PILOT: PROBE-GUIDED IMPUTATION VIA LOCAL SUBSPACE OPTIMIZATION FOR TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Missing values can substantially degrade the forecasting performance of time series foundation models (TSFMs), while conventional impute-then-forecast pipelines optimize reconstruction rather than downstream forecasting. End-to-end task-aware training could address this mismatch but requires differentiating through the TSFM, which may be too large or accessible only as a black box. We propose **P**robe-guided **I**mputation via **L**ocal Subspace **O**ptimization for **T**ime-Series Forecasting (**PILOT**), a black-box-compatible method that improves imputation using only forward queries to the TSFM. PILOT estimates forecast-loss changes in a local optimization subspace using finite differences on the current batch and a probe sample, and applies a robust correction when the proposed update may conflict with the full-training forecasting objective. We establish conflict-inner-product preservation, second-order finite-difference accuracy, finite-population estimation guarantees, and directional protection under confidence-set coverage. Controlled experiments confirm the predicted 90° conflict boundary and ε^2, η^2 scaling, while experiments on five real-world time series show that PILOT improves test forecast MSE in 77 of 100 paired configurations.

1 INTRODUCTION

Time series are central to industrial maintenance (Kashpruk et al., 2023), clinical monitoring (Bernal et al., 2017), environmental forecasting (Karevan & Suykens, 2020) and financial analysis (Bai & Ng, 2008). Forecasting them has long relied on either classical statistical models, which struggle to capture nonlinear, non-stationary, and long-range temporal dependencies, or task-specific deep models, which often generalize poorly across tasks and require costly redesign and retraining for each new task (Lim & Zohren, 2021; Das et al., 2024). Time-series foundation models (TSFMs) seek to reduce this task-specific burden through large-scale pre-training, enabling zero-shot or few-shot forecasting and lightweight adaptation across downstream datasets (Das et al., 2024; Liu et al., 2025b; Ansari et al., 2024; 2025; Woo et al., 2024; Goswami et al., 2024; Liu et al., 2024).

Deploying TSFMs, however, typically assumes a sufficiently complete input context, whereas sensor drift (Li et al., 2023), outages (Bai & Ng, 2008) and interrupted links (Gong et al., 2023) produce missingness, and a forecaster can amplify the resulting input perturbations (Emmanuel et al., 2021). Common approaches fall into three categories: discarding incomplete timestamps, modelling miss-

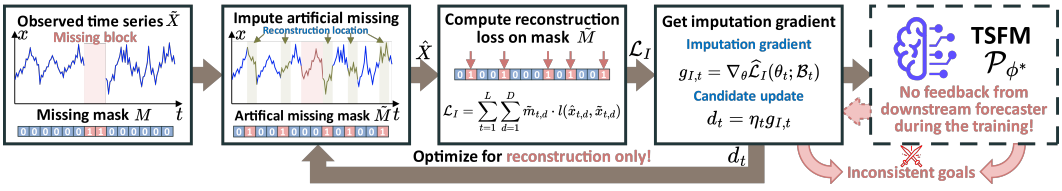


Figure 1: The conventional two-stage impute-then-forecast pipeline. Because the imputer is trained only on reconstruction loss, it receives no feedback about downstream forecast performance.

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.

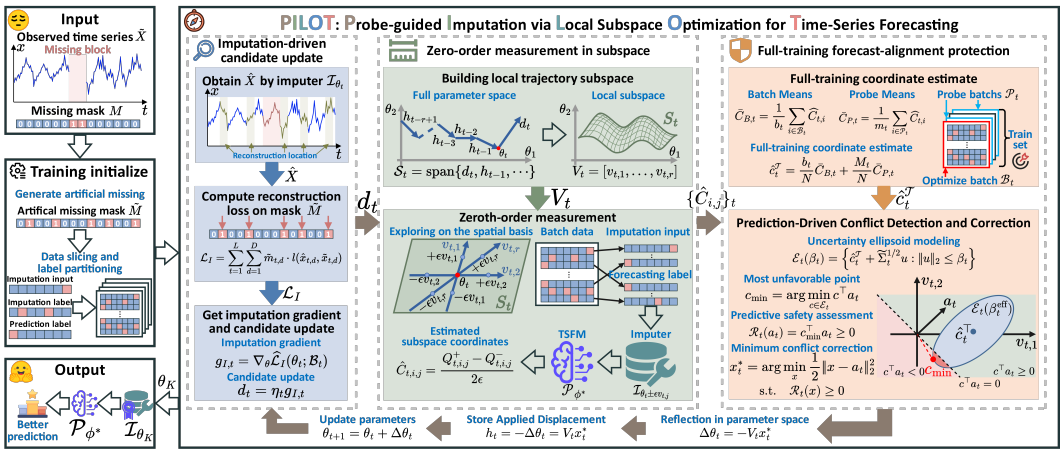


Figure 2: PILOT probes forecast-loss changes in a local parameter subspace, estimates full-training alignment, and releases or corrects the imputer's candidate update. Deployment remains sequential imputation followed by forecasting.

ingness inside the forecasting model as in GRU-D (Che et al., 2018), or completing the series with an imputer and then forecasting. The last approach is modular and naturally compatible with a frozen forecaster (Wen et al., 2022), but **a reconstruction-only imputer receives no feedback from the downstream forecasting objective**, as illustrated in Figure 1.

This objective mismatch is not unique to time series: in a large medical-imaging study, conventional imputation metrics were essentially uncorrelated with downstream performance ($\rho = 0.007$) (Shadbahr et al., 2023). More generally, reconstruction quality is only a proxy for task utility, and a reconstruction-improving update can be misaligned with the downstream objective (Zhou et al., 2024; Whang et al., 2023; Kumar et al., 2024). **For task-aware imputation, a good reconstruction should be not only close to the missing truth, but also compatible with the downstream forecasting objective.**

A direct approach is to optimize the imputer jointly with the forecasting loss when the downstream model is differentiable (Kim et al., 2020; 2023; Hao et al., 2025). That assumption is less attractive for large or externally served TSFMs: backpropagation through the forecaster can be expensive, an endpoint may expose only inference, and modifying an inference-oriented deployment to support gradient propagation weakens the modularity that motivates the impute-then-forecast pipeline in the first place.

We therefore consider a setting in which the TSFM is frozen and queryable but need not expose parameters, gradients, or its computation graph, and propose **P**robe-guided **I**mputation via **L**ocal Subspace **O**ptimization for **T**ime-Series Forecasting (**PILOT**). Figure 2 gives an overview. At each training step, PILOT constructs a local subspace from the optimizer's current candidate descent direction and recently applied directions, then estimates per-window forecast-loss gradient coordinates in that subspace using symmetric finite differences. The current optimization batch is treated as observed after conditioning on the proposed update, while an additional probe set is sampled without replacement from the remaining training windows. This yields a finite-population estimator of the full-training coordinate mean and an associated sampling covariance with the finite-population correction. Because $d_t \in \mathrm{span}(V_t)$, the subspace projection preserves the conflict inner product exactly. PILOT applies the raw optimizer step only when it is robustly feasible for the resulting uncertainty set; otherwise it computes the smallest correction satisfying the robust full-training constraint. PILOT differentiates only through the imputer, introduces no additional parameters, and requires neither access to the forecaster's parameters nor backpropagation through its computation graph.

Our contributions are: (1) **A local formulation of task-aware imputer updates** (Section 3): we characterize when a reconstruction-driven parameter update conflicts with the downstream forecast objective by examining the optimizer's proposed displacement, and express first-order forecast pro-

tection as a directional constraint on that update. (2) **Finite-population partial probing with robust correction**: conditioning on the current optimization batch and sampling additional probe windows without replacement from its complement yields an unbiased estimator of the full-training finite-difference coordinate mean together with its finite-population covariance; the candidate-containing subspace preserves conflict alignment, while sampling uncertainty informs robust feasibility before releasing an update, with central-difference approximation error characterized separately (Theorems 2–4; proofs in Appendix C). (3) **A black-box-compatible training wrapper** that differentiates only through the imputer, uses the frozen forecaster exclusively through forward evaluations, and introduces no trainable parameters; deployment remains sequential imputation followed by forecasting (Algorithm 1 in Appendix D). (4) **Controlled and real-task validation**: the predicted conflict boundary and error scalings on an analytic testbed, and evaluation on five real series with four frozen TSFMs and five imputers, reporting paired test forecast MSE.

2 Problem Formalization

Let $\tilde{X} \in \mathbb{R}^{T\times D}$ denote an incomplete multivariate time series, and let $M \in \{0,1\}^{T\times D}$ be its missingness mask, with $M_{t,d} = 1$ indicating a missing entry. An imputer $\mathcal{I}_\theta$ fills the missing entries while preserving the observed ones, producing $\hat{X} = \mathcal{I}_\theta(\tilde{X}, M)$. A frozen forecaster $\mathcal{P}_{\phi^*}$ then maps an imputed lookback window of length L to a forecast over the next H steps. For a labeled training window i, define the forecast loss induced by the imputer parameters as

$$Q_i(\theta) = \mathcal{L}_P\Big(\mathcal{P}_{\phi^*}(\mathcal{I}_\theta(\tilde{X}_{i,1:L}, M_{i,1:L})), Y_{i,L+1:L+H}\Big). \tag{1}$$

Let $\mathcal{T} = \{1,\dots,N\}$ index the fixed training windows and define the empirical full-training forecast objective

$$Q_{\mathcal{T}}(\theta) = \frac{1}{N}\sum_{i\in\mathcal{T}} Q_i(\theta). \tag{2}$$

The future target $Y_{L+1:L+H}$ is used only during offline training; deployment remains the standard sequential pipeline of imputation followed by forecasting. The imputer itself is trained by artificially masking observed entries and minimizing a reconstruction loss $\mathcal{L}_I(\theta)$, with gradients propagated only through $\mathcal{I}_\theta$. At optimization step t, let d_t denote the positive descent displacement proposed by the reconstruction optimizer, so that the raw parameter update is $\Delta\theta_t^{\text{raw}} = -d_t$.

Checking whether this update decreases $Q_{\mathcal{T}}$ before applying it would require evaluating the frozen forecaster over the full training set at every step, defeating the purpose of a query-efficient black-box wrapper. We therefore allow only a subset of training windows to be queried at each step; the problem is to use these partial observations to detect when the raw update is likely to increase the full training-set forecast objective and to modify the update so that such increases are controlled despite the unqueried windows.

3 Method

3.1 Minimum correction under a forecast constraint

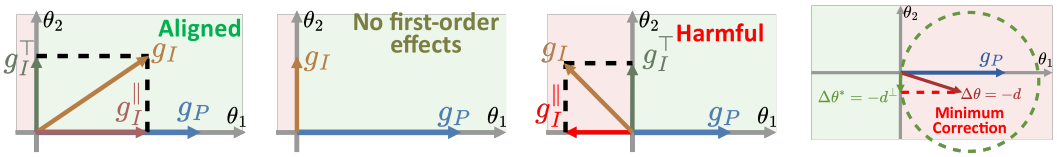
Why reconstruction alone is insufficient. Let X^* denote a complete input, let $\hat{X}$ be its imputed counterpart, and define the imputation error $E_X = \hat{X} - X^*$. Write $\hat{Y}^* = \mathcal{P}_{\phi^*}(X^*)$, $\Delta Y = \mathcal{P}_{\phi^*}(\hat{X}) - \hat{Y}^*$, and $\Delta\mathcal{L}_P = \mathcal{L}_P(\hat{Y}^* + \Delta Y, Y) - \mathcal{L}_P(\hat{Y}^*, Y)$. If the loss has an L_ℓ-Lipschitz gradient, then

$$\Delta\mathcal{L}_P = \big(\nabla_{\hat{Y}}\mathcal{L}_P(\hat{Y}^*, Y)\big)^\top \Delta Y + R_\ell, \qquad |R_\ell| \le \frac{L_\ell}{2}\|\Delta Y\|_F^2. \tag{3}$$

Locally linearizing the forecaster, $\Delta Y \approx (\nabla_X \mathcal{P}_{\phi^*}(X^*))E_X$, yields the **Error propagation equation**

$$\Delta\mathcal{L}_P \approx \langle \nabla_X \mathcal{L}_P(\mathcal{P}_{\phi^*}(X^*), Y), E_X\rangle_F. \tag{4}$$

Thus, to first order, downstream loss depends on the orientation of the imputation error relative to the forecast-loss gradient, not on the reconstruction-error norm alone; Figure 3a illustrates this directional conflict. A reconstruction metric therefore cannot by itself determine whether an imputation perturbation is beneficial or harmful to forecasting.



(a) Gradient conflict. (b) Minimum correction.

Figure 3: Ideal first-order protection in parameter space. Left: a candidate with negative inner product against the forecast gradient harms forecast loss to first order. Right: projection onto the half-space of nonnegative alignment gives the closest safe displacement.

The argument above explains why reconstruction accuracy alone is insufficient, but it is not yet an operational training criterion: the complete input X^* is unavailable at genuinely missing entries, and PILOT modifies imputer parameters rather than imputed inputs directly, changing the imputations and hence the forecast losses. We therefore formulate the downstream protection criterion directly in parameter space.

Predictive protection constraints in parameter space. For each window i, define its parameter-space forecast gradient

$$g_{P,i} = \nabla_\theta Q_i(\theta) = \big(\nabla_\theta \hat{X}_i(\theta)\big)^\top \nabla_{\hat{X}_i} \mathcal{L}_P\big(\mathcal{P}_{\phi^*}(\hat{X}_i(\theta)), Y_i\big). \tag{5}$$

The gradient of the empirical full-training objective is therefore $g_{P,\mathcal{T}} := \nabla_\theta Q_{\mathcal{T}}(\theta) = \frac{1}{N}\sum_{i\in\mathcal{T}} g_{P,i}$. For a parameter update $\Delta\theta$, a first-order expansion gives

$$Q_{\mathcal{T}}(\theta + \Delta\theta) - Q_{\mathcal{T}}(\theta) = g_{P,\mathcal{T}}^\top \Delta\theta + O(\|\Delta\theta\|_2^2), \tag{6}$$

so $g_{P,\mathcal{T}}^\top \Delta\theta \le 0$ is the local condition under which the empirical training forecast objective does not increase to first order.

The ideal safe update. A first-order expansion gives $Q_{\mathcal{T}}(\theta_t - d_t) - Q_{\mathcal{T}}(\theta_t) \approx -g_{P,\mathcal{T},t}^\top d_t$, so if $g_{P,\mathcal{T},t}^\top d_t < 0$, applying the raw update increases the full-training forecast loss to first order. If the gradient were known, we could instead choose the smallest change to the descent displacement that satisfies $g_{P,\mathcal{T},t}^\top x \ge 0$; the applied update $-x$ would then be first-order non-increasing for forecast loss.

Theorem 1 (Minimum-change projection)**.** *For $c \neq 0$ and $a^\top c < 0$, the unique solution of*

$$\min_x \|x - a\|_2^2 \quad \textit{subject to} \quad c^\top x \ge 0 \tag{7}$$

is $x^* = a - \frac{a^\top c}{\|c\|_2^2} c$.

The feasible set is a closed half-space and the objective is strictly convex, so the Euclidean projection is unique (Appendix C.1); with $a = d_t$ and $c = g_{P,\mathcal{T},t}$ this yields the minimum-change safe descent displacement (Figure 3b).

The ideal projection requires the full-training gradient $g_{P,\mathcal{T},t}$, which is unavailable because the forecaster is inference-only; we therefore use forward forecast-loss evaluations to obtain zeroth-order directional measurements.

3.2 Local-subspace zeroth-order measurement

Finite-difference measurements. For a queried training window i, a directional derivative along any unit probe direction $v_{t,j}$ can be estimated by the central finite difference

$$\widehat{C}_{t,i,j} = \frac{Q_i(\theta_t + \varepsilon v_{t,j}) - Q_i(\theta_t - \varepsilon v_{t,j})}{2\varepsilon}. \tag{8}$$

Each direction requires two perturbed evaluations, giving $2r$ directional perturbations for r probes, with $O(\varepsilon^2)$ truncation error under a bounded third directional derivative; forecaster work scales with the number of windows evaluated at each perturbation. Estimating a general gradient with random

directions in the ambient space is, however, query-inefficient: for $u \sim \mathcal{N}(0, I_d)$, the standard estimator has an $O(\varepsilon L\sqrt{d})$ Gaussian-smoothing bias under an L-Lipschitz gradient, and even for affine Q its mean squared error is $\frac{d+1}{k}\|g_P\|^2$, so maintaining fixed relative mean-squared accuracy requires a number of random directions proportional to the parameter dimension (Chen et al., 2019; Gu et al., 2021) (statements and proofs in Appendix C.2). This motivates choosing a small set of directions that retains the candidate's forecast-loss alignment while providing local correction directions.

Candidate-containing local subspace. Let $h_{t-j} = -\Delta\theta_{t-j}$ denote previously applied descent displacements and form

$$\mathcal{S}_t = \mathrm{span}\{d_t, h_{t-1}, \ldots, h_{t-r+1}\}, \tag{9}$$

with an orthonormal basis $V_t \in \mathbb{R}^{d\times r}$ formed with d_t first. Write the candidate and full-training gradient coordinates as $a_t = V_t^\top d_t$ and $c_t^{\mathcal{T}} = V_t^\top g_{P,\mathcal{T},t}$. We include the current candidate so that the subspace retains its first-order alignment with the forecast gradient (Theorem 2); the remaining directions are recent applied displacements from the training trajectory, which may capture more forecast-gradient energy than fresh random directions and provide structured correction directions (Appendix C.4).

Theorem 2 (Subspace preservation and finite-difference accuracy)**.** *Let $P_t = V_tV_t^\top$, $d_t \in \mathcal{S}_t$, and $g_{P,t}^{\mathcal{T}} = \nabla Q_{\mathcal{T}}(\theta_t)$. Then*

$$\langle d_t, P_t g_{P,t}^{\mathcal{T}}\rangle = \langle d_t, g_{P,t}^{\mathcal{T}}\rangle . \tag{10}$$

For each window define $C_{t,i} = V_t^\top \nabla Q_i(\theta_t)$. If, for every window i and every direction $v_{t,j}$, the third directional derivative of Q_i along the segment $[\theta_t - \varepsilon v_{t,j}, \theta_t + \varepsilon v_{t,j}]$ is bounded in absolute value by M_3, then

$$\|\widehat{C}_{t,i} - C_{t,i}\|_\infty \le b_{\varepsilon,t}, \qquad b_{\varepsilon,t} = \frac{M_3\varepsilon^2}{6}. \tag{11}$$

Consequently, if $\bar{C}_{\mathcal{T},t} = N^{-1}\sum_{i\in\mathcal{T}}\widehat{C}_{t,i}$ denotes the conceptual full-training mean of the finite-difference coordinates, then

$$\|\bar{C}_{\mathcal{T},t} - c_t^{\mathcal{T}}\|_2 \le \rho_{\varepsilon,t}, \qquad \rho_{\varepsilon,t} = \sqrt{r}\, b_{\varepsilon,t}. \tag{12}$$

The identity is exact because $P_td_t = d_t$; the coordinate bound follows from the central-difference Taylor remainder (Appendix C.3).

3.3 Finite-population estimation and robust protection

Even with only $2r$ perturbations, querying every training window at every step is too expensive. We condition on the optimizer batch, which is already evaluated to form the candidate, and sample additional probe windows uniformly without replacement from its complement. These probes estimate the full-training coordinate mean and covariance. Let

$$\bar{C}_{B,t} = \frac{1}{b_t}\sum_{i\in\mathcal{B}_t}\widehat{C}_{t,i}, \qquad \bar{C}_{P,t} = \frac{1}{m_t}\sum_{i\in\mathcal{P}_t}\widehat{C}_{t,i}. \tag{13}$$

Because $\mathcal{B}_t$ already determines the optimizer proposal, it is treated as fixed after conditioning on current optimized status $\mathcal{F}_t$ rather than as an independent sample for inference. The estimator of the full-training finite-difference mean is therefore

$$\widehat{c}_t^{\mathcal{T}} = \frac{b_t}{N}\bar{C}_{B,t} + \frac{M_t}{N}\bar{C}_{P,t}. \tag{14}$$

Let $f_t = \frac{m_t}{M_t}$ and let $S_{P,t}$ be the usual sample covariance of the vectors $\{\widehat{C}_{t,i} : i \in \mathcal{P}_t\}$. Define

$$\widehat{\Sigma}_{\mathrm{FP},t} = \left(\frac{M_t}{N}\right)^2 \frac{1-f_t}{m_t} S_{P,t}. \tag{15}$$

Theorem 3 (Finite-population mean and covariance)**.** *Conditionally on $\mathcal{F}_t$, suppose $\mathcal{P}_t$ is sampled uniformly without replacement from $\mathcal{U}_t$. Let $S_{U,t}$ be the finite-population covariance of the conceptual vectors $\{\widehat{C}_{t,i} : i \in \mathcal{U}_t\}$. Then*

$$\mathbb{E}[\widehat{c}_t^{\mathcal{T}} \mid \mathcal{F}_t] = \bar{C}_{\mathcal{T},t}, \tag{16}$$

$$\mathrm{Cov}(\widehat{c}_t^{\mathcal{T}} \mid \mathcal{F}_t) = \left(\frac{M_t}{N}\right)^2 \frac{1-f_t}{m_t} S_{U,t}. \tag{17}$$

Moreover, $\widehat{\Sigma}_{\mathrm{FP},t}$ is conditionally unbiased for this covariance when $m_t \ge 2$.

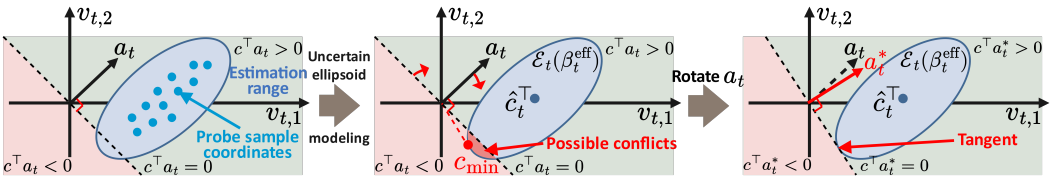


Figure 4: Uncertainty-aware protection in the candidate subspace. Left: the finite-population estimate and covariance define an uncertainty ellipse. Center: the candidate is released only if its worst-case alignment is nonnegative. Right: the robust projection makes the minimum-change correction satisfying the worst-case constraint.

Theorem 3 establishes that the batch-conditioned probe estimate targets the finite-difference coordinates of the entire training set; its covariance measures uncertainty due to querying only part of the residual population, and the factor $1 - f_t$ makes this uncertainty shrink as more residual windows are probed. These mean and covariance estimates supply the center and shape of the uncertainty set used by both protection layers below. The identities follow from sampling without replacement after conditioning and do not require independent time windows (proof in Appendix C.5).

Finite-window sampling can cause two distinct errors: a truly conflicting candidate may appear safe, or a correction based on the estimated mean may leave residual conflict with the full-training gradient. PILOT therefore applies two protection layers. Let

$$\widetilde{\Sigma}_t = \widehat{\Sigma}_{\mathrm{FP},t} + \lambda_t I, \qquad \lambda_t > 0, \tag{18}$$

and suppose a radius β_t is calibrated so that the ellipsoid

$$\mathcal{E}_t(\beta_t) = \left\{ \widehat{c}_t^{\mathcal{T}} + \widetilde{\Sigma}_t^{1/2} u : \ \|u\|_2 \le \beta_t \right\} \tag{19}$$

covers $\bar{C}_{\mathcal{T},t}$ with conditional probability at least $1 - \delta_t$. Define the robust lower alignment

$$\mathcal{R}_t(x) = \widehat{c}_t^{\mathcal{T}\top} x - \beta_t \sqrt{x^\top \widetilde{\Sigma}_t x}. \tag{20}$$

Robust conflict identification. The candidate is released only when $\mathcal{R}_t(a_t) \ge 0$, meaning every finite-difference coordinate in the uncertainty set has nonnegative alignment with a_t; otherwise it is treated as potentially conflicting and sent for correction. This conservative criterion accounts for finite-window sampling uncertainty when deciding whether to release the raw update.

Robust conflict projection. For a candidate that is not released, PILOT computes the minimum-change coordinate whose alignment is nonnegative for every finite-difference coordinate in the uncertainty set:

$$x_t^* = \arg\min_x \frac{1}{2}\|x - a_t\|_2^2 \quad \text{s.t.} \quad \mathcal{R}_t(x) \ge 0. \tag{21}$$

It then applies $\Delta\theta_t = -V_t x_t^*$. Because the constraint uses the worst-case alignment over the uncertainty set, the correction protects alignment with the full-training finite-difference coordinate mean against finite-window sampling uncertainty; a broad uncertainty set can conservatively drive the solution toward zero.

Theorem 4 (Robust full-training directional protection)**.** *For any positive-definite $\widetilde{\Sigma}_t$ and $\beta \ge 0$,*

$$\min_{c \in \{\widehat{c}_t^{\mathcal{T}} + \widetilde{\Sigma}_t^{1/2} u : \|u\|_2 \le \beta\}} c^\top x = \widehat{c}_t^{\mathcal{T}\top} x - \beta \sqrt{x^\top \widetilde{\Sigma}_t x}. \tag{22}$$

If $\mathcal{E}_t(\beta_t)$ has conditional coverage at least $1-\delta_t$, then for the data-dependent coordinate x_t returned by the safe-release rule,

$$\Pr\left(\bar{C}_{\mathcal{T},t}^\top x_t \ge 0 \mid \mathcal{F}_t\right) \ge 1 - \delta_t. \tag{23}$$

Under the finite-difference bound of Theorem 2, the same event implies $(c_t^{\mathcal{T}})^\top x_t \ge -\rho_{\varepsilon,t}\|x_t\|_2$. Thus the true first-order forecast-loss change can have a residual of order $O(\varepsilon^2\|x_t\|_2)$.

Equation 22 identifies $\mathcal{R}_t(x)$ as the worst-case alignment over the uncertainty set: release tests it at a_t, and the projection finds the closest coordinate satisfying it, so either decision protects alignment with the full-training finite-difference mean up to the $O(\varepsilon^2)$ discrepancy stated in the theorem (proof in Appendix C.6; Figure 4).

3.4 Algorithm implementation

At each training step, PILOT forms V_t from the optimizer's raw displacement d_t and recent applied displacements, queries central differences on $\mathcal{B}_t$ and on a probe sample drawn uniformly without replacement from $\mathcal{T} \setminus \mathcal{B}_t$, regularizes the finite-population covariance and calibrates the radius, and evaluates the robust alignment $\mathcal{R}_t(a_t)$. A nonnegative margin releases the raw coordinate; a negative margin triggers the minimum-change robust projection, and the applied displacement enters the history. Rank deficiency or fewer than two probe windows skips protection and applies the raw update. Algorithm 1 in Appendix D gives the complete per-step procedure.

4 Experiments

4.1 Setup

Controlled experiment. In the real tasks, the inference-only forecaster provides neither the full-training forecast gradient nor the full-population effect of alternative updates, so decisions made from finite-difference queries and sampled windows cannot be checked against ground truth. We therefore use an analytically tractable single-step testbed in which these quantities are known. We control the candidate–forecast gradient angle φ, how much of the lateral gradient lies in the local subspace (γ), the perturbation step ε, update size η, window variability σ, probe rate, and miscoverage parameter δ, and record the resulting forecast-loss change, correction rate, measurement error, confidence-set coverage, and intervention rate. Appendix E details the construction, sampling implementation, and sweep ranges.

Real-task. We use five univariate series from the TIME benchmark (Qiao et al., 2026): EWELD Load, Coastal T–S, Port Activity, Amazon Supply Chain, and Current Velocity. We use train/validation/test ranges with a 7:1:2 window-count ratio, context length 256, and forecast horizon 8. Each split receives length-8 synthetic missing blocks at 20% and an independent 20% mask of observed values for imputation supervision. Standardization uses observed training values only. We set standard impute-then-forecast pipeline as **B1** and PILOT as **B2**. Five imputers (SAITS(Du et al., 2023), DeepMVI(Zhang et al., 2021), ModernTCN(Luo & Wang, 2024), TimesNet(Wu et al., 2023), TSLANet(Eldele et al., 2024)) are crossed with four frozen forecasters (Toto2(Khwaja et al., 2026), Chronos-2(Ansari et al., 2025), Sundial(Liu et al., 2025b), TimesFM3(Das et al., 2024)), giving $5 \times 5 \times 4 = 100$ paired configurations. Both branches use random seed 42, AdamW at 10^{-3}, and batch size 128. B2 uses rank 3, $\varepsilon = 10^{-3}$, four probe batches, and miscoverage $\delta = 0.3$.

Component ablation and parameter sensitivity. On Current Velocity with ModernTCN and Chronos-2, we ablate single components: **B3** replaces the historical applied directions with random directions while retaining the current candidate; **B4** replaces the robust minimum-change projection with a point-estimate projection; **B5** replaces the robust release test with a gate based on the estimated mean alignment; and **B6** replaces the minimum-change correction with a step along the estimated forecast-gradient direction. For parameter sensitivity we vary one PILOT setting at a time around the defaults (Appendix G).

4.2 Control experiment results

Figure 5 shows what changes when each controlled factor is varied. In panel (a), increasing the candidate–forecast gradient angle past 90° makes the raw update increase forecast loss, whereas PILOT keeps the change near zero on the conflicting side, matching the predicted conflict boundary. Panel (b) shows the effective correction rate rising from 12.4% at $\gamma = 10^{-3}$ to 98.0% at 0.1 with final forecast loss falling, so additional directions within the local subspace provide useful correction options. Panel (c) confirms the $O(\varepsilon^2)$ central-difference scaling (fitted log–log slope 2.00) with nearly flat sampling error. In panel (d), with all residual windows probed, the raw forecast-loss change grows with slope 1.04 in update size against 2.00 for PILOT, consistent with removing the raw update's first-order loss change and leaving a second-order contribution. Panel (e) shows empirical coverage fluctuating around the nominal $1-\delta$ levels under partial probing and reaching one under full probing. Panel (f) shows interventions concentrating near the boundary (1.0% at 85° vs. 94.2% at 89° at $\delta = 0.1$): when the sampled evidence cannot certify a candidate as nonconflicting,

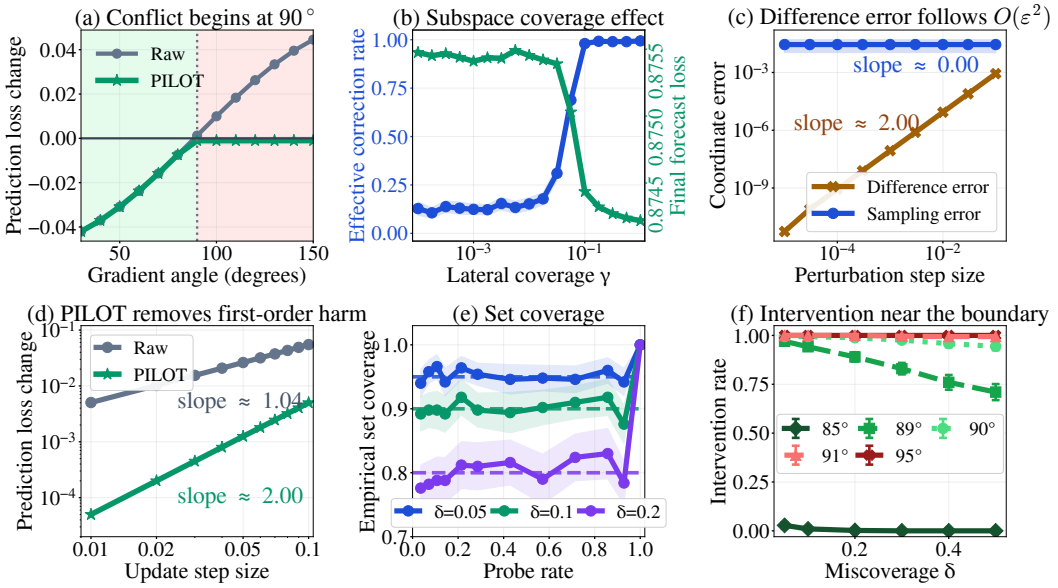


Figure 5: Controlled PILOT diagnostics. (a) Full-population forecast-loss change against the candidate–gradient angle. (b) Effective correction rate and final forecast loss against lateral coverage γ (log scale). (c) Central-difference and sampling errors against perturbation size. (d) Complete forecast-loss changes for Raw and PILOT with all residual windows probed, with fitted log–log slopes. (e) Empirical coverage against the $1-\delta$ reference levels. (f) Intervention rates near the conflict boundary. Shading summarizes repeated finite populations and probe draws (Appendix E).

PILOT's conservative release rule favors intervention over missing a conflict. Appendix E gives the construction and additional results.

4.3 Real-task results

Table 1a reports test forecast MSE comparisons across the full imputer–forecaster product on each dataset; Appendix F lists all 100 pairs. B2 has lower test forecast MSE in 77 pairs, equal MSE in 2, and higher MSE in 21; the mean paired relative change $\Delta = 100 \cdot (\frac{\text{MSE}_{\text{B2}}}{\text{MSE}_{\text{B1}}} - 1)$ is -1.84% and the median is -0.40%. Three patterns stand out. First, the gains are broad but concentrated: the mean magnitude exceeds four times the median, so a subset of pairs carries large improvements while the typical pair changes little. EWELD Load shows the largest average gain (-4.95% with $17/20$ wins); Current Velocity has the fewest wins ($11/20$) and a median near zero; Port Activity wins $18/20$ pairs with a small average (-0.48%). Second, the improvement is not forecaster-specific: mean paired changes range from -0.93% (Sundial) to -2.23% (Toto2), with win counts between 18 and 21 of 25 for every forecaster. Third, the protection is essentially free for reconstruction: pooled test imputation MSE changes by -0.71% on average (median -0.62%), EWELD Load and Port Activity improve in imputation as well (-2.92% and -3.67%), and the only deterioration, on Amazon ($+6.07\%$ mean), has a median near zero, indicating a few outlier pairs rather than a systematic tax.

4.4 Component ablation and parameter sensitivity results

Table 1b compares the component variants on the same configuration. Full PILOT (B2) obtains test forecast MSE 0.39685 against 0.45952 for reconstruction-only B1, and every ablated variant has higher MSE than B2. The margins assign clear roles to the components. The uncertainty-aware release and projection carry the largest share: the point-projection variant B4 (0.45236) retains less than a fifth of B2's improvement over B1. The robust release test contributes next, as the mean-sign gate B5 (0.43566) gives up roughly 62% of the gain. Trajectory-informed directions matter at the basis level, with random directions (B3, 0.44269) recovering only about a quarter of the

(a) Real-task results.

Dataset	Wins	Mean Δ_{P}	Median Δ_{P}	Mean Δ_{I}
EWELD	17/20	−4.95	−1.63	−2.92
Coastal	17/20	−1.46	−0.20	−1.64
Port	18/20	−0.48	−0.32	−3.67
Amazon	14/20	−0.85	−0.69	+6.07
Velocity	11/20	−1.46	−0.11	−1.39
All	77/100	−1.84	−0.40	−0.71

(b) Component ablation results.

Method	Change	MSE	Δ_{B1}
B1	Reconstruction	0.45952	+0.00
B2	Full PILOT	**0.39685**	−13.64
B3	Random directions	0.44269	−3.66
B4	Point projection	0.45236	−1.56
B5	Mean gate	0.43566	−5.19
B6	Direct gradient	0.47712	+3.83

Table 1: Test results. (a) Each dataset pools $5 \times 4 = 20$ imputer–forecaster pairs. Δ_{P} and Δ_{I} are the paired percentage changes from B1 to B2 in forecast and imputation MSE, respectively; the table reports their means and the forecast median. (b) Δ_{B1} is the percent forecast MSE change from B1. Lower MSE is better.

improvement. Directly following the measured forecast gradient (B6, 0.47712) has 3.83% higher MSE than B1, showing that descent along a partially measured forecast gradient need not improve the test forecast objective. In the one-factor scans of Figure 6, every one of the nineteen perturbed settings records a higher MSE than the default, and the default attains the lowest MSE along each axis individually. The probe budget is the most influential axis: two probe batches give 0.54985, worse than not intervening at all, while three and five batches recover to 0.47263 and 0.44042; the miscoverage parameter is comparatively forgiving, with $\delta = 0.2$ (0.39764) essentially matching $\delta = 0.3$, and rank and perturbation step show interior optima around their defaults. These outcomes indicate that probing enough of the training population matters most, with the remaining settings secondary.

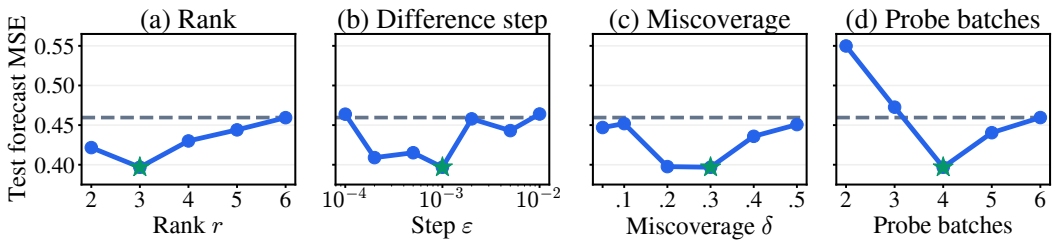


Figure 6: One-factor sensitivity on Current Velocity with ModernTCN and Chronos-2. Each panel changes one PILOT setting and reports test forecast MSE; the green star marks the default B2 setting, and the gray dashed line marks B1 (0.45952).

5 Conclusion

PILOT trains an imputer for a frozen, inference-only forecaster using only forward queries. At each step it measures forecast-gradient coordinates in a candidate-containing local subspace, treats the unqueried training windows as a finite population, and releases the optimizer's update only when robust alignment is nonnegative, otherwise applying the minimum-change robust correction. The subspace preserves the candidate's conflict alignment exactly, and the sampling identities hold conditionally, without independence assumptions on overlapping windows. The controlled experiment reproduces the predicted conflict boundary, the second-order scaling of finite differences, the first- versus second-order step-size scalings of raw and corrected updates, and the nominal coverage of the uncertainty set. On five real series crossed with five imputers and four time-series foundation models, PILOT lowers test forecast MSE in 77 of 100 matched pairs with essentially no imputation cost, and both the component ablation and the one-factor sensitivity scans favour the complete default configuration. Broader multi-seed evaluation, distribution-free calibration of the uncertainty radius, and trajectory-level guarantees that turn single-step protection into monotone improvement are the most direct next steps.

AI USE STATEMENT

The authors used ChatGPT only to assist with English language polishing in selected parts of the manuscript, specifically the Introduction and Related Work sections, and to assist with identifying potentially relevant literature.

All AI-assisted content was carefully reviewed, substantively edited, and verified by the authors. For literature identified with AI assistance, the authors independently verified the existence and bibliographic information of each cited work and checked the original sources to ensure that their content accurately supports the corresponding statements made in the manuscript. The authors take full responsibility for the final content, citations, and scientific validity of the publication.

No AI tools were used to generate experimental results, mathematical proofs, theoretical analyses, or any other technical results presented in this paper.

ETHICS STATEMENT

This work focuses on methodological research and experiments conducted using publicly available datasets(Qiao et al., 2026). It does not involve human subjects, personally identifiable information, or the collection of private data. We are not aware of any immediate ethical concerns specific to the proposed method beyond those generally associated with the use of machine learning models. We encourage responsible use of the proposed approach and adherence to the licenses and terms of use of the datasets and models involved.

REPRODUCIBILITY STATEMENT

We have taken several steps to facilitate the reproducibility of our results. Detailed descriptions of the proposed method and its implementation are provided in the main text and Appendix D. Unless otherwise specified, we report results using the same experimental set described therein. We additionally provide anonymized source code, configuration files, and scripts for reproducing the main experiments at https://anonymous.4open.science/r/ICLR2027_PILOT/. These materials are intended to enable reproduction of the main results reported in the paper.

REFERENCES

Juan Miguel Lopez Alcaraz and Nils Strodthoff. Diffusion-based time series imputation and forecasting with structured state space models. *Trans. Mach. Learn. Res.*, 2023, 2023. URL https://openreview.net/forum?id=hHiIbk7ApW.

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Türkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda-Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=gerNCVqqtR.

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

Jushan Bai and Serena Ng. Forecasting economic time series using targeted predictors. *Journal of Econometrics*, 146(2):304–317, 2008.

James Lopez Bernal, Steven Cummins, and Antonio Gasparrini. Interrupted time series regression for the evaluation of public health interventions: a tutorial. *International journal of epidemiology*, 46(1):348–355, 2017.

Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. BRITS: bidirectional recurrent imputation for time series. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6776–6786, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/734e6bfcd358e25ac1db0a4241b95651-Abstract.html.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085, 2018.

Can Chen, Gabriel L Oliveira, Hossein Sharifi-Noghabi, and Tristan Sylvain. Enhance time series modeling by integrating llm. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024.

Xiangyi Chen, Sijia Liu, Kaidi Xu, Xingguo Li, Xue Lin, Mingyi Hong, and David D. Cox. Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 7202–7213, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/576d026223582a390cd323bef4bad026-Abstract.html.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 793–802. PMLR, 2018. URL http://proceedings.mlr.press/v80/chen18a.html.

Andrea Cini, Ivan Marisca, and Cesare Alippi. Filling the g_ap_s: Multivariate time series imputation by graph neural networks. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=kOu3-S3wJ7.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10148–10167. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/das24c.html.

Wenjie Du, David Côté, and Yan Liu. SAITS: self-attention-based imputation for time series. *Expert Syst. Appl.*, 219:119619, 2023. doi: 10.1016/J.ESWA.2023.119619. URL https://doi.org/10.1016/j.eswa.2023.119619.

Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, and Xiaoli Li. Tslanet: Rethinking transformers for time series representation learning. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12409–12428. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/eldele24a.html.

Tlamelo Emmanuel, Thabiso M. Maupong, Dimane Mpoeleng, Thabo Semong, Banyatsang Mphago, and Oteng Tabona. A survey on missing data in machine learning. *J. Big Data*, 8 (1):140, 2021. doi: 10.1186/S40537-021-00516-9. URL https://doi.org/10.1186/s40537-021-00516-9.

Vincent Fortuin, Dmitry Baranchuk, Gunnar Rätsch, and Stephan Mandt. GP-VAE: deep probabilistic time series imputation. In Silvia Chiappa and Roberto Calandra (eds.), *The 23rd International*

Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy], volume 108 of *Proceedings of Machine Learning Research*, pp. 1651–1661. PMLR, 2020. URL http://proceedings.mlr.press/v108/fortuin20a.html.

Yongshun Gong, Zhibin Li, Jian Zhang, Wei Liu, Yilong Yin, and Yu Zheng. Missing value imputation for multi-view urban statistical data via spatial correlation learning. *IEEE Trans. Knowl. Data Eng.*, 35(1):686–698, 2023. doi: 10.1109/TKDE.2021.3072642. URL https://doi.org/10.1109/TKDE.2021.3072642.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 16115–16152. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/goswami24a.html.

Bin Gu, Xiyuan Wei, Shangqian Gao, Ziran Xiong, Cheng Deng, and Heng Huang. Black-box reductions for zeroth-order gradient algorithms to achieve lower query complexity. *J. Mach. Learn. Res.*, 22:170:1–170:47, 2021. URL https://jmlr.org/papers/v22/20-611.html.

Qi Hao, Runchang Liang, Yue Gao, Hao Dong, Wei Fan, Lu Jiang, and Pengyang Wang. Is precise recovery necessary? A task-oriented imputation approach for time series forecasting on variable subset. *IEEE Trans. Knowl. Data Eng.*, 37(11):6464–6477, 2025. doi: 10.1109/TKDE.2025.3594032. URL https://doi.org/10.1109/TKDE.2025.3594032.

Baoyu Jing, Dawei Zhou, Kan Ren, and Carl Yang. Causality-aware spatiotemporal graph neural networks for spatiotemporal time series imputation. In Edoardo Serra and Francesca Spezzano (eds.), *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM 2024, Boise, ID, USA, October 21-25, 2024*, pp. 1027–1037. ACM, 2024. doi: 10.1145/3627673.3679642. URL https://doi.org/10.1145/3627673.3679642.

Zahra Karevan and Johan A. K. Suykens. Transductive LSTM for time-series prediction: An application to weather forecasting. *Neural Networks*, 125:1–9, 2020. doi: 10.1016/J.NEUNET.2019.12.030. URL https://doi.org/10.1016/j.neunet.2019.12.030.

Nataliia Kashpruk, Cezary Piskor-Ignatowicz, and Jerzy Baranowski. Time series prediction in industry 4.0: a comprehensive review and prospects for future advancements. *Applied sciences*, 13(22):12374, 2023.

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

Jinhee Kim, Taesung Kim, Jang-Ho Choi, and Jaegul Choo. End-to-end multi-task learning of missing value imputation and forecasting in time-series data. In *25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021*, pp. 8849–8856. IEEE, 2020. doi: 10.1109/ICPR48806.2021.9412112. URL https://doi.org/10.1109/ICPR48806.2021.9412112.

Seunghyun Kim, Hyunsu Kim, Eunggu Yun, Hwangrae Lee, Jaehun Lee, and Juho Lee. Probabilistic imputation for time-series classification with missing data. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 16654–16667. PMLR, 2023. URL https://proceedings.mlr.press/v202/kim23m.html.

Sushant Kumar, Sumit Datta, Vishakha Singh, Sanjay Kumar Singh, and Ritesh Sharma. Opportunities and challenges in data-centric AI. *IEEE Access*, 12:33173–33189, 2024. doi: 10.1109/ACCESS.2024.3369417. URL https://doi.org/10.1109/ACCESS.2024.3369417.

Xiao Li, Huan Li, Harry Kai-Ho Chan, Hua Lu, and Christian S. Jensen. Data imputation for sparse radio maps in indoor positioning. In *39th IEEE International Conference on Data Engineering, ICDE 2023, Anaheim, CA, USA, April 3-7, 2023*, pp. 2235–2248. IEEE, 2023. doi: 10.1109/ICDE55515.2023.00173. URL https://doi.org/10.1109/ICDE55515.2023.00173.

Guojun Liang, Prayag Tiwari, Slawomir Nowaczyk, and Stefan Byttner. Higher-order spatio-temporal physics-incorporated graph neural network for multivariate time series imputation. In Edoardo Serra and Francesca Spezzano (eds.), *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM 2024, Boise, ID, USA, October 21-25, 2024*, pp. 1356–1366. ACM, 2024. doi: 10.1145/3627673.3679775. URL https://doi.org/10.1145/3627673.3679775.

Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 18878–18890, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/9d27fdf2477ffbff837d73ef7ae23db9-Abstract.html.

Chenghao Liu, Taha Aksu, Juncheng Liu, Xu Liu, Hanshu Yan, Quang Pham, Silvio Savarese, Doyen Sahoo, Caiming Xiong, and Junnan Li. Moirai 2.0: When less is more for time series forecasting, 2025a. URL https://doi.org/10.48550/arXiv.2511.11698.

Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: Generative pre-trained transformers are large time series models. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32369–32399. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/liu24cb.html.

Yong Liu, Guo Qin, Zhiyuan Shi, Zhi Chen, Caiyin Yang, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Sundial: A family of highly capable time series foundation models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025b. URL https://proceedings.mlr.press/v267/liu25be.html.

Donghao Luo and Xue Wang. Moderntcn: A modern pure convolution structure for general time series analysis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=vpJMJerXHU.

Yonghong Luo, Xiangrui Cai, Ying Zhang, Jun Xu, and Xiaojie Yuan. Multivariate time series imputation with generative adversarial networks. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 1603–1614, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/96b9bff013acedfb1d140579e2fbeb63-Abstract.html.

Yonghong Luo, Ying Zhang, Xiangrui Cai, and Xiaojie Yuan. E2gan: End-to-end generative adversarial network for multivariate time series imputation. In *Proceedings of the 28th international joint conference on artificial intelligence*, pp. 3094–3100. AAAI press Palo Alto, CA, USA, 2019.

Ivan Marisca, Andrea Cini, and Cesare Alippi. Learning to reconstruct missing data from spatiotemporal graphs with sparse observations. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/cf70320e93c08b39b1b29a348097a376-Abstract-Conference.html.

Tong Nie, Guoyang Qin, Wei Ma, Yuewen Mei, and Jian Sun. Imputeformer: Low rankness-induced transformers for generalizable spatiotemporal imputation. In Ricardo Baeza-Yates and Francesco Bonchi (eds.), *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pp. 2260–2271. ACM, 2024. doi: 10.1145/3637528.3671751. URL https://doi.org/10.1145/3637528.3671751.

Zhongzheng Qiao, Sheng Pan, Anni Wang, Viktoriya Zhukova, Yong Liu, Xudong Jiang, Qingsong Wen, Mingsheng Long, Ming Jin, and Chenghao Liu. It's TIME: towards the next generation of time series forecasting benchmarks, 2026. URL https://doi.org/10.48550/arXiv.2602.12147.

Tolou Shadbahr, Michael Roberts, Jan Stanczuk, Julian Gilbey, Philip Teare, Sören Dittmer, Matthew Thorpe, Ramon Viñas Torné, Evis Sala, Pietro Lió, et al. The impact of imputation quality on machine learning classifiers for datasets with missing values. *Communications medicine*, 3 (1):139, 2023.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: conditional score-based diffusion models for probabilistic time series imputation. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 24804–24816, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/cfe8504bda37b575c70ee1a8276f3486-Abstract.html.

Honglin Wen, Pierre Pinson, Jie Gu, and Zhijian Jin. Wind energy forecasting with missing values within a fully conditional specification framework. *CoRR*, abs/2203.08252, 2022. doi: 10.48550/ARXIV.2203.08252. URL https://doi.org/10.48550/arXiv.2203.08252.

Steven Euijong Whang, Yuji Roh, Hwanjun Song, and Jae-Gil Lee. Data collection and quality challenges in deep learning: a data-centric AI perspective. *VLDB J.*, 32(4):791–813, 2023. doi: 10.1007/S00778-022-00775-9. URL https://doi.org/10.1007/s00778-022-00775-9.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53140–53164. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/woo24a.html.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

Jinsung Yoon, William R. Zame, and Mihaela van der Schaar. Estimating missing data in temporal data streams using multi-directional recurrent neural networks. *IEEE Trans. Biomed. Eng.*, 66(5): 1477–1490, 2019. doi: 10.1109/TBME.2018.2874712. URL https://doi.org/10.1109/TBME.2018.2874712.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3fe78a8acf5fda99de95303940a2420c-Abstract.html.

Ying Zhang, Baohang Zhou, Xiangrui Cai, Wenya Guo, Xiaoke Ding, and Xiaojie Yuan. Missing value imputation in multivariate time series with end-to-end generative adversarial networks. *Inf. Sci.*, 551:67–82, 2021. doi: 10.1016/J.INS.2020.11.035. URL https://doi.org/10.1016/j.ins.2020.11.035.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained LM. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/86c17de05579cde52025f9984e6e2ebb-Abstract-Conference.html.

Yuhan Zhou, Fengjiao Tu, Kewei Sha, Junhua Ding, and Haihua Chen. A survey on data quality dimensions and tools for machine learning invited paper. In *IEEE International Conference on Artificial Intelligence Testing, AITest 2024, Shanghai, China, July 15-18, 2024*, pp. 120–131. IEEE, 2024. doi: 10.1109/AITEST62860.2024.00023. URL https://doi.org/10.1109/AITest62860.2024.00023.

A Notation

Table 2: Symbols used in the method and its analysis. A subscript t denotes the current training step.

Symbol	Meaning
$\mathcal{T}, N, Q_i, Q_{\mathcal{T}}$	Training-window index set, its size, per-window forecast loss, and full-training mean forecast loss.
$\theta_t, d_t, \Delta\theta_t$	Current imputer parameters, optimizer-proposed descent displacement, and applied parameter update; the raw update is $-d_t$.
$\mathcal{B}_t, b_t$	Current optimizer batch and its size $\lvert\mathcal{B}_t\rvert$.
$\mathcal{U}_t, M_t$	Residual window set $\mathcal{T} \setminus \mathcal{B}_t$ and its size $N - b_t$.
$\mathcal{P}_t, m_t, f_t$	Probe sample from $\mathcal{U}_t$, its size, and sampling fraction m_t/M_t.
$\mathcal{F}_t$	Information fixed before drawing $\mathcal{P}_t$, including the training windows, current state, optimizer batch, candidate, and basis.
V_t, r, a_t, x_t	Orthonormal local basis, its rank, raw coordinate $V_t^\top d_t$, and applied descent coordinate.
$\widehat{C}_{t,i}, \bar{C}_{\mathcal{T},t}$	Per-window central-difference coordinates and their conceptual full-training mean.
$c_t^{\mathcal{T}}, \widehat{c}_t^{\mathcal{T}}$	Exact full-training gradient coordinates $V_t^\top \nabla Q_{\mathcal{T}}(\theta_t)$ and their probe-based estimate.
$S_{U,t}, \widehat{\Sigma}_{\mathrm{FP},t}$	Residual finite-population covariance and estimated sampling covariance of $\widehat{c}_t^{\mathcal{T}}$.
$\varepsilon, \rho_{\varepsilon,t}$	Central-difference step and bound on full-training coordinate approximation error.
$\delta_t, \beta_t, \mathcal{E}_t$	Nominal miscoverage probability, ellipsoid radius, and coordinate confidence set; nominal coverage is $1 - \delta_t$.
$\mathcal{R}_t(x)$	Worst-case alignment of coordinate x over $\mathcal{E}_t$.

B Related work

B.1 Deep learning models for time-series imputation

Existing deep multivariate imputation methods fall into three architectural families. **Predictive methods** learn a deterministic map from observed to missing values and are simple to train and efficient at inference. Early work is recurrent: GRU-D (Che et al., 2018) introduces time decay for irregular sampling, BRITS (Cao et al., 2018) treats missing values as variables of the computation graph inside a bidirectional RNN, and M-RNN (Yoon et al., 2019) combines a bidirectional RNN with a fully connected network to model temporal and cross-variable dependence separately. Convolutional models such as TimesNet (Wu et al., 2023) reshape one-dimensional series into two dimensions via the FFT and capture multi-periodic variation with two-dimensional convolutions. Graph methods address dependencies between variables: GRIN (Cini et al., 2022) is the first graph recurrent imputation architecture, SPIN (Marisca et al., 2022) adds sparse spatio-temporal attention, Casper (Jing et al., 2024) is causality-aware and HSPGNN (Liang et al., 2024) injects physical priors. Self-attention models are currently the strongest: SAITS (Du et al., 2023) uses diagonally masked self-attention against information leakage, DeepMVI (Zhang et al., 2021) combines Transformers with convolutions to extract long-range periodic and local features, and ImputeFormer (Nie et al., 2024) introduces low-rank induced attention.

Generative methods learn the data distribution, quantify imputation uncertainty and suit high-stakes decisions: GP-VAE (Fortuin et al., 2020) places a Gaussian-process prior in latent space, GRUI-GAN (Luo et al., 2018) and E^{2}GAN (Luo et al., 2019) generate through adversarial training, and diffusion models such as CSDI (Tashiro et al., 2021) and SSSD (Alcaraz & Strodthoff, 2023) model complex distributions. **Foundation-model-driven imputation** pre-trains general temporal representations, as in Timer (Liu et al., 2024) and MOMENT (Goswami et al., 2024), or transfers language models to series (Zhou et al., 2023; Chen et al., 2024).

All of these optimize reconstruction and make little use of downstream feedback. Joint training instead optimizes the imputer and the downstream model together: Kim et al. (Kim et al., 2020) fuse observations with estimates through a gate and train imputation and forecasting jointly, Kim et al. (Kim et al., 2023) pair a generative imputer with a classifier to characterize predictive uncertainty under multiple plausible imputations, and TOI-VSF (Hao et al., 2025) learns imputations that adapt to a variable-subset forecasting target. These methods show that downstream feedback can shape imputation, but they require the downstream model inside the training loop. In our setting the forecaster is frozen and only function values are accessible, so the forecast loss cannot be backpropagated; the contribution of PILOT is the local directional constraint that operates under black-box access, not a restatement of the task-aware objective.

B.2 Time-series foundation models

Time-series foundation models (TSFMs) reduce the per-task burden of forecasting by pre-training a single large model on heterogeneous series and repurposing it across downstream datasets with little or no task-specific training (Lim & Zohren, 2021; Das et al., 2024). Architecturally they fall into several families. Decoder-only language-model-style forecasters quantize the series into tokens and train autoregressively on large corpora: the Chronos family (Ansari et al., 2024; 2025) scales this recipe to zero-shot generalist forecasting, and Toto (Khwaja et al., 2026) adapts the same paradigm toward open scientific modelling of dynamical systems. Patch- and masked-prediction generalists instead frame forecasting as patch-level regression or masked reconstruction: TimesFM (Das et al., 2024), the Moirai family (Woo et al., 2024; Liu et al., 2025a), Timer (Liu et al., 2024) and MOMENT (Goswami et al., 2024) differ in tokenization, objective and context handling but share the premise that one backbone serves arbitrary univariate or multivariate horizons. Diffusion- and flow-based forecasters generate predictive distributions through a stochastic denoising process, exemplified by Sundial (Liu et al., 2025b). A further line repurposes pre-trained language models directly for series, either by treating patches as prompts inside the LLM backbone (Zhou et al., 2023) or by augmenting LLMs with time-series-specific components (Chen et al., 2024), trading architectural novelty for the transfer benefits of language-scale pre-training.

Capabilities and adaptation define the operational interface that downstream methods face. TSFMs are typically evaluated on zero-shot and few-shot generalization across unrelated datasets (Qiao

et al., 2026), often with long context windows and probabilistic outputs (Ansari et al., 2025; Liu et al., 2025b), and are adapted to a task by full fine-tuning, lightweight parameter-efficient tuning, or a task-specific head on frozen representations (Liu et al., 2024; Goswami et al., 2024). Full fine-tuning and parameter-efficient tuning require gradients through the forecaster, whereas training a head on frozen representations requires access to those representations. These interfaces may be unavailable when the deployed TSFM exposes only forward forecasts.

PILOT occupies the complementary corner of this design space. It assumes the TSFM is *frozen* and accessible only through forward queries (matching API-only or inference-optimized deployments where backpropagation is unavailable or deliberately disabled), and it adapts the *imputer* rather than the forecaster. Any member of the above families can serve as the queried model $\mathcal{P}_{\phi^\star}$; our experiments pair five imputers with Sundial, Toto2, Chronos-2 and TimesFM3 and leave their architectures and weights untouched. The TSFM literature asks how to build and adapt one general forecaster; PILOT asks how to train a missing-value pipeline around any such forecaster once it is deployed.

B.3 Gradient conflict and zeroth-order optimization

Multi-task learning studies interference between task gradients. PCGrad (Yu et al., 2020) projects conflicting gradients onto mutually orthogonal directions, though with many tasks successive projections can drift away from the original objective; GradNorm (Chen et al., 2018) reweights tasks by gradient magnitude but does not change any single direction; CAGrad (Liu et al., 2021) searches a minimax update inside a shared trust region. When the objective is reachable only by queries, zeroth-order optimization approximates gradients by finite differences: ZO-AdaMM (Chen et al., 2019) extends adaptive momentum to black-box optimization, and later work reduces query complexity by staged variance reduction (Gu et al., 2021). Such methods minimize the black-box objective itself and estimate full-space descent directions. PILOT instead keeps the imputation loss as the optimized objective and uses black-box queries only to measure forecast-gradient coordinates inside the local subspace of the current candidate update, which separates it both from gradient-surgery methods that need two task gradients and from conventional zeroth-order minimization.

C Proofs and auxiliary derivations

C.1 Proof of Theorem 1

The feasible set is the closed convex half-space $\{x : c^\top x \geq 0\}$ and the objective is strictly convex, so the optimum is unique. The unconstrained point is a and $c^\top a < 0$, hence its Euclidean projection lies on the boundary $c^\top x = 0$. The half-space projection formula gives

$$x^* = a - \frac{c^\top a}{\|c\|_2^2} c. \tag{24}$$

C.2 Gaussian zeroth-order estimation: bias and query cost

The following result quantifies what a full-space random-direction estimator would cost, which is why PILOT restricts measurement to a subspace. Section 3 also uses the exact update alignment satisfied by a locally Lipschitz gradient,

$$Q(\theta - \alpha d_t) - Q(\theta) = -\alpha g_P^\top d_t + O(\alpha^2 \|d_t\|^2), \tag{25}$$

whose first-order term has the sign of $-g_P^\top d_t$.

A full-space random-direction estimator samples $u_1, \ldots, u_k \sim \mathcal{N}(0, I_d)$ independently and forms

$$\hat{g}_P = \frac{1}{k} \sum_{i=1}^{k} \frac{Q(\theta + \varepsilon u_i) - Q(\theta - \varepsilon u_i)}{2\varepsilon} u_i, \tag{26}$$

so that under a local linear approximation each summand is $u_i u_i^\top g_P$ and $\mathbb{E}[u_i u_i^\top] = I_d$.

Theorem 5 (Asymptotic unbiasedness under Gaussian smoothing)**.** *Let $Q : \mathbb{R}^d \to \mathbb{R}$ be differentiable with L-Lipschitz continuous gradient, let $Q_\varepsilon(\theta) = \mathbb{E}_{u \sim \mathcal{N}(0,I)}[Q(\theta + \varepsilon u)]$ be its ε-Gaussian*

smoothing, and let $\hat{g}_P$ be the estimator of Equation 26. Then $\mathbb{E}[\hat{g}_P] = \nabla Q_\varepsilon(\theta)$ and

$$\|\mathbb{E}[\hat{g}_P] - \nabla Q(\theta)\| \le \varepsilon L\, \mathbb{E}[\|u\|] \le \varepsilon L\sqrt{d}, \tag{27}$$

so $\hat{g}_P \to \nabla Q(\theta)$ in expectation as $\varepsilon \to 0$. If Q is three times differentiable with bounded $\|\nabla^3 Q\|$, the smoothing bias tightens to $O(\varepsilon^2 d)$.

Proof. For a single direction let $f(u) = \frac{Q(\theta+\varepsilon u)-Q(\theta-\varepsilon u)}{2\varepsilon}$. The Gaussian Stein identity $\mathbb{E}[f(u)u] = \mathbb{E}[\nabla_u f(u)]$ with $\nabla_u f(u) = \frac{\nabla Q(\theta+\varepsilon u)+\nabla Q(\theta-\varepsilon u)}{2}$ gives

$$\mathbb{E}\left[\tfrac{Q(\theta+\varepsilon u)-Q(\theta-\varepsilon u)}{2\varepsilon} u\right] = \tfrac{1}{2}\mathbb{E}[\nabla Q(\theta+\varepsilon u) + \nabla Q(\theta-\varepsilon u)] = \mathbb{E}[\nabla Q(\theta+\varepsilon u)],$$

where the last step uses the symmetry of the Gaussian distribution. Exchanging expectation and gradient by the Leibniz rule, $\mathbb{E}[\nabla Q(\theta+\varepsilon u)] = \nabla_\theta \mathbb{E}[Q(\theta+\varepsilon u)] = \nabla Q_\varepsilon(\theta)$, and averaging k independent samples yields $\mathbb{E}[\hat{g}_P] = \nabla Q_\varepsilon(\theta)$. The L-Lipschitz continuity of ∇Q gives $\|\nabla Q(\theta + \varepsilon u) - \nabla Q(\theta)\| \le \varepsilon L\|u\|$, hence $\|\mathbb{E}[\hat{g}_P] - \nabla Q(\theta)\| \le \varepsilon L\, \mathbb{E}[\|u\|] \le \varepsilon L\sqrt{d}$. Under the additional third-order bound, $\nabla Q(\theta+\varepsilon u) = \nabla Q(\theta) + \varepsilon \nabla^2 Q(\theta) u + O(\varepsilon^2\|u\|^2)$ and the first-order term cancels in expectation because $\mathbb{E}[u] = 0$, leaving a bias of $O(\varepsilon^2 \mathbb{E}\|u\|^2) = O(\varepsilon^2 d)$. □

For a finite number of queries, let Q be affine so the difference is exact and a single direction gives $z = uu^\top g_P$. With $\mathbb{E}[\|u\|^2 uu^\top] = (d+2)I_d$ from the fourth Gaussian moment, and using $\mathbb{E}[z] = g_P$ together with independence,

$$\mathbb{E}\|z - g_P\|^2 = (d+1)\|g_P\|^2, \tag{28}$$

$$\mathbb{E}\|\hat{g}_P - g_P\|^2 = \frac{d+1}{k}\|g_P\|^2. \tag{29}$$

Even without truncation error the mean squared error grows with the parameter dimension, so a fixed relative accuracy needs a number of directions proportional to d, each costing two forecasts. The subspace construction of Section 3 replaces this with $2r$ queries at a controlled truncation error.

C.3 Proof of Theorem 2

Preservation follows because P_t is an orthogonal projector and $P_t d_t = d_t$:

$$\langle d_t, g^{\mathcal{T}}_{P,t}\rangle = \langle P_t d_t, g^{\mathcal{T}}_{P,t}\rangle = \langle d_t, P_t g^{\mathcal{T}}_{P,t}\rangle .$$

For finite differences, define $h_{i,j}(s) = Q_i(\theta_t + s v_{t,j})$. Taylor expansion with Lagrange remainder at $\pm\varepsilon$ gives

$$\left|\widehat{C}_{t,i,j} - v_{t,j}^\top \nabla Q_i(\theta_t)\right| \le \frac{M_3 \varepsilon^2}{6} = b_{\varepsilon,t}.$$

Taking the maximum over coordinates yields Equation 11. Averaging over the N windows does not enlarge the coordinatewise error, so

$$\|\bar{C}_{\mathcal{T},t} - c_t^{\mathcal{T}}\|_2 \le \sqrt{r}\, \|\bar{C}_{\mathcal{T},t} - c_t^{\mathcal{T}}\|_\infty \le \sqrt{r}\, b_{\varepsilon,t},$$

which is Equation 12.

C.4 Subspace geometry and the value of historical directions

Assume the exact full-training coordinates $c_t^{\mathcal{T}} = V_t^\top g^{\mathcal{T}}_{P,t}$ are known. Since $d_t = V_t a_t$, we have $\|V_t x - d_t\|^2 = \|x - a_t\|^2$ and $(g^{\mathcal{T}}_{P,t})^\top V_t x = (c_t^{\mathcal{T}})^\top x$, so the minimum correction inside the subspace solves $\min_x \frac{1}{2}\|x - a_t\|^2$ subject to $(c_t^{\mathcal{T}})^\top x \ge 0$. When $a_t^\top c_t^{\mathcal{T}} < 0$ the optimum lies on the boundary and

$$x_t^* = a_t - \frac{a_t^\top c_t^{\mathcal{T}}}{\|c_t^{\mathcal{T}}\|^2} c_t^{\mathcal{T}}. \tag{30}$$

In the one-dimensional subspace spanned by d_t alone the correction stops the step. Historical directions enlarge the feasible space so that a non-zero displacement can survive. With $P_t = V_t V_t^\top$,

$$\|V_t x_t^* - d_t\| = \frac{|d_t^\top g^{\mathcal{T}}_{P,t}|}{\|P_t g^{\mathcal{T}}_{P,t}\|} \ge \frac{|d_t^\top g^{\mathcal{T}}_{P,t}|}{\|g^{\mathcal{T}}_{P,t}\|}, \tag{31}$$

so a subspace constraint can require a larger change than the exact full-space projection, and a nested larger subspace weakly reduces the required correction. Writing $q_t = \frac{d_t}{\|d_t\|}$ and $g_{\perp,t} = (I - q_t q_t^\top) g_{P,t}^{\mathcal{T}}$, with the additional $m = r - 1$ orthonormal directions collected in W_t,

$$\|P_t g_{P,t}^{\mathcal{T}}\|^2 = (q_t^\top g_{P,t}^{\mathcal{T}})^2 + \|W_t^\top g_{\perp,t}\|^2. \tag{32}$$

For a uniformly random m-dimensional subspace of $q_t^\perp$, rotational symmetry gives

$$\mathbb{E}\|W_{\text{rand}}^\top g_{\perp,t}\|^2 = \frac{m}{d-1}\|g_{\perp,t}\|^2, \qquad d > 1. \tag{33}$$

Thus historical directions can reduce the correction by capturing forecast-sensitive components of the full-training gradient, although this remains trajectory-dependent and should be assessed empirically.

C.5 Proof of Theorem 3

Condition on $\mathcal{F}_t$. The vectors $\{\widehat{C}_{t,i} : i \in \mathcal{U}_t\}$ are then fixed and $\mathcal{P}_t$ is a simple random sample without replacement of size m_t from the M_t residual windows. Standard finite-population sampling identities, with the population covariance $S_{U,t}$ carrying the $M_t - 1$ denominator that matches the $m_t - 1$ denominator of the sample covariance $S_{P,t}$, give

$$\mathbb{E}[\bar{C}_{P,t} \mid \mathcal{F}_t] = \bar{C}_{U,t}, \qquad \mathrm{Cov}(\bar{C}_{P,t} \mid \mathcal{F}_t) = \frac{1 - f_t}{m_t} S_{U,t},$$

where $\bar{C}_{U,t} = M_t^{-1} \sum_{i \in \mathcal{U}_t} \widehat{C}_{t,i}$. Since

$$\widehat{c}_t^{\mathcal{T}} = \frac{b_t}{N}\bar{C}_{B,t} + \frac{M_t}{N}\bar{C}_{P,t}, \qquad \bar{C}_{\mathcal{T},t} = \frac{b_t}{N}\bar{C}_{B,t} + \frac{M_t}{N}\bar{C}_{U,t},$$

the stated mean and covariance follow. The usual sample covariance $S_{P,t}$ is unbiased for $S_{U,t}$ under simple random sampling without replacement, which proves the last claim. The placement of the correction factor deserves a remark, since it is easy to mis-place. Under approximate normality, the sampling pivot $m_t(\bar{C}_{P,t} - \bar{C}_{U,t})^\top S_{P,t}^{-1}(\bar{C}_{P,t} - \bar{C}_{U,t})$ follows $(1 - f_t)$ times a Hotelling T^2 distribution, because sampling without replacement deflates the covariance of the mean by exactly $1 - f_t$. The Mahalanobis distance induced by $\widehat{\Sigma}_{\text{FP},t}$ divides this pivot by the same factor $1 - f_t$, so the two cancel and the ellipsoid $\mathcal{E}_t(\beta_t)$ with the unadjusted Hotelling radius used in Algorithm 1 has the nominal coverage. In short, the correction factor enters the covariance of the estimator once, through Equation 15, and must not be applied again to the radius.

C.6 Proof of Theorem 4

For the support identity, let $w = \widetilde{\Sigma}_t^{1/2} x$. Then

$$\min_{\|u\|_2 \le \beta} (\widehat{c}_t^{\mathcal{T}} + \widetilde{\Sigma}_t^{1/2} u)^\top x = \widehat{c}_t^{\mathcal{T}\top} x + \min_{\|u\|_2 \le \beta} u^\top w = \widehat{c}_t^{\mathcal{T}\top} x - \beta\|w\|_2,$$

where the last equality is Cauchy–Schwarz, and $\|w\|_2^2 = x^\top \widetilde{\Sigma}_t x$.

Let the event $\mathcal{A}_t = \{\bar{C}_{\mathcal{T},t} \in \mathcal{E}_t(\beta_t)\}$. The safe-release rule and the correction constraint both enforce $\mathcal{R}_t(x_t) \ge 0$, so the support identity yields $\bar{C}_{\mathcal{T},t}^\top x_t \ge 0$ on $\mathcal{A}_t$. Since $\Pr(\mathcal{A}_t \mid \mathcal{F}_t) \ge 1 - \delta_t$, Equation 23 follows. The argument remains valid although x_t is selected from the same probe sample, because the confidence set covers the entire vector $\bar{C}_{\mathcal{T},t}$ simultaneously. The finite-difference bound gives

$$(c_t^{\mathcal{T}})^\top x_t = \bar{C}_{\mathcal{T},t}^\top x_t + (c_t^{\mathcal{T}} - \bar{C}_{\mathcal{T},t})^\top x_t \ge -\rho_{\varepsilon,t}\|x_t\|_2$$

by Cauchy–Schwarz.

Algorithm 1 PILOT at one imputer-training step

Input: imputer $\mathcal{I}_\theta$, frozen forecaster $\mathcal{P}_{\phi^*}$, optimizer batch $\mathcal{B}_t$, history $\mathcal{H}_t$, population size N, rank r, probe size m_t, perturbation ε, miscoverage parameter δ_t

Compute $\mathcal{L}_I$ on $\mathcal{B}_t$ and let the optimizer propose d_t; restore parameters to θ_t while retaining the optimizer state
Form $V_t = \mathrm{Orth}_r([d_t, \mathcal{H}_t])$ and $a_t = V_t^\top d_t$
if $\mathrm{rank}(V_t) = r$ and $m_t \geq 2$ **then**
    Set $M_t = N - |\mathcal{B}_t|$ and sample $\mathcal{P}_t \subset \mathcal{T} \setminus \mathcal{B}_t$ uniformly without replacement
    For each $i \in \mathcal{B}_t \cup \mathcal{P}_t$ and $j = 1, \ldots, r$, query $Q_i(\theta_t \pm \varepsilon v_{t,j})$ using paired random seeds
    Compute $\widehat{C}_{t,i,j} = \frac{1}{2\varepsilon}[Q_i(\theta_t + \varepsilon v_{t,j}) - Q_i(\theta_t - \varepsilon v_{t,j})]$ and stack $\widehat{C}_{t,i} = (\widehat{C}_{t,i,1}, \ldots, \widehat{C}_{t,i,r})^\top$
    Compute $\bar{C}_{B,t} = |\mathcal{B}_t|^{-1} \sum_{i \in \mathcal{B}_t} \widehat{C}_{t,i}$ and $\bar{C}_{P,t} = m_t^{-1} \sum_{i \in \mathcal{P}_t} \widehat{C}_{t,i}$
    Set $\widehat{c}_t^{\mathcal{T}} = (\frac{|\mathcal{B}_t|}{N})\bar{C}_{B,t} + (\frac{M_t}{N})\bar{C}_{P,t}$, $f_t = \frac{m_t}{M_t}$, and let $S_{P,t}$ be the sample covariance of $\{\widehat{C}_{t,i} : i \in \mathcal{P}_t\}$
    Compute $\widehat{\Sigma}_{\mathrm{FP},t} = \frac{(\frac{M_t}{N})^2(1-f_t)S_{P,t}}{m_t}$
    Set $\lambda_t = \max\{\frac{\mathrm{tr}(\widehat{\Sigma}_{\mathrm{FP},t})}{r} \cdot 10^{-8}, 10^{-12}\}$ and $\widetilde{\Sigma}_t = \widehat{\Sigma}_{\mathrm{FP},t} + \lambda_t I$
    if $m_t > r$ **then**
        Set $\beta_t = \sqrt{\frac{r(m_t-1)}{m_t - r} F_{r, m_t - r; 1-\delta_t}}$
    else
        Set $\beta_t = \sqrt{\chi^2_{r;1-\delta_t}}$ ▷ small-sample fallback
    end if
    Compute $\mathcal{R}_t(a_t) = \widehat{c}_t^{\mathcal{T}\top} a_t - \beta_t \sqrt{a_t^\top \widetilde{\Sigma}_t a_t}$
    if $\mathcal{R}_t(a_t) \geq 0$ **then**
        $x_t \leftarrow a_t$ ▷ release under the finite-difference uncertainty set
    else
        Solve $x_t = \arg\min_x \frac{1}{2}\|x - a_t\|_2^2$ subject to $\mathcal{R}_t(x) \geq 0$
    end if
    $\Delta\theta_t \leftarrow -V_t x_t$
else
    $\Delta\theta_t \leftarrow -d_t$ ▷ skip protection when measurements are unavailable
end if
Apply $\Delta\theta_t$ and append $h_t = -\Delta\theta_t$ to $\mathcal{H}_t$

D IMPLEMENTATION DETAILS

Candidate displacement. The optimizer is advanced once from the current parameters to obtain θ_{prop}, the displacement $d_t = \theta_t - \theta_{\mathrm{prop}}$ is recorded, and the parameters are restored while the advanced optimizer state is kept, so all measurements of the step share one parameter position. With AdamW this displacement includes momentum, adaptive scaling and decoupled weight decay; with plain gradient descent it equals ηg_{I_t}. PILOT synchronizes the optimizer's first moment with the applied displacement after each correction.

Differential measurement. The subspace is rebuilt at every step from the current candidate and the most recent $r-1$ applied displacements. After the optimizer batch $\mathcal{B}_t$ has produced d_t and V_t, the probe set $\mathcal{P}_t$ should be drawn uniformly without replacement from $\mathcal{T} \setminus \mathcal{B}_t$. This ordering is important: conditional on $\mathcal{B}_t, d_t, V_t$, it makes the residual training windows a fixed finite population and makes Theorem 3 applicable. During measurement the imputer is put in evaluation mode and both members of a paired query use the same random seed. Each basis direction requires two perturbed evaluations over the queried windows; the number of directional perturbations remains $2r$, while total forecaster work scales with $|\mathcal{B}_t \cup \mathcal{P}_t|$.

Robust solve. Compute $\widehat{\Sigma}_{\mathrm{FP},t}$ from the probe set and regularize it as $\widetilde{\Sigma}_t = \widehat{\Sigma}_{\mathrm{FP},t} + \lambda_t I$; the ridge is numerical stabilization, not sampling covariance. The implementation uses the Hotelling F-based radius, with a χ^2 fallback when the probe count is too small for the subspace rank. The raw coordinate is accepted only when it satisfies the robust constraint. Otherwise Equation 21 is

Table 3: Controlled experiment scan designs by figure panel. In panel (e), probe rate is $m/448$.

Panel	Scan factor	Plotted scan
(a)	Conflict angle φ	30° to 150° in 10° increments; $m = 256$, $\sigma = 0.5$.
(b)	Lateral coverage γ	17 log-spaced values from 10^{-4} to 1; $d = 50$.
(c)	Difference step ε	Nine log-spaced values from 10^{-5} to 10^{-1}; $\kappa = 0.1$.
(d)	Update size η	$0.01, 0.02, \ldots, 0.10$; Raw and full-probe PILOT ($m = 448$).
(e)	Probe size m	$16, 32, 48, 64, 96, 128, 192, 256, 320, 384, 416, 448$; plotted at $\varphi = 120^\circ$, $\sigma = 0.5$, with $\delta \in \{0.05, 0.1, 0.2\}$.
(f)	Miscoverage δ	$\{0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$ at $\varphi \in \{85^\circ, 89^\circ, 90^\circ, 91^\circ, 95^\circ, 120^\circ\}$.

solved in coordinate space; when the uncertainty set contains a neighbourhood of the origin, the only robustly feasible displacement can be zero, which is an intended conservative outcome under the under-correction-averse design.

E Controlled experiment details

E.1 Experimental setup

The controlled imputation loss is $L_I(\theta) = \frac{1}{2}\|\theta - e_1\|_2^2$ at $\theta_0 = 0$, yielding candidate descent displacement $d = -\eta e_1$ and applied raw update $+\eta e_1$. The fixed basis is $V = [e_1, e_2, e_3]$ and

$$q_{\varphi,\gamma} = \cos\varphi\, e_1 + \sin\varphi\big(\gamma e_2 + \sqrt{1-\gamma^2}\, e_4\big), \quad b_i = q_{\varphi,\gamma} + \frac{\kappa}{8}\mathbf{1} + \sigma V \xi_i, \tag{34}$$

where ξ_i are independent standard Gaussian vectors in $\mathbb{R}^3$ centered across the $N = 512$ fixed windows. Each query to window i uses the same target b_i in

$$Q_i(\theta) = \frac{1}{2}\|\theta - b_i\|_2^2 + \frac{\kappa}{4}\sum_{j=1}^{d}(\theta_j + 0.5)^4. \tag{35}$$

Centering gives full-population gradient $g_{P,\mathcal{T}}(0) = -q_{\varphi,\gamma}$ and $g_{P,\mathcal{T}}(0)^\top d = \eta\cos\varphi$. The current batch is a uniform sample of 64 windows; a probe of size m is drawn without replacement from its 448-window complement. The batch coordinates enter the finite-population estimate with weight $\frac{64}{512}$ and probe coordinates with weight $\frac{448}{512}$. Central differences use paired evaluations at $\pm\varepsilon v_j$. The statistical ellipsoid uses the finite-population covariance correction and a Hotelling-style radius at level $1-\delta$. PILOT releases the candidate only if its robust margin is nonnegative; otherwise it computes the minimum-change robust correction.

The default values are $d = 50$, $r = 3$, $\varphi = 120^\circ$, $\gamma = 0.75$, $\eta = 0.05$, $\varepsilon = 10^{-3}$, $\kappa = 0$, $\sigma = 0.5$, $m = 256$, and $\delta = 0.1$. Every stochastic condition uses 50 finite-population seeds and 10 probe draws per population, with methods paired on the same population, optimizer batch, and probe sample.

E.2 Full results

Tables 4–9 give the numerical values for every setting in the six scans underlying Figure 5.

Table 4: Angle scan in Figure 5(a): full-population forecast-loss change $\Delta Q_{\mathcal{T}}$; lower is better.

φ	Raw	PILOT
30°	-0.042051	-0.042051
40°	-0.037052	-0.037052
50°	-0.030889	-0.030889
60°	-0.023750	-0.023750
70°	-0.015851	-0.015851
80°	-0.007432	-0.007432
90°	0.001250	-0.001036
100°	0.009932	-0.001027
110°	0.018351	-0.001020
120°	0.026250	-0.001012
130°	0.033389	-0.001003
140°	0.039552	-0.000993
150°	0.044551	-0.000980

Table 5: Lateral-coverage scan in Figure 5(b) at $\varphi = 120°$. Effective correction removes the true first-order conflict.

γ	Correction (%)	Final loss
0.0001	12.8	0.87548
0.000178	10.6	0.87545
0.000316	13.8	0.87547
0.000562	13.0	0.87544
0.001	12.4	0.87541
0.00178	12.2	0.87544
0.00316	15.4	0.87543
0.00562	13.4	0.87549
0.01	15.2	0.87545
0.0178	17.8	0.87542
0.0316	31.0	0.87539
0.0562	68.8	0.87503
0.1	98.0	0.87444
0.178	99.2	0.87432
0.316	99.0	0.87427
0.562	99.0	0.87424
1	99.4	0.87422

Table 4 shows that Raw and PILOT coincide from $30°$ to $80°$. At the $90°$ first-order boundary, the finite-step Raw loss change becomes positive (0.001250), while PILOT yields -0.001036; by $150°$, Raw reaches 0.044551 and PILOT remains close to -0.001.

Table 5 shows that effective correction stays below 18% through $\gamma = 0.0178$, then rises sharply to 98.0% at $\gamma = 0.1$. Final forecast loss falls overall from 0.87548 at $\gamma = 10^{-4}$ to 0.87422 at $\gamma = 1$, with smaller changes after the correction rate saturates.

Table 6: Perturbation-step scan in Figure 5(c): finite-difference (FD) and total coordinate errors.

ε	FD error	Coord. error
10^{-5}	5.49×10^{-12}	0.02820
3×10^{-5}	7.91×10^{-11}	0.02820
10^{-4}	8.66×10^{-10}	0.02820
3×10^{-4}	7.79×10^{-9}	0.02820
10^{-3}	8.66×10^{-8}	0.02820
3×10^{-3}	7.79×10^{-7}	0.02820
10^{-2}	8.66×10^{-6}	0.02820
3×10^{-2}	7.79×10^{-5}	0.02820
10^{-1}	8.66×10^{-4}	0.02824

Table 7: Update-size scan in Figure 5(d): full-population forecast-loss changes with full probing ($m = 448$).

η	Raw	PILOT
0.01	0.005050	0.000050
0.02	0.010200	0.000200
0.03	0.015450	0.000450
0.04	0.020800	0.000800
0.05	0.026250	0.001250
0.06	0.031800	0.001800
0.07	0.037450	0.002450
0.08	0.043200	0.003200
0.09	0.049050	0.004050
0.10	0.055000	0.005000

Table 6 shows that the isolated finite-difference error rises with ε, as expected from its second-order scaling, but the total coordinate error stays near 0.0282 throughout the sweep. Sampling error therefore dominates the measured coordinate error at these perturbation sizes.

Table 7 shows that increasing η tenfold raises the Raw loss change from 0.005050 to 0.055000, while the PILOT change rises from 0.000050 to 0.005000. PILOT remains lower at every tested step size, and its residual increase follows the second-order pattern shown in Figure 5(d).

Table 8: Probe-size scan in Figure 5(e) at $\varphi = 120°$, $\sigma = 0.5$: empirical confidence-set coverage.

m	$m/448$	$\delta = 0.05$	$\delta = 0.10$	$\delta = 0.20$
16	0.036	0.940	0.892	0.776
32	0.071	0.958	0.898	0.782
48	0.107	0.966	0.898	0.788
64	0.143	0.942	0.892	0.788
96	0.214	0.964	0.918	0.812
128	0.286	0.954	0.898	0.810
192	0.429	0.946	0.894	0.816
256	0.571	0.948	0.902	0.790
320	0.714	0.946	0.910	0.824
384	0.857	0.960	0.918	0.830
416	0.929	0.942	0.876	0.784
448	1.000	1.000	1.000	1.000

Table 9: Miscoverage-parameter scan in Figure 5(f): PILOT intervention rates (%), including 120°.

δ	85°	89°	90°	91°	95°	120°
0.05	2.8	97.0	99.4	100.0	100.0	100.0
0.10	1.0	94.2	99.4	100.0	100.0	100.0
0.20	0.2	89.0	98.8	100.0	100.0	100.0
0.30	0.0	83.0	97.6	99.6	100.0	100.0
0.40	0.0	76.0	95.8	99.4	100.0	100.0
0.50	0.0	71.0	94.4	99.4	100.0	100.0

Table 8 shows that under partial probing, empirical coverage fluctuates around the nominal levels rather than increasing monotonically with m; for example, at $\delta = 0.10$ it is 0.918 for $m = 384$ and 0.876 for $m = 416$. Full probing ($m = 448$) covers the finite-population target in every decision.

Table 9 shows that raising δ from 0.05 to 0.50 reduces interventions at $89°$ from 97.0% to 71.0% and at $85°$ from 2.8% to zero. At $91°$ and above, intervention remains at least 99.4%, so the confidence setting matters most near the conflict boundary.

F Real-task details

F.1 Dataset description

Table 10 lists the five complete univariate TIME series used in the paired experiment.

Table 10: Selected TIME series in the main paired experiment. All are complete before synthetic masking.

Dataset	Domain	Sampling interval
EWELD Load	Electricity load	15 min
Coastal T–S	Coastal observation	15 min
Port Activity	Port activity	1 day
Amazon Supply Chain	Supply chain	1 day
Current Velocity	Water current	10 min

F.2 Data processing and training

All five complete series receive length-8 synthetic missing blocks at rate 0.2 and an independent artificial missing mask at rate 0.2 on observed entries in each split. Observed training values determine the z-score statistics, which are then applied to validation and test data. The time series are windowed and divided into train, validation, and test sets at a 7:1:2 window-count ratio, with look-back 256 and horizon 8. AdamW uses learning rate 10^{-3} and batch size 128. At random seed 42, training has patience 5 on validation forecast loss and no explicit epoch cap. The final continuation state is evaluated on the test split. PILOT uses rank 3, $\varepsilon = 10^{-3}$, four probe batches, $\delta = 0.3$.

F.3 Metrics and reporting conventions

For each matched imputer–forecaster pair we report final test forecast and imputation MSE. For $X \in \{\mathrm{P}, \mathrm{I}\}$, the signed relative change is $\Delta_X = 100 \cdot (\mathrm{MSE}_{X,\mathrm{B2}}/\mathrm{MSE}_{X,\mathrm{B1}} - 1)$ in percent; negative values favour PILOT. Pairs and tabulated percentage changes are computed from unrounded MSE; the displayed MSE values are rounded to five decimal places.

F.4 FULL RESULTS

Table 11: Real-task full results. P_k and I_k denote forecast and imputation MSE for Bk, and both Δ columns are percentage changes. In the two Δ columns, dark green marks improvements above 1% and light green improvements below 1%; dark and light red mark degradations by the same thresholds. Zero changes are unshaded. Dataset abbreviations: EWELD Load (EWELD), Coastal T–S (Coastal), Port Activity (Port), Amazon Supply Chain (Amazon), and Current Velocity (Velocity).

Dataset	Imputer	TSFM	P_1	P_2	Δ_P	I_1	I_2	Δ_I
EWELD	SAITS	Toto2	0.38241	0.30556	-20.09	1.41845	1.26011	-11.16
EWELD	SAITS	Chronos-2	0.23758	0.19030	-19.90	1.41845	1.33968	-5.55
EWELD	SAITS	Sundial	0.23761	0.23673	-0.37	1.34783	1.33525	-0.93
EWELD	SAITS	TimesFM3	0.29161	0.25327	-13.15	1.41845	1.30371	-8.09
EWELD	DeepMVI	Toto2	0.18218	0.17959	-1.42	0.37677	0.38359	+1.81
EWELD	DeepMVI	Chronos-2	0.13603	0.13451	-1.12	0.40451	0.38306	-5.30
EWELD	DeepMVI	Sundial	0.18077	0.17879	-1.10	0.46325	0.42092	-9.14
EWELD	DeepMVI	TimesFM3	0.21049	0.20952	-0.46	0.46325	0.44961	-2.94
EWELD	ModernTCN	Toto2	0.08002	0.08577	+7.19	0.60613	0.54631	-9.87
EWELD	ModernTCN	Chronos-2	0.06224	0.06110	-1.83	0.29953	0.28763	-3.97
EWELD	ModernTCN	Sundial	0.06153	0.06153	+0.00	0.60613	0.60613	0.00
EWELD	ModernTCN	TimesFM3	0.08508	0.07259	-14.68	0.43941	0.44168	+0.52
EWELD	TimesNet	Toto2	0.15715	0.13497	-14.11	0.52786	0.51385	-2.65
EWELD	TimesNet	Chronos-2	0.05670	0.05964	+5.18	0.20963	0.19887	-5.14
EWELD	TimesNet	Sundial	0.11511	0.10461	-9.12	0.52786	0.48964	-7.24
EWELD	TimesNet	TimesFM3	0.11099	0.10748	-3.16	0.19777	0.22316	+12.84
EWELD	TSLANet	Toto2	0.05199	0.05178	-0.39	0.18568	0.18406	-0.88
EWELD	TSLANet	Chronos-2	0.05188	0.05179	-0.19	0.19480	0.19251	-1.17
EWELD	TSLANet	Sundial	0.06265	0.05750	-8.22	0.18178	0.18825	+3.56
EWELD	TSLANet	TimesFM3	0.06196	0.06068	-2.07	0.18215	0.17646	-3.12
Coastal	SAITS	Toto2	1.17422	1.17413	-0.01	0.98960	0.98960	0.00
Coastal	SAITS	Chronos-2	1.02312	1.02443	+0.13	0.98995	1.01939	+2.97
Coastal	SAITS	Sundial	1.01225	1.01203	-0.02	0.98842	0.98208	-0.64
Coastal	SAITS	TimesFM3	1.06662	1.06542	-0.11	0.98782	0.99243	+0.47
Coastal	DeepMVI	Toto2	1.18145	1.17909	-0.20	0.98106	0.98649	+0.55
Coastal	DeepMVI	Chronos-2	1.01535	1.01722	+0.18	1.08202	1.03453	-4.39
Coastal	DeepMVI	Sundial	1.01997	1.01948	-0.05	1.23078	1.21497	-1.28
Coastal	DeepMVI	TimesFM3	1.07569	1.07086	-0.45	0.98106	0.99153	+1.07
Coastal	ModernTCN	Toto2	1.61258	1.50201	-6.86	21.29431	20.02471	-5.96
Coastal	ModernTCN	Chronos-2	1.12421	1.10489	-1.72	21.29431	18.03243	-15.32
Coastal	ModernTCN	Sundial	1.30192	1.18325	-9.12	21.29431	13.33047	-37.40
Coastal	ModernTCN	TimesFM3	1.35683	1.27178	-6.27	21.29431	22.64402	+6.34
Coastal	TimesNet	Toto2	1.23095	1.22580	-0.42	1.24727	1.25248	+0.42
Coastal	TimesNet	Chronos-2	1.01232	1.00212	-1.01	1.33681	1.53406	+14.76
Coastal	TimesNet	Sundial	1.02531	1.02494	-0.04	1.35326	1.38511	+2.35
Coastal	TimesNet	TimesFM3	1.11285	1.08439	-2.56	1.24725	1.18737	-4.80
Coastal	TSLANet	Toto2	1.22445	1.22192	-0.21	1.02375	1.03053	+0.66
Coastal	TSLANet	Chronos-2	1.01700	1.01320	-0.37	1.02375	1.05406	+2.96
Coastal	TSLANet	Sundial	1.02267	1.02181	-0.08	1.02375	1.06998	+4.52
Coastal	TSLANet	TimesFM3	1.08130	1.08196	+0.06	1.02375	1.02341	-0.03
Port	SAITS	Toto2	1.04107	1.04027	-0.08	1.02194	1.01895	-0.29
Port	SAITS	Chronos-2	1.07374	1.06855	-0.48	0.98797	0.97996	-0.81
Port	SAITS	Sundial	1.03807	1.03339	-0.45	1.02904	1.01899	-0.98
Port	SAITS	TimesFM3	1.03567	1.03602	+0.03	1.00413	1.00964	+0.55
Port	DeepMVI	Toto2	1.02610	1.02602	-0.01	1.01160	1.00748	-0.41
Port	DeepMVI	Chronos-2	1.07896	1.07398	-0.46	1.02121	1.00953	-1.14
Port	DeepMVI	Sundial	1.03584	1.03452	-0.13	1.01160	1.00556	-0.60
Port	DeepMVI	TimesFM3	1.02960	1.02926	-0.03	1.01160	1.01078	-0.08
Port	ModernTCN	Toto2	1.12489	1.08688	-3.38	21.94022	15.85492	-27.74
Port	ModernTCN	Chronos-2	1.03383	1.04151	+0.74	1.77000	1.91099	+7.97
Port	ModernTCN	Sundial	1.04484	1.04326	-0.15	1.77000	1.73355	-2.06

Continued from previous page

Dataset	Imputer	TSFM	P_1	P_2	Δ_P	I_1	I_2	Δ_I
Port	ModernTCN	TimesFM3	1.08666	1.06716	-1.79	21.94022	7.20153	-67.18
Port	TimesNet	Toto2	1.04011	1.03587	-0.41	1.40884	1.41263	+0.27
Port	TimesNet	Chronos-2	1.04285	1.03972	-0.30	1.21630	1.08971	-10.41
Port	TimesNet	Sundial	1.03543	1.03204	-0.33	1.40884	1.68836	+19.84
Port	TimesNet	TimesFM3	1.03882	1.03130	-0.72	1.40867	1.50512	+6.85
Port	TSLANet	Toto2	1.04036	1.03691	-0.33	1.10257	1.10154	-0.09
Port	TSLANet	Chronos-2	1.04684	1.03846	-0.80	1.10257	1.10871	+0.56
Port	TSLANet	Sundial	1.03744	1.03439	-0.29	1.10257	1.12006	+1.59
Port	TSLANet	TimesFM3	1.03967	1.03648	-0.31	1.10257	1.11082	+0.75
Amazon	SAITS	Toto2	0.47166	0.46648	-1.10	1.76323	1.74338	-1.13
Amazon	SAITS	Chronos-2	0.38713	0.36469	-5.80	1.80078	1.84259	+2.32
Amazon	SAITS	Sundial	0.45802	0.45746	-0.12	1.80036	1.80518	+0.27
Amazon	SAITS	TimesFM3	0.36937	0.37020	+0.22	1.76215	1.80142	+2.23
Amazon	DeepMVI	Toto2	0.51080	0.50313	-1.50	1.69476	1.87292	+10.51
Amazon	DeepMVI	Chronos-2	0.41030	0.37144	-9.47	1.83529	1.96811	+7.24
Amazon	DeepMVI	Sundial	0.36363	0.38913	+7.01	1.66223	2.65063	+59.46
Amazon	DeepMVI	TimesFM3	0.34914	0.33484	-4.10	1.56877	1.99855	+27.40
Amazon	ModernTCN	Toto2	0.26016	0.23957	-7.91	0.77769	0.70280	-9.63
Amazon	ModernTCN	Chronos-2	0.22045	0.24024	+8.98	0.41694	0.62125	+49.00
Amazon	ModernTCN	Sundial	0.29300	0.29017	-0.97	0.77769	0.74853	-3.75
Amazon	ModernTCN	TimesFM3	0.22141	0.22052	-0.40	0.77769	0.69751	-10.31
Amazon	TimesNet	Toto2	0.22245	0.21998	-1.11	0.26324	0.24532	-6.81
Amazon	TimesNet	Chronos-2	0.22157	0.22524	+1.65	0.25999	0.23723	-8.75
Amazon	TimesNet	Sundial	0.23023	0.22881	-0.62	0.25614	0.25011	-2.35
Amazon	TimesNet	TimesFM3	0.18966	0.19059	+0.49	0.25616	0.25788	+0.67
Amazon	TSLANet	Toto2	0.26315	0.26513	+0.75	0.34694	0.32801	-5.46
Amazon	TSLANet	Chronos-2	0.22541	0.22480	-0.27	0.27568	0.27139	-1.56
Amazon	TSLANet	Sundial	0.29738	0.29514	-0.75	0.34694	0.36046	+3.90
Amazon	TSLANet	TimesFM3	0.23684	0.23196	-2.06	0.34694	0.37545	+8.22
Velocity	SAITS	Toto2	0.50886	0.50439	-0.88	0.74259	0.72621	-2.21
Velocity	SAITS	Chronos-2	0.53794	0.53774	-0.04	0.74259	0.74207	-0.07
Velocity	SAITS	Sundial	0.49189	0.49678	+0.99	0.71864	0.76476	+6.42
Velocity	SAITS	TimesFM3	0.48841	0.48901	+0.12	0.74259	0.72728	-2.06
Velocity	DeepMVI	Toto2	0.49071	0.49405	+0.68	0.61728	0.60354	-2.23
Velocity	DeepMVI	Chronos-2	0.53874	0.52718	-2.15	0.61728	0.60025	-2.76
Velocity	DeepMVI	Sundial	0.49191	0.49417	+0.46	0.61728	0.59729	-3.24
Velocity	DeepMVI	TimesFM3	0.48488	0.47734	-1.55	0.61728	0.58271	-5.60
Velocity	ModernTCN	Toto2	0.45042	0.45059	+0.04	0.47240	0.47043	-0.42
Velocity	ModernTCN	Chronos-2	0.45952	0.39685	-13.64	0.70325	0.61995	-11.84
Velocity	ModernTCN	Sundial	0.51732	0.51732	+0.00	0.70325	0.70325	0.00
Velocity	ModernTCN	TimesFM3	0.42629	0.43199	+1.34	0.47240	0.44784	-5.20
Velocity	TimesNet	Toto2	0.41619	0.41477	-0.34	0.82565	0.87877	+6.43
Velocity	TimesNet	Chronos-2	0.46527	0.46135	-0.84	0.82565	0.88136	+6.75
Velocity	TimesNet	Sundial	0.42214	0.42286	+0.17	0.82565	0.88857	+7.62
Velocity	TimesNet	TimesFM3	0.42948	0.42871	-0.18	0.82596	0.83892	+1.57
Velocity	TSLANet	Toto2	0.37056	0.35696	-3.67	0.29772	0.27616	-7.24
Velocity	TSLANet	Chronos-2	0.42503	0.39197	-7.78	0.29772	0.27286	-8.35
Velocity	TSLANet	Sundial	0.39747	0.39766	+0.05	0.34907	0.34319	-1.68
Velocity	TSLANet	TimesFM3	0.36540	0.35837	-1.92	0.34208	0.32979	-3.59

G Component ablation and parameter sensitivity details

G.1 Component ablation

All variants share the same training and testing setup with the real-task on Current Velocity with ModernTCN and Chronos-2, using random seed 42. B3 replaces recent applied directions with

random directions, B4 uses point projection, B5 replaces robust release by a mean-sign gate, and B6 follows the measured forecast-gradient direction. Table 1b gives their final test forecast MSE.

G.2 Parameter sensitivity

The baseline B2 parameters are rank 3, $\varepsilon = 10^{-3}$, $\delta = 0.3$, and four probe batches. Each row in Table 12 changes one parameter from that baseline while retaining the same data, models, and seed.

Table 12: Parameter sensitivity test forecast MSE on Current Velocity / ModernTCN / Chronos-2.

Parameter	Setting	Test forecast MSE
B1 reference	–	0.45952
B2 reference	Default B2	0.39685
Rank r	2	0.42162
Rank r	4	0.43002
Rank r	5	0.44400
Rank r	6	0.45939
Difference step ε	0.0001	0.46380
Difference step ε	0.0002	0.40898
Difference step ε	0.0005	0.41514
Difference step ε	0.002	0.45790
Difference step ε	0.005	0.44305
Difference step ε	0.01	0.46402
Miscoverage δ	0.05	0.44696
Miscoverage δ	0.1	0.45189
Miscoverage δ	0.2	0.39764
Miscoverage δ	0.4	0.43587
Miscoverage δ	0.5	0.45066
Probe batches	2	0.54985
Probe batches	3	0.47263
Probe batches	5	0.44042
Probe batches	6	0.45952

Across the scanned miscoverage parameters, $\delta = 0.3$ gives 0.39685 and $\delta = 0.2$ gives 0.39764; the other tested values give 0.43587–0.45189. Increasing rank from 2 to 6 gives 0.42162, 0.43002, 0.44400, and 0.45939 against the rank-3 reference of 0.39685. Using two, three, five, or six probe batches gives 0.54985, 0.47263, 0.44042, and 0.45952, respectively. The additional finite-difference steps remain above the default reference, with MSE ranging from 0.40898 to 0.46402.

H Limitations and broader impact

Range of the first-order approximation. Fix the label Y and let $F(X) = \mathcal{L}_P(\mathcal{P}_{\phi^*}(X), Y)$. If the forecaster is twice continuously differentiable in its input, the loss is twice continuously differentiable in the forecast and the relevant derivatives are bounded, then $\nabla_X F$ is locally Lipschitz and

$$|F(X^* + E_X) - F(X^*) - \langle \nabla_X F(X^*), E_X \rangle_F| \le \frac{L_F}{2} \|E_X\|_F^2, \tag{36}$$

so the linear approximation behind Equation 4 carries an $O(\|E_X\|_F^2)$ error and gives only a local directional explanation when the missing rate is high, when early training leaves a large imputation error, or when the imputation leaves the neighbourhood. The second approximation, $\delta\hat{X} \approx \nabla_\theta \hat{X} \Delta\theta$, requires a small enough step: with an excessive learning rate, an abnormal gradient norm or a strongly nonlinear imputer, the projected step may no longer correspond to a linear input perturbation, and the local directional constraint loses explanatory power.

Statistical boundary of finite-population protection. The revised analysis targets the empirical full-training forecast gradient rather than a superpopulation mean of correlated windows. Condi-

tional on the current optimization batch and proposal, sampling probe indices uniformly without replacement makes the unqueried residual windows a fixed finite population, so the unbiasedness and finite-population covariance in Theorem 3 do not require independence of overlapping time-series windows. The probability guarantee in Theorem 4, however, still depends on how the radius β_t is calibrated. A Gaussian/Hotelling-style ellipsoid is a design-based large-sample approximation rather than an exact finite-sample statement for an arbitrary finite population. If exact nominal coverage is required, one can assume bounded coordinate ranges and replace the ellipsoid by a constraint set built from simultaneous coordinatewise finite-population concentration bounds, a confidence box or any ellipsoid enclosing it; this is more conservative but gives a distribution-free confidence set. In either case the robustness claim concerns the full-training empirical objective at the current step, not held-out generalization.

Optimizer boundary. Theorems 1 and 2 are stated for an arbitrary candidate displacement, so the single-step geometry applies to the actual AdamW parameter difference; but only the first moment is synchronized with the applied displacement, while the second moment is updated from the uncorrected gradient, so the optimizer state is not perfectly consistent with the corrected step. Single-step optimality therefore does not extend to the multi-step trajectory, and the recorded history follows the applied displacements, not the proposed ones.

No guarantee of end-to-end improvement. On the confidence-set coverage event, the robust rule ensures $\bar{C}_{\mathcal{T},t}^{\top} x_t \geq 0$ for the full-training finite-difference mean. For the true gradient, Theorem 4 gives only $(c_t^{\mathcal{T}})^{\top} x_t \geq -\rho_{\varepsilon,t}\|x_t\|_2$, leaving a possible first-order residual of order $O(\varepsilon^2\|x_t\|_2)$. It still does not imply a finite-step decrease unless curvature is controlled. If $Q_{\mathcal{T}}$ has an L_t-Lipschitz gradient, a sufficient finite-step condition for the exact gradient is $(c_t^{\mathcal{T}})^{\top} x_t \geq \frac{L_t}{2}\|x_t\|_2^2$; the current algorithm does not estimate such an L_t. Nor does training-set protection imply monotone multi-step convergence or better test performance, since optimizer-state mismatch, non-convex trajectories and train–test shift remain.