-1} \gamma^{t-1-s} \tilde{\boldsymbol{z}}_s \tilde{\boldsymbol{z}}_s^\top \preceq (\delta + B_z^2 h)\boldsymbol{I}.$$

By (B3)–(B4), $\|\boldsymbol{x}_t\| \le 2R$. Therefore

$$2|\boldsymbol{\Delta}_t^\top \boldsymbol{A}_{t-1}\boldsymbol{x}_t| \le 2\|\boldsymbol{\Delta}_t\| \, \|\boldsymbol{A}_{t-1}\|_{\mathrm{op}} \|\boldsymbol{x}_t\| \le 4R(\delta + B_z^2 h)\|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|.$$

Summing (21) over $t = 2, \dots, T$,

$$S_1 \le \delta R^2 + 4R(\delta + B_z^2 h) P_T(\boldsymbol{u}_{1:T}).$$

Combining this with (19) proves (16). If $\delta \le B_z^2 h$, then $\delta + B_z^2 h \le 2B_z^2 h$, giving (17). □

Remark A.2 (About bounded iterates and projection)**.** *The exact identity* (8) *is what makes Lemma A.1 possible, and a Euclidean projection after the RLS update would generally destroy it. An $\boldsymbol{A}_t$-metric projection, as in online Newton methods, can be analysed, but the implemented recursion is not projected. We therefore state the deterministic bound under the bounded-iterate condition (B4). The same proof goes through under the weaker bounds $|\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t| \le D$ and $\|\boldsymbol{w}_t - \boldsymbol{u}_t\| \le 2R$ for the comparator class under consideration.*

A.6 Extension to the implemented covariance-inflated recursion

Theorem A.3 analyses the idealised recursion (7). The implementation runs (3), which adds $\varepsilon \boldsymbol{I}$ to the covariance $\boldsymbol{P}_t = \boldsymbol{A}_t^{-1}$. This subsection closes that gap: we show the entire proof architecture of Appendices A.3–A.5 survives verbatim once the two matrices that the inflated recursion distinguishes are kept separate, and that the only place new work is required is the log-determinant inequality. Throughout, $\varepsilon > 0$ is fixed and we write $\varepsilon = \varepsilon_k$ for the expert under consideration.

The two precision matrices. Let $\boldsymbol{P}_{t-1}$ be the stored covariance entering step t in Algorithm 1, with $\boldsymbol{P}_0 = \delta^{-1}\boldsymbol{I}$, and set

$$\tilde{\boldsymbol{A}}_{t-1} := \boldsymbol{P}_{t-1}^{-1}, \qquad \tilde{\boldsymbol{A}}_0 = \delta \boldsymbol{I}.$$

Define the *update precision*

$$\boldsymbol{A}'_t \;:=\; \gamma \tilde{\boldsymbol{A}}_{t-1} + \tilde{\boldsymbol{z}}_t \tilde{\boldsymbol{z}}_t^\top. \tag{22}$$

A direct computation identifies every line of the implemented update with $\boldsymbol{A}'_t$: the gain and the pre-inflation covariance are

$$\boldsymbol{K}_t = \frac{\boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t}{\gamma + \tilde{\boldsymbol{z}}_t^\top \boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t} = \boldsymbol{A}_t'^{-1}\tilde{\boldsymbol{z}}_t, \qquad \boldsymbol{P}_{t|t} = \gamma^{-1}\Big(\boldsymbol{P}_{t-1} - \frac{\boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top \boldsymbol{P}_{t-1}}{\gamma + \tilde{\boldsymbol{z}}_t^\top \boldsymbol{P}_{t-1}\tilde{\boldsymbol{z}}_t}\Big) = \boldsymbol{A}_t'^{-1}, \tag{23}$$

the second equality being the Sherman–Morrison identity applied to (22). The inflation step then produces the matrix carried to the next step,

$$\boldsymbol{P}_t = \boldsymbol{A}_t'^{-1} + \varepsilon \boldsymbol{I}, \qquad \text{i.e.} \qquad \tilde{\boldsymbol{A}}_t = \big(\boldsymbol{A}_t'^{-1} + \varepsilon \boldsymbol{I}\big)^{-1}. \tag{24}$$

Thus the parameter update uses $\boldsymbol{A}'_t$, while the potential function and the carry-over to step $t+1$ use $\tilde{\boldsymbol{A}}_t$; the two differ at order ε and must not be interchanged. With $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \boldsymbol{K}_t(y_t - \tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t)$, (23) gives the analogue of (8),

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \tfrac{1}{2}\boldsymbol{A}_t'^{-1}\boldsymbol{g}_t, \qquad \boldsymbol{g}_t = 2(\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t)\tilde{\boldsymbol{z}}_t, \qquad \text{equivalently} \qquad \boldsymbol{g}_t = 2\boldsymbol{A}'_t(\boldsymbol{w}_t - \boldsymbol{w}_{t+1}). \tag{25}$$

Two elementary consequences of inflating the covariance. Because $\boldsymbol{A}_t'^{-1}+\varepsilon\boldsymbol{I} \succeq \boldsymbol{A}_t'^{-1}$, inverting reverses the order and

$$\tilde{\boldsymbol{A}}_t \preceq \boldsymbol{A}_t'. \tag{26}$$

Moreover $\boldsymbol{A}_t'^{-1} + \varepsilon\boldsymbol{I} \succeq \varepsilon\boldsymbol{I}$ gives a new upper bound, absent for the idealised recursion,

$$\tilde{\boldsymbol{A}}_t \preceq \varepsilon^{-1}\boldsymbol{I}. \tag{27}$$

Inflating $\boldsymbol{P}$ therefore truncates the spectrum of the precision matrix from above at $1/\varepsilon$: it caps overconfident precision directions, equivalently it floors the effective step size. A ridge term $\boldsymbol{A}_t + \varepsilon\boldsymbol{I}$ acts at the other end and bounds $\lambda_{\min}$ from below. The cap gives no lower bound on $\lambda_{\min}(\boldsymbol{A}_t')$, so (27) does not discharge (B4).

Boundedness assumptions. We keep (B1)–(B3) and replace (B4) by its exact analogue for the implemented recursion:

(B4$^{\varepsilon}$) The iterates of the inflated recursion (3) satisfy $\|\boldsymbol{w}_t\|_2 \le R_\varepsilon$ for all t, and we set $D_\varepsilon := B_y + B_z R_\varepsilon$.

As in the idealised case, the proof only ever uses $|\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t| \le D_\varepsilon$ and $\|\boldsymbol{w}_t - \boldsymbol{u}_t\|_2 \le R + R_\varepsilon$; we write $2R$ for the latter when $R_\varepsilon \le R$.

Spectral envelopes. Both matrices admit envelopes, and each has two available bounds, the second of which is new.

Lemma A.4 (Spectral envelopes under inflation)**.** *Assume (B1), let $h = (1-\gamma)^{-1}$, and suppose $\varepsilon\delta \le 1$. For all $t \ge 1$,*

$$\boldsymbol{A}_t' \preceq \Lambda_\varepsilon'\,\boldsymbol{I}, \quad \Lambda_\varepsilon' := \min\{\delta + B_z^2 h,\ \gamma\varepsilon^{-1} + B_z^2\}, \qquad \tilde{\boldsymbol{A}}_t \preceq \Lambda_\varepsilon\,\boldsymbol{I}, \quad \Lambda_\varepsilon := \min\{\delta + B_z^2 h,\ \varepsilon^{-1}\}, \tag{28}$$

and, in trace,

$$\operatorname{tr}(\boldsymbol{A}_t') \ \le\ \gamma^t \delta d + B_z^2 \sum_{j=0}^{t-1} \gamma^j \ \le\ \gamma^t \delta d + B_z^2 h. \tag{29}$$

The bound $\tilde{\boldsymbol{A}}_t \preceq \Lambda_\varepsilon \boldsymbol{I}$ and the first entry of each minimum hold without the condition $\varepsilon\delta \le 1$; the condition is needed only for the entry $\gamma\varepsilon^{-1} + B_z^2$ at $t = 1$, and under the deployed schedule (51) *it holds with large margin.*

Proof. For the first entry of each minimum, combine (26) with induction: $\lambda_{\max}(\boldsymbol{A}_1') \le \gamma\delta + B_z^2$, and if $\lambda_{\max}(\boldsymbol{A}_{t-1}') \le \gamma^{t-1}\delta + B_z^2 \sum_{j=0}^{t-2}\gamma^j$ then $\tilde{\boldsymbol{A}}_{t-1} \preceq \boldsymbol{A}_{t-1}'$ gives $\lambda_{\max}(\boldsymbol{A}_t') \le \gamma\lambda_{\max}(\boldsymbol{A}_{t-1}') + B_z^2 \le \gamma^t\delta + B_z^2 \sum_{j=0}^{t-1}\gamma^j \le \delta + B_z^2 h$; the same bound for $\tilde{\boldsymbol{A}}_t$ follows from (26). For the second entry, (27) gives $\tilde{\boldsymbol{A}}_{t-1} \preceq \varepsilon^{-1}\boldsymbol{I}$ for $t \ge 2$, while for $t = 1$ the same bound $\tilde{\boldsymbol{A}}_0 = \delta\boldsymbol{I} \preceq \varepsilon^{-1}\boldsymbol{I}$ is exactly the condition $\varepsilon\delta \le 1$; substituting into (22) yields $\boldsymbol{A}_t' \preceq (\gamma\varepsilon^{-1} + B_z^2)\boldsymbol{I}$ for all $t \ge 1$. The trace bound (29) follows by the same induction, since $\operatorname{tr}(\tilde{\boldsymbol{A}}_{t-1}) \le \operatorname{tr}(\boldsymbol{A}_{t-1}')$ by (26) and $\operatorname{tr}(\tilde{\boldsymbol{z}}_t\tilde{\boldsymbol{z}}_t^\top) = \|\tilde{\boldsymbol{z}}_t\|_2^2 \le B_z^2$. □

Both entries of Λ_ε' grow linearly in h. The shrinkage map admits a sharper stationary envelope: the scalar recursion dominating $\lambda_{\max}(\boldsymbol{A}_t')$ has a unique positive fixed point, which is an explicit root of a quadratic.

Lemma A.5 (Steady-state envelope)**.** *Assume (B1) and $\varepsilon > 0$, and let*

$$a_\varepsilon := \frac{-(1-\gamma-\varepsilon B_z^2) + \sqrt{(1-\gamma-\varepsilon B_z^2)^2 + 4\varepsilon B_z^2}}{2\varepsilon} \tag{30}$$

be the unique positive solution of $a = \gamma a/(1+\varepsilon a) + B_z^2$. Then, for all $t \ge 1$,

$$\boldsymbol{A}_t' \preceq \Lambda_{\varepsilon,\sharp}'\,\boldsymbol{I}, \qquad \tilde{\boldsymbol{A}}_t \preceq \min\{\Lambda_{\varepsilon,\sharp}', \varepsilon^{-1}\}\,\boldsymbol{I}, \qquad \Lambda_{\varepsilon,\sharp}' := \max\{\gamma\delta + B_z^2,\ a_\varepsilon\}. \tag{31}$$

If moreover $\varepsilon\delta \le 1$, then $\Lambda_{\varepsilon,\sharp}' \le \Lambda_\varepsilon'$, and Λ_ε', Λ_ε may be replaced by $\Lambda_{\varepsilon,\sharp}'$, $\min\{\Lambda_{\varepsilon,\sharp}', \varepsilon^{-1}\}$ throughout Lemma A.7 and Theorem A.8, since those results use the envelopes only as upper bounds on $\lambda_{\max}(\boldsymbol{A}_t')$ and $\lambda_{\max}(\tilde{\boldsymbol{A}}_t)$.

Proof. Write $f(x) = x/(1+\varepsilon x)$ and $g(u) = \gamma f(u) + B_z^2$, both increasing on $[0,\infty)$. Since $\tilde{\boldsymbol{A}}_t$ is the spectral function f applied to $\boldsymbol{A}'_t$, (24) gives $\lambda_{\max}(\tilde{\boldsymbol{A}}_t) = f(\lambda_{\max}(\boldsymbol{A}'_t))$, and (22) gives, for $t \geq 2$, $\lambda_{\max}(\boldsymbol{A}'_t) \leq \gamma\lambda_{\max}(\tilde{\boldsymbol{A}}_{t-1}) + B_z^2 = g(\lambda_{\max}(\boldsymbol{A}'_{t-1}))$. Clearing denominators, $g(a) = a$ is the quadratic $\varepsilon a^2 + (1-\gamma-\varepsilon B_z^2)a - B_z^2 = 0$; the product of its roots is $-B_z^2/\varepsilon < 0$, so there is a unique positive fixed point, given by (30). The map $u \mapsto g(u) - u$ has derivative $\gamma/(1+\varepsilon u)^2 - 1 < 0$, so it is strictly decreasing and $g(u) \leq u$ for all $u \geq a_\varepsilon$; in particular $g(m) \leq m$ for $m := \Lambda'_{\varepsilon,\sharp}$. Induction now gives $\lambda_{\max}(\boldsymbol{A}'_t) \leq m$: the base case is $\lambda_{\max}(\boldsymbol{A}'_1) \leq \gamma\delta + B_z^2 \leq m$, and if $\lambda_{\max}(\boldsymbol{A}'_{t-1}) \leq m$ then $\lambda_{\max}(\boldsymbol{A}'_t) \leq g(m) \leq m$. The envelope for $\tilde{\boldsymbol{A}}_t$ follows from $f(x) \leq \min\{x, \varepsilon^{-1}\}$. For the domination: the fixed-point equation gives $a_\varepsilon(1-\gamma) \leq B_z^2$, i.e. $a_\varepsilon \leq B_z^2 h$, and $f \leq \varepsilon^{-1}$ gives $a_\varepsilon \leq \gamma\varepsilon^{-1} + B_z^2$; combining with $\gamma\delta + B_z^2 \leq \delta + B_z^2 h$ and, under $\varepsilon\delta \leq 1$, $\gamma\delta + B_z^2 \leq \gamma\varepsilon^{-1} + B_z^2$, yields $\Lambda'_{\varepsilon,\sharp} \leq \Lambda'_\varepsilon$. □

Remark A.3 (The steady-state envelope under the deployed schedule)**.** *Under* (51) *with* $\alpha = 1$*, substituting* $\varepsilon = \varepsilon_0/h$ *and* $1-\gamma = 1/h$ *into the quadratic and multiplying through by* h *gives the exact coupled form*

$$a_\varepsilon = \frac{-(1-\varepsilon_0 B_z^2) + \sqrt{(1-\varepsilon_0 B_z^2)^2 + 4\varepsilon_0 B_z^2 h}}{2\varepsilon_0} = B_z\sqrt{h/\varepsilon_0}\,\big(1+o(1)\big) \qquad (h \to \infty),$$

so the tracking constant in Theorem A.8 improves from $O(h)$ *to* $O(\sqrt{h})$ *at the long-memory end of the grid. The* $\sqrt{h}$ *regime is entered once* $4\varepsilon_0 B_z^2 h \gtrsim (1-\varepsilon_0 B_z^2)^2$*: at moderate* h *when* $\varepsilon_0 B_z^2 \gtrsim 1$*, beyond practical grids when* $\varepsilon_0 B_z^2 \ll 1$*. As in Remark A.6, the sharper tracking rate is available only where the precision cap is active.*

A.6.1 The one-step identity is unchanged

Lemma A.6 (One-step identity under covariance inflation)**.** *For every* $\boldsymbol{u} \in \mathbb{R}^d$ *and every* $t \geq 1$*, the iterates of* (3) *satisfy*

$$\ell_t(\boldsymbol{w}_t) - \ell_t(\boldsymbol{u}) = \gamma\|\boldsymbol{w}_t - \boldsymbol{u}\|^2_{\tilde{\boldsymbol{A}}_{t-1}} - \|\boldsymbol{w}_{t+1} - \boldsymbol{u}\|^2_{\boldsymbol{A}'_t} + \frac{1}{4}\boldsymbol{g}_t^\top \boldsymbol{A}_t'^{-1}\boldsymbol{g}_t, \tag{32}$$

and consequently, by (26)*,*

$$\ell_t(\boldsymbol{w}_t) - \ell_t(\boldsymbol{u}) \leq \gamma\|\boldsymbol{w}_t - \boldsymbol{u}\|^2_{\tilde{\boldsymbol{A}}_{t-1}} - \|\boldsymbol{w}_{t+1} - \boldsymbol{u}\|^2_{\tilde{\boldsymbol{A}}_t} + \frac{1}{4}\boldsymbol{g}_t^\top \boldsymbol{A}_t'^{-1}\boldsymbol{g}_t. \tag{33}$$

Proof. The proof of Lemma A.1 applies verbatim with $\boldsymbol{A}_t$ replaced by $\boldsymbol{A}'_t$ and $\boldsymbol{A}_{t-1}$ by $\tilde{\boldsymbol{A}}_{t-1}$. Indeed, (11) is an identity for squared loss and does not involve the recursion. The polarisation step (12) uses only the Newton form, which holds in the form (25). Finally, the definition (22) of $\boldsymbol{A}'_t$ gives the quadratic-form decomposition

$$\|\boldsymbol{w}_t - \boldsymbol{u}\|^2_{\boldsymbol{A}'_t} = \gamma\|\boldsymbol{w}_t - \boldsymbol{u}\|^2_{\tilde{\boldsymbol{A}}_{t-1}} + \big(\tilde{\boldsymbol{z}}_t^\top(\boldsymbol{w}_t - \boldsymbol{u})\big)^2,$$

which cancels the curvature term $\big(\tilde{\boldsymbol{z}}_t^\top(\boldsymbol{w}_t - \boldsymbol{u})\big)^2$ exactly as before. Note the decomposition holds because $\boldsymbol{A}'_t$ is built from $\tilde{\boldsymbol{A}}_{t-1}$ by the unmodified forgetting-plus-rank-one step; the inflation acts only afterwards, in (24). Inequality (33) then follows since (26) implies $-\|\boldsymbol{\varepsilon}\|^2_{\boldsymbol{A}'_t} \leq -\|\boldsymbol{\varepsilon}\|^2_{\tilde{\boldsymbol{A}}_t}$ for every $\boldsymbol{\varepsilon}$. □

Inequality (33) is the structural reason the extension goes through: with the potential $\Phi_t(\boldsymbol{u}) = \|\boldsymbol{w}_{t+1} - \boldsymbol{u}\|^2_{\tilde{\boldsymbol{A}}_t}$, the inflation step is free: it only decreases the subtracted potential and so weakens the inequality in the admissible direction.

A.6.2 Log-determinant inequality with a shrinkage step in the chain

This is the one ingredient that genuinely changes. Lemma A.2 telescopes $\log\det$ along a chain in which the matrix leaving step t is the matrix entering step $t+1$. Under (3) the chain is broken: $\boldsymbol{A}'_t$ is produced but $\tilde{\boldsymbol{A}}_t \preceq \boldsymbol{A}'_t$ is carried forward. The determinant lost at each splice is what the next lemma accounts for.

Lemma A.7 (Log-determinant inequality under covariance inflation)**.** *Let $\boldsymbol{A}'_t$ and $\tilde{\boldsymbol{A}}_t$ be as in* (22)–(24) *with $\tilde{\boldsymbol{A}}_0 = \delta \boldsymbol{I}$, $\delta > 0$, $\gamma = 1 - 1/h$, and $h \geq 2$. Under (B1) and $\varepsilon\delta \leq 1$,*

$$\sum_{t=1}^{T} \tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t'^{-1} \tilde{\boldsymbol{z}}_t \;\leq\; d \log\left(\frac{\Lambda'_\varepsilon}{\delta}\right) + \frac{2dT}{h} + \underbrace{\min\Big\{(T-1)\, d \log\big(1+\varepsilon\Lambda'_\varepsilon\big),\; \varepsilon h\big(\gamma\delta d + B_z^2(T-1)\big)\Big\}}_{\textit{inflation leakage}}, \tag{34}$$

with Λ'_ε as in (28)*. The two entries of the minimum are complementary: the second is smaller by up to a factor d when $\varepsilon\Lambda'_\varepsilon \leq 1$, the first grows only logarithmically in ε when $\varepsilon\Lambda'_\varepsilon \gg 1$.*

Proof. Put $q_t := \gamma^{-1} \tilde{\boldsymbol{z}}_t^\top \tilde{\boldsymbol{A}}_{t-1}^{-1} \tilde{\boldsymbol{z}}_t \geq 0$. Applying Sherman–Morrison to (22) and contracting with $\tilde{\boldsymbol{z}}_t$ gives, exactly as in (14),

$$\tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t'^{-1} \tilde{\boldsymbol{z}}_t = \frac{q_t}{1+q_t},$$

while the matrix-determinant lemma gives $\det \boldsymbol{A}'_t = \gamma^d \det(\tilde{\boldsymbol{A}}_{t-1})(1+q_t)$, i.e.

$$\log(1+q_t) = \log\det \boldsymbol{A}'_t - \log\det \tilde{\boldsymbol{A}}_{t-1} + d\log(1/\gamma).$$

Using $q/(1+q) \leq \log(1+q)$ and summing,

$$\sum_{t=1}^{T} \tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t'^{-1} \tilde{\boldsymbol{z}}_t \leq \sum_{t=1}^{T} \big[\log\det \boldsymbol{A}'_t - \log\det \tilde{\boldsymbol{A}}_{t-1}\big] + dT\log(1/\gamma). \tag{35}$$

The sum in (35) is not telescoping, because $\tilde{\boldsymbol{A}}_t \neq \boldsymbol{A}'_t$. Re-grouping it as a telescoping part plus the per-step splices,

$$\sum_{t=1}^{T} \big[\log\det \boldsymbol{A}'_t - \log\det \tilde{\boldsymbol{A}}_{t-1}\big] = \log\det \boldsymbol{A}'_T - \log\det \tilde{\boldsymbol{A}}_0 + \sum_{t=1}^{T-1} \big[\log\det \boldsymbol{A}'_t - \log\det \tilde{\boldsymbol{A}}_t\big].$$

By (24) each splice is computable exactly:

$$\log\det \boldsymbol{A}'_t - \log\det \tilde{\boldsymbol{A}}_t = \log\det\big(\boldsymbol{A}'_t(\boldsymbol{A}_t'^{-1} + \varepsilon \boldsymbol{I})\big) = \log\det\big(\boldsymbol{I} + \varepsilon \boldsymbol{A}'_t\big) \leq d\log\big(1+\varepsilon\Lambda'_\varepsilon\big),$$

the last step by Lemma A.4. Alternatively, since $\log\det(\boldsymbol{I}+\varepsilon\boldsymbol{A}) = \sum_i \log(1+\varepsilon\lambda_i(\boldsymbol{A})) \leq \varepsilon \operatorname{tr}(\boldsymbol{A})$, the trace envelope (29) bounds the total splice cost by

$$\sum_{t=1}^{T-1} \log\det\big(\boldsymbol{I}+\varepsilon\boldsymbol{A}'_t\big) \leq \varepsilon \sum_{t=1}^{T-1} \big(\gamma^t \delta d + B_z^2 h\big) \leq \varepsilon\big(\gamma h\, \delta d + B_z^2 h(T-1)\big) = \varepsilon h\big(\gamma\delta d + B_z^2(T-1)\big),$$

using $\sum_{t\geq 1} \gamma^t = \gamma h$. Taking the smaller of the two bounds gives the leakage term in (34). For the telescoped endpoints, Lemma A.4 gives $\log\det \boldsymbol{A}'_T \leq d\log\Lambda'_\varepsilon$ and $\log\det \tilde{\boldsymbol{A}}_0 = d\log\delta$, so $\log\det \boldsymbol{A}'_T - \log\det \tilde{\boldsymbol{A}}_0 \leq d\log(\Lambda'_\varepsilon/\delta)$. Finally $h \geq 2$ gives $\log(1/\gamma) = -\log(1-1/h) \leq 2/h$. Substituting the three displays into (35) proves (34). □

Remark A.4 (Inflation as excess forgetting)**.** *The leakage term admits an equivalent and more interpretable reading. Since $\tilde{\boldsymbol{A}}_t = \boldsymbol{A}'_t(\boldsymbol{I}+\varepsilon\boldsymbol{A}'_t)^{-1} \succeq \boldsymbol{A}'_t/(1+\varepsilon\Lambda'_\varepsilon)$ by Lemma A.4, the recursion* (22) *obeys*

$$\boldsymbol{A}'_t \;\succeq\; \gamma_\varepsilon \boldsymbol{A}'_{t-1} + \tilde{\boldsymbol{z}}_t \tilde{\boldsymbol{z}}_t^\top, \qquad \gamma_\varepsilon := \frac{\gamma}{1+\varepsilon\Lambda'_\varepsilon} \leq \gamma,$$

so $\boldsymbol{A}'_t \succeq \boldsymbol{\Sigma}_t$ where $\boldsymbol{\Sigma}_t = \gamma_\varepsilon \boldsymbol{\Sigma}_{t-1} + \tilde{\boldsymbol{z}}_t \tilde{\boldsymbol{z}}_t^\top$, $\boldsymbol{\Sigma}_0 = \delta \boldsymbol{I}$, is an idealised forgotten-RLS precision at the shortened factor γ_ε. Hence $\tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t'^{-1} \tilde{\boldsymbol{z}}_t \leq \tilde{\boldsymbol{z}}_t^\top \boldsymbol{\Sigma}_t^{-1} \tilde{\boldsymbol{z}}_t$, and Lemma A.2 applies to $\boldsymbol{\Sigma}_t$ as a black box (provided $\gamma_\varepsilon \geq 1/2$, i.e. $h_\varepsilon \geq 2$, as that lemma requires), giving (34) *with $dT\log(1/\gamma_\varepsilon) = dT\log(1/\gamma) + dT\log(1+\varepsilon\Lambda'_\varepsilon)$ in place of the last two terms. To leading order, covariance inflation therefore runs the idealised recursion at a shorter effective memory $h_\varepsilon = (1-\gamma_\varepsilon)^{-1} \leq h$. This makes precise the statement in Appendix B.4 that too much inflation shortens the memory of a slow expert, and it is the deterministic counterpart of the Kalman reading given there, in which $\varepsilon\boldsymbol{I}$ is process noise that lets the coefficients move at each step.*

A.6.3 DYNAMIC REGRET OF THE IMPLEMENTED RECURSION

Theorem A.8 (Dynamic regret under covariance inflation)**.** *Assume (B1)–(B3) and (B4$^\varepsilon$) with $R_\varepsilon \le R$, let $h = (1-\gamma)^{-1} \ge 2$, let $\varepsilon > 0$ with $\varepsilon\delta \le 1$, and let $\boldsymbol{u}_{1:T}$ be any comparator path satisfying (B3). Then the predictions $\check{y}_t^{(h,\varepsilon)} = \tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t$ of the implemented recursion* (3) *satisfy*

$$L_T(\check{y}^{(h,\varepsilon)}) - L_T(\boldsymbol{u}_{1:T}) \;\le\; \Psi_h^\varepsilon(T, P_T(\boldsymbol{u}_{1:T})), \tag{36}$$

where, with $\Lambda_\varepsilon, \Lambda'_\varepsilon$ as in (28) *and $D_\varepsilon = B_y + B_z R_\varepsilon$,*

$$\Psi_h^\varepsilon(T,P) = \underbrace{\delta R^2}_{\text{initialisation}} + \underbrace{D_\varepsilon^2 d \log\!\left(\frac{\Lambda'_\varepsilon}{\delta}\right)}_{\text{log term}} + \underbrace{\frac{2D_\varepsilon^2 dT}{h} + D_\varepsilon^2\,\mathcal{E}_\varepsilon(T)}_{\text{finite-memory term}} + \underbrace{4R\Lambda_\varepsilon P}_{\text{tracking term}}, \tag{37}$$

$$\mathcal{E}_\varepsilon(T) := \min\Big\{ dT\log\big(1+\varepsilon\Lambda'_\varepsilon\big),\; \varepsilon h\big(\delta d + B_z^2 T\big)\Big\}. \tag{38}$$

If $R_\varepsilon > R$, the statement holds with $4R\Lambda_\varepsilon$ replaced by $2(R+R_\varepsilon)\Lambda_\varepsilon$ in the tracking term. With the first entry of the minimum in (38)*, the two finite-memory summands may equivalently be written, by Remark A.4, as the single term $D_\varepsilon^2 dT\log(1/\gamma_\varepsilon)$ at the shortened effective scale h_ε. As $\varepsilon \downarrow 0$ each of the four groups converges to the corresponding group of* (6) *with $D_\varepsilon \to D$.*

Proof. Apply Lemma A.6 in the form (33) with $\boldsymbol{u} = \boldsymbol{u}_t$ and sum over t:

$$L_T(\check{y}^{(h,\varepsilon)}) - L_T(\boldsymbol{u}_{1:T}) \le \underbrace{\sum_{t=1}^{T}\Big[\gamma\|\boldsymbol{w}_t - \boldsymbol{u}_t\|^2_{\tilde{\boldsymbol{A}}_{t-1}} - \|\boldsymbol{w}_{t+1} - \boldsymbol{u}_t\|^2_{\tilde{\boldsymbol{A}}_t}\Big]}_{S_1^\varepsilon} + \underbrace{\frac{1}{4}\sum_{t=1}^{T}\boldsymbol{g}_t^\top \boldsymbol{A}_t'^{-1}\boldsymbol{g}_t}_{S_2^\varepsilon}.$$

Bounding S_2^ε. By (B1), (B2) and (B4$^\varepsilon$), $|\tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t - y_t| \le D_\varepsilon$, so $\boldsymbol{g}_t^\top \boldsymbol{A}_t'^{-1}\boldsymbol{g}_t \le 4D_\varepsilon^2\,\tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t'^{-1}\tilde{\boldsymbol{z}}_t$. Lemma A.7 then gives

$$S_2^\varepsilon \le D_\varepsilon^2 \sum_{t=1}^{T}\tilde{\boldsymbol{z}}_t^\top \boldsymbol{A}_t'^{-1}\tilde{\boldsymbol{z}}_t \le D_\varepsilon^2 d\log\!\left(\frac{\Lambda'_\varepsilon}{\delta}\right) + \frac{2D_\varepsilon^2 dT}{h} + D_\varepsilon^2\,\mathcal{E}_\varepsilon(T),$$

where we enlarged $T-1$ to T and $\gamma\delta d$ to δd in both entries of the leakage minimum.

Bounding S_1^ε. The sum in S_1^ε is telescoping in the potential $\Phi_t(\boldsymbol{u}) = \|\boldsymbol{w}_{t+1} - \boldsymbol{u}\|^2_{\tilde{\boldsymbol{A}}_t}$, because both the subtracted term at step t and the added term at step $t+1$ use the same matrix $\tilde{\boldsymbol{A}}_t$. Shifting the second sum exactly as in (20),

$$S_1^\varepsilon = \gamma\|\boldsymbol{w}_1 - \boldsymbol{u}_1\|^2_{\tilde{\boldsymbol{A}}_0} - \|\boldsymbol{w}_{T+1} - \boldsymbol{u}_T\|^2_{\tilde{\boldsymbol{A}}_T} + \sum_{t=2}^{T}\Big[\gamma\|\boldsymbol{w}_t - \boldsymbol{u}_t\|^2_{\tilde{\boldsymbol{A}}_{t-1}} - \|\boldsymbol{w}_t - \boldsymbol{u}_{t-1}\|^2_{\tilde{\boldsymbol{A}}_{t-1}}\Big].$$

The terminal term is non-positive and is dropped. Since $\boldsymbol{w}_1 = \boldsymbol{0}$, $\tilde{\boldsymbol{A}}_0 = \delta\boldsymbol{I}$, $\gamma \le 1$ and $\|\boldsymbol{u}_1\| \le R$, the first term is at most δR^2. For the bracket, writing $\boldsymbol{x}_t = \boldsymbol{w}_t - \boldsymbol{u}_t$ and $\boldsymbol{\Delta}_t = \boldsymbol{u}_t - \boldsymbol{u}_{t-1}$, the computation (21) is algebraically identical with $\boldsymbol{A}_{t-1}$ replaced by $\tilde{\boldsymbol{A}}_{t-1}$ and yields the bound $2|\boldsymbol{\Delta}_t^\top \tilde{\boldsymbol{A}}_{t-1}\boldsymbol{x}_t|$. By Lemma A.4, $\|\tilde{\boldsymbol{A}}_{t-1}\|_{\mathrm{op}} \le \Lambda_\varepsilon$, and by (B3) and (B4$^\varepsilon$), $\|\boldsymbol{x}_t\| \le 2R$; hence $2|\boldsymbol{\Delta}_t^\top \tilde{\boldsymbol{A}}_{t-1}\boldsymbol{x}_t| \le 4R\Lambda_\varepsilon\|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|$. Summing over $t = 2,\dots,T$ gives $S_1^\varepsilon \le \delta R^2 + 4R\Lambda_\varepsilon P_T(\boldsymbol{u}_{1:T})$. Adding the two bounds proves (36).

Degeneration as $\varepsilon \downarrow 0$. By (28), $\Lambda_\varepsilon \uparrow \delta + B_z^2 h$ and $\Lambda'_\varepsilon \uparrow \delta + B_z^2 h$, so the log term tends to $D^2 d\log(1 + B_z^2 h/\delta)$, the leakage summand tends to 0, and the tracking term tends to $4R(\delta + B_z^2 h)P$, recovering (6) group by group. □

Remark A.5 (Which terms ε improves and which it degrades)**.** *Comparing* (37) *with* (6) *term by term: the initialisation term is unchanged; the log term is never larger, since $\Lambda'_\varepsilon \le \delta + B_z^2 h$, and is strictly smaller when the cap $\gamma\varepsilon^{-1} + B_z^2$ is active; the tracking term is never larger and improves by the factor $\Lambda_\varepsilon/(\delta + B_z^2 h)$, which is $\varepsilon^{-1}/(\delta + B_z^2 h)$ when the cap binds; and the finite-memory term is the only one that degrades, by the leakage $D_\varepsilon^2\,\mathcal{E}_\varepsilon(T)$. Capping the precision from above limits how*

strongly the expert can lock onto a stale direction, which the tracking term measures, and discards curvature, which the finite-memory term measures. Separately, ε enters through the constant D_ε, which multiplies the log and finite-memory terms. On the RTE-FR stream the realised supremum of $\|\boldsymbol{w}_t\|_2$ is largest at $c = 1$ for every finite h (Appendix B.9), so inflation improves R_ε, although (B4$^\varepsilon$) remains an assumption.

Remark A.6 (The exponent $\alpha = 1$ and when the bound is informative)**.** *Under the deployed schedule* (51) *with $\alpha = 1$, i.e. $\varepsilon = \varepsilon_0/h$, the first entry of the minimum defining Λ'_ε gives*

$$\varepsilon\Lambda'_\varepsilon \;\le\; \frac{\varepsilon_0}{h}\big(\delta + B_z^2 h\big) \;=\; \varepsilon_0 B_z^2 + \frac{\varepsilon_0\delta}{h} \;\xrightarrow[h\to\infty]{}\; \varepsilon_0 B_z^2,$$

so the per-step leakage rate is asymptotically independent of γ. The second entry of (38) *makes this exact and non-asymptotic: $\varepsilon h = \varepsilon_0$ under $\alpha = 1$, so*

$$\mathcal{E}_\varepsilon(T) \;\le\; \varepsilon_0\big(\delta d + B_z^2 T\big) \qquad \textit{uniformly over the } \gamma\textit{-grid},$$

i.e. the implemented recursion incurs, on top of Ψ_h, at most the same additive overhead $\varepsilon_0 D_\varepsilon^2(\delta d + B_z^2 T)$ at every memory scale. This is a precise version of the scale-matching heuristic offered for $\alpha = 1$ in Appendix B.4: $\alpha = 1$ is the unique exponent for which the excess forgetting induced by inflation neither vanishes nor blows up along the grid. It also shows what the bound requires to be informative. The leakage is dominated by the idealised finite-memory term $2D^2 dT/h$ only when $\varepsilon_0 B_z^2 \lesssim 2d/h$ (second entry), a condition that is γ-free on its left side and is most demanding at the long-memory end of the grid. The bound is therefore stated for design vectors on an $O(1)$ scale. Which entry of the minimum in (28) *is active is governed by the same product: $\gamma\varepsilon^{-1} + B_z^2$ is the smaller entry when $\varepsilon_0 B_z^2 \gtrsim 1$, so ε tightens the log and tracking terms in the same regime in which it loosens the finite-memory term. The two effects cannot be obtained simultaneously, and no choice of ε_0 makes $\Psi_h^\varepsilon \le \Psi_h$ in every term at once.*

A.6.4 From single scales to the aggregate

Remark A.7 (Scope of the aggregation layer)**.** *Theorem A.8 concerns the single-scale EWLS experts. The aggregation layer itself is untouched by inflation: the point-mass case of Lemma A.11 requires only bounded losses, which clipping supplies, and is indifferent to how the expert predictions are generated. What does require attention is the statement of the oracle inequality: Theorem A.14 assumes the conditions of Theorem A.3, which concern the idealised recursion, for each $h \in \mathcal{H}$, so its Ψ_h branch does not, as written, apply to a pool generated by* (3)*. The following corollary closes this gap; its proof is verbatim that of Theorem A.14.*

Corollary A.9 (Oracle inequality for the implemented pool)**.** *Let each EWLS expert $h \in \mathcal{H}$ be generated by the implemented recursion* (3) *with inflation $\varepsilon_h > 0$, and assume, for each $h \in \mathcal{H}$, the conditions of Theorem A.8 at scale h with $\varepsilon = \varepsilon_h$. Assume outcomes lie in $[-B, B]$, and all predictions entering MLpol are clipped to $[-B, B]$. Then, writing f_j and $\check{y}^{(h,\varepsilon_h)}$ for the clipped raw and EWLS prediction sequences, for every comparator path $\boldsymbol{u}_{1:T}$ satisfying (B3),*

$$L_T(\hat{y}) \le \min\left\{ \min_{1\le j\le M} L_T(f_j),\; L_T(\boldsymbol{u}_{1:T}) + \min_{h\in\mathcal{H}} \Psi_h^{\varepsilon_h}\big(T, P_T(\boldsymbol{u}_{1:T})\big) \right\} + \Omega_{M+K,T}(B). \tag{39}$$

Under the deployed schedule (51)*, $\varepsilon_h = \varepsilon_0/h$.*

Proof. By the point-mass special case of Lemma A.11 applied to the enlarged pool, which uses only that all clipped losses lie in $[0, 4B^2]$ whatever produces the expert predictions,

$$L_T(\hat{y}) \le \min\left\{ \min_{1\le j\le M} L_T(f_j), \min_{h\in\mathcal{H}} L_T\big(\check{y}^{(h,\varepsilon_h)}\big) \right\} + \Omega_{M+K,T}(B).$$

By Lemma A.10, clipping cannot increase squared loss, so Theorem A.8 gives, for every $h \in \mathcal{H}$,

$$L_T\big(\check{y}^{(h,\varepsilon_h)}\big) \le L_T(\boldsymbol{u}_{1:T}) + \Psi_h^{\varepsilon_h}\big(T, P_T(\boldsymbol{u}_{1:T})\big).$$

Taking the minimum over $h \in \mathcal{H}$ proves (39). □

Remark A.8 (Geometric grids)**.** *Corollary A.15 extends in the same way. Its proof evaluates Ψ_h at a grid point h_k near $h^\star$ and controls only the leading terms $C_1 dT/h + C_2 hP$; the corresponding terms of $\Psi_h^{\varepsilon_h}$ are no larger, since $D_\varepsilon \le D$ under (B4$^\varepsilon$) with $R_\varepsilon \le R$, and $4R\Lambda_\varepsilon \le 4R(\delta + B_z^2 h) \le C_2 h$ for $\delta \le B_z^2 h$. The terms set aside in* (46) *then additionally include the inflation leakage, which under the deployed schedule is bounded by the h-free quantity $\varepsilon_0 D_\varepsilon^2(\delta d + B_z^2 T)$ and therefore does not affect the choice of grid point.*

Remark A.9 (The no-forgetting endpoint $\gamma = 1$)**.** *Theorem A.8 requires $h = (1-\gamma)^{-1}$ finite, but the deployed grid carries a $\gamma = 1$ expert, for which the schedule* (51) *sets $\varepsilon_h = 0$, and an uncoupled grid can pair $\gamma = 1$ with $\varepsilon > 0$. Both are covered by reading the proofs at the endpoint. For $\gamma = 1$ and $\varepsilon \ge 0$ with $\varepsilon\delta \le 1$, set*

$$\Lambda'_{1,\varepsilon} = \min\{\delta + B_z^2 T,\ \varepsilon^{-1} + B_z^2\}, \qquad \Lambda_{1,\varepsilon} = \min\{\delta + B_z^2 T,\ \varepsilon^{-1}\},$$

with the second entries read as ∞ at $\varepsilon = 0$. Then, for every comparator path satisfying (B3),

$$L_T(\check{y}) - L_T(\boldsymbol{u}_{1:T}) \ \le\ \delta R^2 + D_\varepsilon^2 d \log\left(\frac{\Lambda'_{1,\varepsilon}}{\delta}\right) + D_\varepsilon^2\, \mathcal{E}_\varepsilon^{(1)}(T) + 2(R + R_\varepsilon)\, \Lambda_{1,\varepsilon}\, P_T(\boldsymbol{u}_{1:T}),$$

$$\mathcal{E}_\varepsilon^{(1)}(T) = \min\Big\{(T-1)\, d \log\big(1 + \varepsilon\Lambda'_{1,\varepsilon}\big),\ \varepsilon(T-1)\big(\delta d + \tfrac{1}{2} B_z^2 T\big)\Big\}.$$

Only three steps of the proofs change. The geometric sum in Lemma A.4 becomes $\sum_{j<t} 1 = t$, so $B_z^2 h$ is replaced by $B_z^2 T$ in both envelopes; the same induction in general gives $\delta + B_z^2 \min\{h, t\}$, so this is the same envelope read at $h \wedge T = T$. In Lemma A.7 the term $dT \log(1/\gamma)$ is exactly 0, so the finite-memory term $2D_\varepsilon^2 dT/h$ disappears, and the trace entry of the splice bound loses its geometric truncation and sums to $\varepsilon(T-1)(\delta d + B_z^2 T/2)$. The assembly in Theorem A.8 is unchanged. At $\varepsilon = 0$ the cap entries and the leakage vanish and the bound is the dynamic regret of online ridge without forgetting, with tracking constant $4R(\delta + B_z^2 T)$ growing in T. At $\varepsilon > 0$ the tracking constant is at most $2(R + R_\varepsilon)\varepsilon^{-1}$, independent of T: at this endpoint the inflation is the only tracking mechanism. With this remark, Corollary A.9 runs over the full deployed grid, the $\gamma = 1$ expert instantiated by the $\varepsilon = 0$ case.

Remark A.10 (The persistent ridge)**.** *The implementation carries one further device. After the inflation step, the carried precision gains $(1-\gamma)\lambda_0 \boldsymbol{I}$ and the coefficients are shrunk consistently:*

$$\boldsymbol{H}_t = \boldsymbol{I} + (1-\gamma)\lambda_0 \boldsymbol{P}_t, \qquad (\boldsymbol{w}_{t+1}, \boldsymbol{P}_t) \leftarrow (\boldsymbol{H}_t^{-1}\boldsymbol{w}_{t+1}, \boldsymbol{H}_t^{-1}\boldsymbol{P}_t),$$

where $\boldsymbol{P}_t$ on the right is the covariance after inflation and before ridge. This is the update produced by a phantom observation $\boldsymbol{w} = \boldsymbol{0}$ of precision $(1-\gamma)\lambda_0$ at each step. Its purpose is conditioning: the accumulated floor keeps the precision along directions the data never excites at level $\gamma\lambda_0$, so exactly duplicated or collinear pool members cannot wind the covariance up, and with (27) *the precision spectrum is bounded on both sides. The statements above are given at $\lambda_0 = 0$. For the comparator, the phantom rows cost at most $(1-\gamma)\lambda_0 R^2$ per step; we do not track the resulting constants here. Empirically the fixed level $\lambda = 10^{-12}$ moves the deployed overall RMSE by at most 0.2% relative to $\lambda = 0$ on either stream, while removing the ridge in the presence of one exactly duplicated expert breaks the layer outright (Appendix F).*

A.7 Clipping and bounded expert losses

The MLpol bound below requires bounded outcomes and bounded expert predictions. We use the standard clipping device.

Lemma A.10 (Clipping does not increase squared loss)**.** *Let $|y| \le B$, and define*

$$\mathrm{clip}_B(a) := \max\{-B, \min\{a, B\}\}.$$

Then

$$\big(\mathrm{clip}_B(a) - y\big)^2 \le (a - y)^2 \qquad \text{for all } a \in \mathbb{R}.$$

Proof. If $a \in [-B, B]$, the claim is equality. If $a > B$, then $\mathrm{clip}_B(a) = B \ge y$, so

$$|B - y| \le |a - y|.$$

The case $a < -B$ is symmetric. □

Thus the clipped forgotten-RLS expert has cumulative squared loss no larger than the un-clipped expert analysed in Theorem A.3, provided the target lies in $[-B, B]$.

A.8 MLpol regret over a finite pool

We analyse the adaptive-rate update actually used by the implementation, (5). It is a polynomially weighted forecaster in the MLpol family (Gaillard et al., 2014); the shared maximum correction in its learning rates is included explicitly below.

Write $r_{t,j} = 2(\hat{y}_t - y_t)(\hat{y}_t - \tilde{y}_{t,j})$, $R_{t,j} = \sum_{s\le t} r_{s,j}$, $V_{t,j} = \sum_{s\le t} r_{s,j}^2$, and $b_t = \max_{s\le t,j} r_{s,j}^2$, with all four quantities initially zero. Set $D_{t,j} = 1 + V_{t,j} + b_t$ and $\eta_{t,j} = D_{t,j}^{-1}$. The prediction weights are proportional to $\eta_{t-1,j}[R_{t-1,j}]_+$, with a uniform fallback. For bounded predictions and outcomes in $[-B, B]$, put

$$G = 8B^2, \qquad A_T = \sum_{j=1}^{N} \big[G^2 + \ln(1 + V_{T,j})\big], \\ \Omega_{N,T}(B) = \sqrt{N\big[1 + (T+1)G^2\big]\big[G^2 + \ln(1 + TG^2)\big]}. \tag{40}$$

For fixed B, $\Omega_{N,T}(B) = O_B(\sqrt{NT\ln(1+T)})$. The constant 1 in the implemented learning-rate denominator ties the explicit bound to the units of the loss, which are standardized units in the experiments.

Lemma A.11 (Adaptive MLpol regret, distributional form)**.** *Assume all expert predictions and outcomes lie in $[-B, B]$. For any fixed $q \in \Delta_N$, let $y_t^q = \sum_j q_j \tilde{y}_{t,j}$. Then*

$$L_T(\hat{y}) - L_T(y^q) \le \sqrt{A_T \sum_j q_j^2 D_{T,j}} \le \Omega_{N,T}(B)\|q\|_2. \tag{41}$$

In particular,

$$L_T(\hat{y}) \le \min_j L_T(\tilde{y}_{\cdot,j}) + \Omega_{N,T}(B). \tag{42}$$

For a nonempty expert subset S and its uniform average $\bar{y}_t^S = |S|^{-1}\sum_{j\in S}\tilde{y}_{t,j}$,

$$L_T(\hat{y}) \le L_T(\bar{y}^S) + \frac{\Omega_{N,T}(B)}{\sqrt{|S|}}. \tag{43}$$

Proof. Use the weighted potential $\Phi_t = \sum_j \eta_{t,j}[R_{t,j}]_+^2$. The rates are nonincreasing. The inequality $[x+a]_+^2 \le [x]_+^2 + 2[x]_+ a + a^2$ gives

$$\Phi_t - \Phi_{t-1} \le 2\sum_j \eta_{t-1,j}[R_{t-1,j}]_+ r_{t,j} + \sum_j \eta_{t-1,j} r_{t,j}^2 = \sum_j \eta_{t-1,j} r_{t,j}^2.$$

The equality follows from $\sum_j p_{t,j} r_{t,j} = 0$; when all positive parts vanish the cross term is also zero. Since $|r_{t,j}| \le G$ and $\eta_{t-1,j} \le (1 + V_{t-1,j})^{-1}$, it remains to bound a scalar sum. For $a > 0$ and $0 \le v \le G^2$, using $x - \ln(1+x) \le x^2/(1+x)$ at $x = v/a$ yields

$$\frac{v}{a} \le \ln\frac{a+v}{a} + G^2\left(\frac{1}{a} - \frac{1}{a+v}\right).$$

With $a = 1 + V_{t-1,j}$ and $v = r_{t,j}^2$, both terms telescope, so $\Phi_T \le \sum_j[\ln(1 + V_{T,j}) + G^2] = A_T$. Convexity of squared loss and weighted Cauchy–Schwarz now give

$$L_T(\hat{y}) - L_T(y^q) \le \sum_j q_j R_{T,j} \le \left(\sum_j q_j^2 D_{T,j}\right)^{1/2} \Phi_T^{1/2}.$$

Finally $V_{T,j} \le TG^2$ and $b_T \le G^2$ give the second inequality. Point masses and uniform distributions on S give the two special cases. □

A.9 AGGREGATION: THE SECOND-ORDER MLPOL BOUND

The same potential argument gives a bound that retains each comparator's observed squared excess losses and covers the shared maximum correction of the implemented rates.

Lemma A.12 (Second-order bound for the implemented rates)**.** *Under the boundedness conditions of Lemma A.11, every expert k satisfies*

$$L_T(\hat{y}) - L_T(\tilde{y}_{\cdot,k}) \le \sqrt{(1 + V_{T,k} + b_T)\sum_{j=1}^{N}[G^2 + \ln(1 + V_{T,j})]} \le \Omega_{N,T}(B), \qquad G = 8B^2.$$

Proof. Take $q = e_k$ in (41). □

Corollary A.13 (Oracle inequality for the deployed pool)**.** *Let the pool consist of the M clipped raw forecasts and the $K = |\Gamma|\,|\mathcal{C}|$ clipped EWLS experts of the deployed (γ, c) grid, each expert (γ, c) run with inflation $\varepsilon_{\gamma,c} = c(1-\gamma)/\widehat{B}_z^2$ and satisfying the conditions of Theorem A.8 at its own parameters. Then, for every comparator path $\boldsymbol{u}_{1:T}$ satisfying (B3), the MLpol aggregate satisfies*

$$L_T(\hat{y}) \le \min\left\{\min_{1\le j\le M} L_T(f_j),\ L_T(\boldsymbol{u}_{1:T}) + \min_{\gamma\in\Gamma,\, c\in\mathcal{C}} \Psi_{h_\gamma}^{\varepsilon_{\gamma,c}}\big(T, P_T(\boldsymbol{u}_{1:T})\big)\right\} + \Delta_{N,T},$$

with $\Delta_{N,T}$ the bound of Lemma A.12 for the best-performing pool member.

Proof. Lemma A.12 bounds $L_T(\hat{y}) - \min_k L_T(\text{expert } k)$ by $\Delta_{N,T}$; it requires only bounded gradient-trick excess losses, regardless of how the expert predictions are produced. By Lemma A.10, clipping cannot increase squared loss, so Theorem A.8 bounds each EWLS expert's loss by $L_T(\boldsymbol{u}_{1:T}) + \Psi_{h_\gamma}^{\varepsilon_{\gamma,c}}(T, P_T(\boldsymbol{u}_{1:T}))$. Taking the minimum over the pool proves the claim. □

A.10 ORACLE INEQUALITY FOR THE BASE+EWLS POOL

Let $\mathcal{H} = \{h_1, \dots, h_K\}$ be a finite set of nominal EWLS scales. For each $h \in \mathcal{H}$, let $\check{y}^{(h)}$ denote the corresponding forgotten-RLS adaptation sequence, with predictions clipped to $[-B, B]$ before aggregation. The enlarged pool is

$$\mathcal{I}_{\mathcal{H}} = \{f_1, \dots, f_M\} \cup \{\check{y}^{(h)} : h \in \mathcal{H}\}.$$

Let $N = M + K$, and let Ψ_h be as in (6).

Theorem A.14 (Oracle inequality for multi-scale online adaptation)**.** *Assume the conditions of Theorem A.3 hold for each $h \in \mathcal{H}$, and assume outcomes lie in $[-B, B]$. All predictions entering MLpol are clipped to $[-B, B]$. Then, writing f_j and $\check{y}^{(h)}$ for the clipped raw and EWLS prediction sequences, for every comparator path $\boldsymbol{u}_{1:T}$ satisfying (B3),*

$$L_T(\hat{y}) \le \min\left\{\min_{1\le j\le M} L_T(f_j),\ L_T(\boldsymbol{u}_{1:T}) + \min_{h\in\mathcal{H}} \Psi_h(T, P_T(\boldsymbol{u}_{1:T}))\right\} + \Omega_{M+K,T}(B). \tag{44}$$

Proof. By the point-mass special case of Lemma A.11 applied to the enlarged pool,

$$L_T(\hat{y}) \le \min\left\{\min_{1\le j\le M} L_T(f_j), \min_{h\in\mathcal{H}} L_T(\check{y}^{(h)})\right\} + \Omega_{M+K,T}(B).$$

By Lemma A.10, clipping cannot increase squared loss. Therefore Theorem A.3 gives, for every $h \in \mathcal{H}$,

$$L_T(\check{y}^{(h)}) \le L_T(\boldsymbol{u}_{1:T}) + \Psi_h(T, P_T(\boldsymbol{u}_{1:T})).$$

Taking the minimum over $h \in \mathcal{H}$ proves (44). □

A.11 GEOMETRIC GRIDS AND UNKNOWN ADAPTATION SCALE

The leading h-dependent terms in Ψ_h are

$$\frac{C_1 dT}{h} + C_2 hP,$$

where $P = P_T(\boldsymbol{u}_{1:T})$. If $P = 0$, the leading expression is minimised by the largest available memory scale, corresponding to the no-drift case. For $P > 0$, the continuous minimizer is

$$h^\star = \sqrt{\frac{C_1 dT}{C_2 P}},$$

with minimum value $2\sqrt{C_1 C_2 dTP}$.

Corollary A.15 (Geometric-grid adaptation)**.** *Let $P = P_T(\boldsymbol{u}_{1:T}) > 0$, and let $\mathcal{H}$ be a geometric grid over $[h_{\min}, h_{\max}]$ with ratio $\rho > 1$. Suppose $h^\star \in [h_{\min}, h_{\max}]$. Then there exists $h_k \in \mathcal{H}$ such that*

$$\frac{C_1 dT}{h_k} + C_2 h_k P \le C_\rho \sqrt{dTP}, \tag{45}$$

where C_ρ depends only on ρ, C_1, C_2. Consequently, ignoring logarithmic and initialization terms,

$$L_T(\hat{y}) \le \min\left\{ \min_j L_T(f_j),\ L_T(\boldsymbol{u}_{1:T}) + C_\rho \sqrt{dT\, P_T(\boldsymbol{u}_{1:T})} \right\} + \Omega_{M+K,T}(B). \tag{46}$$

Proof. Since $\mathcal{H}$ is geometric and contains $h^\star$ within its range, there is a grid point h_k with

$$h^\star \le h_k \le \rho h^\star \quad \text{or} \quad h^\star/\rho \le h_k \le h^\star.$$

In either case,

$$\frac{h_k}{h^\star} \le \rho, \qquad \frac{h^\star}{h_k} \le \rho.$$

Therefore

$$\begin{aligned} \frac{C_1 dT}{h_k} + C_2 h_k P &\le \rho \left(\frac{C_1 dT}{h^\star} + C_2 h^\star P \right) \\ &= 2\rho \sqrt{C_1 C_2 dTP}. \end{aligned}$$

Absorbing $2\rho\sqrt{C_1 C_2}$ into C_ρ proves (45). Substituting this bound into Theorem A.14 gives (46). □

A.12 LOCAL-BAND REFINEMENT FOR DENSE EWLS GRIDS

The oracle inequality above compares MLpol with the best single EWLS scale in a finite grid, which gives a worst-case aggregation overhead of order $\sqrt{T(M+K)\ln(1+T)}$ for fixed B. This point-oracle view can be pessimistic for dense EWLS grids. Nearby forgetting factors often generate very similar prediction sequences, and the useful comparator may be a local band of time scales. The distributional form of Lemma A.11 makes this precise.

Let the enlarged pool contain M base experts and K EWLS experts, so that $N = M + K$. For a non-empty subset $S \subseteq \{1, \dots, K\}$ of EWLS scales, define the band-averaged EWLS prediction

$$\bar{y}_t^S = \frac{1}{|S|} \sum_{k \in S} \tilde{y}_t^{(h_k)}.$$

Corollary A.16 (Band-oracle bound for EWLS grids)**.** *Assume all base predictions, EWLS predictions after clipping, and outcomes lie in $[-B, B]$. Then for every non-empty EWLS band $S \subseteq \{1, \dots, K\}$,*

$$L_T(\hat{y}) \le L_T(\bar{y}^S) + \frac{\Omega_{M+K,T}(B)}{\sqrt{|S|}}. \tag{47}$$

Proof. Apply Lemma A.11 with $N = M + K$ to the distribution q^S that assigns mass $1/|S|$ to the EWLS experts in S and zero mass to all other experts, including the M base experts. Then

$$\|q^S\|_2^2 = \frac{1}{|S|},$$

and the convex-mixture prediction induced by q^S is precisely $\bar{y}_t^S$. Substituting this into (41) gives (47). □

We next relate the band-average comparator to a single central forgetting scale. The following continuity statement is for the idealised forgotten-RLS recursion analysed above, without covariance inflation. It is used only to justify why nearby forgetting factors in a dense grid should not be viewed as unrelated experts. It is convenient to parametrize forgetting factors by

$$\theta = \log h, \qquad h = (1-\gamma)^{-1}.$$

Write $\check{y}^{(\theta)}$ for the clipped EWLS prediction sequence with $h = e^\theta$.

Lemma A.17 (Continuity of finite-horizon EWLS predictions)**.** *Fix $T < \infty$, $\delta > 0$, and a compact interval $\Theta \subset \mathbb{R}$. For every $t \le T$, the idealised EWLS coefficient $\boldsymbol{w}_t^{(\theta)}$ and prediction $\check{y}_t^{(\theta)}$ are continuous functions of $\theta \in \Theta$. If Θ is compact and $\gamma(\theta) = 1 - e^{-\theta}$ stays in $(0,1)$, then the map is Lipschitz: there exists $L_\theta < \infty$, depending on the realised finite sequence and on Θ, such that*

$$\max_{1\le t\le T} |\check{y}_t^{(\theta)} - \check{y}_t^{(\theta')}| \le L_\theta |\theta - \theta'| \qquad \text{for all } \theta, \theta' \in \Theta. \tag{48}$$

Proof. For the idealised EWLS recursion,

$$\boldsymbol{A}_t(\gamma) = \gamma^t \delta \boldsymbol{I} + \sum_{s=1}^{t} \gamma^{t-s} \tilde{\boldsymbol{z}}_s \tilde{\boldsymbol{z}}_s^\top, \qquad \boldsymbol{b}_t(\gamma) = \sum_{s=1}^{t} \gamma^{t-s} y_s \tilde{\boldsymbol{z}}_s.$$

Both $\boldsymbol{A}_t(\gamma)$ and $\boldsymbol{b}_t(\gamma)$ are polynomial functions of γ. Moreover $\boldsymbol{A}_t(\gamma) \succeq \gamma^t \delta \boldsymbol{I} \succ 0$ for $\gamma \in (0,1]$, so $\boldsymbol{A}_t(\gamma)^{-1}$ is continuous wherever $\gamma > 0$. Hence

$$\boldsymbol{w}_{t+1}^{(\gamma)} = \boldsymbol{A}_t(\gamma)^{-1} \boldsymbol{b}_t(\gamma)$$

is continuous in γ, and therefore in θ through the continuous map $\gamma(\theta) = 1 - e^{-\theta}$. On a compact interval contained in $(0,1)$, the derivative is bounded, so the map is Lipschitz. The prediction $\check{y}_t^{(\theta)} = \tilde{\boldsymbol{z}}_t^\top \boldsymbol{w}_t^{(\theta)}$, followed by clipping, preserves continuity and Lipschitzness. □

Corollary A.18 (Local-band control for dense EWLS grids)**.** *Assume the Lipschitz condition* (48)*. Fix a target scale $\theta^\star \in \Theta$ and a radius $r > 0$. Let*

$$S_r(\theta^\star) = \{k \in \{1, \dots, K\} : |\theta_k - \theta^\star| \le r\}, \qquad \theta_k = \log h_k,$$

and assume $S_r(\theta^\star) \neq \emptyset$. Then

$$L_T(\hat{y}) \le L_T(\check{y}^{(\theta^\star)}) + 4BL_\theta T r + \frac{\Omega_{M+K,T}(B)}{\sqrt{|S_r(\theta^\star)|}}. \tag{49}$$

Consequently, combining with Theorem A.3, for $h^\star = e^{\theta^\star}$,

$$L_T(\hat{y}) \le L_T(\boldsymbol{u}_{1:T}) + \Psi_{h^\star}(T, P_T(\boldsymbol{u}_{1:T})) + 4BL_\theta T r + \frac{\Omega_{M+K,T}(B)}{\sqrt{|S_r(\theta^\star)|}}. \tag{50}$$

Proof. For every $k \in S_r(\theta^\star)$, the Lipschitz condition gives

$$\max_{1\le t\le T} |\check{y}_t^{(h_k)} - \check{y}_t^{(\theta^\star)}| \le L_\theta r.$$

Therefore the band average satisfies

$$\max_{1\le t\le T} |\bar{y}_t^{S_r(\theta^\star)} - \check{y}_t^{(\theta^\star)}| \le L_\theta r.$$

For clipped predictions and outcomes in $[-B, B]$, squared loss is $4B$-Lipschitz in the prediction:

$$|(a-y)^2 - (b-y)^2| \le 4B|a-b|, \qquad a, b, y \in [-B, B].$$

Hence

$$L_T(\bar{y}^{S_r(\theta^\star)}) \le L_T(\check{y}^{(\theta^\star)}) + 4BL_\theta T r.$$

Combining this inequality with Corollary A.16 proves (49). The final display follows by applying Theorem A.3 to the EWLS scale $h^\star = e^{\theta^\star}$. □

Algorithm 1 Implemented Multi-Time-Scale EWLS with Adaptive MLpol

Require: M base experts; K pairs $(\gamma_k, \varepsilon_k)$; ridge λ; training RMS $s > 0$; optional clipping radius B

Work with targets and base forecasts divided by s; return $s\hat{y}_t$
Set $N = M + K$, $d = M + 1$, $n_0 = M + 5$; $R_{0,j} = V_{0,j} = b_0 = 0$, $\eta_{0,j} = 1$, $p_{1,j} = 1/N$
Set cold-start statistics $\boldsymbol{S}^{(k)} = \mathbf{0}$, $\boldsymbol{b}^{(k)} = \mathbf{0}$ for every k
for $t = 1, \dots, T$ **do**
  Observe base forecasts $\boldsymbol{z}_t$; set $\phi_t = (\boldsymbol{z}_t^\top, 1)^\top$
  if $t = n_0 + 1$ **then**
    for $k = 1, \dots, K$ **do**
      $a_k = 10^{-12}(\mathrm{tr}(\boldsymbol{S}^{(k)})/d + 10^{-12})$; $\boldsymbol{P}^{(k)} = (\boldsymbol{S}^{(k)} + a_k \boldsymbol{I})^{-1}$; $\boldsymbol{w}^{(k)} = \boldsymbol{P}^{(k)} \boldsymbol{b}^{(k)}$
    end for
  end if
  $\tilde{y}_{t,j} = z_{t,j}$ for $j \le M$
  $\tilde{y}_{t,M+k} = M^{-1} \sum_j z_{t,j}$ if $t \le n_0$; otherwise $\tilde{y}_{t,M+k} = \phi_t^\top \boldsymbol{w}^{(k)}$
  Optionally clip candidate predictions to $[-B, B]$ (off in experiments)
  Predict $\hat{y}_t = \sum_j p_{t,j} \tilde{y}_{t,j}$, then observe y_t
  for $k = 1, \dots, K$ **do**
    if $t \le n_0$ **then**
      $\boldsymbol{S}^{(k)} \leftarrow \gamma_k \boldsymbol{S}^{(k)} + \phi_t \phi_t^\top$; $\boldsymbol{b}^{(k)} \leftarrow \gamma_k \boldsymbol{b}^{(k)} + y_t \phi_t$
    else
      $a \leftarrow \gamma_k + \phi_t^\top \boldsymbol{P}^{(k)} \phi_t$; $\boldsymbol{K} \leftarrow \boldsymbol{P}^{(k)} \phi_t / a$
      $\boldsymbol{w}^{(k)} \leftarrow \boldsymbol{w}^{(k)} + \boldsymbol{K}(y_t - \phi_t^\top \boldsymbol{w}^{(k)})$
      $\boldsymbol{P}^{(k)} \leftarrow \gamma_k^{-1}(\boldsymbol{P}^{(k)} - \boldsymbol{K} \phi_t^\top \boldsymbol{P}^{(k)}) + \varepsilon_k \boldsymbol{I}$; symmetrize $\boldsymbol{P}^{(k)}$
      $\boldsymbol{H} \leftarrow \boldsymbol{I} + (1 - \gamma_k)\lambda \boldsymbol{P}^{(k)}$; $\boldsymbol{w}^{(k)} \leftarrow \boldsymbol{H}^{-1} \boldsymbol{w}^{(k)}$; $\boldsymbol{P}^{(k)} \leftarrow \boldsymbol{H}^{-1} \boldsymbol{P}^{(k)}$; symmetrize $\boldsymbol{P}^{(k)}$
    end if
  end for
  $r_{t,j} = 2(\hat{y}_t - y_t)(\hat{y}_t - \tilde{y}_{t,j})$; $R_{t,j} = R_{t-1,j} + r_{t,j}$; $V_{t,j} = V_{t-1,j} + r_{t,j}^2$
  $b_t = \max\{b_{t-1}, \max_j r_{t,j}^2\}$; $\eta_{t,j} = (1 + V_{t,j} + b_t)^{-1}$
  $p_{t+1,j} \propto \eta_{t,j} [R_{t,j}]_+$; use uniform weights when the normalizer is zero
end for

B Experimental protocol and implementation details

B.1 The algorithm

Algorithm 1 is the recursion run in the experiments, with the causal cold start, the adaptive learning rates, the covariance inflation, and the persistent ridge. The EWLS update is not projected. The clipping line is off in every reported experiment; it is there because the MLpol bound of Appendix A is stated for clipped predictions.

B.2 RTE-FR Load data and splits

Raw series and daily target. We use the public French national electricity-load data described in Section 5. The native short-term load-forecasting problem is intraday: load is observed at half-hourly resolution, and production systems typically forecast a vector of future half-hourly loads. In this paper we focus on a scalar daily-load benchmark. For each calendar day t, the target is the average load over the intraday half-hourly observations,

$$y_t = \frac{1}{n_t} \sum_{q=1}^{n_t} \mathrm{Load}_{t,q},$$

where $\mathrm{Load}_{t,q}$ denotes the q-th half-hourly load observation on day t. On ordinary days $n_t = 48$. The target is therefore an average power, and RMSE is in MW. This aggregation removes the need to model the intraday load shape and focuses the evaluation on non-stationarity in the daily demand level.

Weather covariates. All methods receive the same temperature covariates, built from realised temperatures. This keeps differences in weather-forecast quality out of the comparison between adaptation methods and makes the benchmark ex post in the weather; a deployment would feed forecast temperatures to the same pipeline.

Feature availability and leakage control. The feature vector contains calendar variables, holiday indicators, temperature variables, and lagged load features. Calendar and holiday variables are known before the target day, and lagged load features use strictly previous days only. During the test period every method first predicts y_t, then observes y_t, and only then updates its EWLS states and MLpol weights.

Train, validation, and test periods. The data span 2012-01-01 to 2023-09-30. Hyperparameters are selected only within the pre-test period, with 2018 as the validation window for base-model tuning (Appendix B.3). The online test period, 2019-01-01 to 2023-09-30, is split into six regimes: pre-COVID (to 2020-03-16), lockdown (to 2020-05-11), recovery (to 2021-12-31), energy crisis (to 2022-09-30), demand reduction (to 2023-03-31), and 2023 (to 2023-09-30). No observation from the test period is used for base-model tuning, EWLS-grid selection, or nuisance-parameter selection.

B.3 Base-model features and tuning

Scope. Of the $M = 5$ base predictors, hyperparameters are set as follows:

- *Lag-1*: no hyperparameters.
- *GAM*: specification after Gaillard et al. (2016), fitted with `pyGAM LinearGAM`; the formula is given below.
- *Ridge, XGBoost, LightGBM*: tuned on a held-out 2018 validation window within the training period, using the Optuna-based protocol described below.

The release includes the frozen configurations and a runnable Optuna search with the recorded search spaces and budgets. Historical trial logs are not available, so a fresh search may not return the same selected trial; the tables are reproduced from the frozen configurations or the shipped predictions.

GAM specification. The executed class is `GAMForecaster_original` (`pyGAM 0.12.0`); the second GAM class in the code is not used. Its additive load-forecasting structure is inspired by Goude et al. (2014); Gaillard et al. (2016):

$$\begin{aligned}\eta_t = \beta_0 &+ s_{10}(\mathrm{Time}_t) + s^{\mathrm{cp}}_{50}(\mathrm{toy}_t) + s_{10}(\mathrm{Temp}_t) + \mathrm{f}(\mathrm{WeekDay}_t) + \mathrm{f}(\mathrm{BH}_t) \\ &+ s_{10}(\mathrm{Load}_{t-1}) + s_{10}(\mathrm{Load}_{t-7}) + s_{10}(\mathrm{Temp}_t^{s99}).\end{aligned}$$

Here s_k denotes a pyGAM spline term with k basis functions and s^{cp}_{50} the cyclic P-spline term for time of year. The two factor terms encode weekday and bank holiday. The temperature feature uses exponential smoothing factor 0.99. `LinearGAM.gridsearch` selects a shared smoothing penalty from $\{0, 0.1, \ldots, 0.9\}$ by generalized cross-validation (GCV) on the fit window. The structure, basis dimensions, and penalty grid are fixed before 2019; there are no tensor-product interactions in the executed specification.

Train/validation split. Each tuned model is fit on 2012-01-01 to 2017-12-31 and scored on a 2018 validation window (2018-01-01 to 2018-12-31) by RMSE on the one-step-ahead prediction task. After tuning, the selected configuration is refit on the full 2012–2018 window for evaluation on the 2019–2023 test period.

Separation of convergence and modelling hyperparameters. Convergence controls, such as `max_iter` and `tol` of the linear models, are fixed a priori, so the search does not mix optimisation budget with modelling capacity.

Search algorithm and trial budget. We use Optuna (Akiba et al., 2019) with the Tree-structured Parzen Estimator (TPE) sampler and seed 42, without pruning. Trial budgets in the supplied configuration are 1000 for Ridge (and optional Lasso) and 300 for each tree ensemble. Each invocation adds that many trials; fresh searches and resumed studies are distinguished in the reproduction guide.

Search spaces. Table 5 lists the per-model search spaces.

Table 5: Per-model Optuna search spaces used for base-model hyperparameter tuning. "log": log-uniform sampling. "step s": discretised with stride s.

Model	Hyperparameter	Range	Notes
Ridge	`alpha`	$[10^{-4}, 10^{4}]$	log
XGBoost	`n_estimators`	[100, 2000]	step 100
	`learning_rate`	$[10^{-3}, 0.3]$	log
	`max_depth`	[2, 10]	integer
	`min_child_weight`	$[10^{-2}, 20]$	log
	`subsample`	[0.3, 1.0]	uniform
	`colsample_bytree`	[0.3, 1.0]	uniform
	`reg_alpha`	$[10^{-8}, 10]$	log
	`reg_lambda`	$[10^{-8}, 10]$	log
LightGBM	`n_estimators`	[100, 2000]	step 100
	`learning_rate`	$[10^{-3}, 0.3]$	log
	`num_leaves`	[7, 255]	log
	`max_depth`	$[-1, 12]$	integer (-1 = unlimited)
	`min_child_samples`	[5, 100]	step 5
	`subsample`	[0.5, 1.0]	uniform
	`colsample_bytree`	[0.5, 1.0]	uniform
	`reg_alpha`	$[10^{-8}, 10]$	log
	`reg_lambda`	$[10^{-8}, 10]$	log

Foundation-model baselines. Both foundation-model wrappers use random seed 42 and the MPS backend. TabICL sets `n_estimators=10` and `kv_cache=True`; TabPFN uses its library defaults. The online variants are refit at each test step on the expanding sample observed before the prediction, and no task-specific hyperparameter is tuned.

B.4 EWLS IMPLEMENTATION AND COVARIANCE INFLATION

Algorithm 1 adds two devices to the recursion of Section 3: a batch cold start, and the persistent ridge of Remark A.10, applied after the inflation step.

Cold start for EWLS experts. Algorithm 1 uses a short cold-start period of $M+5$ test observations for the EWLS experts. During this period, each EWLS adaptation expert outputs the uniform average of the raw base predictions,

$$\check{y}_t^{(\gamma_k)} = \frac{1}{M}\sum_{j=1}^{M} f^{(j)}(\boldsymbol{x}_t), \qquad t \le M+5,$$

while MLpol treats these predictions like ordinary expert predictions and updates its cumulative pseudo-regrets normally. After observing the first $M+5$ labels, for each γ_k we initialise each EWLS state by solving the corresponding batch EWLS problem on those cold-start observations, obtaining the pre-update coefficient $\boldsymbol{w}_{M+6}^{(k)}$ and covariance $\boldsymbol{P}_{M+5}^{(k)}$. From step $M+6$ onward, the EWLS experts use the RLS recursion in Algorithm 1. Thus the cold start affects only the internal EWLS states; the MLpol layer is active from the first test day and all updates remain causal.

Covariance inflation with γ-dependent scaling. Expert k runs the covariance recursion (3) with $(\gamma, \varepsilon) = (\gamma_k, \varepsilon_k)$. Under the Kalman-filter interpretation of RLS, adding $\varepsilon_k \boldsymbol{I}$ to the covariance corresponds to a process-noise covariance in the implicit state equation $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \boldsymbol{\eta}_t$, with $\mathrm{Cov}(\boldsymbol{\eta}_t) = \varepsilon_k \boldsymbol{I}$: it sets the expert's prior scale for how much the combination coefficients may move from one step to the next.

A single ε shared by all experts does not fit the γ-grid. At large γ the expert is meant to keep a long memory, and an inflation sized for the short-memory experts would dominate its slow forgetting and shorten that memory. We therefore couple the inflation to the forgetting scale,

$$\varepsilon_k \;=\; \varepsilon_0\,(1-\gamma_k)^{\alpha}, \qquad \alpha = 1. \tag{51}$$

The exponent matches scales. An expert with forgetting factor γ carries discounted mass $\sum_{\ell \geq 0} \gamma^\ell = 1/(1-\gamma)$, so when the feature directions are of comparable scale its weighted Gram matrix is of order $(1-\gamma)^{-1}$ and its inverse of order $1-\gamma$; $\alpha = 1$ keeps the added covariance at the same fraction of that inverse across the grid. Remark A.6 makes this precise: $\alpha = 1$ is the only exponent for which the inflation cost in the oracle inequality is the same additive overhead at every memory length. The scalar ε_0 itself is hedged over the values of c (Section 3); the validation study behind the hedge is in Appendix B.5.

Relation to the theoretical analysis. Appendix A.6 analyses this recursion directly (Theorem A.8 and Corollary A.9). The inflation does not discharge the bounded-iterate condition. It shortens the effective memory: Remark A.4 identifies the inflated recursion with an idealised one at a factor $\gamma_\varepsilon \leq \gamma$. We therefore read $h(\gamma) = 1/(1-\gamma)$ as a *nominal* memory: a smaller h adapts faster, but h is not a number of days.

B.5 Hedging versus selecting c

Could one select the inflation on pre-test data and deploy the winner? The deployed layer hedges over all six values of $c \in \mathcal{C} = \{1, 10^2, \ldots, 10^{10}\}$ (Section 3); this study selects one.

Protocol. For each value of c we run the single-c aggregate: the five raw bases plus the γ-grid EWLS sub-pool at $\varepsilon_0 = c/\widehat{B}_z^2$, with the coupling, ridge, and MLpol layer of the deployed configuration. Each stream provides one validation leg and one test leg.

- **RTE-FR.** The five bases are refit on 2012–2017 with the hyperparameters of Appendix B.3, so 2018 is a genuine out-of-sample year for them; the layer cold-starts on 2018-01-01 and is scored on the 365 validation days. The test leg runs the same single-c aggregate on the deployed 2019–2023 stream.
- **NYC taxi.** The deployed stream already runs through 2018 as a burn-in year (bases frozen on 2011–2017), so one run yields both legs: the 2018 score and the 2019–2023 score.

Table 6: Overall RMSE of the single-c aggregate for each value of c, on the validation year and on the test period of each stream (RTE-FR in MW, NYC taxi in daily trips). **Bold**: best value of c per column. The last row is the deployed hedge over all six values.

	RTE-FR		NYC taxi	
	val. 2018	test 2019–23	val. 2018	test 2019–23
$c = 1$	687.3	728.8	34585.1	11114.3
$c = 10^2$	**685.2**	**686.9**	31960.5	**9690.5**
$c = 10^4$	704.1	715.2	**30917.8**	9866.2
$c = 10^6$	712.0	726.0	30963.6	9979.5
$c = 10^8$	712.1	726.2	30965.8	9981.2
$c = 10^{10}$	712.0	725.9	30963.4	9979.2
Hedge over $\mathcal{C}$ (deployed)	699.5	686.2	31084.1	9599.3

Why a validation year cannot pick c. Validation ranks the values of c correctly on RTE-FR and inverts the ranking on NYC taxi, where its choice is 2.8% worse than the hedge on the test period. The value of c matters most at breaks: the hindsight-best cell of each regime has $c = 10^{10}$ in the RTE-FR lockdown and in the NYC collapse, and a smaller c in every other regime (Appendix G). A validation year without a break sees little of this. On RTE-FR, $c = 1$ is 2.1 MW behind the best value in 2018 and 41.9 MW behind it in the test period, which contains the lockdown. Even a correct ranking fixes one value of c for the whole test period, while the best value changes with the regime. The hedge fixes none: its weights move between cells as the regime changes (Section 6).

B.6 Adaptive single-model references

Tables 1 and 2 include two adaptive references that update a single affine combination of the frozen bases online. Rolling OLS refits ordinary least squares of y on $(\boldsymbol{z}_t, 1)$ every day on the last W stream

days, playing the uniform mean of the bases until $M + 3$ days are available; one window serves every direction. VIKING (de Vilmarest & Wintenberger, 2024) is the variational Bayesian Kalman filter on the same design, run on the standardized stream with an intercept. Its observation-noise variance stays at the training residual variance of the uniform mean of the bases (with $\rho_a = 0$ the variational update of this variance is inactive), and the isotropic process-noise variance is learned online from the initial value $\exp(\hat{b}_0)$ with prior variance Σ_0 and random-walk variance ρ_b. The process noise inflates the covariance as in (3) with $\gamma = 1$ and ε equal to the ratio of process to observation noise, so VIKING has the ordered memory lengths of Section 3 at a single learned inflation. Nuisance parameters are selected by validation RMSE on the 2018 leg of Appendix B.5: $W \in \{20, 60, 120, 365, 730\}$ and $(\rho_b, \hat{b}_0, \Sigma_0) \in \{10^{-4}, 10^{-3}, 10^{-2}\} \times \{-14, -12, \dots, 0\} \times \{10^{-5}, 10^{-4}, \dots, 1\}$. On RTE-FR the selection picks $W = 120$ and $(\rho_b, \hat{b}_0, \Sigma_0) = (10^{-2}, -10, 1)$, and both references cold-start on 2019-01-01. On NYC taxi it picks $W = 365$ and $(10^{-3}, -2, 10^{-5})$, and both references run on from the 2018 burn-in. At the selected parameters the learned ratio barely moves at the breaks: on RTE-FR it is 0.0059 on the day before the lockdown and at most 0.0071 during it, and on NYC taxi it stays at 0.0211 through the collapse. These values equal $\varepsilon_0 = c/\widehat{B}_z^2$ with $c \approx 0.04$ and $c \approx 0.13$, below the smallest value of $\mathcal{C}$. The bootstrap intervals in the captions of Tables 1 and 2 use 30-day blocks and 2,000 replicates.

B.7 Target scale: the standardized recipe and its ablations

The layer runs on the standardized stream (Section 3). Table 7 rescales the inputs and removes the intercept column on both streams.

Table 7: Overall RMSE under rescalings and ablations of the deployed configuration (RTE-FR in MW, NYC taxi in daily trips). The raw-units rows use the ridge level 10^{-2} matched to raw target units; the deployed rows use the standardized recipe of Section 3.

Configuration	RTE-FR	NYC taxi
Deployed (standardized, $\lambda = 10^{-12}$)	686.2	9599
No intercept column	702.2	9977
No intercept at $c \in \{10^8, 10^{10}\}$ only	688.4	9651
Raw units ($\lambda = 10^{-2}$)	690.1	9749
Raw units, all inputs $\times 10^8$	700.0	9947
Standardized target, ridge left at 10^{-2}	1508.1	9610

The inflation and the ridge must be compatible. Once the covariance sits at its ε floor, the ridge step shrinks the coefficients by a factor $1 + (1 - \gamma)\lambda\varepsilon$ per step, so the design needs $\varepsilon_0\lambda \ll 1$. Standardizing the target multiplies ε_0 by the squared target scale. With the ridge left at its raw-units value the product reaches 10^7 at the largest values of c, and the cells at $c \geq 10^4$ shrink their coefficients to zero. MLpol starts from uniform weights, so RTE-FR, which is scored from its first day, carries these cells until the losses move the weight away, and its pre-COVID error grows fourfold; NYC taxi is scored after its burn-in year and is unaffected. The deployed recipe keeps the product below 2×10^{-3} for every c.

The intercept matters at the smaller values of c. Removing the intercept column everywhere is significant on RTE-FR (paired bootstrap interval $[+8, +24]$ MW). Removing it only at $c \in \{10^8, 10^{10}\}$ changes little: at those values the cap binds in every direction, all directions share one short memory, and over short windows the intercept is nearly collinear with the consensus level. At the smaller values of c the offset keeps a longer memory than the consensus gain, long enough for the seasonal movement of the level to separate additive from multiplicative drift (Section 3).

Standardization removes the unit dependence. Multiplying all raw inputs by 10^8 changes the raw-units result only through the intercept column, which does not rescale with them: without that column the two raw-unit runs agree to within 0.01% in overall RMSE. Standardizing fixes the scale of the constant coordinate.

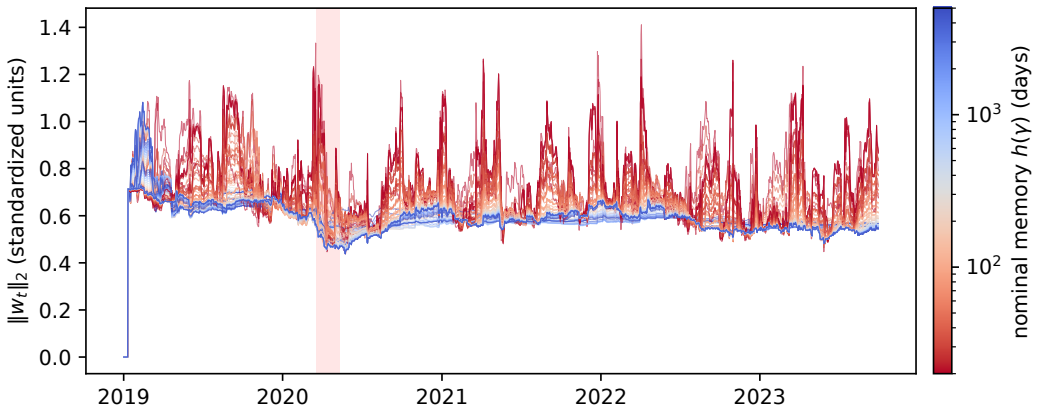


Figure 4: **Empirical bounded-iterate diagnostic on RTE-FR Load.** Coefficient-norm trajectories $\|\boldsymbol{w}_t^{(\gamma,c)}\|_2$ of the 90 finite-forgetting EWLS cells of the deployed (γ, c) grid over the 2019–2023 test period, in standardized units, coloured by nominal memory $h(\gamma)$ (warm = short, cool = long); the shaded band marks the first COVID-19 lockdown.

B.8 Compute and reproducibility

Compute. The base-model fits and aggregation exhibits run on the CPU of an Apple M4 Pro workstation (14 cores, 48 GB unified memory). The foundation-model reference streams use its MPS GPU backend. The Optuna workflow is supplied separately; its full tuning history and timing are not part of the released run record. The deployed online-aggregation pipeline over the 1,734-day test period with $N = 101$ experts runs end-to-end in about 3 s (1.5 ms per step); the recursion is dominated by $d \times d$ linear-algebra updates with $d = M + 1 = 6$ and is not the computational bottleneck. The paired moving block bootstrap of Appendix C.3 ($R = 2{,}000$, all windows and methods) runs in seconds.

B.9 Empirical bounded-iterate diagnostic

Theorem 1 assumes bounded EWLS iterates, condition (B4$^{\varepsilon}$) of Theorem A.8. On the standardized stream the coefficients are dimensionless, so the condition can be checked directly. Figure 4 shows the coefficient norms of the 90 finite-forgetting cells of the deployed grid over all $T = 1{,}734$ test days; the six $\gamma = 1$ cells, not shown, peak at 1.08. The running supremum is

$$\sup_{t,\gamma,c} \|\boldsymbol{w}_t^{(\gamma,c)}\|_2 = 1.41,$$

attained by the fastest cell at $c = 1$ near the onset of the lockdown, and at every finite h the supremum is largest at $c = 1$, the weakest inflation. The iterate reacts to the break and stays bounded by a small constant.

C Detailed RTE-FR results and statistical significance

C.1 The (γ, c) expert family in full

Table 8 reports the overall RMSE of every standalone EWLS cell in the deployed pool: 16 forgetting factors (15 on the geometric grid over $h \in [20, 5000]$ plus the $\gamma = 1$ endpoint) times the six values of c, each cold-started with the stream as in the deployed run. Per-regime values for all 101 cells are the input of the selection study (Table 3 and Appendix G).

Each column of Table 8 sweeps the forgetting factor at one value of c, and the columns trace the mechanism of Section 6 cell by cell. At $c = 1$ the inflation is nearly inactive, so the level must be tracked through forgetting and the column favours the shortest memories, as in the $\varepsilon = 0$ panel of Figure 5. At $c = 10^2$ the inflation carries the fast directions, γ sets the memory of the contrasts only,

Table 8: Overall RMSE (MW) of each standalone EWLS cell on RTE-FR 2019–2023, by nominal memory $h(\gamma)$ (rows) and c (columns). **Bold**: family minimum. The $\gamma = 1$ row is constant in c because the coupling $\varepsilon_k = \varepsilon_0(1 - \gamma_k)$ sets zero inflation at the endpoint. For reference: the deployed hedge reaches 686.2, MLpol on the 96 EWLS cells alone 691.3, and MLpol on the five raw bases alone 987.1.

$h(\gamma)$	$c = 1$	$c = 10^2$	$c = 10^4$	$c = 10^6$	$c = 10^8$	$c = 10^{10}$
20	734.4	781.8	813.3	813.6	813.4	801.5
30	733.4	761.0	801.7	802.7	802.7	798.4
44	740.9	744.3	794.6	796.5	796.5	795.1
65	753.0	731.4	790.7	793.3	793.4	792.9
97	767.6	721.3	788.1	792.0	792.1	791.9
144	784.2	713.6	786.0	791.4	791.5	791.5
213	802.6	708.2	783.7	791.2	791.3	791.3
316	822.4	705.3	780.8	791.1	791.3	791.3
469	843.3	**704.9**	777.2	791.0	791.3	791.3
696	865.0	706.9	772.6	790.9	791.3	791.3
1032	887.3	711.4	766.8	790.7	791.3	791.3
1531	910.0	718.3	759.8	790.5	791.4	791.4
2273	933.3	727.4	751.7	790.1	791.4	791.4
3367	956.9	738.6	742.9	789.6	791.4	791.4
5000	980.8	751.8	733.9	788.8	791.4	791.4
∞ ($\gamma = 1$)	1181.5	1181.5	1181.5	1181.5	1181.5	1181.5

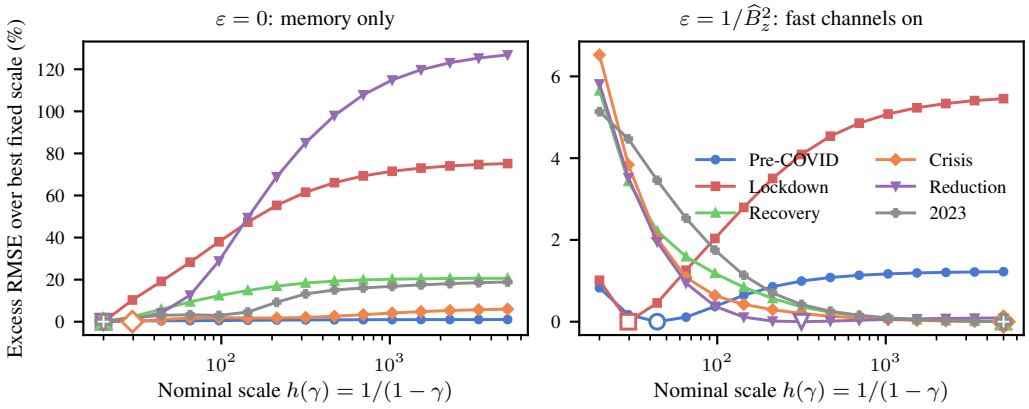


Figure 5: **One speed cannot serve two directions.** Excess RMSE of standalone EWLS experts at fixed γ, relative to each regime's best fixed scale (white markers), with the inflation toggled. *Left* ($\varepsilon = 0$): the level drift present in every regime forces every regime to the shortest scale. *Right* ($\varepsilon = 1/\widehat{B}_z^2$): the fast channels absorb the level and the offset, and the remaining contrast preferences disperse across regimes with an order of magnitude smaller excess. Figure 3c reports the minimum and the worst point of each curve.

and the column favours a longer memory. From $c = 10^6$ on the cap binds almost at once along the level, the offset, and every model contrast, so the nominal memory no longer matters. No cell is best in every regime (Appendix G), and the hedge improves significantly on the best fixed cell (Table 12).

C.2 Foundation-model references: the time-index rerun

Table 9 reports the foundation-model reference grid on the 2019–2021 sub-window. Each variant refits daily on an expanding in-context window (train through 2018, then 746 online steps) and varies two features: a global time index (row position in the stream), and the Oxford Government Response Index (GRI), an external COVID policy covariate. Without either feature the inputs never reveal where the present sits in time, so in-context learning cannot down-weight pre-shift rows, and both models fail in the lockdown. Either feature restores that, which is why the time index alone recovers

most of what the policy covariate provides; with the time index present, the covariate helps mainly in the lockdown. MELO runs the same window at under 2 ms per step on the same machine.

Table 9: Foundation-model reference grid on RTE-FR Load, 2019-01-01–2021-01-15. All variants are refit daily on an expanding context window; "time index" is the global row position, "GRI" the Oxford Government Response Index. Wall-clock is mean seconds per online step (Apple M-series, MPS backend).

Model	Features	Pre-lockdown	Lockdown	Post-lockdown	Overall	s/step
TabICL	–	661.8	2962.4	730.4	1046.8	2.9
TabICL	+GRI	658.0	979.0	713.2	705.4	2.9
TabICL	+time index	622.5	1033.0	594.0	653.4	2.9
TabICL	+time index, +GRI	622.4	985.4	573.5	641.6	3.0
TabPFN	–	653.7	3035.5	680.1	1048.2	3.1
TabPFN	+GRI	696.7	926.2	741.7	731.4	3.1
TabPFN	+time index	621.2	1289.9	616.8	692.8	3.2
TabPFN	+time index, +GRI	629.5	937.2	650.4	664.4	3.2
MELO (deployed, no external signal)		662.9	1063.0	650.9	697.2	–

The same grid on a differenced target separates the two models. Table 10 fits each variant to the daily change $\Delta y_t = y_t - y_{t-1}$ with the same features and reconstructs $\hat{y}_t = y_{t-1} + \widehat{\Delta y}_t$. Differencing is a fixed stationarising prior, and it competes with whatever adaptation channel the model already has. TabICL, whose in-context window tracks the level once the time index reveals where the present sits, loses from the transform wherever the time index is present. TabPFN, whose prior expects a stable target distribution, gains in every variant, and its differenced variant with both features is the strongest reference of the campaign. Under differencing the time index still helps TabPFN, so the two devices complement each other. Wrapping this variant alone in the deployed layer raises its overall RMSE by 1.2%: it sits at the end of the ordering in Section 6, with nothing left for the layer to repair. The pool still absorbs it as one more expert (Appendix I).

Table 10: Foundation-model grid on the differenced target $\Delta y_t = y_t - y_{t-1}$; same features, window and protocol as Table 9, forecasts reconstructed as $\hat{y}_t = y_{t-1} + \widehat{\Delta y}_t$ before scoring.

Model	Features	Pre-lockdown	Lockdown	Post-lockdown	Overall	s/step
TabICL	–	650.8	2897.8	673.4	1015.9	2.9
TabICL	+GRI	645.5	991.3	743.6	710.4	3.0
TabICL	+time index	604.3	1879.8	639.3	785.8	3.0
TabICL	+time index, +GRI	606.4	972.4	697.0	671.2	3.2
TabPFN	–	645.1	2637.4	650.9	953.7	3.5
TabPFN	+GRI	639.0	754.6	757.6	689.7	3.4
TabPFN	+time index	607.2	1223.8	658.5	689.3	3.2
TabPFN	+time index, +GRI	605.9	829.7	616.2	628.9	3.2

C.3 BOOTSTRAP UNCERTAINTY FOR THE MAIN COMPARISON

MLpol is deterministic given its inputs and the bases are fit once, so Table 1 carries no seed-level variability. The remaining uncertainty is sampling variability of the finite test trajectory: on a different but comparable stretch of the same length, how far would the numbers move? We quantify it with the paired moving-block bootstrap quoted in Section 5.1, extended here to every regime and every method.

Protocol. For each method the basic observation is the daily squared-error sequence. Within each window (overall and the six regimes) we draw block start positions uniformly, concatenate blocks of B consecutive days, and truncate to the window length; the replicate statistic is the RMSE of the resampled sequence. Blocks are $B = 30$ days everywhere except the 56-day lockdown, where $B = 7$ keeps the set of valid block starts non-trivial. All methods share the same block draws within a replicate, so pairwise differences are paired. We use $R = 2{,}000$ replicates and seed 42. The bootstrap resamples the per-step sequences behind Table 1, so the point estimates match that table.

Table 11: Per-method RMSE (MW) with 95% moving-block bootstrap CIs ($R = 2{,}000$, $B = 30$ days, $B = 7$ inside the lockdown). CIs are rounded to the nearest MW.

Method	Overall	Pre-COVID	Lockdown	Recovery	Crisis	Reduction	2023
MELO (deployed)	686.2 [624, 743]	662.9 [532, 728]	1063.0 [636, 1296]	655.7 [560, 733]	660.5 [514, 810]	815.8 [688, 944]	580.9 [475, 651]
MLpol on EWLS only	691.3 [625, 747]	684.7 [536, 742]	1072.0 [649, 1331]	655.6 [563, 732]	657.7 [511, 807]	816.9 [685, 948]	575.4 [471, 649]
MLpol on Base only	987.1 [864, 1124]	747.0 [606, 846]	2244.2 [1516, 2700]	882.3 [735, 984]	775.1 [604, 883]	1419.4 [1302, 1606]	948.2 [778, 1075]
Best fixed cell (hindsight)	704.9 [636, 766]	693.8 [552, 751]	1209.0 [793, 1449]	652.3 [561, 728]	682.1 [540, 825]	815.6 [666, 959]	593.5 [491, 665]
Lag-1 (base)	3455.5 [3311, 3590]	3918.6 [3607, 4135]	2626.3 [2168, 2887]	3420.3 [3219, 3615]	3193.1 [2955, 3359]	3531.4 [3198, 3874]	2849.9 [2638, 2966]
Ridge (base)	2072.6 [1944, 2194]	1999.0 [1790, 2148]	3116.8 [2442, 3533]	1938.3 [1743, 2069]	1930.6 [1751, 2147]	2527.0 [2106, 2877]	1973.3 [1725, 2171]
GAM (base)	1336.0 [1173, 1543]	946.6 [801, 1063]	3201.3 [2365, 3948]	1179.4 [979, 1302]	1545.0 [1329, 1704]	1488.8 [1308, 1784]	1139.5 [881, 1374]
XGBoost (base)	1931.4 [1545, 2305]	842.8 [700, 951]	3972.8 [3033, 4873]	1341.4 [1073, 1545]	1220.9 [993, 1341]	4031.4 [3601, 4710]	2127.9 [1706, 2357]
LightGBM (base)	2082.9 [1672, 2480]	821.9 [643, 949]	4410.9 [3634, 5290]	1460.2 [1159, 1672]	1315.5 [1079, 1421]	4221.4 [3780, 4891]	2498.3 [2005, 2812]

Pairwise differences against MELO. Table 12 reports $\Delta = \mathrm{RMSE}_m - \mathrm{RMSE}_{\mathrm{MELO}}$ with the 95% CI of its paired bootstrap distribution. The paired CIs are much narrower than the marginal CIs above because the methods share the same days' noise. For the γ-grid-only ($\varepsilon_0 = 0$) row of Table 1 the overall difference is $+120.8$ MW, with interval $[+69, +188]$.

Table 12: Paired $\Delta\mathrm{RMSE} = \mathrm{RMSE}_m - \mathrm{RMSE}_{\mathrm{MELO}}$ (MW) with 95% paired bootstrap CIs; positive Δ favours MELO. † marks CIs that cover zero; block lengths and replicates as in Table 11.

Comparator m	Overall	Pre-COVID	Lockdown	Recovery	Crisis	Reduction	2023
MLpol on EWLS only	$+5.1^\dagger$ [−3, +7]	$+21.7^\dagger$ [−2, +25]	$+9.0^\dagger$ [−9, +63]	$-0.1^\dagger$ [−5, +6]	$-2.9^\dagger$ [−7, +1]	$+1.1^\dagger$ [−4, +6]	$-5.5^\dagger$ [−10, +4]
MLpol on Base only	+300.9 [+212, +411]	+84.1 [+31, +171]	+1181.2 [+610, +1745]	+226.6 [+126, +313]	+114.5 [+29, +149]	+603.6 [+485, +804]	+367.3 [+248, +481]
Best fixed cell (hindsight)	+18.6 [+2, +32]	$+30.8^\dagger$ [−1, +35]	+146.0 [+35, +300]	$-3.4^\dagger$ [−19, +12]	+21.6 [+2, +41]	$-0.2^\dagger$ [−34, +22]	$+12.6^\dagger$ [−9, +37]
Lag-1 (base)	+2769.2 [+2630, +2893]	+3255.7 [+3027, +3451]	+1563.4 [+1023, +2087]	+2764.6 [+2589, +2934]	+2532.6 [+2291, +2698]	+2715.6 [+2477, +3003]	+2269.0 [+2053, +2420]
Ridge (base)	+1386.4 [+1281, +1486]	+1336.1 [+1141, +1545]	+2053.9 [+1697, +2302]	+1282.6 [+1145, +1390]	+1270.0 [+1021, +1512]	+1711.2 [+1366, +1966]	+1392.3 [+1190, +1595]
GAM (base)	+649.7 [+508, +835]	+283.6 [+199, +404]	+2138.3 [+1585, +2900]	+523.7 [+355, +647]	+884.5 [+626, +1094]	+673.0 [+479, +1008]	+558.6 [+348, +791]
XGBoost (base)	+1245.2 [+882, +1596]	+179.9 [+110, +285]	+2909.8 [+2258, +3827]	+685.7 [+467, +862]	+560.3 [+388, +643]	+3215.6 [+2786, +3908]	+1547.0 [+1189, +1737]
LightGBM (base)	+1396.6 [+1008, +1767]	+159.0 [+52, +276]	+3348.0 [+2802, +4260]	+804.6 [+551, +989]	+654.9 [+459, +764]	+3405.6 [+2973, +4086]	+1917.4 [+1482, +2202]

MELO beats base-only MLpol in every window. Keeping the raw bases next to the EWLS experts makes no significant difference in any window; they are in the pool for the safety bound of Section 4. The hedge beats the hindsight-best fixed cell overall and in the lockdown, although that cell is chosen with test-period knowledge: a fixed cell keeps one memory through the break, while the hedge moves its weight to the fast cells when the break arrives (Appendix H).

D CROSS-DOMAIN EVALUATION: NYC YELLOW-TAXI DAILY RIDERSHIP

This appendix gives the data, protocol, and full results for the NYC yellow-taxi stream of Section 5.2.

Data. Daily trip counts for 2009–2023 are aggregated from the yearly NYC Open Data yellow-taxi trip-record datasets by server-side daily aggregation of pickup timestamps; rows whose pickup date falls outside the dataset's calendar year are discarded as meter artefacts. December 2011 is absent from the source dataset; the series is reindexed on the full calendar and the affected lag windows are dropped. Four months of the aggregated counts are inconsistent with their neighbours and are kept as fetched: February and April 2011 and July 2012 hold 8–21% of the adjacent months' level, and March 2018 about twice that level. The first three lie in the training window of the bases, and March 2018 in the 2018 burn-in year, which is also the validation year of Appendices B.5 and B.6; the test period 2019–2023 contains no such month. Daily weather (maximum and minimum temperature, precipitation, snowfall) comes from the NOAA GHCN Central Park station (USW00094728).

Protocol. The task is one-step-ahead prediction of the daily count. Features are calendar variables (day-of-week, month, federal holidays and their adjacent days), realised same-day weather as in the RTE-FR benchmark, lagged counts at $1, 7, 14, 28$ days, and rolling means over 7 and 28 days. The base pool contains $M = 5$ forecasters trained once on 2011–2017 and then frozen: a seasonal-naive lag-7 predictor, an OLS on calendar and weather only, and ridge, XGBoost, and LightGBM on the full feature set. The 2018 stream is the burn-in year of the layer and the validation year of the two adaptive references (Appendix B.6). The layer runs the deployed configuration of Section 3, with $\widehat{B}_z^2$ computed on this stream's training window and $K = 78$ EWLS experts: 13 forgetting factors ($h \in [20, 2000]$ plus the $\gamma = 1$ endpoint) times the six values of c. Online evaluation runs over 2019-01-01 to 2023-12-31 ($T = 1{,}826$ days) in four regimes: *pre-COVID* (to 2020-03-11), *collapse* (to 2020-06-30), *partial recovery* (to 2021-12-31), and *new normal* (to 2023-12-31).

Results. MELO improves on base-only MLpol in every regime and on both adaptive references overall (Table 13). MLpol on the raw pool trails the seasonal-naive base in the recovery regime

Table 13: Per-regime and overall RMSE (daily trips) on the NYC yellow-taxi test period 2019–2023. **Bold**: best per column among non-hindsight methods. The italicised row selects its window on the test period.

Method *Regime length (days)*	Pre-COVID *436*	Collapse *111*	Recovery *549*	New normal *730*	Overall *1826*
Calendar+weather OLS (base)	183573	414489	335011	302754	298123
XGBoost (base)	28970	293158	172476	105250	137103
LightGBM (base)	39723	251919	154792	110109	127615
Ridge (base)	22830	37107	30338	28276	28364
Seasonal-naive lag-7 (base)	29903	35548	10730	18106	21354
MLpol on Base only	20729	31183	11181	15508	17188
Rolling OLS (W=365, val-selected)	15653	23625	12337	11419	13796
VIKING (val-selected)	**13092**	13716	6574	10118	10309
γ-grid only (ε_0=0)	15120	18266	7345	10710	11702
Rolling OLS (W=60, hindsight)	*14538*	*16749*	*7131*	*11650*	*11708*
MELO (deployed)	13231	**9284**	**6096**	**9213**	**9599**

because its weights follow cumulative regret: ridge had built a large lead before the collapse, and the weight on the seasonal-naive predictor climbs only from 0.47 to 0.76 across the regime, so the mixture still carries about a quarter of ridge, whose recovery error is almost three times that of the seasonal-naive predictor. Under a paired moving-block bootstrap on the daily squared-loss sequences (30-day blocks, 2,000 replicates), the 95% intervals for the overall RMSE differences are $[+5692, +9352]$ against base-only MLpol, $[+7393, +15699]$ against the best single base, $[+2902, +5474]$ against the val-selected rolling OLS, and $[+1246, +2958]$ against the γ-grid-only ($\varepsilon_0 = 0$) row; all exclude zero.

The raw bases degrade in the order of Section 6. The calendar-and-weather OLS carries the broken exogenous relation and is worst throughout. The tree ensembles extrapolate flat outside the training support, which the ride-hailing decline had already left before 2020. Ridge extrapolates linearly and partly corrects itself, and the seasonal-naive predictor re-anchors within a week. The layer restores calibration on top of all of them.

E Robustness to base-pool diversity: a heterogeneity sweep

The deployed RTE-FR pool has $M = 5$ members spanning autoregressive, linear, additive, and tree families. This appendix asks how much of the reported gain depends on that composition. We enumerate every non-empty subset of the five bases and rerun the complete deployed pipeline on each: MLpol on the subset alone, MLpol on the EWLS pool built from the subset, and MLpol on the combined pool. The plug-in $\widehat{B}_z^2$ is recomputed on each subset's training window; the grids, coupling, ridge, and aggregator are unchanged (31 subsets, RTE-FR 2019–2023). For each pool size $H \in \{1, \dots, 5\}$ we report the mean and the 2.5%–97.5% percentile range across the $\binom{5}{H}$ subsets, so the band shows the sensitivity to which members happen to be available.

RMSE versus pool size. Figure 6 reports overall RMSE against H. Base-only aggregation needs many members: a simplex over frozen members cannot correct a bias they share and can only average their errors. The wrapped pools need few, because the layer re-calibrates whichever members are present, and nine of the ten two-member wrapped pools already beat the five-member base-only aggregate. The exception pairs the two weakest members, Lag-1 and ridge. At every H the hedge tracks the hindsight-best cell of the family the subset generates.

The two sources of diversity are partial substitutes. Figure 7(a) splits the gain. The gain of the layer over base-only aggregation shrinks as members are added, because the raw pool then supplies part of the diversity the layer would otherwise provide; it stays positive for all 31 subsets (panel b). Keeping the raw bases next to the EWLS experts adds a few MW at every $H \geq 2$, independently of the subset and without significance in any window (Appendix C.3).

Family heterogeneity does not replace adaptation under the break. Figure 8 splits the sweep by regime. In the calm regimes base-only aggregation closes most of its gap as H grows; in the

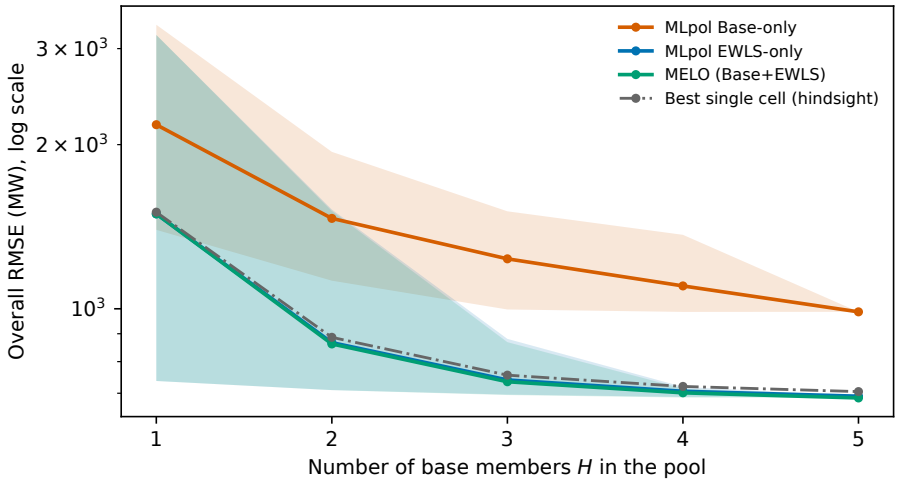


Figure 6: Overall RMSE as a function of the number of base members H (log scale). Curves are means over all $\binom{5}{H}$ subsets; shaded bands are the 2.5%–97.5% percentile range across subsets. The grey dash-dot reference is the hindsight-best standalone EWLS cell per subset.

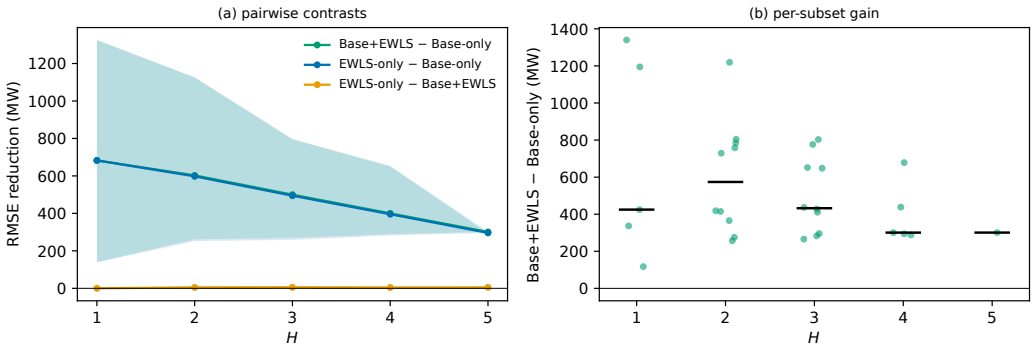


Figure 7: *(a)* Pairwise contrasts against H, means with 2.5%–97.5% bands: Base+EWLS vs. Base-only (green), EWLS-only vs. Base-only (blue), and EWLS-only vs. Base+EWLS (orange). *(b)* Per-subset Base+EWLS gains (points) with per-H medians (bars).

lockdown and the 2022–23 shocks it stays far above the wrapped pools at every H. Frozen members fail together under a break however diverse the pool, so only the adaptation layer can track it.

Marginal members. Table 14 adds one member at a time to the deployed pool and reruns the layer unchanged. A member helps when it carries structure the pool does not span. The temperature-only and calendar-only XGBoost members carry too little information to help. The no-lag variants keep the exogenous response and drop the lag features through which the other members re-anchor, so they add a direction the pool lacks; their gain overall comes with a loss in the lockdown, where a member without lags is the most wrong.

Deliberately overfit members. The last two rows probe overfit members. Multiplying the tuned XGBoost's boosting rounds by ten fits the training window 2.8 times more closely (train RMSE 202 against 567 MW) and is slightly worse out of sample. In the pool it helps, with a mean MLpol weight of 3.5%: the layer uses the extra residual diversity such a member carries and has no need to suppress it. Raising the depth to eight and removing all regularisation produces a member that memorises the training window outright (train RMSE 0.17 MW) and is still only 1% worse out of sample; its mean weight stays near the uniform level, and the aggregate does not change significantly. Overfitting does

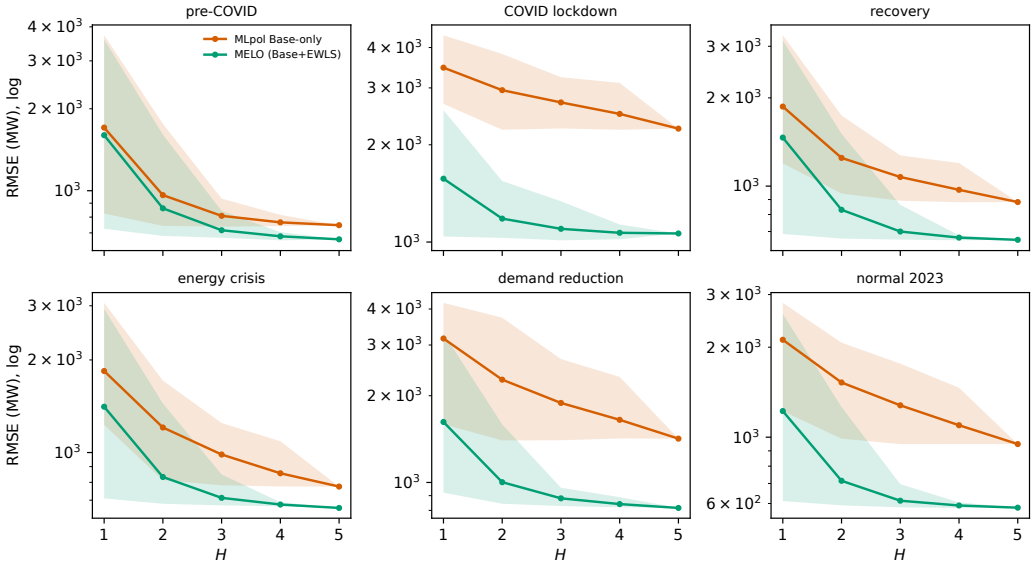


Figure 8: Per-regime RMSE as a function of H (log scale), means with 2.5%–97.5% bands.

Table 14: Adding one member to the deployed five-member pool (RTE-FR 2019–2023, deployed pipeline rerun unchanged; five-member reference 686.2). Δ is the reduction in overall RMSE; brackets are paired moving-block bootstrap 95% CIs (30-day blocks, 2,000 replicates). Feature-ablated variants reuse the tuned parent hyperparameters with a reduced feature list; the overfit variants keep the parent's features and learning rate.

Added member	raw overall	pool overall	Δ overall
Temperature-only XGBoost	4727.1	687.7	-1.5 $[-4.9, +1.9]$
Calendar-only XGBoost	5569.8	693.0	-6.7 $[-10.7, -0.5]$
No-lag XGBoost	3402.8	674.1	$+12.1$ $[+0.5, +21.7]$
No-temperature XGBoost	1817.1	682.3	$+4.0$ $[-5.4, +10.2]$
No-lag LightGBM	3407.3	664.7	$+21.6$ $[+5.0, +35.9]$
All five variants	—	667.7	$+18.6$ $[+2.7, +32.5]$
Overfit XGBoost (rounds $\times 10$)	1990.1	675.5	$+10.7$ $[+4.2, +15.8]$
Memorising XGBoost (depth 8, no regularisation)	1950.8	689.0	-2.7 $[-8.0, +1.9]$

not manufacture a harmful member on these streams. The members that do harm are miscalibrated or degenerate ones: the NYC pool carries a calendar-and-weather OLS at fourteen times the overall RMSE of its best member (Appendix D), and exact duplicates are covered in Appendix F; the layer absorbs both.

Summary. Admitting every available causal base is a safe default: the gain is present at every pool composition tested, and an added member is nearly free at aggregation time.

F Immunity to duplicated experts

The layer must also survive whatever the pool contains. Duplicating one expert exactly spans a direction with no squared input; under forgetting the precision along it decays geometrically and the covariance winds up without bound. Without the ridge the deployed layer fails outright: at $\lambda = 0$ the duplicated pool returns overall RMSE 2.1×10^6. With $\lambda = 10^{-12}$ the same run returns 685.7, indistinguishable from the clean pool's 686.2: the wound-up covariance is capped at $1/\lambda$ by construction, and the stability of the layer no longer rests on floating-point cancellation. A near-duplicate needs no rescue, since even small differences between the copies excite their contrast

direction. Where the ridge is not needed it changes nothing: on the clean pool the test-period RMSE is flat in λ from 0 to 10^{-10} on both streams.

G HEDGING VS. WINDOW SELECTION VS. HINDSIGHT ORACLE: FULL TABLES

This appendix expands Table 3. The pool is the deployed one on each stream (RTE-FR: 5 raw bases + 96 EWLS experts on the (γ, c) grid; NYC: 5 raw bases + 78 EWLS experts), cold-started with the stream, identical to the pool that MELO hedges over. Select-W plays, each day, the cell with the lowest mean squared error over the last W days (expanding window during the first W days). Before the first label it deterministically uses the first raw base in the declared pool order; the oracle splices the hindsight-best cell of each regime, i.e. it is granted both the per-regime argmin and clairvoyant break timing; the best fixed cell is the hindsight argmin of the overall column.

On RTE-FR the hindsight-best cells are $(c{=}10^2, h{=}469)$ pre-COVID and in the crisis, $(c{=}10^2, h{\approx}10^3)$ in recovery and demand reduction, and $(c{=}10^2, h{=}1531)$ in 2023: five of six regimes pick $c = 10^2$ and differ only in memory, while the lockdown picks the fastest cell at the largest value, $(c{=}10^{10}, h{=}20)$. On NYC they are $(c{=}10^2, h{=}247)$ pre-COVID, $(c{=}10^{10}, h{=}70)$ in the collapse, $(c{=}10^2, h{=}162)$ in partial recovery, and $(c{=}10^4, h{=}2000)$ in the new normal.

Table 15: RTE-FR 2019–2023, RMSE in MW, all selection windows. Switch counts: $W{=}15$: 791, $W{=}30$: 632, $W{=}60$: 531, $W{=}90$: 390, $W{=}180$: 279, $W{=}365$: 153 (in 1,734 days).

Method	Pre-COVID	Lockdown	Recovery	Crisis	Reduction	2023	Overall
Select $W{=}15$	745.3	994.1	690.3	734.3	901.7	619.9	740.7
Select $W{=}30$	702.6	988.2	694.3	719.0	851.3	632.5	723.4
Select $W{=}60$	729.6	1155.4	667.8	705.3	824.8	621.8	723.2
Select $W{=}90$	752.5	1078.5	675.5	688.5	837.0	626.2	727.1
Select $W{=}180$	747.9	1120.2	667.3	688.2	834.2	590.2	721.8
Select $W{=}365$	742.3	1116.7	674.5	684.8	815.1	591.6	719.8
Best fixed cell $(c{=}10^2, h{=}469)$	693.8	1209.0	652.3	682.1	815.6	593.5	704.9
Oracle splice (hindsight)	693.8	978.6	648.4	682.1	809.6	588.0	690.7
MELO (deployed hedge)	662.9	1063.0	655.7	660.5	815.8	580.9	686.2

Table 16: NYC yellow-taxi test period 2019–2023, RMSE in daily trips, all selection windows. Switch counts: $W{=}15$: 777, $W{=}30$: 566, $W{=}60$: 425, $W{=}90$: 348, $W{=}180$: 250, $W{=}365$: 143 (in 1,826 evaluation days).

Method	Pre-COVID	Collapse	Recovery	New normal	Overall
Select $W{=}15$	14040	5839	6392	9976	10060
Select $W{=}30$	13780	6817	5916	10516	10144
Select $W{=}60$	13889	8130	6029	9910	10013
Select $W{=}90$	13952	7379	6031	9905	9997
Select $W{=}180$	13414	8350	5957	9882	9844
Select $W{=}365$	13496	8641	6163	9638	9828
Best fixed cell $(c{=}10^2, h{=}107)$	13099	8781	5883	9499	9599
Oracle splice (hindsight)	13028	5777	5876	9463	9420
MELO (deployed hedge)	13231	9284	6096	9213	9599

On both streams the hedge ties the oracle splice within bootstrap noise (Section 6) and beats every window rule. On NYC taxi almost all of the splice's point advantage comes from the collapse. The short windows ($W \leq 30$) come closest to the splice there, switch cells most often, and are the two worst windows overall: switching fast helps at the break and hurts in the calm regimes, where the cells are near-tied and the switches follow noise.

H WEIGHT DYNAMICS: PER-EXPERT DETAILS

Section 6 states that at lockdown onset the aggregator moves weight into the EWLS experts with nominal memory $h < 100$. This appendix shows the weight trajectories behind that sentence, from a

diagnostics-instrumented rerun of the deployed 101-member pipeline; the predictions are identical to Table 1. The controlled single-expert pair of Section 6 is Figure 9; the figures below concern the deployed pool.

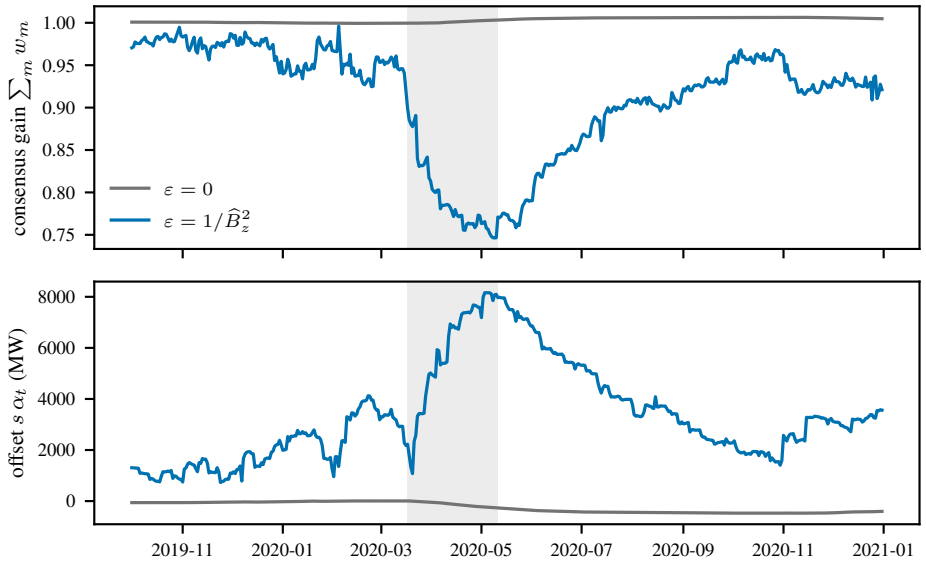


Figure 9: **The inflation creates the fast channels.** One EWLS expert at fixed $\gamma = 1$ on the five-member RTE-FR pool, run from 2012 on the standardized stream, with the per-step covariance inflation toggled between $\varepsilon = 0$ and $\varepsilon = 1/\widehat{B}_z^2$ ($c = 1$ without the coupling). *Top:* consensus gain $\sum_m w_m$. *Bottom:* offset $s\,\alpha_t$ in MW. Shading marks the first lockdown. At $\varepsilon = 1/\widehat{B}_z^2$ the consensus gain and the offset absorb the break within days; at $\varepsilon = 0$ the expert is frozen.

In the top panel, base-only aggregation falls behind the best fixed cell in the lockdown and again through the 2022 shocks, while the two aggregates that contain the EWLS experts end ahead of it; for MELO the difference is significant (Table 12).

The bucket allocation in the bottom panel is the online counterpart of the per-regime optima in Figure 3. In the six weeks before the break the fast bucket ($h < 100$) holds 29% of the mass and averages 44% through the lockdown, while the base band collapses from 26% to 8%. The slow bucket ($h \geq 1000$) then rebuilds through the calm years, from 21% in the lockdown to 39% in 2023, and the base band stays below 12% after the break. The reallocation is loss-driven; MLpol sees no regime label.

Figure 11 shows where the fast cells' weight comes from at lockdown onset: from the base band and from the $\gamma = 1$ cells, which lose almost all of theirs. A moving comparator shortens the optimal memory in the tracking bound (Theorem 1), and the weights follow.

Inside the base band the weights follow the same logic (Figure 12). Before the break the band is carried by the feature-based members, with Lag-1 at 3%. During the lockdown Lag-1, which re-anchors on yesterday's load, takes 68% of what remains: while the frozen feature-to-load relations are broken it is the one base worth keeping, the role the fast cells play among the EWLS experts. From 2021 on the band concentrates on GAM, and the tree models stay near zero through the 2022–23 shocks, in line with their collapse in the demand-reduction regime (Table 1).

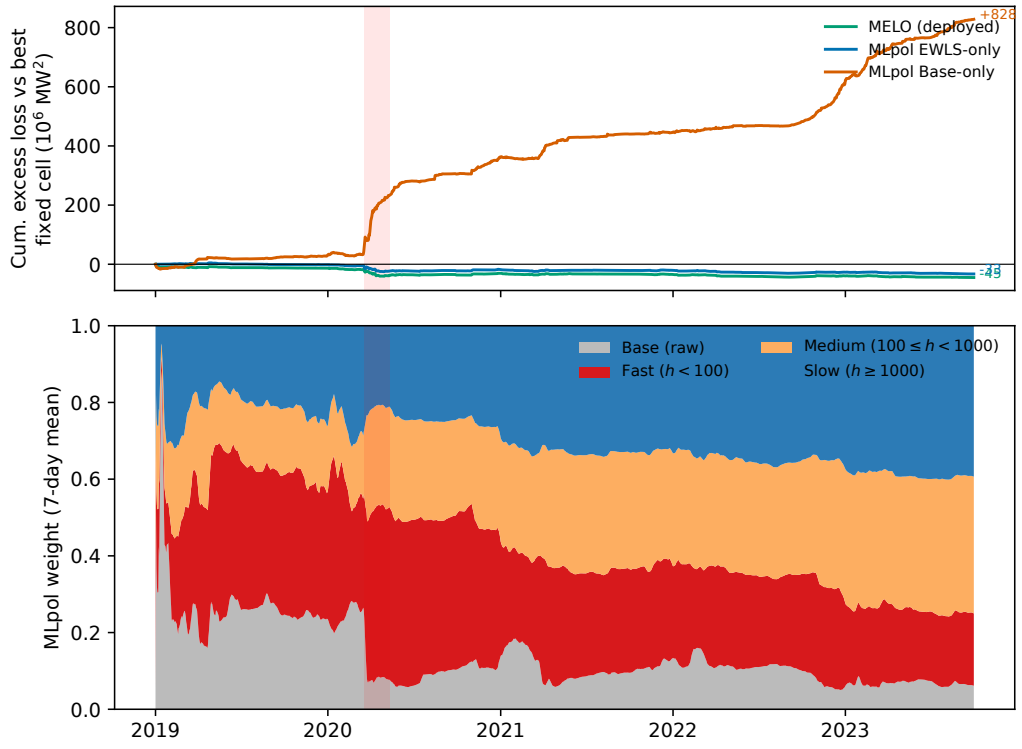


Figure 10: **Aggregate regret and bucketed weights.** *Top:* cumulative excess squared loss of the three aggregates relative to the hindsight-best fixed cell of Table 8 ($h \approx 470$, $c = 10^2$, overall RMSE 704.9), in 10^6 MW^2; values below zero beat that cell. *Bottom:* MLpol weights (7-day rolling mean) with the 96 EWLS cells summed into three buckets by nominal memory and the five raw bases summed into one band. Shading marks the first lockdown.

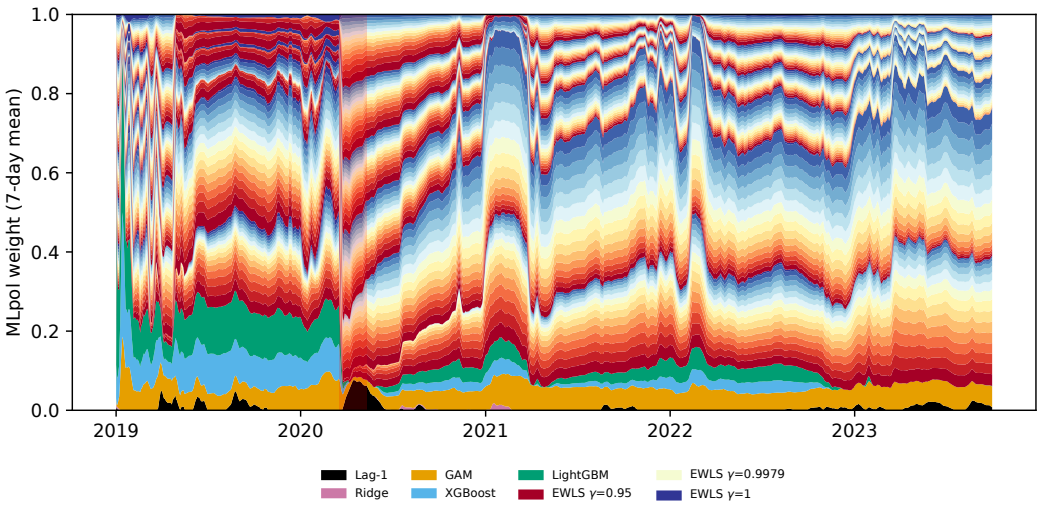


Figure 11: **Fully disaggregated weight trajectory.** All 101 experts as individual strata (7-day rolling mean): the five raw bases in distinct colours at the bottom, the 96 EWLS cells above coloured by forgetting factor on a red-to-blue ramp (red = fastest, $h = 20$; blue = $\gamma = 1$), with the same shade at all six values of c. Shading marks the first lockdown; the opening weeks show the uniform-weight transient.

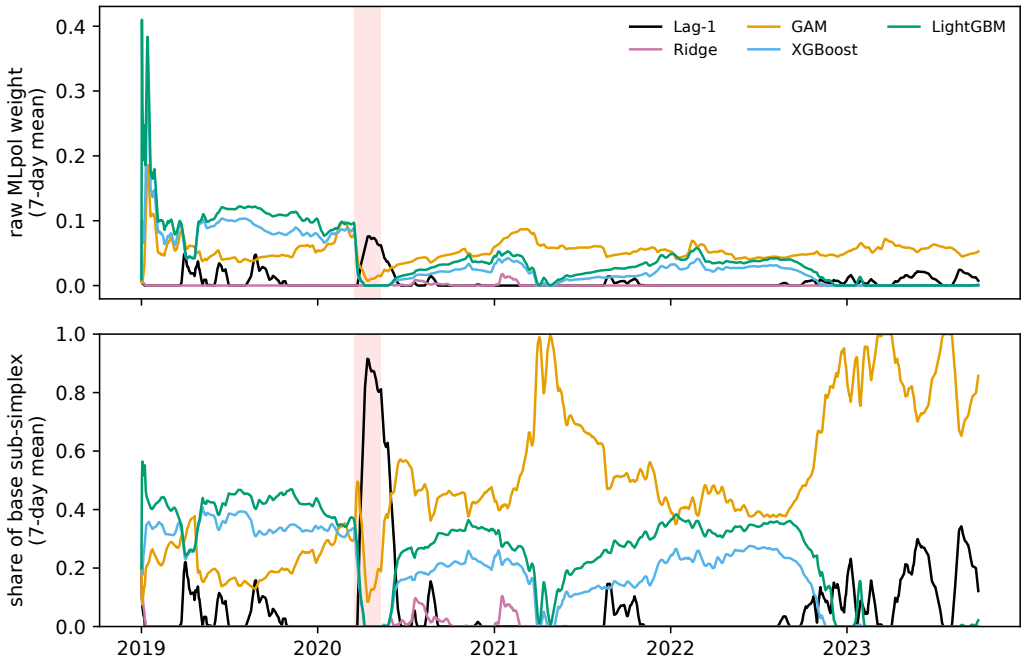


Figure 12: **Within-base reallocation.** *Top:* raw MLpol weight of each of the five raw bases. *Bottom:* the same weights normalised to their sum, i.e. each base's share of the base band. 7-day rolling mean; shading marks the first lockdown.

I EXTENSIBILITY: SIGNAL-INFORMED FORECASTS AS ADDITIONAL EXPERTS

The main experiments use frozen bases and no external regime signal. Any causal forecast whose loss can be evaluated online can join the pool, as a raw expert in the final MLpol layer and optionally as a coordinate of the EWLS design vector.

The oracle inequality covers such an addition. Suppose that a new causal predictor m_X is added to an existing pool of N experts. Applying Lemma A.11 to the enlarged pool gives an aggregate $\hat{y}^{N+1}$ satisfying

$$\sum_{t=1}^{T} \ell_t(\hat{y}_t^{N+1}) \;\le\; \sum_{t=1}^{T} \ell_t(m_X(t)) + \Omega_{N+1,T}(B),$$

where Ω is defined in (40), under the same boundedness conditions as in the MLpol oracle inequality. Adding a signal-informed forecast thus gives the aggregate a new comparator, at the cost of raising the finite-pool overhead from $\Omega_{N,T}(B)$ to $\Omega_{N+1,T}(B)$. The enlarged pool need not beat the old one in finite samples, and the signal enters without any change to the algorithm.

Causal scaling for added foundation models. The frozen base pool uses training-window second moments. The FM streams contain no training-window forecasts, so we fix the FM's normalized second moment at one before deployment: a single-FM wrap uses $\widehat{B}_z^2 = 2$ (one FM plus the intercept), and adding an FM to the five-base design uses $\widehat{B}_{z,6}^2 = \widehat{B}_{z,5}^2 + 1$. Neither calculation reads future FM forecasts or targets, and the target RMS still comes only from the RTE training window.

Table 17 adds the strongest level-target reference of Table 9, the daily-refit TabICL with a global time index and the Oxford Government Response Index. How the reference enters decides how much of its strength survives aggregation. As a raw expert, in the base-only pool or in the deployed one, it can only be weighted against the others, and the aggregate matches the reference. In the EWLS design vector it becomes a coordinate of the affine combination, which the layer re-weights jointly with the bases, and the aggregate beats both the reference and MELO alone, most clearly in the lockdown. A paired moving-block bootstrap (30-day blocks, 2,000 replicates) gives $[+7, +66]$ MW for the overall RMSE difference between the reference and this integration. With the differenced TabPFN reference (Table 10) the same integration reaches 594.1 against the reference's 628.9, with interval $[-9, +62]$ MW: the order is the same, but the difference is no longer significant.

Table 17: Extensibility experiment with the strongest level-target foundation-model reference (TabICL, time index and GRI; Table 9), 2019-01-01–2021-01-15. "Raw only" adds the TabICL forecast only as an expert in the final MLpol pool. "Via EWLS" also includes it among the inputs to the EWLS layer. All aggregation rows use the deployed configuration.

Method	Pre-lockdown	Lockdown	Post-lockdown	Overall
MLpol on Base only	747.0	2244.2	957.9	1007.1
TabICL (time index, GRI)	622.4	985.4	573.5	641.6
MLpol on Base + TabICL	595.0	1051.3	579.5	635.8
MELO (deployed)	662.9	1063.0	650.9	697.2
MELO + TabICL raw only	613.2	1013.2	563.8	636.7
MELO + TabICL via EWLS	582.3	900.9	536.4	597.8