NOT ALL VARIABLES AGREE: RELIABILITY-AWARE VARIABLE-WISE GRADIENT SURGERY FOR MULTIVARIATE TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

In data-driven training, multivariate time-series forecasting is usually optimized with a scalar loss averaged over samples, variables, and horizons. This averaging is convenient, but the optimizer sees only the aggregated gradient, which does not reveal whether the variable-wise contributions align or oppose one another. To quantify how often this disagreement arises, we measure the variable-wise gradients directly and find that 30.6% of their pairwise cosine similarities are negative on average across seven datasets. However, conflict and harm are not the same thing. Under shared training 35 of the 64 variables do worse than a full-input single-target oracle, and the harmed fraction is not reliably predicted by how often gradients conflict. We propose Per-Variable Surgery (PV-Surgery), an optimizer-side training strategy for backbones with cache-compatible layers. One backward pass builds variable-wise gradient proxies from output-side signals and keeps the pointwise forecasting loss. Reliability-aware selection targets layers whose proxy sums closely approximate their shared-gradient slices. Conditional pooling forms anchor and conflict pools without dropping variables. Common-direction surgery aligns variable or pooled gradients with their normalized mean and restores input norms to avoid reweighting. In experiments across five backbones, seven datasets, and four horizons, PV-Surgery lowers MSE by 3.61% and MAE by 2.93% on average. For multivariate forecasting, this indicates that the variable-wise structure hidden by mean-loss training is a usable optimization signal.

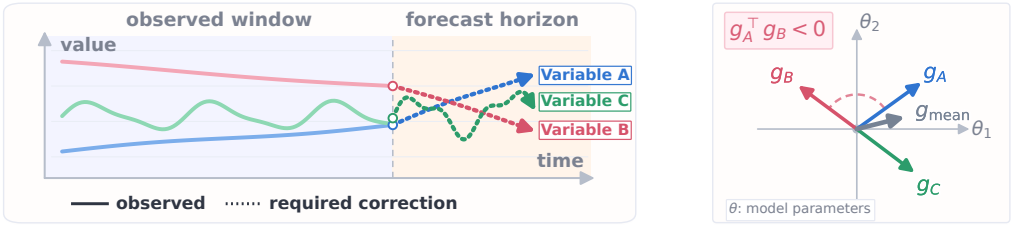
1 INTRODUCTION

Multivariate time-series forecasting predicts the future values of multiple variables from a shared history of all variables (Lai et al., 2018; Lim & Zohren, 2020). Forecasting architectures differ widely, but most are trained with a similar mean-loss objective. Prediction errors are averaged over samples, variables, and horizons to form a scalar loss, which is optimized through backpropagation (Nie et al., 2023; Liu et al., 2024; Wang et al., 2024). This objective is convenient and model-agnostic, but it also collapses heterogeneous variable-wise learning signals into one shared update.

The three averages behind this scalarization do not play the same role. Averaging over samples estimates an expectation over the data distribution, whereas averaging over variables and horizons assumes that heterogeneous targets and future steps can be merged into one optimization signal without losing important structure. In practice, however, variables differ in scale, noise, predictability, dynamics, and cross-variable dependency (Lai et al., 2018; Liu et al., 2024; Nochumsohn et al., 2025). Consequently, an update that helps one variable can be neutral or harmful for another, but the mean-loss gradient does not reveal which case holds (Nochumsohn et al., 2025; Liu et al., 2021).

More concretely, let $L_d(\theta)$ denote the loss induced by variable d, averaged over samples and horizons. Standard training minimizes $L_{\text{mean}}(\theta) = D^{-1}\sum_{d=1}^{D} L_d(\theta)$ and updates parameters with $g_{\text{mean}} = \nabla_\theta L_{\text{mean}}(\theta) = D^{-1}\sum_{d=1}^{D} g_d$, where $g_d = \nabla_\theta L_d(\theta)$. This aggregation is many-to-one, so different sets of variable-wise gradients can produce the same update. For $D = 2$, $g_1 = u + v$ and $g_2 = u - v$ produce $g_{\text{mean}} = u$, hiding the disagreement component v. When $\|v\|_2 > \|u\|_2$, the inner product $\langle g_1, g_2\rangle = \|u\|_2^2 - \|v\|_2^2$ turns negative, so an update along $-g_1$ reduces L_1 but increases L_2 to first order. Moreover, under any shared update $\theta \leftarrow \theta - \eta g_{\text{shared}}$ with step size $\eta > 0$,

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.



(a) Variable-specific future corrections.

(b) Mean-loss gradient averaging.

Figure 1: Conceptual illustration of variable-wise gradient conflict. (a) Different variables can require corrections in different directions over the same forecast horizon. (b) These corrections translate into variable-wise gradients that disagree in parameter space, while the mean-loss update shown here coincides with none of them.

the first-order change of L_d is $-\eta\langle g_d, g_{\text{shared}}\rangle$, whose sign cannot be checked from the aggregate alone. The problem is therefore not that mean loss is invalid, but that its single gradient hides the variable-wise directions needed to distinguish disagreement from actual harm.

Figure 1(a) shows three variables requiring corrections in different directions over the same horizon, and Figure 1(b) shows their parameter-space gradients, two of which oppose each other while their mean coincides with none of them. We call this *variable-wise gradient conflict*, following the view that a negative inner product encodes incompatible descent directions (Yu et al., 2020; Nochumsohn et al., 2025). A natural reaction is to remove every conflict, as in gradient surgery (Yu et al., 2020). In multivariate forecasting, our diagnosis suggests otherwise. Conflicts are common but not uniformly harmful, so intervention should be selective, variable-aware, and anchored to the mean-loss update.

In this paper, we propose PV-Surgery, a reliability-aware optimizer-side framework. At each training step, one backward pass builds variable-wise gradient proxies from output-side signals. PV-Surgery then selects surgery layers based on the normalized error between each proxy sum and its matching reference-gradient slice, conditionally forms anchor and conflict pools without dropping variables, and aligns variable or pooled gradients with the normalized mean of their input directions before restoring their norms to avoid reweighting. PV-Surgery requires no changes to cache-compatible forecasting backbones or per-variable losses. We make the following three contributions:

- We analyze variable-wise conflict under standard mean-loss training using exact gradients. Although 30.6% of pairwise cosines are negative on average, conflict frequency is weakly related to harm. Shared training underperforms a full-input single-target oracle for 35 of 64 variables, while mean alignment does not reliably identify helpful partners.
- We propose PV-Surgery, which reconstructs variable-wise gradient proxies in one backward pass, targets reconstruction-consistent layers, pools without dropping variables, and corrects directions while restoring surgery-input norms over the selected subspace.
- We conduct experiments across five backbones, seven datasets, and four horizons, showing that PV-Surgery lowers MSE by 3.61% and MAE by 2.93% on average and that variable-wise structure hidden by mean-loss training is a usable optimization signal in these settings.

2 Related Work

2.1 Multivariate Time-Series Forecasting

Multivariate forecasting has largely advanced through backbone design, spanning efficient attention (Zhou et al., 2021), patch tokens (Nie et al., 2023), variate tokens (Liu et al., 2024), exogenous cross-attention (Wang et al., 2024), series decomposition (Wu et al., 2021), multi-scale convolution (Wang et al., 2023), sample convolution (Liu et al., 2022), and linear baselines (Zeng et al., 2023). These approaches differ in architecture but retain the same mean-loss objective, whereas we intervene in the update it produces. MTLinear is the closest prior work, connecting multivariate forecasting to multi-task learning through variate gradient angle, correlation-based grouping, and gradient scal-

ing for linear models (Nochumsohn et al., 2025). MTLinear fixes variable groups in advance and rescales gradient magnitudes, whereas we retain all variables, pool their gradients by directional compatibility at each step, and correct directions while restoring input norms across backbones.

2.2 Forecasting Objectives and Variable-Wise Learning

Forecasting objectives have expanded beyond point-wise MSE and MAE to include shape and time distortion losses (Guen & Thome, 2019), frequency-domain losses (Wang et al., 2025), transformation-invariant criteria (Lee et al., 2022), patch-wise structural losses (Kudrat et al., 2025), and selective timestep masking (Fu et al., 2025). These objectives change which errors are emphasized during training, but they still aggregate per-variable errors into one scalar. Reweighting the loss therefore does not reveal whether the corresponding gradients agree, cancel, or interfere.

2.3 Gradient Conflict and Multi-Task Optimization

Gradient conflict is studied in multi-task learning, where methods balance losses (Chen et al., 2018), seek Pareto-stationary descent directions (Sener & Koltun, 2018), project conflicting task gradients (Yu et al., 2020), maximize the worst-case improvement (Liu et al., 2021), or aggregate task gradients by bargaining (Navon et al., 2022). Multivariate forecasting exposes its objectives differently because variables are components scalarized inside one loss rather than separate tasks, a setting that MTLinear connects to multi-task learning (Nochumsohn et al., 2025). We therefore construct variable-wise gradient proxies for these operators rather than applying them to the aggregate.

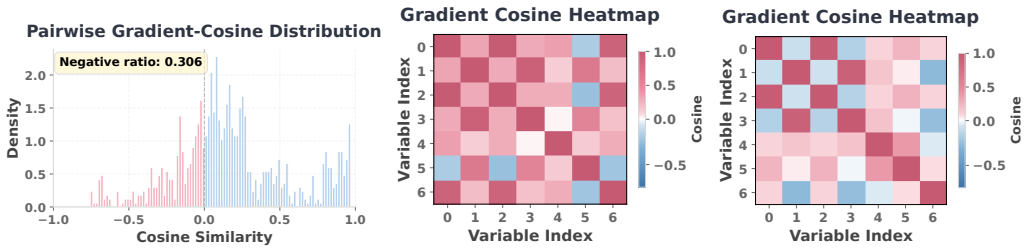
3 Diagnosing Variable-Wise Gradient Conflict

We analyze the frequency of variable-wise gradient conflict under standard multivariate forecasting training and its relation to the fraction of variables harmed under shared training. We also ask whether mean pairwise alignment reliably identifies helpful training partners. Together, these analyses assess whether pairwise cosine statistics provide a sound basis for deciding when to intervene.

Diagnostic setup. We train iTransformer with the standard mean-loss objective on all seven benchmarks, at prediction length 96 for the four ETT datasets, Weather, and Exchange, and at 24 for ILI. At analysis checkpoints we decompose the scalar objective into variable-induced losses L_d, set $g_d^{(t)} = \nabla_\theta L_d(\theta_t)$, and form the pairwise cosine matrix over nonzero gradients,

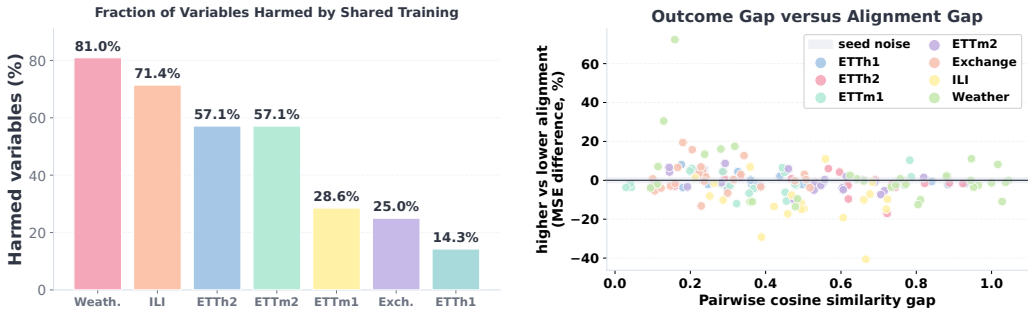
$$C_{ij}^{(t)} = \frac{\langle g_i^{(t)}, g_j^{(t)} \rangle}{\|g_i^{(t)}\|_2 \|g_j^{(t)}\|_2}. \tag{1}$$

A pair is conflicting at checkpoint t when $C_{ij}^{(t)} < 0$. Checkpoint summaries discard the first half of recorded checkpoints as warmup because early updates are unstable, but this filter does not apply to separately trained oracle or partner-group results. The diagnostics expose variable-wise structure hidden by the baseline objective and are not used to tune PV-Surgery.



(a) Post-warmup cosine distribution. (b) Early checkpoint. (c) Late checkpoint.

Figure 2: A representative ETTh1 run from the variable-wise gradient-cosine diagnostic. (a) Distribution of the post-warmup pairwise cosines, of which 30.6% fall below zero for this run. (b) An early checkpoint, which lies inside the discarded warmup half and is shown only to expose the change over training. (c) A late checkpoint. Both heatmaps show that conflict is structured across variable pairs rather than being uniform, so one summary number would hide which pairs disagree.



(a) Harmed variables by dataset.

(b) Outcome gap versus alignment gap.

Figure 3: (a) Fraction of variables for which shared training is worse than the target-specific oracle. (b) Across 162 target-and-size cells, the relative MSE difference between the two alignment groups versus the pairwise-alignment gap. Negative values favor higher alignment. The shaded $\pm 1.1\%$ band summarizes repeated-fit variation for fixed target-partner groups (Appendix C).

3.1 Variable-wise conflicts are frequent

Across the seven datasets, 30.6% of the pairwise cosine similarities are negative on average, with per-dataset results in Appendix B and the ETTh1 case shown in Figure 2. On the six datasets other than ILI, most post-warmup checkpoints contain at least one conflicting pair. The disagreement is also intermittent, since no pair is negative at every post-warmup checkpoint. Therefore, the standard scalar objective hides a substantial amount of variable-wise disagreement. However, this does not yet justify gradient surgery, since a negative cosine says only that two descent directions disagree at one checkpoint, not which variable is harmed or whether sharing is worse than separate training.

3.2 Conflict and harm are not the same object

To separate disagreement from harm, we compare the shared model with full-input single-target oracles, each optimized and evaluated on one target. We define the oracle gap as $\Delta_d^{\text{oracle}} = \text{MSE}_{\text{shared},d} - \text{MSE}_{\text{oracle},d}$. A positive Δ_d^{oracle} means variable d is better predicted by its target-specific oracle. Across the seven datasets 35 of the 64 variables have a positive oracle gap, and Figure 3(a) reports the fraction for every dataset. At the dataset level, the negative-cosine ratio and the harmed fraction correlate at -0.29 ($p = 0.53$), so a larger conflict rate does not reliably coincide with a larger harmed fraction. ILI has the lowest conflict rate, whereas Weather has the highest harmed fraction (Table 4). A method that removes every negative cosine similarity would therefore risk destroying useful sharing for variables that improve under the shared model.

3.3 Alignment alone does not reliably identify helpful partners

For each reported target we rank the remaining variables by their post-warmup mean pairwise gradient cosine in the baseline model. For $k \in \{2, 3, 4\}$, we train with its $k - 1$ highest-ranked or lowest-ranked partners, which form disjoint sets. Both conditions use matched replicate seeds, keep the full multivariate input, restrict the loss to the target and its partners, and are scored by held-out target MSE. Appendix C gives the full protocol. Higher alignment wins $100/162$ cells, and the all-cell median signed contrast is -1.4%, but the benchmark-averaged contrast is not distinguishable from zero. Figure 3(b) shows that a larger alignment gap does not reliably identify the better partner set, consistent with Appendix Figure 13, where ILI favors higher alignment at every group size but five of the other six datasets change sign across sizes. Group composition still matters, since the median absolute within-cell contrast is 3.5%, over three times the 1.1% variation observed when the same target and partner set are retrained with different random seeds.

Together, the evidence shows that negative pairwise cosine similarities are common, but their frequency does not reveal which variables shared training harms. Partner composition can materially change a target's error, but mean gradient alignment does not reliably predict whether the change helps or hurts. The comparisons are associational rather than causal, and they motivate conditional, reliability-aware intervention rather than removing conflicts uniformly across variable subsets.

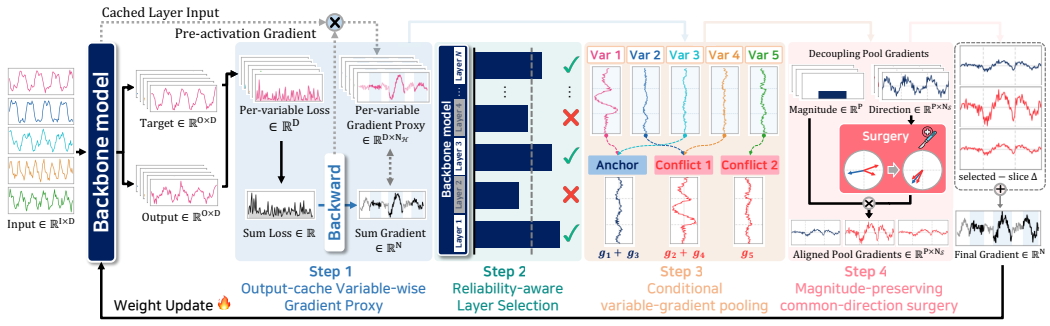


Figure 4: Overview of PV-Surgery. **Step 1: Output-cache variable-wise gradient proxy** reconstructs variable-wise proxy gradients from a single sum-loss backward pass and cached layer signals. **Step 2: Reliability-aware layer selection** selects slices by normalized proxy-sum error with output fallback. **Step 3: Conditional variable-gradient pooling** forms anchor and conflict pool gradients without dropping variables. **Step 4: Magnitude-preserving common-direction surgery** aligns variable or pooled gradients while restoring their pre-surgery norms to avoid reweighting.

4 Proposed Method

PV-Surgery is an optimizer-side gradient transformation that leaves the forecasting architecture, prediction target, and per-variable loss unchanged. Figure 4 gives an overview of PV-Surgery, which consists of four stages that form a dependency chain. Computing exact per-variable gradients would require D costly backward passes, so PV-Surgery instead builds variable-wise gradient proxies from cached layer signals in one pass. Variable mixing can shift proxy sums from reference-gradient slices, so PV-Surgery selects layers by relative error and falls back to the output layer. Section 3.3 shows that mean alignment does not reliably identify helpful partners, which is why variables are pooled conditionally at each step without dropping any of them. Finally, surgery aligns each variable or pooled gradient with the normalized mean direction, then restores its pre-surgery norm so the correction does not reweight the surgery inputs. Appendix D gives the training pseudocode.

4.1 Problem Setup

For a batch of B windows, D variables, and horizon length H, let $\hat{Y}, Y \in \mathbb{R}^{B\times H\times D}$ be the predicted and true future values and φ be the pointwise forecasting loss. The loss induced by variable d is

$$L_d(\theta) = \frac{1}{BH}\sum_{b=1}^{B}\sum_{h=1}^{H}\varphi\left(\hat{Y}_{b,h,d}, Y_{b,h,d}\right). \tag{2}$$

Let $L_\Sigma = \sum_{d=1}^{D} L_d$ and $L_{\text{mean}} = L_\Sigma/D$. With $g_d = \nabla_\theta L_d$ and $g_{\text{mean}} = \nabla_\theta L_{\text{mean}}$, PV-Surgery's exact single-backward reference gradient is

$$g_0 = \nabla_\theta L_\Sigma = \sum_{d=1}^{D} g_d = D g_{\text{mean}}. \tag{3}$$

PV-Surgery backpropagates L_Σ once, avoiding D backward passes and a dense $D \times |\theta|$ gradient tensor. At the output boundary, variable d's cache row equals its g_d slice without $1/D$ scaling because only that output slice enters L_d, and the rows sum to the matching g_0 slice without rescaling.

4.2 Output-cache variable-wise gradient proxy

One backward pass materializes only g_0, so we must estimate variable-wise gradients from same-pass layer inputs and output-side backward signals. We hook cache-compatible linear layers $\mathcal{H}$ to collect these quantities and convert them to a common variable-axis form by moving an explicit axis or reshaping batch-folded layouts. For a layer ℓ with $W_\ell \in \mathbb{R}^{F_{\text{out}}\times F_{\text{in}}}$, the caches keep a variable axis, $Z_\ell \in \mathbb{R}^{B\times D\times M_\ell\times F_{\text{in}}}$ and $E_\ell \in \mathbb{R}^{B\times D\times M_\ell\times F_{\text{out}}}$, and flattening the batch and position axes gives $Z_{\ell,d} \in \mathbb{R}^{B\,M_\ell\times F_{\text{in}}}$ and $E_{\ell,d} \in \mathbb{R}^{B\,M_\ell\times F_{\text{out}}}$. The weight and bias gradients are then

$$G^W_{\ell,d} = E^\top_{\ell,d} Z_{\ell,d}, \qquad G^b_{\ell,d} = E^\top_{\ell,d}\mathbf{1}_{B\,M_\ell}. \tag{4}$$

When a bias is present, it is concatenated with the flattened weight gradient.

$$G_\ell = \mathrm{stack}_{d=1}^{D}\big(\mathrm{concat}\big(\mathrm{vec}(G_{\ell,d}^{W}), G_{\ell,d}^{b}\big)\big) \in \mathbb{R}^{D\times N_\ell}. \tag{5}$$

Equation 4 is the per-variable chain-rule gradient at the output boundary and a proxy after variable mixing, motivating the reliability test. If a hooked layer lacks a variable-axis form for that batch, $G_\ell = 0_{D\times N_\ell}$. Here M_ℓ counts positions sharing W_ℓ within a variable and N_ℓ cached parameters.

4.3 Reliability-aware layer selection

Reconstruction consistency varies by layer and backbone and can shift during training, so the method selects its layer set dynamically at each step rather than fixing it in advance. For every hooked layer $\ell \in \mathcal{H}$, we compare the summed proxy with the matching slice of the single-backward reference.

$$g_{\mathrm{proxy},\ell} = \sum_{d=1}^{D} G_\ell[d,\cdot], \qquad e_\ell = \frac{\|g_{\mathrm{proxy},\ell} - g_{0,\ell}\|_2}{\|g_{0,\ell}\|_2 + \epsilon}. \tag{6}$$

Reconstruction reliability $r_\ell = (1+e_\ell)^{-1}$ scores proxy-sum agreement, while normalized slice norm $q_\ell = \|g_{0,\ell}\|_2/(\|g_0\|_2+\epsilon)$ measures reference-slice magnitude. Let $\mathcal{H}_{\mathrm{elig}}$ be the unprotected hooked layers. The median-reliability gate retains its non-output layers at or above median reliability and includes the output layer. For the resulting $\mathcal{H}_{\mathrm{gate}}$, we set $\mathcal{A} = \{\ell \in \mathcal{H}_{\mathrm{gate}} \mid q_\ell > 0\}$ and compute

$$\pi_\ell = \frac{q_\ell}{\sum_{j\in\mathcal{A}} q_j}, \quad \mathcal{E}_{\mathrm{layer}} = -\sum_{\ell\in\mathcal{A}} \pi_\ell \log \pi_\ell, \quad K_{\mathrm{eff}} = \left\lceil e^{\mathcal{E}_{\mathrm{layer}}} \right\rceil. \tag{7}$$

It takes the K_{eff} largest-q_ℓ layers and the output layer that pass the reconstruction-error check, falling back to the output layer if none do (Appendix D). Here $e^{\mathcal{E}_{\mathrm{layer}}}$ is π's exponentiated Shannon entropy, or order-one Hill number (Hill, 1973), and K_{eff} is its ceiling. The selected set $\mathcal{S}$ then gives

$$G_{\mathcal{S}} = \mathrm{concat}_{\ell\in\mathcal{S}}\, G_\ell \in \mathbb{R}^{D\times N_{\mathcal{S}}}, \qquad N_{\mathcal{S}} = \sum_{\ell\in\mathcal{S}} N_\ell \le N_{\mathcal{H}}. \tag{8}$$

Here $N_{\mathcal{H}} = \sum_{\ell\in\mathcal{H}} N_\ell$ counts all cached parameters. Fixing the support before pooling or direction change keeps surgery out of layers where the cache decomposition is not reconstruction-consistent, and leaves later stages operating on $N_{\mathcal{S}}$ rather than $N_{\mathcal{H}}$ parameters.

4.4 Conditional variable-gradient pooling

Treating every variable gradient as a surgery objective would make direction surgery reconcile all pairs, although Section 3 shows that a negative cosine alone does not reliably imply harmful sharing. We keep every variable but reduce the objective count. With $G_{\mathrm{pre}} = G_{\mathcal{S}}$, each row contains one variable's proxy gradient over the selected parameter slices, and $g_{0,\mathcal{S}}$ is the exact sum-loss gradient over those same slices:

$$g_{\mathrm{ref}} = g_{0,\mathcal{S}}, \qquad a_d = \cos(G_{\mathrm{pre}}[d,\cdot], g_{\mathrm{ref}}). \tag{9}$$

Pooling requires two finite, nonzero rows with a negative cosine, not just a row opposed to g_{ref}. Rows with $a_d \ge 0$ are aligned to the reference and the rest conflict with it. An anchor combines at least two aligned rows whose sum has nonnegative cosine with each. Each conflicting row greedily joins the pool whose current sum is most aligned with it, provided their cosine is positive. The partition $M_{\mathrm{pool}} \in \{0,1\}^{P\times D}$ maps D rows to P objectives while preserving their aggregate gradient:

$$G_{\mathrm{pool}} = M_{\mathrm{pool}} G_{\mathrm{pre}} \in \mathbb{R}^{P\times N_{\mathcal{S}}}, \qquad \sum_{k=1}^{P} G_{\mathrm{pool}}[k,\cdot] = \sum_{d=1}^{D} G_{\mathrm{pre}}[d,\cdot]. \tag{10}$$

Without pairwise conflict, the pooled branch skips direction surgery and remains $G_p^{\mathrm{post}} = G_{\mathrm{pre}}$. The subsequent safe candidate-selection step may retain it over the corrected unpooled candidate.

4.5 Magnitude-preserving common-direction surgery

Pooling reduces the objective count, but pooled gradients can still disagree, so their directions must be adjusted and the unpooled and pooled branches remain candidates. For unpooled u and pooled p,

the inputs are $G_{\text{in}}^{(u)} = G_{\text{pre}}$ and, when pooling is active, $G_{\text{in}}^{(p)} = G_{\text{pool}}$. For $c \in \{u, p\}$, let P_c be its objective count and decompose each nonzero objective into its magnitude and unit direction:

$$m_i^{(c)} = \|G_{\text{in}}^{(c)}[i, \cdot]\|_2, \qquad \hat{g}_i^{(c)} = \frac{G_{\text{in}}^{(c)}[i, \cdot]}{m_i^{(c)}}. \tag{11}$$

Objectives with $m_i^{(c)} = 0$ take $\hat{g}_i^{(c)} = 0$. Let $\bar{g}^{(c)} = P_c^{-1} \sum_i \hat{g}_i^{(c)}$. The operator aligns each objective with $\bar{g}^{(c)}$ while preserving its input magnitude, and makes no change when $\|\bar{g}^{(c)}\|_2 = 0$.

$$\widetilde{G}_{\text{in}}^{(c)}[i, \cdot] = \begin{cases} m_i^{(c)} \, \bar{g}^{(c)} / \|\bar{g}^{(c)}\|_2, & \|\bar{g}^{(c)}\|_2 > 0, \\ G_{\text{in}}^{(c)}[i, \cdot], & \|\bar{g}^{(c)}\|_2 = 0, \end{cases} \quad \text{and} \quad G_u^{\text{post}} = \widetilde{G}_{\text{in}}^{(u)}. \tag{12}$$

Pool restoration and safe candidate selection. For active pooling, pool k distributes the change made by surgery equally across its member set $\mathcal{M}_k$.

$$G_p^{\text{post}}[d, \cdot] = G_{\text{pre}}[d, \cdot] + \frac{\widetilde{G}_{\text{in}}^{(p)}[k, \cdot] - G_{\text{pool}}[k, \cdot]}{|\mathcal{M}_k|}, \qquad d \in \mathcal{M}_k. \tag{13}$$

This restores D variable rows while preserving each post-surgery pool sum, but not individual row norms. The descent-lexicographic selector compares G_u^{post} with G_p^{post}. For candidate c, let $g_c = \sum_d G_c^{\text{post}}[d, \cdot]$. For each finite, nonzero variable-wise gradient, the selector computes the following scores:

$$\gamma_{c,d} = \cos(G_{\text{pre}}[d, \cdot], g_c), \qquad v_{c,d} = \mathbf{1}[\gamma_{c,d} < 0]. \tag{14}$$

It first minimizes $\sum_d v_{c,d}$, then compares the sorted cosine vectors from worst to best. Exact or numerically ambiguous ties retain the unpooled candidate. This ranks candidates and does not guarantee a descent-compatible update for every variable when both contain violations.

Final gradient assembly. For selected G_{post}, $\Delta_{\mathcal{S}}$ is the change in the aggregate selected-subspace gradient:

$$\Delta_{\mathcal{S}} = \sum_{d=1}^{D} G_{\text{post}}[d, \cdot] - \sum_{d=1}^{D} G_{\text{pre}}[d, \cdot]. \tag{15}$$

For each hooked layer ℓ, let Ω_ℓ be its flattened parameter indices, and let $\Omega_{\mathcal{S}} = \bigcup_{\ell \in \mathcal{S}} \Omega_\ell$. The full-sum assembly inserts every hooked-slice proxy sum and then adds the selected-subspace correction.

$$g_{\text{final}} \leftarrow g_0, \quad g_{\text{final}}[\Omega_\ell] \leftarrow \sum_{d=1}^{D} G_\ell[d, \cdot] \;\; (\ell \in \mathcal{H}), \quad g_{\text{final}}[\Omega_{\mathcal{S}}] \leftarrow g_{\text{final}}[\Omega_{\mathcal{S}}] + \Delta_{\mathcal{S}}. \tag{16}$$

Thus, non-hooked parameters retain g_0. Unselected hooked slices use their variable-wise proxy sums, and selected slices add the corresponding $\Delta_{\mathcal{S}}$ entries.

5 Experiments

5.1 Experimental setup

Datasets. We evaluate PV-Surgery on ETTh1, ETTh2, ETTm1, ETTm2, Weather, and Exchange using an input length of 96 and prediction lengths $\{96, 192, 336, 720\}$. For ILI, the input length is 36 and the prediction lengths are $\{24, 36, 48, 60\}$. All methods share chronological splits. Appendices E and A give settings and dataset statistics.

Baselines. We compare standard MSE training and PV-Surgery on DLinear (Zeng et al., 2023), iTransformer (Liu et al., 2024), MICN (Wang et al., 2023), SCINet (Liu et al., 2022), and TimeXer (Wang et al., 2024). We additionally compare four forecasting-specific methods, TILDE-Q (Lee et al., 2022), FreDF (Wang et al., 2025), PSLoss (Kudrat et al., 2025), and Selective Learning (Fu et al., 2025). All use the common protocol of Appendix E, with iTransformer for this comparison.

Metrics. We report test mean squared error (MSE) and mean absolute error (MAE), both averaged over test windows, horizons, and variables. Exact definitions are provided in Appendix E.

5.2 MAIN RESULTS

Tables 1 and 6 report all 140 standard-MSE comparisons, with Weather and ILI in Appendix F due to the page limit. Separately, Table 2 compares PV-Surgery with four forecasting-specific methods using iTransformer across five datasets. Appendix G covers all backbones and benchmarks.

Table 1: Backbone comparison between standard MSE training and PV-Surgery on five of the seven benchmarks. Within each backbone and metric, the better value between MSE and +PV is bolded using unrounded scores. Weather and ILI use the same protocol and are reported in Table 6.

Models		DLinear				iTransformer				MICN				SCINet				TimeXer			
Training setup		MSE		+PV		MSE		+PV		MSE		+PV		MSE		+PV		MSE		+PV	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.462	0.444	**0.461**	**0.440**	0.454	0.447	**0.451**	**0.442**	0.552	0.530	**0.492**	**0.497**	0.490	0.464	**0.473**	**0.449**	**0.460**	0.454	0.461	**0.450**
	192	0.514	0.476	**0.514**	**0.471**	0.517	0.485	**0.508**	**0.476**	0.594	0.553	**0.561**	**0.543**	0.545	0.495	**0.530**	**0.483**	**0.511**	0.485	0.518	**0.485**
	336	0.559	0.506	**0.556**	**0.501**	0.561	0.513	**0.550**	**0.504**	**0.688**	**0.623**	0.700	0.630	0.590	0.521	**0.574**	**0.510**	0.564	**0.511**	**0.564**	0.515
	720	0.651	0.573	**0.641**	**0.561**	0.671	0.583	**0.661**	**0.575**	**0.753**	**0.671**	0.798	0.700	0.700	0.588	**0.687**	**0.581**	0.693	0.590	**0.691**	**0.581**
	Avg	0.546	0.500	**0.543**	**0.493**	0.551	0.507	**0.543**	**0.499**	0.647	0.594	**0.638**	**0.593**	0.581	0.517	**0.566**	**0.506**	**0.557**	0.510	0.558	**0.508**
ETTh2	96	0.239	0.329	**0.229**	**0.313**	0.239	0.323	**0.229**	**0.314**	0.238	0.333	**0.230**	**0.320**	0.254	0.334	**0.237**	**0.317**	0.233	0.319	**0.231**	**0.316**
	192	0.329	0.393	**0.290**	**0.357**	0.303	0.366	**0.287**	**0.355**	0.315	0.388	**0.291**	**0.369**	0.311	0.370	**0.297**	**0.358**	**0.294**	0.360	0.295	**0.358**
	336	0.437	0.462	**0.341**	**0.396**	0.356	0.401	**0.334**	**0.388**	0.452	0.472	**0.352**	**0.412**	0.356	0.402	**0.346**	**0.391**	**0.341**	0.394	0.342	**0.390**
	720	0.650	0.582	**0.470**	**0.484**	0.457	0.460	**0.446**	**0.452**	0.672	0.583	**0.513**	**0.520**	0.470	0.472	**0.447**	**0.454**	0.449	0.457	**0.446**	**0.451**
	Avg	0.414	0.442	**0.332**	**0.388**	0.339	0.388	**0.324**	**0.377**	0.419	0.444	**0.346**	**0.405**	0.347	0.394	**0.332**	**0.380**	0.329	0.382	**0.328**	**0.379**
ETTm1	96	0.388	0.395	**0.383**	**0.389**	0.407	0.415	**0.402**	**0.404**	0.445	0.472	**0.430**	**0.460**	0.448	0.428	**0.421**	**0.414**	0.432	0.421	**0.396**	**0.403**
	192	0.447	0.426	**0.445**	**0.421**	0.480	0.457	**0.461**	**0.439**	0.495	0.511	**0.470**	**0.487**	**0.488**	0.453	0.490	**0.446**	0.477	0.453	**0.455**	**0.429**
	336	0.507	0.459	**0.505**	**0.454**	0.520	0.477	**0.517**	**0.473**	0.567	0.570	**0.508**	**0.513**	**0.548**	0.483	0.551	**0.475**	0.521	0.479	**0.505**	**0.462**
	720	0.575	0.502	**0.573**	**0.499**	0.618	0.531	**0.575**	**0.510**	0.627	0.596	**0.576**	**0.562**	**0.614**	0.521	0.628	**0.516**	0.580	0.515	**0.564**	**0.503**
	Avg	0.479	0.446	**0.476**	**0.441**	0.506	0.470	**0.488**	**0.457**	0.533	0.537	**0.496**	**0.505**	0.524	0.471	**0.523**	**0.463**	0.502	0.467	**0.480**	**0.449**
ETTm2	96	0.160	0.268	**0.155**	**0.253**	0.157	0.258	**0.150**	**0.247**	0.166	0.276	**0.146**	**0.248**	0.155	0.253	**0.150**	**0.246**	0.151	0.249	**0.149**	**0.247**
	192	0.209	0.308	**0.197**	**0.287**	0.205	0.295	**0.195**	**0.283**	0.202	0.303	**0.190**	**0.285**	0.202	0.290	**0.193**	**0.278**	0.200	0.287	**0.193**	**0.282**
	336	0.262	0.350	**0.239**	**0.319**	0.252	0.328	**0.240**	**0.317**	0.266	0.355	**0.231**	**0.316**	0.248	0.322	**0.236**	**0.309**	0.239	0.316	**0.235**	**0.312**
	720	0.348	0.408	**0.306**	**0.365**	0.322	0.373	**0.310**	**0.365**	0.389	0.434	**0.331**	**0.389**	0.321	0.368	**0.308**	**0.357**	0.312	0.367	**0.307**	**0.360**
	Avg	0.245	0.334	**0.225**	**0.306**	0.234	0.313	**0.224**	**0.303**	0.256	0.342	**0.224**	**0.310**	0.231	0.308	**0.222**	**0.297**	0.226	0.305	**0.221**	**0.300**
Exchange	96	0.105	0.234	**0.100**	**0.226**	0.108	0.236	**0.105**	**0.231**	0.123	0.262	**0.109**	**0.242**	0.120	0.249	**0.116**	**0.244**	**0.113**	0.239	0.113	**0.239**
	192	0.224	0.349	**0.191**	**0.323**	0.222	0.344	**0.211**	**0.335**	0.219	0.352	**0.212**	**0.348**	0.223	0.347	**0.222**	**0.346**	0.216	0.337	**0.213**	**0.335**
	336	0.425	0.498	**0.321**	**0.425**	0.397	0.463	**0.390**	**0.460**	0.498	0.525	**0.378**	**0.470**	**0.405**	**0.470**	0.409	0.473	0.396	0.459	**0.382**	**0.452**
	720	**0.652**	**0.640**	0.847	0.724	1.099	0.800	**1.075**	**0.792**	2.931	1.400	**1.106**	**0.793**	**1.101**	**0.804**	1.103	0.806	1.105	0.798	**1.010**	**0.763**
	Avg	**0.352**	0.430	0.365	**0.424**	0.457	0.461	**0.446**	**0.454**	0.943	0.635	**0.451**	**0.463**	**0.462**	0.468	0.463	**0.467**	0.457	0.458	**0.429**	**0.447**

PV-Surgery obtains lower MSE in 111 of the 140 settings and lower MAE in 119, both counted on unrounded scores, and averaging the relative change within each setting gives 3.61% in MSE and 2.93% in MAE. The improvement is largest on MICN, at 8.50% in MSE and 5.67% in MAE, and on DLinear, at 5.38% and 5.01%. PV-Surgery therefore improves the majority of settings for every backbone and dataset, although the direction and size of the average change depend on both.

Table 2: Comparison of PV-Surgery with four forecasting-specific methods on iTransformer for five of the seven benchmarks. The MSE row denotes standard MSE training. Best results are bolded and second-best results are underlined using unrounded scores.

Dataset		ETTh1				ETTh2				ETTm1				ETTm2				Exchange			
Forecast length		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720
MSE	MSE	0.454	0.517	0.561	0.671	0.239	0.303	0.356	0.457	0.407	0.480	0.520	0.618	0.157	0.205	0.252	0.322	0.108	0.222	0.397	1.099
	MAE	0.447	0.485	0.513	0.583	0.323	0.366	0.401	0.460	0.415	0.457	0.477	0.531	0.258	0.295	0.328	0.373	0.236	0.344	0.463	0.800
TILDE-Q	MSE	0.469	0.528	0.564	0.669	0.233	0.287	0.354	0.473	**0.402**	0.465	0.537	0.622	0.156	0.200	0.247	0.311	0.111	0.219	0.428	1.109
	MAE	0.455	0.489	0.510	0.577	0.315	0.353	0.400	0.466	0.404	0.440	0.483	0.534	0.251	0.285	0.320	0.362	0.239	0.341	0.481	0.806
FreDF	MSE	**0.451**	0.512	0.560	0.662	0.235	0.295	0.350	0.450	0.410	0.475	0.553	0.603	0.154	0.197	0.245	**0.309**	0.112	0.223	0.462	1.132
	MAE	0.445	0.478	0.508	0.576	0.317	0.360	0.397	0.455	0.408	0.452	0.492	0.524	0.250	0.284	0.319	0.362	0.241	0.344	0.502	0.819
PSLoss	MSE	0.458	0.513	0.558	0.665	0.234	0.291	0.360	0.473	0.429	0.499	0.538	0.626	0.153	0.196	**0.239**	0.310	0.113	0.213	0.413	1.146
	MAE	0.448	0.480	0.509	0.577	0.316	0.355	0.406	0.469	0.421	0.459	0.482	0.535	0.247	**0.281**	**0.314**	**0.361**	0.241	0.338	0.476	0.825
SL	MSE	0.465	0.527	0.560	0.665	**0.229**	**0.284**	0.348	**0.438**	0.404	**0.458**	0.521	0.590	0.153	0.199	0.243	0.312	0.108	**0.211**	0.393	1.228
	MAE	0.449	0.485	**0.501**	**0.568**	**0.312**	**0.349**	0.393	**0.449**	**0.400**	**0.432**	**0.465**	**0.505**	0.250	0.285	0.317	0.362	0.232	**0.332**	0.460	0.856
PV(Ours)	MSE	0.451	**0.508**	**0.550**	**0.661**	0.229	0.287	**0.334**	0.446	0.402	0.461	**0.517**	**0.575**	**0.150**	**0.195**	0.240	0.310	**0.105**	0.211	**0.390**	**1.075**
	MAE	**0.442**	**0.476**	0.504	0.575	0.314	0.355	**0.388**	0.452	0.404	0.439	0.473	0.510	**0.247**	0.283	0.317	0.365	**0.231**	0.335	**0.460**	**0.792**

Across the five datasets in Table 2, PV-Surgery ranks first or second in $19/20$ MSE cells and $17/20$ MAE cells. No method is best throughout. Some competing methods perform better in individual ETT settings, while PV-Surgery is particularly competitive at longer horizons and on Exchange. This pattern supports the diagnostic premise that intervention should preserve useful sharing rather than treat every disagreement as harmful. PV-Surgery acts on the update rather than the loss and can therefore complement these methods instead of replacing them.

5.3 Ablation study

The full configuration gives the lowest MSE and MAE. Removing the output-cache proxy or conflict pooling causes the largest degradations. Because the output-cache ablation also disables cache-dependent layer selection, pooling, and candidate comparison, its large performance drop reflects several coupled components being removed together. Removing conflict pooling degrades both metrics much more than removing anchor pooling, consistent with its role in grouping variables into fewer, directionally coherent objectives when conflicts are frequent. Without layer selection, surgery runs on every hooked layer, lowering accuracy and raising mean wall-clock training time from 332.45 to 642.98 seconds. Appendix H reports dataset-level results.

Table 3: Core component ablation on MICN–Exchange, averaged over four horizons. Δ is the relative change against the full recipe. Appendix H reports the same table for every dataset.

Category	Module	MSE		MAE		Time (s)
		value	Δ%	value	Δ%	
Gradient proxy	w/o output-cache proxy	0.937	+107.7	0.629	+35.7	338.92
Layer selection	w/o layer selection	0.538	+19.1	0.482	+4.0	642.98
Variable pooling	w/o conflict pooling	0.858	+90.0	0.607	+31.0	253.42
	w/o anchor pooling	0.506	+12.1	0.478	+3.2	272.47
	w/o variable pooling	0.497	+10.0	0.474	+2.2	277.40
Direction surgery	w/o magnitude decoupling	0.512	+13.5	0.493	+6.3	333.05
	w/o safe candidate selection	0.478	+6.0	0.478	+3.1	312.27
PV-Surgery (full)		**0.451**	–	**0.463**	–	332.45

5.4 Mechanism evidence

We directly test the mechanism by comparing exact per-variable gradients with cache-reconstructed proxies across the main-result settings (Figure 5). Proxies are nearly exact at the output boundary. Fidelity declines after variable mixing, but nonzero selected-slice proxies remain close. Appendix J details the protocol and tests whether surgery improves first-order update behavior through conflict mass, which measures pairwise disagreement among reconstructed proxy rows, and descent violations, which count variable objectives pushed uphill by the update. PV-Surgery lowers violations in most runs without increasing conflict mass. The appendix also analyzes oracle-gap reduction, where 23 of 29 harmed variables close part of their oracle gap (Figure 17). Appendix K gives qualitative results.

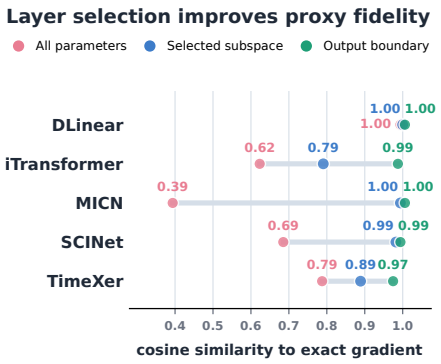


Figure 5: Cached-to-exact gradient cosine by backbone. Selected-slice cosine excludes zero proxies.

6 Conclusion

Multivariate forecasters optimize a loss averaged across variables, leaving the effect of the shared update on each variable unclear. We find that pairwise gradient conflict is common and most variables underperform a full-input single-target oracle. But conflict frequency is weakly related to harm, and mean alignment does not reliably identify helpful partners. PV-Surgery addresses this problem with a single-backward optimizer-side transformation. The method constructs variable-wise gradient proxies, restricts surgery to a reconstruction-consistent subspace, pools variables conditionally, and corrects directions while preserving surgery-input norms. Across matched settings, it reduces MSE by 3.61% and MAE by 2.93% on average, with every module contributing to both metrics. Mechanism analyses show that selected proxies track variable-wise gradients and forecasts for harmed variables often improve. Together, the results support selective, reliability-aware intervention over uniform conflict removal. PV-Surgery has two main limitations. The method raises median per-epoch cost to $1.54\times$ the mean-loss baseline (Appendix L), and gains can weaken when proxy fidelity is low, selected layers mix variables, or variable relationships differ across datasets. Future work should reduce this overhead and improve proxy construction under stronger variable mixing and more diverse cross-variable structures without sacrificing useful sharing.

AI USE STATEMENT

In this work, we used generative AI tools for research ideation and execution. They provided feedback on the method and experimental design, assisted with parts of the method implementation, analysis code, and plotting code, and helped interpret experimental results. For retrieval and discovery, they helped locate candidate references, each of which we read and verified before citing. For drafting the paper, they helped polish prose, translate draft passages for internal review, and draft passages of some sections from our outlines and results, which we then revised. We did not use these tools to originate the research question or the central research idea. We reviewed all AI-assisted text and code and take responsibility for the final text, claims, and artifacts.

ETHICS STATEMENT

The experiments use established public forecasting benchmarks and do not collect personal data or make individual-level decisions. Benchmark accuracy does not establish reliability under distribution shift or safety in operational use. Any high-stakes deployment would require domain-specific validation, monitoring, and appropriate human oversight.

REPRODUCIBILITY STATEMENT

Section 4 and Algorithm 1 specify the loss scaling, cache construction, layer-selection rule, pooling and restoration operations, candidate comparison, and final-gradient assembly. Appendix E reports the forecasting protocol and optimization settings. Tables 1 and 6 cover every one of the $5 \times 7 \times 4 = 140$ main comparisons, while Table 3 and Appendix H jointly cover all 140 matched ablation settings per variant. Except for the multi-seed analyses in Appendix C, all reported runs use the single seed 42, as specified in Appendix E. The accompanying anonymous supplementary material provides the source code, configuration files, execution scripts, and result-processing scripts.

REFERENCES

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.

Yisong Fu, Zezhi Shao, Chengqing Yu, Yujie Li, Zhulin An, Qi (Cheems) Wang, Yongjun Xu, and Fei Wang. Selective learning for deep time series forecasting. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/8cf54ff53f44835b9bdab2c546a1ca6d-Abstract-Conference.html.

Vincent Le Guen and Nicolas Thome. Shape and time distortion loss for training deep time series forecasting models. 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. 4191–4203, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/466accbac9a66b805ba50e42ad715740-Abstract.html.

Mark O Hill. Diversity and evenness: a unifying notation and its consequences. *Ecology*, 54(2): 427–432, 1973.

Dilfira Kudrat, Zongxia Xie, Yanru Sun, Tianyu Jia, and Qinghua Hu. Patch-wise structural loss for time series forecasting. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Fe-

lix 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, 2025. URL https://proceedings.mlr.press/v267/kudrat25a.html.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz (eds.), *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pp. 95–104. ACM, 2018. doi: 10.1145/3209978.3210006. URL https://doi.org/10.1145/3209978.3210006.

Hyunwook Lee, Chunggi Lee, Hongkyu Lim, and Sungahn Ko. TILDE-Q: A transformation invariant loss function for time-series forecasting. *CoRR*, abs/2210.15050, 2022. doi: 10.48550/ARXIV.2210.15050. URL https://doi.org/10.48550/arXiv.2210.15050.

Bryan Lim and Stefan Zohren. Time series forecasting with deep learning: A survey. *CoRR*, abs/2004.13408, 2020. URL https://arxiv.org/abs/2004.13408.

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.

Minhao Liu, Ailing Zeng, Muxi Chen, Zhijian Xu, Qiuxia Lai, Lingna Ma, and Qiang Xu. Scinet: Time series modeling and forecasting with sample convolution and interaction. 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/266983d0949aed78a16fa4782237dea7-Abstract-Conference.html.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=JePfAI8fah.

Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16428–16446. PMLR, 2022. URL https://proceedings.mlr.press/v162/navon22a.html.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. 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=Jbdc0vTOcol.

Liran Nochumsohn, Hedi Zisling, and Omri Azencot. A multi-task learning approach to linear multivariate forecasting. In Yingzhen Li, Stephan Mandt, Shipra Agrawal, and Mohammad Emtiyaz Khan (eds.), *International Conference on Artificial Intelligence and Statistics, AISTATS 2025, Mai Khao, Thailand, 3-5 May 2025*, volume 258 of *Proceedings of Machine Learning Research*, pp. 2638–2646. PMLR, 2025. URL https://proceedings.mlr.press/v258/nochumsohn25a.html.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. 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. 525–536, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/432aca3a1e345e339f35a30c8f65edce-Abstract.html.

Dmitry Senushkin, Nikolay Patakin, Arseny Kuznetsov, and Anton Konushin. Independent component alignment for multi-task learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 20083–20093. IEEE, 2023. doi: 10.1109/CVPR52729.2023.01923. URL https://doi.org/10.1109/CVPR52729.2023.01923.

Hao Wang, Lichen Pan, Yuan Shen, Zhichao Chen, Degui Yang, Yifei Yang, Sen Zhang, Xinggao Liu, Haoxuan Li, and Dacheng Tao. Fredf: Learning to forecast in the frequency domain. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=4A9IdSa1ul.

Huiqiang Wang, Jian Peng, Feihu Huang, Jince Wang, Junhui Chen, and Yifei Xiao. MICN: multi-scale local and global context modeling for long-term series forecasting. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=zt53IDUR1U.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. Timexer: Empowering transformers for time series forecasting with exogenous variables. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/0113ef4642264adc2e6924a3cbbdf532-Abstract-Conference.html.

Zirui Wang, Yulia Tsvetkov, Orhan Firat, and Yuan Cao. Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=F1vEjWK-lH_.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. 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. 22419–22430, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/bcc0d400288793e8bdcd7c19a8ac0c2b-Abstract.html.

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.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In Brian Williams, Yiling Chen, and Jennifer Neville (eds.), *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pp. 11121–11128. AAAI Press, 2023. doi: 10.1609/AAAI.V37I9.26317. URL https://doi.org/10.1609/aaai.v37i9.26317.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February*

2-9, 2021, pp. 11106–11115. AAAI Press, 2021. doi: 10.1609/AAAI.V35I12.17325. URL https://doi.org/10.1609/aaai.v35i12.17325.

A Dataset Diagnostics

Table 4 summarizes the datasets and the diagnostic quantities used to contextualize PV-Surgery. In addition to common dataset metadata, we report conflict-oriented indicators tied to the optimization issue studied in this paper. The negative cosine ratio measures how often variable-wise gradients disagree, the mean cosine measures average alignment, and the harmed-variable fraction is the share of variables for which shared training performs worse than the full-input single-target oracle. Diagnostics are computed with iTransformer at prediction length 96, except for ILI where prediction length 24 is used.

Table 4: Dataset statistics and conflict-oriented diagnostics. T is the number of timestamps and D is the number of target variables. Neg. cos. is the percentage of negative pairwise variable-gradient cosines and Mean cos. is their average. Both follow the protocol of Section 3 and are computed after dropping the warmup half of the recorded checkpoints. Harmed vars. is the percentage of variables for which shared training is worse than the full-input single-target oracle on the held-out test split.

Dataset	Frequency	T	D	Neg. cos. (%)	Mean cos.	Harmed vars. (%)
ETTh1	1 hour	17,420	7	30.6	0.178	14.3
ETTh2	1 hour	17,420	7	30.4	0.281	57.1
ETTm1	15 min	69,680	7	30.8	0.172	28.6
ETTm2	15 min	69,680	7	29.9	0.246	57.1
Weather	10 min	52,696	21	37.5	0.133	81.0
Exchange	1 day	7,588	8	39.3	0.148	25.0
ILI	1 week	966	7	15.9	0.508	71.4

B Gradient-Conflict Figures for All Datasets

Figure 2 presents ETTh1 in the main text. Figures 6–11 use the same three-panel layout, consisting of the post-warmup cosine distribution and heatmaps from an early and a late checkpoint, for the remaining six benchmarks. ETTh1, ETTh2, ETTm1, ETTm2, Weather, and Exchange use prediction length 96, and ILI uses 24.

Conflict is intermittent. No variable pair carries a negative cosine at every post-warmup checkpoint in any of the seven benchmarks. The most persistent pair is on ETTh2 and is negative at 72% of its checkpoints, while the median pair ranges from 11% to 43% across the seven benchmarks (Table 5). When each pair is averaged across its post-warmup checkpoints, the resulting cosines span $[-0.12, 0.98]$. Thus, even the lowest-alignment pairs are only weakly opposed on average, and this average hides changes in sign over training. A static pair-level rule could therefore act after the relation has changed. To account for this behavior, the method in Section 4 instead recomputes the relevant gradient relations from the current batch and reevaluates layer support at every step.

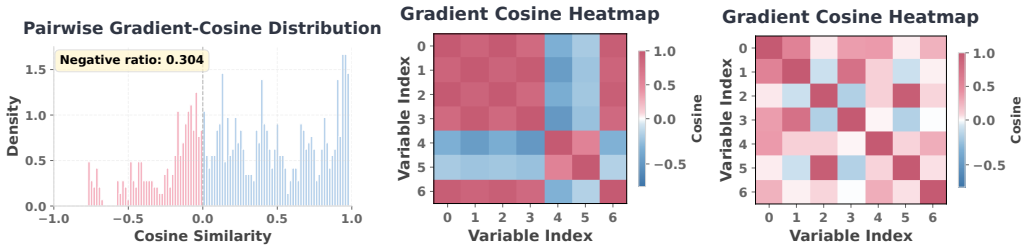
Table 5: Persistence of variable-pair conflict over training. The Checkpoints column gives the number of post-warmup analysis points, and the Pairs column gives the number of variable pairs in each benchmark. Persistence is the share of those checkpoints at which a pair has a negative cosine, reported for the most persistent pair and for the median pair. No entry reaches 100%, so no pair conflicts throughout.

Dataset	Checkpoints	Pairs	Persistence (%)	
			Most persistent	Median
ETTh1	46	21	50.0	28.3
ETTh2	39	21	71.8	20.5
ETTm1	48	21	50.0	35.4
ETTm2	37	21	48.6	40.5
Weather	44	210	68.2	40.9
Exchange	38	28	55.3	43.4
ILI	150	21	37.3	11.3

C Group-Composition Diagnostics for All Datasets

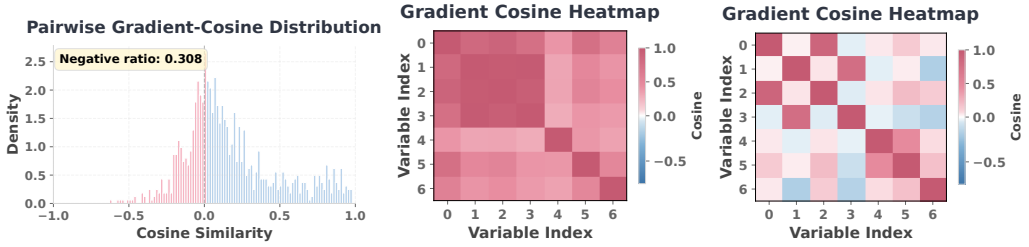
Group-composition protocol. For each reported target variable, we rank the remaining variables by their post-warmup mean pairwise gradient cosine in the baseline run. For each total group size $k \in \{2, 3, 4\}$, we train one model with the $k-1$ highest-ranked partners and another with the $k-1$ lowest-ranked partners. The two partner sets are disjoint in every reported cell. Each model still receives every variable as input. Only the supervised loss is restricted to the target and its selected partners, and evaluation uses the target's held-out per-variable test MSE. The 162 target-and-size cells cover every target in the four ETT benchmarks, Exchange, and ILI, plus 11 of Weather's 21 targets. Higher- and lower-alignment conditions use matched replicate seeds, three per cell, which totals 972 group-training runs. We report the seed-averaged signed contrast $100(\mathrm{MSE}_{\mathrm{high}} - \mathrm{MSE}_{\mathrm{low}})/\mathrm{MSE}_{\mathrm{low}}$. Negative values favor the higher-alignment set. A cell counts as a higher-alignment win when its signed contrast is negative. The median signed contrast retains each cell's direction, whereas the median absolute within-cell contrast removes the sign and measures the size of the partner-set difference. For every fixed dataset, target, group size, and partner condition, we express the standard deviation of test MSE across the three seeds as a percentage of their mean. To express seed variation on the scale of a two-run difference, we multiply this coefficient of variation by $\sqrt{2}$, which is the standard deviation of the difference under an equal-variance independence approximation. We use it as a descriptive reference rather than as a paired uncertainty estimate. The median over the 324 fixed partner conditions is 1.1%.

Figure 12 splits Figure 3(b) by dataset, and Figure 13 reports the higher- minus lower-alignment contrast for every benchmark and group size. Higher-alignment sets win 100 of the 162 cells, and the median signed contrast across cells is -1.4%. A two-sided one-sample t-test over the seven benchmark-level mean contrasts does not reject a zero mean ($p = 0.45$), although five of the seven means are negative. Excluding ILI shifts the pooled contrast from -0.9% to $+0.4\%$. The median absolute within-cell contrast is 3.5%, which exceeds the 1.1% descriptive seed-variation reference above, so partner choice can materially change target error even though mean alignment does not reliably identify the better set. Across all 162 cells, Spearman's correlation between the alignment gap and MSE difference is -0.19. The arithmetic mean of the seven within-benchmark correlations is -0.16, and a two-sided one-sample t-test of those correlations against zero gives $p = 0.07$. Figure 14 reports the seed-42 Weather extension at the six tested group sizes $k \in \{2, 3, 4, 5, 8, 11\}$, including the disjoint-set limit $k = 11$ for $D = 21$.



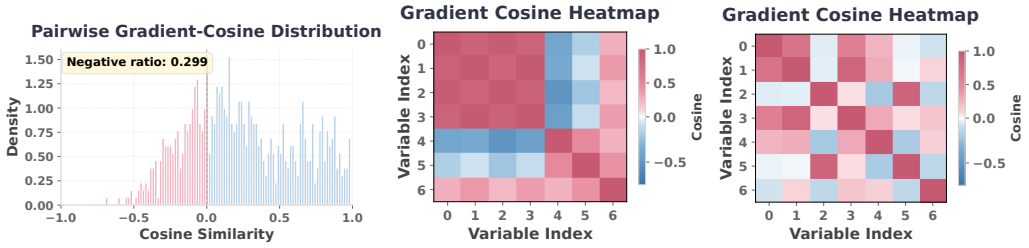
(a) Post-warmup cosine distribution. (b) Early checkpoint. (c) Late checkpoint.

Figure 6: A representative ETTh2 run from the variable-wise gradient-cosine diagnostic at prediction length 96. (a) Distribution of the post-warmup pairwise cosines, of which 30.4% fall below zero for this run. (b) An early checkpoint, which lies inside the discarded warmup half and is shown only to expose the change over training. (c) A late checkpoint. Both heatmaps show that conflict is structured across variable pairs rather than being uniform, so one summary number would hide which pairs disagree.



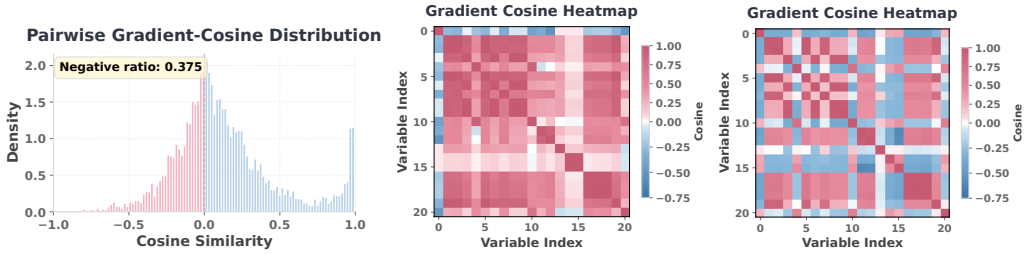
(a) Post-warmup cosine distribution. (b) Early checkpoint. (c) Late checkpoint.

Figure 7: A representative ETTm1 run from the variable-wise gradient-cosine diagnostic at prediction length 96. (a) Distribution of the post-warmup pairwise cosines, of which 30.8% fall below zero for this run. (b) An early checkpoint, which lies inside the discarded warmup half and is shown only to expose the change over training. (c) A late checkpoint. Both heatmaps show that conflict is structured across variable pairs rather than being uniform, so one summary number would hide which pairs disagree.



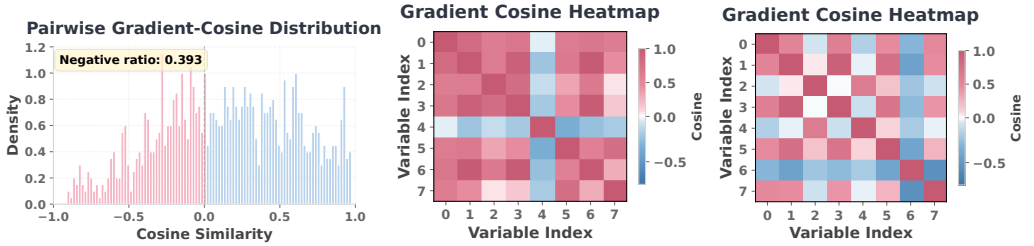
(a) Post-warmup cosine distribution. (b) Early checkpoint. (c) Late checkpoint.

Figure 8: A representative ETTm2 run from the variable-wise gradient-cosine diagnostic at prediction length 96. (a) Distribution of the post-warmup pairwise cosines, of which 29.9% fall below zero for this run. (b) An early checkpoint, which lies inside the discarded warmup half and is shown only to expose the change over training. (c) A late checkpoint. Both heatmaps show that conflict is structured across variable pairs rather than being uniform, so one summary number would hide which pairs disagree.



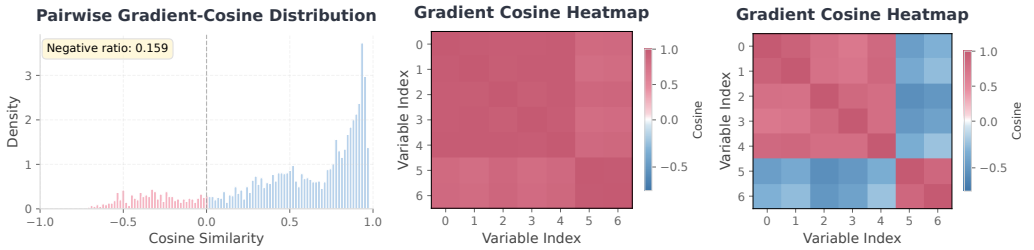
(a) Post-warmup cosine distribution. (b) Early checkpoint. (c) Late checkpoint.

Figure 9: A representative Weather run from the variable-wise gradient-cosine diagnostic at prediction length 96. (a) Distribution of the post-warmup pairwise cosines, of which 37.5% fall below zero for this run. (b) An early checkpoint, which lies inside the discarded warmup half and is shown only to expose the change over training. (c) A late checkpoint. Both heatmaps show that conflict is structured across variable pairs rather than being uniform, so one summary number would hide which pairs disagree.



(a) Post-warmup cosine distribution. (b) Early checkpoint. (c) Late checkpoint.

Figure 10: A representative Exchange run from the variable-wise gradient-cosine diagnostic at prediction length 96. (a) Distribution of the post-warmup pairwise cosines, of which 39.3% fall below zero for this run. (b) An early checkpoint, which lies inside the discarded warmup half and is shown only to expose the change over training. (c) A late checkpoint. Both heatmaps show that conflict is structured across variable pairs rather than being uniform, so one summary number would hide which pairs disagree.



(a) Post-warmup cosine distribution. (b) Early checkpoint. (c) Late checkpoint.

Figure 11: A representative ILI run from the variable-wise gradient-cosine diagnostic at prediction length 24. (a) Distribution of the post-warmup pairwise cosines, of which 15.9% fall below zero for this run. (b) An early checkpoint, which lies inside the discarded warmup half and is shown only to expose the change over training. (c) A late checkpoint. Both heatmaps show that conflict is structured across variable pairs rather than being uniform, so one summary number would hide which pairs disagree.

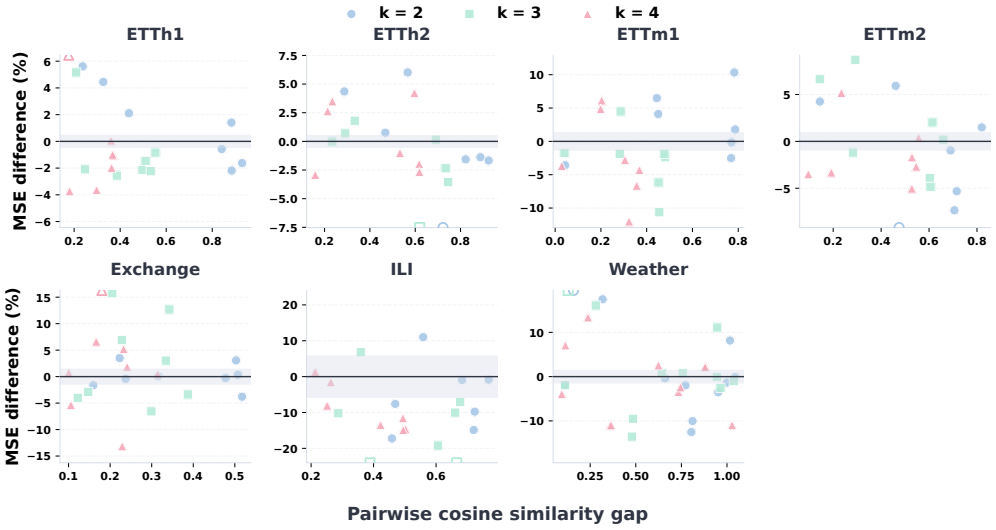


Figure 12: Figure 3(b) split by dataset, with the three group sizes marked separately. Each plot holds the 21 to 33 target-and-size cells of one benchmark.

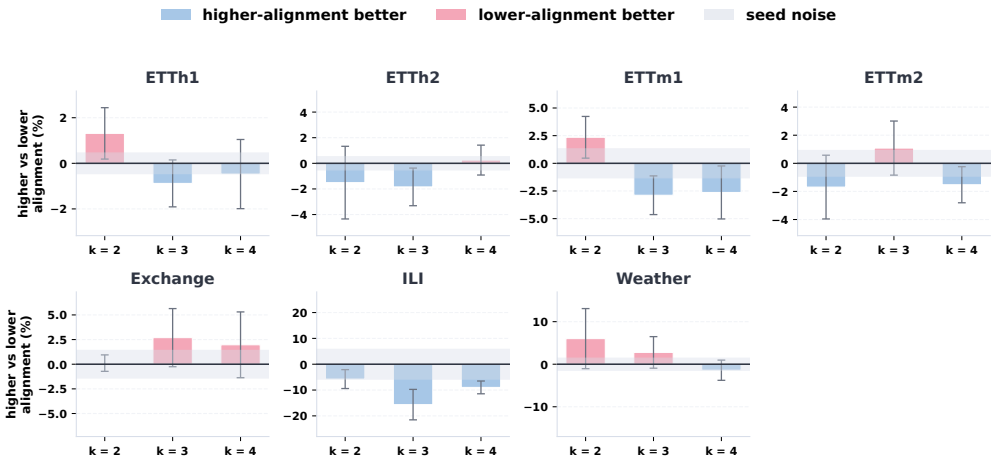


Figure 13: Percentage difference between the higher- and lower-alignment partner groups' test MSE, taken relative to the lower-alignment group and averaged over the targets of each benchmark. Negative means the higher-alignment group did better. Vertical ranges differ across panels. The grey band is that benchmark's seed noise and the error bars are the standard error across its targets.

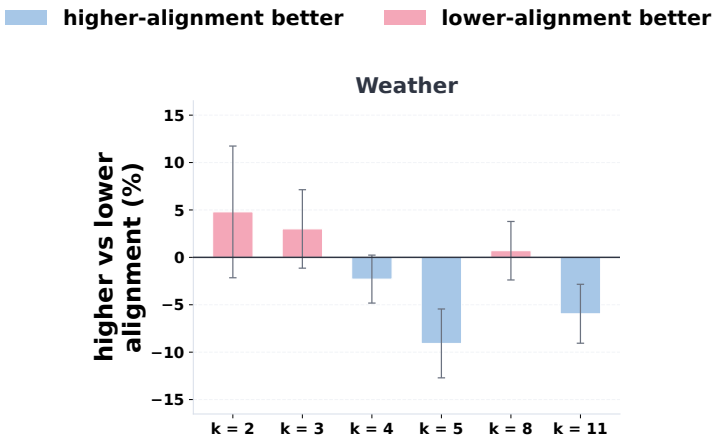


Figure 14: Weather at the six tested group sizes $k \in \{2, 3, 4, 5, 8, 11\}$, using seed 42. Bars show the mean higher- versus lower-alignment MSE contrast across 11 targets, and error bars show the standard error across targets. Negative values favor the higher-alignment group. The contrast does not change monotonically with group size.

D PV-SURGERY TRAINING PROCEDURE

Algorithm 1 makes the four stages in Figure 4 explicit for one training batch. The reference gradient and output-cache signals come from the same single backward pass on the summed variable loss. Because no rescaling is applied to this sum loss, its raw reference gradient satisfies $\|g_0\|_2 = D\|g_{\text{mean}}\|_2$. Baseline and PV-Surgery runs use the same AdamW hyperparameters. The remaining operations transform cached gradient matrices and assemble the selected-slice correction. Let $\mathcal{H}_{\text{run}} \subseteq \mathcal{H}$ denote the layers whose proxy rows materialize for the current batch. We construct their observed proxy matrices and set $G_\ell = 0_{D\times N_\ell}$ for $\ell \in \mathcal{H} \setminus \mathcal{H}_{\text{run}}$. Layer scoring and final assembly use this zero-extended collection over $\mathcal{H}$, while parameters outside the hooked set retain their reference-gradient values.

Algorithm 1 PV-Surgery Training

Input: batch (X, Y), D variables, backbone f_θ, cache-compatible hooked layers $\mathcal{H}$
Output: optimizer gradient g_{final}
// Step 1: Output-cache variable-wise gradient proxy
$\hat{Y} \leftarrow f_\theta(X)$ {hooks cache $\{Z_\ell\}_{\ell\in\mathcal{H}}$}
$L_\Sigma \leftarrow \sum_{d=1}^{D} L_d(\hat{Y}_{:,:,d}, Y_{:,:,d})$ $\quad g_0 \leftarrow \text{Backward}(L_\Sigma)$ {Eqs. 2, 3}
$\mathcal{H}_{\text{run}} \leftarrow \{\ell \in \mathcal{H} : Z_\ell \text{ and } E_\ell \text{ materialize for this batch}\}$
$G_\ell[d,\cdot] \leftarrow \text{concat}\big(\text{vec}(E_{\ell,d}^\top Z_{\ell,d}), E_{\ell,d}^\top \mathbf{1}\big)$ for $\ell \in \mathcal{H}_{\text{run}}$ {Eqs. 4, 5}
$G_\ell \leftarrow 0_{D\times N_\ell}$ for $\ell \in \mathcal{H} \setminus \mathcal{H}_{\text{run}}$
// Step 2: Reliability-aware layer selection
$(e_\ell, r_\ell, q_\ell) \leftarrow \text{LayerScores}(G_\ell, g_{0,\ell}, g_0)$ for $\ell \in \mathcal{H}$ {Eq. 6}
$\mathcal{H}_{\text{elig}} \leftarrow \{\ell \in \mathcal{H} : \ell \text{ is not protected}\}$
$\rho \leftarrow \text{median}\{r_\ell : \ell \in \mathcal{H}_{\text{elig}}\}$
$\mathcal{H}_{\text{gate}} \leftarrow \{\ell \in \mathcal{H}_{\text{elig}} : r_\ell \geq \rho \text{ or } \ell = \ell_{\text{out}}\}$
$\mathcal{A} \leftarrow \{\ell \in \mathcal{H}_{\text{gate}} : q_\ell > 0\}$
if $\mathcal{A} \neq \varnothing$ **then**
  $\pi_\ell \leftarrow q_\ell / \sum_{j\in\mathcal{A}} q_j$ $\quad K_{\text{eff}} \leftarrow \lceil \exp(-\sum_{\ell\in\mathcal{A}} \pi_\ell \log \pi_\ell) \rceil$
  $\mathcal{T} \leftarrow \text{Top}_{K_{\text{eff}}}(\mathcal{A}, q) \cup (\mathcal{A} \cap \{\ell_{\text{out}}\})$
  $\mathcal{S} \leftarrow \{\ell \in \mathcal{T} : e_\ell \leq 1\}$ {Eq. 7}
else
  $\mathcal{S} \leftarrow \varnothing$
end if
$\mathcal{S} \leftarrow \text{OutputFallback}(\mathcal{S}, \mathcal{H}_{\text{elig}}, \ell_{\text{out}})$
$G_{\mathcal{S}} \leftarrow \text{concat}_{\ell\in\mathcal{S}}\, G_\ell$ {Eq. 8}
// Step 3: Conditional variable-gradient pooling
$G_{\text{pre}} \leftarrow G_{\mathcal{S}}$ $\quad g_{\text{ref}} \leftarrow g_{0,\mathcal{S}}$
$\mathcal{V} \leftarrow \{d : G_{\text{pre}}[d] \text{ is finite and } \|G_{\text{pre}}[d]\|_2 > \epsilon\}$
$(a, M_{\text{pool}}, \text{active}) \leftarrow \text{ConditionalAnchorConflict}(G_{\text{pre}}, g_{\text{ref}}; \text{valid} = \mathcal{V}, \text{order} = (a_d, d))$ {Eq. 9}
if active **then**
  $G_{\text{pool}} \leftarrow M_{\text{pool}} G_{\text{pre}}$ {Eq. 10}
end if
// Step 4: Magnitude-preserving common-direction surgery
$G_{\text{in}}^{(u)} \leftarrow G_{\text{pre}}$, and $G_{\text{in}}^{(p)} \leftarrow G_{\text{pool}}$ when pooling is active
for each available candidate $c \in \{u, p\}$ **do**
  $\widetilde{G}_{\text{in}}^{(c)} \leftarrow \text{CommonDirection}_{\text{preserve}}(G_{\text{in}}^{(c)})$ {Eqs. 11, 12}
end for
$G_u^{\text{post}} \leftarrow \widetilde{G}_{\text{in}}^{(u)}$
if active **then**
  $G_p^{\text{post}} \leftarrow \text{PoolRestore}(G_{\text{pre}}, M_{\text{pool}}, \widetilde{G}_{\text{in}}^{(p)})$ {Eq. 13}
else
  $G_p^{\text{post}} \leftarrow G_{\text{pre}}$
end if
$G_{\text{post}} \leftarrow \text{DescentLexicographicSelect}^{u\text{-tie}}(G_{\text{pre}}, G_u^{\text{post}}, G_p^{\text{post}}; \mathcal{V})$ {Eq. 14}
$\Delta_{\mathcal{S}} \leftarrow \sum_d G_{\text{post}}[d,\cdot] - \sum_d G_{\text{pre}}[d,\cdot]$ {Eq. 15}
$g_{\text{final}} \leftarrow \text{Assemble}(g_0, \{G_\ell\}_{\ell\in\mathcal{H}}, \mathcal{S}, \Delta_{\mathcal{S}})$ {Eq. 16}

Pooling and candidate-selection conventions. Cosine and pairwise-conflict tests use only rows that are finite and have norm greater than the numerical tolerance. Cosines within a machine-scale

tolerance of zero are treated as zero. A zero or non-finite row is assigned $a_d = 0$, remains a singleton objective, and does not activate pooling. A zero or non-finite reference yields the identity pooling plan. When pooling is active, conflict rows are processed in ascending reference-alignment order, with the variable index breaking ties. Each row joins the existing conflict pool with the largest cosine only when that cosine is positive. Otherwise, it starts a new pool. Candidate scoring uses the same valid pre-surgery rows. A candidate with a zero or non-finite aggregate is invalid. If only one candidate is valid, it is selected. If both are invalid or numerically tied, the unpooled candidate is retained.

Numerical tolerance for common-direction surgery. Equation 12 defines the ideal map using an exact zero test. In the implementation, we use the first branch only when $\|\bar{g}^{(c)}\|_2 > \epsilon$ and otherwise retain the input, where $\epsilon > 0$ is a small numerical tolerance. This avoids normalizing a near-zero mean direction.

Proxy-supported hooked-block updates. PV-Surgery treats the hooked parameter blocks as the region in which an update should remain attributable to variable-indexed signals. When $\ell \in \mathcal{H}_{\mathrm{run}}$, every hooked-block contribution is constructed from observed proxy rows and can therefore pass through the same reconstruction, pooling, and direction checks as the other variable-wise inputs. When $\ell \notin \mathcal{H}_{\mathrm{run}}$, the single-backward slice $g_{0,\ell}$ is still available, but it contains only the aggregate of the unresolved variable contributions. Substituting that slice would reintroduce an update whose variable-wise sources cannot be inspected by the method. We therefore impose a blockwise evidence-support constraint that permits a nonzero hooked-block data-gradient contribution only when its variable-indexed rows materialize. Under this constraint, zero is the only admissible current data-gradient contribution on an unsupported block and is also the minimum-norm choice. This is a conservative abstention rule rather than a claim that a zero data-gradient contribution is a better descent direction than $g_{0,\ell}$. Parameters outside $\mathcal{H}$ are not assigned to the variable-aware region and continue to use g_0. A zero-extended block remains zero even if it enters $\mathcal{S}$, because its columns are zero for every surgery input and the norm-preserving direction operator cannot create support on those columns.

This zero data-gradient contribution is injected as an explicit zero loss-gradient tensor. It is therefore not the same as removing the parameters from AdamW or setting their gradients to `None`. AdamW still advances their optimizer state and applies decoupled weight decay. If a block has accumulated momentum from earlier nonzero updates, that state can continue to affect later parameter values while it decays. The rule should consequently be understood as withholding the current data-gradient update on an unsupported hooked block, not as permanently freezing the block.

Layer-selection thresholds. Although the cache supports explicit variable axes and supported batch-folded layouts, the configured protection rule excludes patch and value embedding parameters from direction-surgery selection to avoid changing the shared input representation before downstream variable mixing. If registered as hooked blocks, these parameters still follow the proxy-sum assembly rule above. Excluding them from $\mathcal{S}$ prevents direction-surgery correction on those slices but does not restore g_0. Their data-gradient contribution is the sum of the variable-indexed proxy rows when those rows materialize for the current batch, and zero when no such variable-axis proxy materializes. Among the remaining candidates the reliability gate keeps every non-output layer whose r_ℓ reaches the median over the eligible hooked layers, and the output layer is exempt from this gate. Equation 7 is evaluated on q_ℓ, which also orders the layers before the K_{eff} cut. A layer that survives the cut is eligible when its relative reconstruction error satisfies $e_\ell \leq 1$, equivalently $r_\ell \geq 1/2$. This means that its proxy-sum error does not exceed $\|g_{0,\ell}\|_2 + \epsilon$, which marks, up to numerical tolerance, the boundary at which the reconstruction error begins to exceed the magnitude of the gradient being reconstructed. If nothing is left, PV-Surgery falls back to the output layer.

Properties of effective coverage. The unrounded quantity $e^{\mathcal{E}_{\mathrm{layer}}}$ in Equation 7 is the exponential of the Shannon entropy of π and therefore the order-one Hill number (Hill, 1973) of the normalized slice-norm distribution. Because $\pi_\ell = q_\ell / \sum_{j \in \mathcal{A}} q_j$, it represents the fraction of the total gated slice norm carried by layer ℓ. The entropy $\mathcal{E}_{\mathrm{layer}}$ measures how evenly these fractions are distributed, and exponentiation converts this log-scale measure into an effective number of contributing layers. This number is one when a single layer carries all of the slice norm and m when the norm is uniform

over m layers. An intermediate value is the number of equally contributing layers that would produce the same concentration as the observed distribution. For example, $\pi = (0.8, 0.1, 0.1)$ gives $e^{\mathcal{E}_{\text{layer}}} \approx 1.89$ and therefore $K_{\text{eff}} = 2$ after taking the ceiling. The selector uses this ceiling as an entropy-derived cutoff for the largest-norm layers, while the reliability and reconstruction criteria still determine which layers are eligible. The Hill number is invariant to a common rescaling of the layer slice norms, so a backbone whose layer gradients are uniformly larger does not receive a wider support. It is recomputed at each step, allowing the support to contract when one layer dominates and to widen when the slice-norm distribution becomes diffuse. A layer whose share vanishes contributes $\pi_\ell \log \pi_\ell \to 0$ and has a vanishing effect on the effective count. The choice of order also matters. Order zero counts every gated layer equally, while order two emphasizes dominant layers more strongly. Order one retains the Shannon weighting in which each entropy contribution is weighted by its normalized slice-norm share π_ℓ. We avoid a fixed top-k because the number of hooked layers differs by backbone, from one to seventeen in our experiments, so a new backbone would otherwise need its own value.

E Detailed Experimental Configuration

Forecasting protocol. For ETTh1, ETTh2, ETTm1, ETTm2, Weather, and Exchange, we use an input length of 96, a decoder-label length of 48, and prediction lengths $\{96, 192, 336, 720\}$. ILI uses input and decoder-label lengths of 36 and 18. Its main, ablation, mechanism, and objective-comparison runs use prediction lengths $\{24, 36, 48, 60\}$, while the diagnostic, oracle, and partner-group analyses in Section 3 use 24. Every method uses the chronological train, validation, and test splits provided by the forecasting framework. Training windows are not shuffled because Selective Learning tracks residual statistics across consecutive windows (Fu et al., 2025).

Optimization. Unless otherwise noted for the multi-seed analyses in Appendix C, each reported run uses seed 42. Training uses a batch size of 64 for at most 30 epochs and stops after seven consecutive validation epochs without improvement. We use AdamW with a fixed learning rate of 10^{-4}, weight decay 5×10^{-4}, and a gradient-norm limit of 1.

Specialized-objective settings. In our objective comparison, TILDE-Q uses $\alpha = 0.5$ and $\gamma = 0$, which removes the amplitude term. FreDF weights the temporal MSE and frequency-domain terms by 0.5 each. Its frequency MAE is the mean modulus of the complex RFFT difference. PSLoss uses $\lambda = 3$, limits the adaptive patch length to 24, and enables Gradient-based Dynamic Weighting with automatic target-layer selection and an equal-weight fallback. Selective Learning uses $(r_u, r_a) = (0.3, 0.3)$, $(0.1, 0.6)$, $(0.2, 0.2)$, $(0.2, 0.5)$, $(0.1, 0.2)$, $(\text{off}, 0.9)$, and $(0.1, 0.1)$ for ETTh1, ETTh2, ETTm1, ETTm2, Weather, Exchange, and ILI, respectively. Its anomaly mask uses a pretrained DLinear estimator for each dataset and horizon. TILDE-Q, FreDF, and PSLoss select checkpoints using their configured validation objectives, whereas Selective Learning uses validation MSE. Early stopping monitors validation MSE with patience 7 for all four methods.

Diagnostic logging. The shared-model diagnostics in Section 3 record gradients every $\max(1, \lfloor 30S/200 \rfloor)$ optimizer steps, where S is the number of steps in one epoch. This interval targets approximately 200 snapshots over a complete 30-epoch run. Early stopping reduces the total in most benchmarks, while ILI has an interval of one step because each epoch is short. Before removing the warmup half, the runs contain 74 to 300 snapshots. The post-warmup counts reported in Appendix B are therefore 37 to 150.

Backbone hyperparameters. DLinear uses a moving-average window of 25 and shared rather than variable-specific trend and seasonal heads. iTransformer uses $d_{\text{model}} = 512$, $d_{\text{ff}} = 512$, 8 attention heads, 3 encoder layers, GELU activations, and dropout 0.1. TimeXer uses $d_{\text{model}} = 256$, $d_{\text{ff}} = 512$, 8 attention heads, 2 encoder layers, patch length 16, GELU activations, dropout 0.1, and input normalization. MICN uses $d_{\text{model}} = 512$, $d_{\text{ff}} = 2048$, 8 heads, 1 decoder layer, convolution kernels $\{12, 16\}$, and dropout 0.05. SCINet uses one stack and dropout 0.1. Standard MSE training and PV-Surgery use the same backbone settings.

Hardware. We run the experiments on NVIDIA TITAN RTX GPUs with 24 GB of memory.

Metric definitions. We compute MSE and MAE on standardized forecasts and targets without applying the inverse transformation. Let N_{test} denote the number of test windows. The reported metrics are

$$\begin{aligned} \text{MSE} &= \frac{1}{N_{\text{test}} HD} \sum_{n=1}^{N_{\text{test}}} \sum_{h=1}^{H} \sum_{d=1}^{D} (\hat{Y}_{n,h,d} - Y_{n,h,d})^2, \\ \text{MAE} &= \frac{1}{N_{\text{test}} HD} \sum_{n=1}^{N_{\text{test}}} \sum_{h=1}^{H} \sum_{d=1}^{D} |\hat{Y}_{n,h,d} - Y_{n,h,d}|. \end{aligned} \tag{17}$$

F Backbone Results for Weather and ILI

The page limit restricts Table 1 to five of the seven benchmarks. Table 6 reports the omitted Weather and ILI results, completing the same comparison over five backbones, seven datasets, and four prediction lengths. Each MSE and +PV pair uses the same data split, backbone configuration, and optimization protocol.

Table 6: Backbone comparison between standard MSE training and PV-Surgery on Weather and ILI. Within each backbone and metric, the better value between MSE and +PV is bolded using unrounded scores. The protocol and horizons follow Table 1.

Models		DLinear				iTransformer				MICN				SCINet				TimeXer			
Training setup		MSE		+PV		MSE		+PV		MSE		+PV		MSE		+PV		MSE		+PV	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Weather	96	**0.203**	0.266	0.203	**0.245**	0.183	0.222	**0.182**	**0.222**	0.398	0.442	**0.318**	**0.383**	**0.166**	**0.212**	0.170	0.215	**0.164**	**0.208**	0.180	0.223
	192	0.241	0.302	**0.241**	**0.292**	0.231	**0.263**	0.231	0.265	**0.403**	**0.447**	0.435	0.460	0.218	0.259	**0.218**	**0.257**	**0.208**	**0.251**	0.233	0.269
	336	0.286	0.338	**0.283**	**0.325**	0.285	0.300	**0.281**	**0.299**	**0.380**	**0.422**	0.436	0.468	0.280	0.303	**0.278**	**0.300**	**0.261**	**0.290**	0.283	0.302
	720	0.349	0.385	**0.345**	**0.368**	0.358	0.349	**0.353**	**0.345**	0.496	0.501	**0.479**	**0.493**	0.361	0.356	**0.357**	**0.352**	**0.340**	**0.342**	0.356	0.350
	Avg	0.270	0.323	**0.268**	**0.308**	0.264	0.283	**0.262**	**0.283**	0.419	0.453	**0.417**	**0.451**	0.256	0.282	**0.256**	**0.281**	**0.243**	**0.273**	0.263	0.286
ILI	24	3.828	1.464	**3.590**	**1.414**	**2.018**	**0.878**	2.190	0.905	**3.169**	**1.237**	3.190	1.243	3.946	1.405	**3.945**	**1.400**	2.798	**0.990**	**2.723**	1.047
	36	3.801	1.435	**3.490**	**1.358**	**2.173**	0.945	2.293	**0.943**	2.996	1.228	**2.985**	**1.220**	3.970	1.417	**3.778**	**1.369**	**2.204**	**0.931**	2.619	1.056
	48	3.808	1.420	**3.446**	**1.330**	**2.099**	**0.940**	2.283	0.941	3.269	1.284	**3.228**	**1.272**	3.977	1.434	**3.698**	**1.371**	**2.250**	**0.954**	2.542	1.041
	60	4.175	1.446	**3.760**	**1.359**	2.579	1.107	**2.136**	**0.932**	**3.273**	**1.267**	3.364	1.283	3.961	1.433	**3.639**	**1.368**	2.572	1.058	**2.406**	**1.009**
	Avg	3.903	1.441	**3.571**	**1.365**	**2.217**	0.967	2.225	**0.930**	**3.177**	**1.254**	3.191	1.254	3.963	1.422	**3.765**	**1.377**	**2.456**	**0.983**	2.572	1.038

G Loss-Baseline Results for All Backbones

The page limit restricts Table 2 to iTransformer on five of the seven benchmarks. Tables 7–11 report all seven benchmarks separately for each backbone, completing 280 backbone, dataset, prediction-length, and metric cells.

No method is best in every cell. Counting a tie for each tied method, PV-Surgery gives the lowest value in 91 of the 280 cells and Selective Learning in 77. These tables compare the methods as alternatives and do not evaluate combined objectives. The specialized methods change the forecasting loss, whereas PV-Surgery retains pointwise MSE and modifies the optimizer update. The two mechanisms can therefore be combined in principle, but their joint effect is not evaluated here.

H Dataset-Wise Component Ablations

Table 3 reports the component ablation for MICN on Exchange. Tables 12–18 give the corresponding result for each dataset. Every entry averages the five backbones and four prediction lengths, yielding 20 matched settings per dataset. Relative changes are computed against the full configuration using the unrounded averages.

For every removal, the mean of the seven dataset-level relative MSE changes is positive, so each module contributes on average. Removing a component improves MSE on some datasets, but the largest such gain is 0.9%. Every ablated variant is also at least 2.1% worse than the full configuration on another dataset. Overall, the full configuration has lower MSE in 37 of the 49 dataset-variant comparisons. It is not uniformly best, but retaining all modules avoids the larger and more frequent degradations and provides the most stable performance across datasets.

Table 7: Loss-baseline comparison on DLinear for all seven benchmarks. The MSE row denotes standard MSE training. Best results are bolded and second-best results are underlined using unrounded scores.

Dataset		ETTh1				ETTh2				ETTm1				ETTm2				Weather				Exchange				ILI			
Forecast length		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	24	36	48	60
MSE	MSE	0.462	0.514	0.559	0.651	0.239	0.329	0.437	0.650	0.388	0.447	0.507	0.575	0.160	0.209	0.262	0.348	0.203	0.241	0.286	0.349	0.105	0.224	0.425	**0.652**	3.828	3.801	3.808	4.175
	MAE	0.444	0.476	0.506	0.573	0.329	0.393	0.462	0.582	0.395	0.426	0.459	0.502	0.268	0.308	0.350	0.408	0.266	0.302	0.338	0.385	0.234	0.349	0.498	**0.640**	1.464	1.435	1.420	1.446
TILDE-Q	MSE	0.467	0.516	0.558	0.647	0.233	0.293	0.354	0.524	**0.383**	**0.445**	0.510	0.576	0.158	0.203	0.246	0.317	0.200	0.238	0.282	0.343	0.116	0.213	0.432	0.888	4.014	3.685	3.538	3.777
	MAE	0.443	0.472	**0.501**	0.567	0.317	0.361	0.408	0.514	**0.387**	**0.420**	0.455	**0.496**	0.258	0.294	0.325	0.375	0.242	0.281	0.321	0.371	0.250	0.348	0.509	0.760	1.529	1.410	1.352	1.365
FreDF	MSE	**0.459**	**0.511**	**0.555**	0.644	0.235	0.308	0.391	0.591	0.386	0.446	0.506	0.574	0.157	0.202	0.248	0.319	0.200	**0.238**	0.281	0.344	0.107	0.208	0.430	0.790	3.810	3.734	3.797	4.146
	MAE	**0.439**	0.471	0.501	0.566	0.323	0.378	0.437	0.550	0.392	0.424	0.458	0.501	0.260	0.295	0.331	0.380	0.249	0.286	0.320	0.370	0.239	0.345	0.497	0.700	1.466	1.422	1.427	1.456
PSLoss	MSE	0.462	0.513	0.557	0.648	0.238	0.308	0.390	0.626	0.385	0.445	0.505	0.573	0.156	**0.197**	0.242	0.317	0.202	0.238	**0.281**	**0.341**	0.120	0.226	0.388	0.700	3.961	3.775	3.577	3.763
	MAE	0.441	0.474	0.504	0.570	0.325	0.376	0.433	0.560	0.391	0.421	0.455	0.497	0.254	**0.286**	0.322	0.377	**0.242**	**0.277**	**0.311**	**0.359**	0.256	0.358	0.478	0.655	1.503	1.434	1.362	1.368
SL	MSE	0.460	0.514	0.558	0.650	0.244	0.323	0.395	0.623	0.392	0.452	0.511	0.578	0.160	0.216	0.275	0.378	**0.200**	0.240	0.288	0.348	0.105	0.219	0.406	0.675	3.872	3.819	3.753	4.158
	MAE	0.442	0.476	0.506	0.574	0.333	0.389	0.437	0.563	0.397	0.428	0.460	0.505	0.268	0.315	0.361	0.428	0.259	0.301	0.341	0.386	0.236	0.348	0.487	0.649	1.459	1.425	1.396	1.437
PV(Ours)	MSE	0.461	0.514	0.556	**0.641**	**0.229**	**0.290**	**0.341**	**0.470**	0.383	0.445	**0.505**	**0.573**	**0.155**	**0.197**	**0.239**	**0.306**	0.203	0.241	0.283	0.345	**0.100**	**0.191**	**0.321**	0.847	**3.590**	**3.490**	**3.446**	**3.760**
	MAE	0.440	**0.471**	**0.501**	**0.561**	**0.313**	**0.357**	**0.396**	**0.484**	0.389	0.421	**0.454**	0.499	**0.253**	0.287	**0.319**	**0.365**	0.245	0.292	0.325	0.368	**0.226**	**0.323**	**0.425**	0.724	**1.414**	**1.358**	**1.330**	**1.359**

Table 8: Loss-baseline comparison on iTransformer for all seven benchmarks. The MSE row denotes standard MSE training. Best results are bolded and second-best results are underlined using unrounded scores.

Dataset		ETTh1				ETTh2				ETTm1				ETTm2				Weather				Exchange				ILI			
Forecast length		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	24	36	48	60
MSE	MSE	0.454	0.517	0.561	0.671	0.239	0.303	0.356	0.457	0.407	0.480	0.520	0.618	0.157	0.205	0.252	0.322	0.183	0.231	0.285	0.358	0.108	0.222	0.397	1.099	2.018	2.173	**2.099**	2.579
	MAE	0.447	0.485	0.513	0.583	0.323	0.366	0.401	0.460	0.415	0.457	0.477	0.531	0.258	0.295	0.328	0.373	0.222	0.263	0.300	0.349	0.236	0.344	0.463	0.800	0.878	0.945	0.940	1.107
TILDE-Q	MSE	0.469	0.528	0.564	0.669	0.233	0.287	0.354	0.473	**0.402**	0.465	0.537	0.622	0.156	0.200	0.247	0.311	0.180	0.231	0.283	**0.350**	0.111	0.219	0.428	1.109	1.969	2.035	2.143	2.548
	MAE	0.455	0.489	0.510	0.577	0.315	0.353	0.400	0.466	0.404	0.440	0.483	0.534	0.251	0.285	0.320	0.362	0.217	0.261	0.297	**0.342**	0.239	0.341	0.481	0.806	0.928	0.931	**0.934**	1.095
FreDF	MSE	**0.451**	0.512	0.560	0.662	0.235	0.295	0.350	0.450	0.410	0.475	0.553	0.603	0.154	0.197	0.245	**0.309**	0.178	0.227	0.283	0.354	0.112	0.223	0.462	1.132	2.047	2.079	2.272	2.507
	MAE	0.445	0.478	0.508	0.576	0.317	0.360	0.397	0.455	0.408	0.452	0.492	0.524	0.250	0.284	0.319	0.362	0.216	0.257	0.297	0.346	0.241	0.344	0.502	0.819	**0.872**	**0.905**	0.984	1.081
PSLoss	MSE	0.458	0.513	0.558	0.665	0.234	0.291	0.360	0.473	0.429	0.499	0.538	0.626	0.153	0.196	**0.239**	0.310	**0.177**	**0.225**	**0.278**	0.352	0.113	0.213	0.413	1.146	**1.880**	**2.025**	2.300	2.562
	MAE	0.448	0.480	0.509	0.577	0.316	0.355	0.406	0.469	0.421	0.459	0.482	0.535	0.247	**0.281**	**0.314**	**0.361**	**0.214**	**0.254**	**0.293**	0.342	0.241	0.338	0.476	0.825	0.899	0.927	1.003	1.119
SL	MSE	0.465	0.527	0.560	0.665	**0.229**	**0.284**	0.348	**0.438**	0.404	**0.458**	0.521	0.590	0.153	0.199	0.243	0.312	0.182	0.236	0.280	0.356	0.108	**0.211**	0.393	1.228	2.324	2.312	2.324	2.348
	MAE	0.449	0.485	**0.501**	**0.568**	**0.312**	**0.349**	0.393	**0.449**	**0.400**	**0.432**	**0.465**	**0.505**	0.250	0.285	0.317	0.362	0.219	0.264	0.297	0.346	0.232	**0.332**	0.460	0.856	0.972	0.985	0.982	1.018
PV(Ours)	MSE	0.451	**0.508**	**0.550**	**0.661**	0.229	0.287	**0.334**	0.446	0.402	0.461	**0.517**	**0.575**	**0.150**	**0.195**	0.240	0.310	0.182	0.231	0.281	0.353	**0.105**	0.211	**0.390**	**1.075**	2.190	2.293	2.283	**2.136**
	MAE	**0.442**	**0.476**	0.504	0.575	0.314	0.355	**0.388**	0.452	0.404	0.439	0.473	0.510	**0.247**	0.283	0.317	0.365	0.222	0.265	0.299	0.345	**0.231**	0.335	**0.460**	**0.792**	0.905	0.943	0.941	**0.932**

Table 9: Loss-baseline comparison on MICN for all seven benchmarks. The MSE row denotes standard MSE training. Best results are bolded and second-best results are underlined using unrounded scores.

Dataset		ETTh1				ETTh2				ETTm1				ETTm2				Weather				Exchange				ILI			
Forecast length		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	24	36	48	60
MSE	MSE	0.552	0.594	0.688	0.753	0.238	0.315	0.452	0.672	0.445	0.495	0.567	0.627	0.166	0.202	0.266	0.389	0.398	0.403	0.380	0.496	0.123	0.219	0.498	2.931	**3.169**	2.996	3.269	**3.273**
	MAE	0.530	0.553	0.623	0.671	0.333	0.388	0.472	0.583	0.472	0.511	0.570	0.596	0.276	0.303	0.355	0.434	0.442	0.447	0.422	0.501	0.262	0.352	0.525	1.400	**1.237**	1.228	1.284	**1.267**
TILDE-Q	MSE	**0.480**	**0.529**	**0.556**	0.916	0.237	0.300	0.403	0.656	0.408	0.449	0.505	**0.548**	0.153	0.200	0.241	0.308	0.299	0.400	0.506	0.506	0.117	0.215	0.419	3.289	4.109	3.027	3.145	3.475
	MAE	0.485	0.518	**0.534**	0.747	0.325	0.371	0.445	0.579	0.434	0.464	0.501	**0.533**	0.257	0.295	0.324	0.373	0.364	0.431	0.501	0.501	0.252	0.353	0.498	1.426	1.477	1.245	1.265	1.303
FreDF	MSE	0.501	0.548	0.579	**0.699**	0.241	0.326	0.396	0.735	0.423	0.449	0.516	0.570	0.148	0.191	0.236	**0.301**	0.325	0.322	**0.340**	0.451	0.119	**0.210**	0.416	3.067	3.585	2.999	**3.054**	3.286
	MAE	0.504	0.533	0.552	**0.630**	0.332	0.396	0.446	0.618	0.454	0.471	0.510	0.554	0.252	0.286	0.322	**0.368**	0.386	0.384	**0.388**	0.469	0.254	0.350	0.495	1.426	1.325	1.236	**1.228**	1.274
PSLoss	MSE	0.492	**0.529**	0.558	0.753	0.236	0.309	0.402	0.630	0.428	0.475	0.516	0.924	0.149	**0.187**	0.233	0.302	0.282	0.337	0.423	0.480	0.129	0.243	0.446	1.649	3.231	2.991	3.141	3.347
	MAE	0.493	0.517	0.534	0.666	0.328	0.381	0.450	0.566	0.450	0.483	0.505	0.761	0.256	**0.283**	0.320	0.371	0.357	0.394	0.453	0.486	0.268	0.373	0.505	0.939	1.255	1.230	1.250	1.271
SL	MSE	0.489	0.532	0.624	0.769	0.236	0.301	0.385	0.615	**0.397**	**0.444**	**0.503**	0.611	0.153	0.209	0.258	0.384	**0.247**	**0.321**	0.376	**0.435**	**0.108**	0.220	0.448	1.353	3.317	**2.975**	3.185	3.586
	MAE	**0.466**	**0.494**	0.586	0.659	0.326	0.371	0.431	0.565	**0.408**	**0.442**	**0.468**	0.555	0.257	0.308	0.345	0.434	**0.322**	**0.378**	0.416	**0.464**	**0.240**	0.351	0.494	0.854	1.251	**1.208**	1.243	1.310
PV(Ours)	MSE	0.492	0.561	0.700	0.798	**0.230**	**0.291**	**0.352**	**0.513**	0.430	0.470	0.508	0.576	**0.146**	0.190	**0.231**	0.331	0.318	0.435	0.436	0.479	0.109	0.212	**0.378**	**1.106**	3.190	2.985	3.228	3.364
	MAE	0.497	0.543	0.630	0.700	**0.320**	**0.369**	**0.412**	**0.520**	0.460	0.487	0.513	0.562	**0.248**	0.285	**0.316**	0.389	0.383	0.460	0.468	0.493	0.242	**0.348**	**0.470**	**0.793**	1.243	1.220	1.272	1.283

Table 10: Loss-baseline comparison on SCINet for all seven benchmarks. The MSE row denotes standard MSE training. Best results are bolded and second-best results are underlined using unrounded scores.

Dataset		ETTh1				ETTh2				ETTm1				ETTm2				Weather				Exchange				ILI			
Forecast length		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	24	36	48	60
MSE	MSE	0.490	0.545	0.590	0.700	0.254	0.311	0.356	0.470	0.448	0.488	0.548	0.614	0.155	0.202	0.248	0.321	0.166	0.218	0.280	0.361	0.120	0.223	0.405	1.101	3.946	3.970	3.977	3.961
	MAE	0.464	0.495	0.521	0.588	0.334	0.370	0.402	0.472	0.428	0.453	0.483	0.521	0.253	0.290	0.322	0.368	0.212	0.259	0.303	0.356	0.249	0.347	0.470	0.804	1.405	1.417	1.434	1.433
TILDE-Q	MSE	0.485	0.540	0.585	0.696	0.238	0.301	0.353	0.454	0.441	0.475	0.534	0.603	0.148	0.194	0.240	0.312	0.162	0.217	0.277	0.352	0.116	0.218	0.399	1.092	4.101	4.126	3.969	3.921
	MAE	0.457	0.488	0.514	0.583	0.320	0.362	0.397	0.457	0.414	0.437	0.468	0.511	0.243	0.279	0.313	0.359	0.208	0.256	0.298	0.346	0.246	0.342	0.464	0.804	1.463	1.448	1.432	1.428
FreDF	MSE	**0.470**	**0.526**	**0.571**	**0.680**	**0.234**	**0.293**	**0.345**	**0.447**	0.444	0.481	0.531	0.589	**0.146**	**0.192**	**0.236**	**0.307**	0.161	0.213	0.271	0.352	0.112	0.216	0.393	1.093	**3.713**	**3.699**	3.842	3.939
	MAE	**0.447**	**0.479**	**0.505**	**0.575**	**0.316**	**0.356**	0.392	0.454	0.419	0.445	0.471	0.504	**0.243**	0.278	**0.309**	**0.356**	0.206	0.253	0.295	0.347	0.242	0.340	0.460	0.802	**1.359**	**1.365**	1.419	1.434
PSLoss	MSE	0.477	0.533	0.578	0.684	0.240	0.299	0.346	0.449	0.450	0.480	0.542	0.607	0.148	0.194	0.237	0.309	0.161	**0.211**	0.271	**0.351**	0.122	0.225	0.401	**1.086**	3.951	3.799	**3.646**	**3.553**
	MAE	0.453	0.485	0.512	0.580	0.320	0.359	0.393	0.457	0.422	0.440	0.472	0.509	0.246	0.281	0.311	0.358	0.205	0.250	0.293	**0.345**	0.253	0.347	0.466	**0.797**	1.399	1.368	**1.359**	**1.355**
SL	MSE	0.476	0.533	0.577	0.683	0.235	0.297	0.348	0.453	**0.389**	**0.449**	**0.513**	**0.585**	0.152	0.196	0.241	0.314	**0.160**	0.212	**0.270**	0.353	**0.111**	**0.212**	**0.385**	1.181	3.980	3.959	3.924	3.888
	MAE	0.449	0.482	0.507	0.576	0.318	0.359	0.394	0.457	**0.391**	**0.422**	**0.455**	**0.495**	0.250	0.282	0.316	0.361	**0.203**	**0.250**	**0.293**	0.348	**0.238**	**0.336**	**0.457**	0.836	1.406	1.413	1.423	1.420
PV(Ours)	MSE	0.473	0.530	0.574	0.687	0.237	0.297	0.346	0.447	0.421	0.490	0.551	0.628	0.150	0.193	0.236	0.308	0.170	0.218	0.278	0.357	0.116	0.222	0.409	1.103	3.945	3.778	3.698	3.639
	MAE	0.449	0.483	0.510	0.581	0.317	0.358	**0.391**	**0.454**	0.414	0.446	0.475	0.516	0.246	**0.278**	0.309	0.357	0.215	0.257	0.300	0.352	0.244	0.346	0.473	0.806	1.400	1.369	1.371	1.368

Table 11: Loss-baseline comparison on TimeXer for all seven benchmarks. The MSE row denotes standard MSE training. Best results are bolded and second-best results are underlined using unrounded scores.

Dataset		ETTh1				ETTh2				ETTm1				ETTm2				Weather				Exchange				ILI			
Forecast length		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	24	36	48	60
MSE	MSE	**0.460**	**0.511**	0.564	0.693	0.233	0.294	0.341	0.449	0.432	0.477	0.521	0.580	0.151	0.200	0.239	0.312	0.164	**0.208**	**0.261**	**0.340**	0.113	0.216	0.396	1.105	2.798	**2.204**	**2.250**	2.572
	MAE	0.454	0.485	0.511	0.590	0.319	0.360	0.394	0.457	0.421	0.453	0.479	0.515	0.249	0.287	0.316	0.367	0.208	0.251	0.290	0.342	0.239	0.337	0.459	0.798	0.990	**0.931**	**0.954**	1.058
TILDE-Q	MSE	0.474	0.525	0.574	0.693	0.238	0.295	0.338	**0.436**	0.417	0.460	0.517	0.566	**0.148**	0.194	0.237	**0.305**	0.163	0.216	0.275	0.352	0.122	0.227	0.456	1.171	2.615	2.503	2.520	2.547
	MAE	0.455	0.488	0.514	0.592	0.323	0.359	0.389	**0.448**	0.407	0.437	0.471	0.510	**0.244**	**0.280**	**0.312**	**0.357**	0.209	0.258	0.298	0.348	0.250	0.348	0.502	0.836	1.035	1.022	1.028	1.040
FreDF	MSE	0.475	0.556	0.658	0.802	0.243	0.300	0.348	0.448	0.452	0.510	0.575	0.661	0.155	0.202	0.244	0.316	0.165	0.224	0.281	0.361	0.133	0.256	0.478	1.226	2.659	2.526	2.640	2.658
	MAE	0.461	0.510	0.562	0.642	0.330	0.366	0.398	0.456	0.440	0.471	0.501	0.552	0.257	0.292	0.320	0.367	0.216	0.270	0.307	0.360	0.262	0.373	0.515	0.857	1.010	1.031	1.085	1.090
PSLoss	MSE	0.468	0.522	0.579	0.699	0.237	0.295	0.342	0.438	0.417	0.465	0.517	0.572	0.152	0.196	0.238	0.308	0.165	0.219	0.282	0.357	0.137	0.254	0.452	1.192	2.710	2.316	2.671	2.541
	MAE	0.459	0.494	0.520	0.602	0.321	0.360	0.391	0.448	0.416	0.445	0.477	0.512	0.252	0.285	0.315	0.360	0.216	0.262	0.304	0.352	0.268	0.372	0.499	0.844	1.044	0.973	1.090	1.081
SL	MSE	0.475	0.529	0.572	0.703	**0.226**	**0.283**	**0.329**	0.438	0.407	**0.455**	0.522	0.600	0.153	0.198	0.241	0.322	**0.162**	0.211	0.263	0.342	**0.108**	0.222	0.412	1.208	**2.539**	2.584	2.450	**2.392**
	MAE	**0.448**	**0.480**	**0.507**	0.586	**0.311**	**0.350**	**0.380**	0.454	0.407	**0.425**	**0.460**	**0.502**	0.252	0.284	0.315	0.368	**0.204**	**0.249**	**0.288**	**0.341**	**0.229**	0.340	0.472	0.845	**0.986**	1.038	1.018	**1.001**
PV(Ours)	MSE	0.461	0.518	**0.564**	**0.691**	0.231	0.295	0.342	0.446	**0.396**	0.455	**0.505**	**0.564**	0.149	**0.193**	**0.235**	0.307	0.180	0.233	0.283	0.356	0.113	**0.213**	**0.382**	**1.010**	2.723	2.619	2.542	2.406
	MAE	0.450	0.485	0.515	**0.581**	0.316	0.358	0.390	0.451	**0.403**	0.429	0.462	0.503	0.247	0.282	0.312	0.360	0.223	0.269	0.302	0.350	0.239	**0.335**	**0.452**	**0.763**	1.047	1.056	1.041	1.009

Table 12: Core component ablation on ETTh1, averaged over 20 matched backbone–prediction-length settings. Δ is the relative change against the full configuration. Bold indicates the best score and underlining the second best using unrounded averages.

Category	Module	MSE		MAE		Time (s)
		value	Δ%	value	Δ%	
Gradient proxy	w/o output-cache proxy[a]	0.574	+0.8	0.525	+0.9	192.97
Layer selection	w/o layer selection	0.569	−0.2	0.520	−0.0	288.11
Variable pooling	w/o conflict pooling	0.576	+1.1	0.524	+0.8	197.07
	w/o anchor pooling	0.568	−0.2	0.519	−0.0	213.68
	w/o variable pooling	**0.568**	−0.3	0.519	−0.1	198.92
Direction surgery	w/o magnitude decoupling	0.568	−0.3	**0.519**	−0.2	225.96
	w/o safe candidate selection	0.578	+1.5	0.525	+0.9	194.06
PV-Surgery (full)		0.570	–	0.520	–	213.65

[a] Removing the output-cache proxy also disables cache-dependent layer selection, pooling, and candidate comparison.

Table 13: Core component ablation on ETTh2, averaged over 20 matched backbone–prediction-length settings. Δ is the relative change against the full configuration. Bold indicates the best score and underlining the second best using unrounded averages.

Category	Module	MSE		MAE		Time (s)
		value	Δ%	value	Δ%	
Gradient proxy	w/o output-cache proxy[a]	0.367	+10.5	0.407	+5.6	238.37
Layer selection	w/o layer selection	0.343	+3.0	0.391	+1.5	338.14
Variable pooling	w/o conflict pooling	0.340	+2.2	0.390	+1.2	265.51
	w/o anchor pooling	0.344	+3.4	0.393	+1.9	252.61
	w/o variable pooling	0.343	+3.0	0.392	+1.7	239.77
Direction surgery	w/o magnitude decoupling	0.335	+0.6	0.387	+0.3	256.50
	w/o safe candidate selection	0.333	+0.2	0.386	+0.0	258.56
PV-Surgery (full)		**0.333**	–	**0.386**	–	274.73

[a] Removing the output-cache proxy also disables cache-dependent layer selection, pooling, and candidate comparison.

Table 14: Core component ablation on ETTm1, averaged over 20 matched backbone–prediction-length settings. Δ is the relative change against the full configuration. Bold indicates the best score and underlining the second best using unrounded averages.

Category	Module	MSE		MAE		Time (s)
		value	$\Delta\%$	value	$\Delta\%$	
Gradient proxy	w/o output-cache proxy[a]	0.511	+3.7	0.474	+2.3	792.11
Layer selection	w/o layer selection	0.496	+0.7	0.463	+0.1	1075.03
Variable pooling	w/o conflict pooling	<u>0.493</u>	+0.1	0.464	+0.2	902.63
	w/o anchor pooling	0.498	+1.0	0.467	+0.8	915.37
	w/o variable pooling	0.494	+0.4	0.465	+0.4	822.81
Direction surgery	w/o magnitude decoupling	0.498	+1.1	<u>0.463</u>	+0.0	906.55
	w/o safe candidate selection	0.497	+0.8	0.464	+0.3	919.78
PV-Surgery (full)		**0.493**	–	**0.463**	–	857.04

[a] Removing the output-cache proxy also disables cache-dependent layer selection, pooling, and candidate comparison.

Table 15: Core component ablation on ETTm2, averaged over 20 matched backbone–prediction-length settings. Δ is the relative change against the full configuration. Bold indicates the best score and underlining the second best using unrounded averages.

Category	Module	MSE		MAE		Time (s)
		value	$\Delta\%$	value	$\Delta\%$	
Gradient proxy	w/o output-cache proxy[a]	0.226	+1.4	0.308	+1.6	925.66
Layer selection	w/o layer selection	0.223	+0.2	0.303	+0.1	1126.85
Variable pooling	w/o conflict pooling	<u>0.222</u>	−0.7	<u>0.302</u>	−0.4	984.73
	w/o anchor pooling	0.228	+2.2	0.308	+1.6	1077.18
	w/o variable pooling	0.223	−0.0	0.304	+0.3	968.47
Direction surgery	w/o magnitude decoupling	0.222	−0.5	0.303	−0.1	1105.34
	w/o safe candidate selection	**0.221**	−0.9	**0.301**	−0.6	1000.83
PV-Surgery (full)		0.223	–	0.303	–	938.45

[a] Removing the output-cache proxy also disables cache-dependent layer selection, pooling, and candidate comparison.

Table 16: Core component ablation on Weather, averaged over 20 matched backbone–prediction-length settings. Δ is the relative change against the full configuration. Bold indicates the best score and underlining the second best using unrounded averages.

Category	Module	MSE		MAE		Time (s)
		value	$\Delta\%$	value	$\Delta\%$	
Gradient proxy	w/o output-cache proxy[a]	0.301	+2.8	0.327	+1.8	1138.48
Layer selection	w/o layer selection	**0.291**	−0.6	**0.321**	−0.4	1887.39
Variable pooling	w/o conflict pooling	0.298	+1.6	0.326	+1.3	1212.13
	w/o anchor pooling	0.307	+4.8	0.332	+3.1	1094.43
	w/o variable pooling	0.308	+4.9	0.332	+3.1	993.80
Direction surgery	w/o magnitude decoupling	0.301	+2.7	0.326	+1.3	1103.28
	w/o safe candidate selection	<u>0.292</u>	−0.4	<u>0.321</u>	−0.2	1132.23
PV-Surgery (full)		0.293	–	0.322	–	1286.21

[a] Removing the output-cache proxy also disables cache-dependent layer selection, pooling, and candidate comparison.

Table 17: Core component ablation on Exchange, averaged over 20 matched backbone–prediction-length settings. Δ is the relative change against the full configuration. Bold indicates the best score and underlining the second best using unrounded averages.

Category	Module	MSE		MAE		Time (s)
		value	Δ%	value	Δ%	
Gradient proxy	w/o output-cache proxy[a]	0.528	+22.6	0.486	+7.8	106.10
Layer selection	w/o layer selection	0.461	+7.0	0.461	+2.2	172.68
Variable pooling	w/o conflict pooling	0.516	+19.8	0.484	+7.2	97.14
	w/o anchor pooling	0.436	+1.2	0.452	+0.1	101.68
	w/o variable pooling	<u>0.434</u>	+0.7	**0.450**	−0.2	101.73
Direction surgery	w/o magnitude decoupling	0.448	+3.9	0.461	+2.2	113.14
	w/o safe candidate selection	0.440	+2.1	0.457	+1.3	106.53
PV-Surgery (full)		**0.431**	–	<u>0.451</u>	–	115.19

[a] Removing the output-cache proxy also disables cache-dependent layer selection, pooling, and candidate comparison.

Table 18: Core component ablation on ILI, averaged over 20 matched backbone–prediction-length settings. Δ is the relative change against the full configuration. Bold indicates the best score and underlining the second best using unrounded averages.

Category	Module	MSE		MAE		Time (s)
		value	Δ%	value	Δ%	
Gradient proxy	w/o output-cache proxy[a]	<u>3.05</u>	−0.4	**1.18**	−0.7	12.80
Layer selection	w/o layer selection	**3.04**	−0.8	<u>1.19</u>	−0.5	19.48
Variable pooling	w/o conflict pooling	3.08	+0.4	1.20	+0.4	15.07
	w/o anchor pooling	3.09	+0.7	1.20	+0.3	15.03
	w/o variable pooling	3.09	+0.7	1.20	+0.3	14.43
Direction surgery	w/o magnitude decoupling	3.13	+2.0	1.22	+2.0	13.79
	w/o safe candidate selection	3.09	+0.7	1.20	+0.6	14.12
PV-Surgery (full)		3.07	–	1.19	–	15.21

[a] Removing the output-cache proxy also disables cache-dependent layer selection, pooling, and candidate comparison.

I MULTI-TASK GRADIENT OPERATORS

Multi-task optimization offers several ways to combine gradients that disagree across objectives. We apply five operators discussed in Section 2 to the variable-wise proxies constructed by PV-Surgery. These methods normally act on a small set of task gradients exposed directly by a multi-task model, while a forecaster combines D variable objectives into one scalar loss. This comparison asks whether native multi-task operators can use the reconstructed variable-wise rows without PV-specific pooling or rescaling. Its purpose is to test transferability rather than establish a definitive ranking.

Native-operator protocol. Each native operator receives the same unpooled variable-wise gradient proxies on the selected subspace. We hold output-cache reconstruction, effective-coverage layer selection, selected-layer concatenation, and full-sum assembly fixed. Pooling, candidate selection, magnitude restoration, and PV-specific wrappers or rescaling are disabled, so each operator retains its native aggregation rule. Table 19 reports the 28 matched MICN settings from seven datasets and four prediction lengths using seed 42. Standard MSE training and the unchanged full PV-Surgery configuration provide the two reference rows.

The operators. **MGDA** treats the objectives as a multi-objective problem and takes the minimum-norm point of the convex hull of their gradients, $\min_{\alpha\in\Delta} \| \sum_i \alpha_i g_i \|_2^2$, which descends on every objective at once unless that minimum is zero, in which case the current point is already Pareto stationary (Sener & Koltun, 2018). **PCGrad** works pair by pair. For each objective, it compares its gradient with the others in random order and, whenever the cosine is negative, projects the gradient onto the normal plane of the conflicting one, leaving non-conflicting pairs untouched (Yu et al., 2020). **GradVac** generalizes PCGrad's rule from "repair negative cosines" to "reach a target cosine", tracking an exponential moving average of each pair's observed similarity and using it as the target, so PCGrad is the special case of a zero target applied only under conflict (Wang et al., 2021). **CAGrad** maximizes the worst-case objective improvement inside a ball of radius $c\|\bar{g}\|$ around the average gradient $\bar{g} = D^{-1}\sum_i g_i$, which keeps the update anchored to the mean-loss direction, recovers plain descent at $c = 0$, and approaches MGDA as $c \to \infty$ (Liu et al., 2021). We use $c = 0.5$. **Aligned-MTL** reads the condition number of the gradient matrix as a joint measure of conflict and dominance, eigendecomposes the objective-by-objective Gram matrix, and rescales its singular values to the smallest one, leaving principal components orthogonal and equal in magnitude (Senushkin et al., 2023).

Native-operator settings. PCGrad and GradVac visit peers in random order, and GradVac uses an exponential-moving-average coefficient of 0.01. MGDA runs at most 250 minimum-norm iterations with tolerance 10^{-5}. CAGrad uses $c = 0.5$, rescale mode 1, and SLSQP with at most 250 iterations and tolerance 10^{-10}. Aligned-MTL uses the smallest positive singular value and has no tuned hyperparameter.

The operators differ in both direction and aggregation scale. PCGrad and GradVac sum their corrected rows, MGDA keeps its minimum-norm convex-combination scale, CAGrad uses its mean-anchored native rescaling, and Aligned-MTL uses its spectral scale. We preserve these native rules, so this is a method-level comparison rather than a direction-only ablation. Our common-direction operator restores the norm of each surgery input after changing its direction. The inputs are variable rows in the unpooled branch and pool rows in the pooled branch, so this restoration does not preserve every individual variable-row norm after pooling.

Table 19: Native multi-task gradient operators on MICN over seven benchmarks and four prediction lengths. Each native row receives the same unpooled variable-wise proxies on the selected subspace and retains its own aggregation rule. PV-Surgery uses the unchanged full configuration. The MSE row denotes standard MSE training. Best results are bolded and second-best results are underlined using unrounded scores.

Dataset		ETTh1				ETTh2				ETTm1				ETTm2				Weather				Exchange				ILI			
Forecast length		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	24	36	48	60
MSE	MSE	0.552	0.594	**0.688**	0.753	0.238	0.315	0.452	0.672	0.445	0.495	0.567	0.627	0.166	0.202	0.266	0.389	0.398	0.403	**0.380**	0.496	0.123	0.219	0.498	2.931	**3.169**	2.996	3.269	3.273
	MAE	0.530	0.553	**0.623**	0.671	0.333	0.388	0.472	0.583	0.472	0.511	0.570	0.596	0.276	0.303	0.355	0.434	0.442	0.447	**0.422**	0.501	0.262	0.352	0.525	1.400	**1.237**	1.228	1.284	1.267
MGDA	MSE	0.497	0.641	0.718	0.790	0.236	0.295	**0.349**	**0.436**	0.447	0.473	**0.495**	**0.551**	0.151	**0.189**	0.237	0.320	0.355	**0.375**	0.385	0.555	0.122	0.226	0.401	**0.857**	3.454	2.991	**3.224**	3.394
	MAE	0.496	0.599	0.641	0.697	0.324	0.370	0.413	**0.475**	0.472	0.487	0.494	**0.538**	0.253	**0.280**	0.315	0.383	0.398	**0.424**	0.426	0.539	0.261	0.361	0.491	**0.743**	1.300	**1.215**	**1.271**	1.286
PCGrad	MSE	0.497	0.559	0.770	0.773	0.230	0.291	0.365	0.603	0.427	0.475	0.517	0.599	0.147	0.193	0.237	**0.295**	0.312	0.446	0.430	0.523	0.108	0.217	0.386	1.143	3.483	2.983	3.234	3.324
	MAE	0.502	0.542	0.671	0.686	0.321	**0.366**	0.424	0.562	0.454	0.489	0.522	0.582	0.251	0.291	0.324	**0.361**	0.378	0.471	0.459	0.525	0.242	0.352	0.475	0.822	1.306	1.221	1.274	1.271
GradVac	MSE	0.495	0.556	0.767	**0.728**	0.233	0.296	0.369	0.600	0.434	0.468	0.513	0.587	0.148	0.195	0.233	0.296	**0.304**	0.422	0.424	0.512	0.108	0.215	0.385	1.142	3.477	**2.980**	3.232	3.321
	MAE	0.500	0.540	0.670	**0.656**	0.325	0.371	0.427	0.561	0.462	0.485	0.519	0.573	0.253	0.290	0.319	**0.361**	**0.370**	0.449	0.458	0.518	0.241	0.350	0.479	0.814	1.305	1.220	1.273	1.272
CAGrad	MSE	0.510	**0.555**	0.770	0.769	0.234	0.318	0.386	0.647	0.419	0.484	0.515	0.604	0.151	0.209	0.243	0.303	0.341	0.447	0.437	0.512	0.115	0.220	0.427	1.399	3.281	2.985	3.233	**3.185**
	MAE	0.506	**0.539**	0.671	0.682	0.325	0.392	0.451	0.587	0.448	0.498	0.521	0.588	0.256	0.305	0.329	0.372	0.402	0.470	0.464	0.517	0.251	0.354	0.491	0.870	1.261	1.219	1.274	**1.240**
Aligned-MTL	MSE	**0.482**	0.587	0.721	0.810	0.232	0.295	0.360	0.601	**0.417**	**0.447**	0.496	0.578	0.148	0.189	**0.228**	0.393	0.401	0.378	0.464	0.513	**0.105**	0.245	0.436	2.729	3.430	2.984	3.251	3.429
	MAE	**0.487**	0.566	0.645	0.708	0.321	0.374	0.418	0.563	**0.448**	**0.458**	**0.493**	0.558	0.250	0.283	**0.310**	0.445	0.428	0.426	0.483	0.518	**0.236**	0.378	0.501	1.367	1.293	1.216	1.278	1.299
PV-Surgery (Ours)	MSE	0.492	0.561	0.700	0.798	**0.230**	**0.291**	0.352	0.513	0.430	0.470	0.508	0.576	**0.146**	0.190	0.231	0.331	0.318	0.435	0.436	**0.479**	0.109	**0.212**	**0.378**	1.106	3.190	2.985	3.228	3.364
	MAE	0.497	0.543	0.630	0.700	**0.320**	0.369	**0.412**	0.520	0.460	0.487	0.513	0.562	**0.248**	0.285	0.316	0.389	0.383	0.460	0.468	**0.493**	0.242	**0.348**	**0.470**	0.793	1.243	1.220	1.272	1.283

All five native operators reduce MSE on average relative to standard training over the 28 settings. The mean per-setting reductions are 7.59% for MGDA, 6.87% for PCGrad, 7.55% for GradVac, 4.82% for CAGrad, and 3.51% for Aligned-MTL. The unchanged PV-Surgery configuration gives the largest average reduction at 8.50%. The native operators outperform standard training in 19, 21, 22, 19, and 19 settings, respectively, but each also outperforms PV-Surgery in between 5 and 11 settings. The comparison therefore does not establish a uniform ranking. Instead, the average gains of all five operators show that the reconstructed variable-wise proxies provide a meaningful training signal that methods designed for explicit task gradients can use, while the per-setting differences show that the way this signal is aggregated still affects the outcome.

J Mechanism Diagnostics

Test error alone does not show whether the internal steps of PV-Surgery behave as intended. We therefore examine the variable-wise gradient proxies reconstructed from cached layer signals and the optimizer update assembled from these proxies. The first analysis measures proxy fidelity by comparing the reconstructed proxy rows with exact gradients obtained by separate per-variable backward passes. The second measures pairwise conflict and descent violations before and after surgery. A descent violation occurs when the update passed to the optimizer has a negative inner product with a reconstructed variable gradient, indicating that a small step along the negative update would increase that variable's loss to first order. A separate test-set analysis then asks whether PV-Surgery narrows the oracle gaps observed under standard shared training.

Measurement protocol. We run two diagnostic versions of the 140 settings in Table 1. Both use the full PV-Surgery configuration and the same training protocol as the reported runs. They enable different measurements at fixed intervals, but the diagnostic results are never used to form the optimizer update. For proxy fidelity, which measures directional agreement between a proxy and its exact gradient, we compute exact per-variable gradients every twenty optimizer steps and compare them with the reconstructed proxy rows using cosine similarity. These additional backward passes provide exact reference gradients only for measurement, while training continues to use the proxy-based update. We report fidelity at the output boundary, over all hooked parameters $\Omega_{\mathcal{H}}$, and on the selected subspace Ω_S. A separate set of runs records conflict mass, the average negative-cosine magnitude across reconstructed variable pairs, and descent violations every fifty steps.

At $12{,}256$ of the $45{,}897$ logged steps, the selected-slice proxy rows have zero norm and the conditional gate skips the intervention. Cosine similarity is undefined for these zero vectors. Our primary selected-subspace statistic is therefore conditional on a nonzero selected-slice proxy. Consequently, five runs have no defined selected-subspace cosine after conditioning, leaving 135 runs for the primary statistic. If the zero entries are instead assigned a cosine of zero, the selected-subspace average is 0.739 over all 140 runs. The output-boundary and all-parameter statistics use all runs without this conditioning.

Proxy fidelity. The average cosine with the exact gradient is 0.990 at the output boundary, 0.936 on the selected subspace when its proxy is nonzero, and 0.698 over all parameters. Figure 15 shows the same conditional comparison for each benchmark. Output-boundary fidelity ranges from 0.981 to 0.996. The gap is larger inside the backbone, where variable-mixing operations make the cache an approximation. Weather has the lowest all-parameter fidelity at 0.508, which rises to 0.862 after layer selection. ETTh1 rises from 0.806 to 0.977. The conditional selected-subspace value exceeds the all-parameter value on every benchmark. Layer selection therefore concentrates active interventions on parameters for which the reconstructed gradients are more faithful, although the selected rows are not exact in every model.

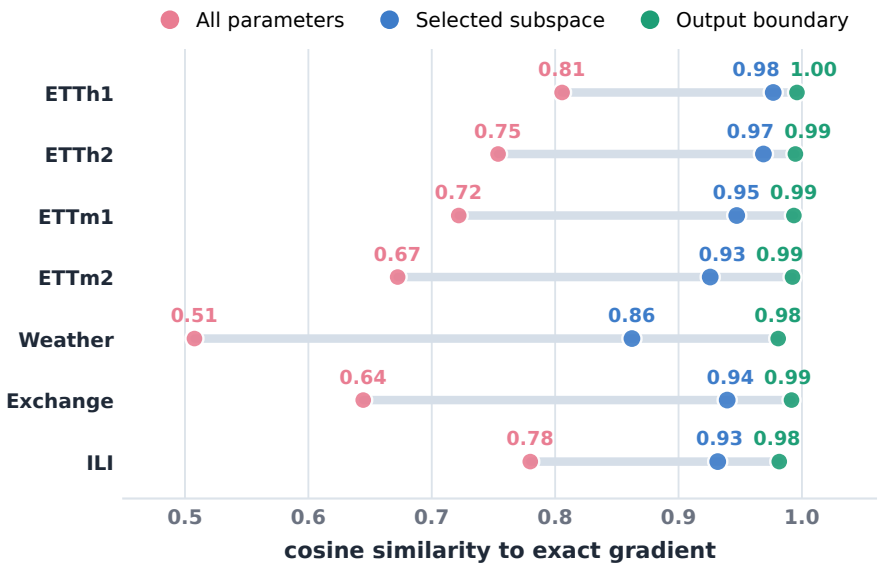
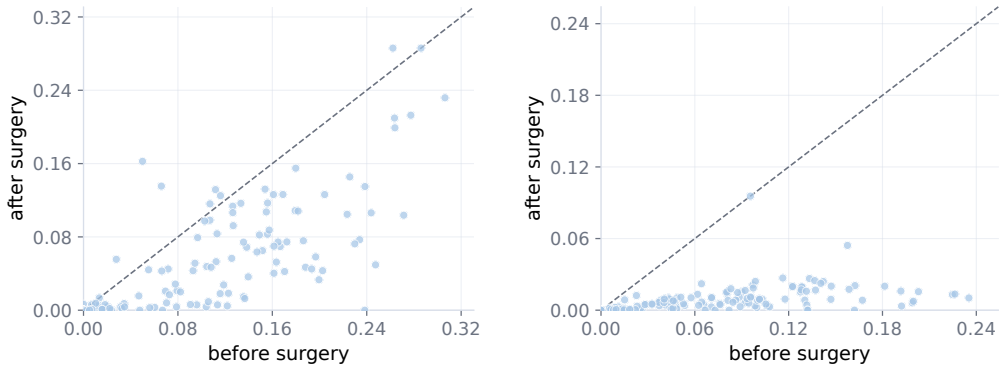


Figure 15: Cosine similarity between reconstructed proxy rows and exact per-variable gradients for each benchmark. Each marker first averages logged steps within a run and then averages the available runs for that benchmark. Each benchmark contributes 20 runs from five backbones and four prediction lengths. Output-boundary and all-parameter markers use all 20 runs. Selected-subspace markers use 17 Exchange runs, 18 ILI runs, and all 20 runs elsewhere after conditioning on nonzero proxy entries. The smaller counts occur when every logged selected-slice proxy in a run has zero norm, leaving no defined selected-subspace cosine.

Effect on the reconstructed gradients. Figure 16 compares the sum-loss update with the final update in each of the 140 mechanism runs. Conflict mass is the mean of $\max(-\cos_{ij}, 0)$ over all variable pairs, so a cosine of -0.4 contributes 0.4 while a nonnegative cosine contributes zero. A larger conflict mass therefore indicates stronger overall directional opposition among the reconstructed variable gradients. The descent violation ratio is the fraction of reconstructed variable gradients that have a negative inner product with the update. A ratio of 0.2, for example, means that a small step along the negative update would increase the losses of two out of ten reconstructed variable objectives to first order. These are optimization diagnostics defined on the reconstructed rows. They should not be read as direct measurements of test-set harm.

Mean conflict mass falls from 0.075 to 0.009. It decreases in 129 runs, remains unchanged in 11, and increases in none. This one-sided shift directly verifies the intended operation of PV-Surgery. The mean descent violation ratio falls from 0.097 to 0.049. It decreases in 103 runs, remains unchanged in 29, and increases in 8. Descent violations are a stricter downstream test because they also depend on how the corrected rows combine into the final update. Their mean is nearly halved and they decrease in most runs, showing that conflict correction usually carries through to an update that is better aligned with the reconstructed variable objectives.



(a) Descent-violation ratio.

(b) Conflict mass.

Figure 16: Descent violations and conflict mass before and after surgery over the 140 matched settings. (a) shows the descent-violation ratio, and (b) shows conflict mass. Each point is one run, with the horizontal and vertical axes giving the values before and after surgery. Points below the diagonal indicate a reduction.

Test-set oracle gap. Figure 17(a) returns to the oracle gap from Section 3. The comparison contains 57 iTransformer variables from the six benchmarks with matched baseline, PV-Surgery, and full-input single-target oracle runs. Shared training harms 29 of these variables. PV-Surgery reduces the gap for 23 and closes it for 12. Among the remaining 28 variables, whose baseline gap is not positive, PV-Surgery increases test MSE for 10 and creates a positive oracle gap for 3. Across all 57 variables, the mean gap decreases by 0.0040 MSE. The aggregate improvement therefore includes both repaired gaps and a smaller number of regressions.

Frozen variable-pair subsets. Figure 17(b) tests a stricter hypothesis using separately trained fixed variable-pair subsets. PV-Surgery improves test MSE on only 5 of 22 conflicting pairs, compared with 16 of 28 aligned pairs. Its mean MSE change relative to the baseline is also unfavorable for both groups, at $+0.0085$ for conflicting pairs and $+0.0021$ for aligned pairs. Persistent pairwise conflict is therefore not a reliable marker of the subsets on which surgery will help. This negative result agrees with Section 3, where conflict identifies disagreement but not harm under shared training.

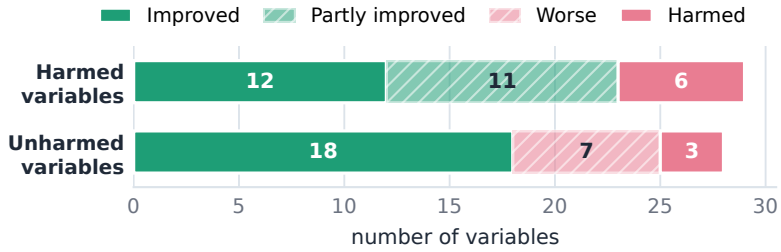
K Qualitative Case Studies

Aggregate MSE does not show how two forecasts differ within a window. We therefore inspect four settings with prediction length 96, namely MICN on ETTm2, iTransformer on ETTm2, MICN on ETTh1, and DLinear on ETTh2. Together they cover three backbones and two ETT benchmarks. Each figure shows the six load variables HUFL, HULL, MUFL, MULL, LUFL, and LULL together with oil temperature OT.

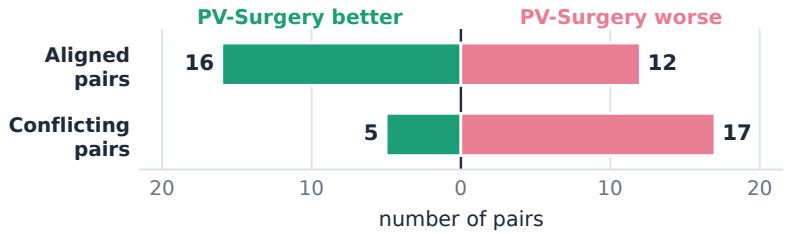
Protocol. For each setting, the baseline and PV-Surgery runs use the same data split and test windows. Their saved ground-truth arrays are identical. Since the variables span different vertical ranges, we summarize the error in each panel using a range-normalized mean absolute error. For window n and variable d, let $\hat{y}^{\mathrm{MSE}}$ and $\hat{y}^{\mathrm{PV}}$ denote the forecasts from standard MSE training and PV-Surgery. For either forecast $\hat{y}$, the panel distance is

$$\delta_{n,d}(\hat{y}) = \frac{\frac{1}{H}\sum_{h=1}^{H}\left|\hat{y}_{n,h,d} - y_{n,h,d}\right|}{\max\{y, \hat{y}^{\mathrm{MSE}}, \hat{y}^{\mathrm{PV}}\}_{n,\cdot,d} - \min\{y, \hat{y}^{\mathrm{MSE}}, \hat{y}^{\mathrm{PV}}\}_{n,\cdot,d}}. \tag{18}$$

The denominator is the range spanned by the ground truth and both forecasts within the same panel. A value of 0.10 means that the average absolute error is one tenth of this range. The plotted vertical axes retain the standardized target and forecast values. We use $\delta_{n,d}$ only as a scale-normalized scalar summary to compare the two forecasts within the same window and variable. The smaller of $\delta_{n,d}(\hat{y}^{\mathrm{MSE}})$ and $\delta_{n,d}(\hat{y}^{\mathrm{PV}})$ identifies the forecast with the lower average vertical error relative to



(a) Effect on the Section 3 oracle gap.



(b) Frozen variable-pair subsets.

Figure 17: Test-set evidence for the effect of PV-Surgery. (a) Outcomes for the 57 variables, grouped by their oracle gap under standard shared training. For harmed variables, Improved closes the gap, Partly improved reduces it without closing it, and Harmed leaves it unreduced. For unharmed variables, Improved does not increase test MSE, Worse increases it without creating a positive gap, and Harmed creates one. (b) Numbers of fixed variable-pair subsets on which PV-Surgery gives lower or higher test MSE than standard training.

their common displayed range. We also report the cosine between the predicted and true trajectories after subtracting their respective horizon means. This direction cosine measures temporal shape independently of the mean level. A negative value indicates that the predicted trajectory moves in the opposite direction from the target.

Window selection. We manually select one representative example from each setting in which the effect of PV-Surgery is clearly visible. Together, the examples illustrate level correction, reduced drift, and cases in which panel distance improves without a higher direction cosine. They are descriptive and do not enter any aggregate estimate.

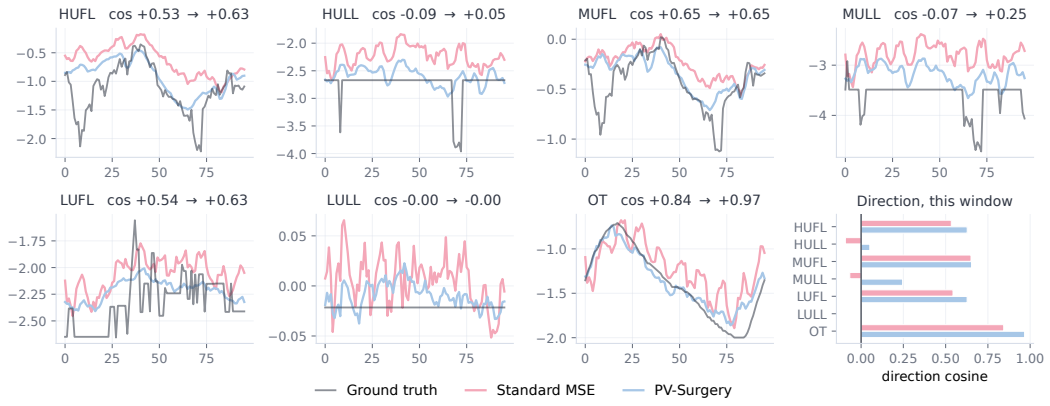


Figure 18: MICN on ETTm2. Panel titles report the direction cosine for standard MSE training and PV-Surgery. The final panel collects the same values.

MICN on ETTm2. Across the full test set, PV-Surgery lowers MSE by 12.52% and raises the mean direction cosine from 0.422 to 0.510. Window-level MSE improves in 19,377 of 23,409 windows. The number of variables with negative direction cosine decreases in 7,726 windows and increases in 2,534. In Figure 18, the baseline forecast is generally shifted above the ground truth, while PV-Surgery reduces this offset. The panel distance falls from 0.25 to 0.13 for HUFL, from 0.24 to 0.10 for HULL, and from 0.35 to 0.18 for MULL. HULL and MULL also change from negative to positive direction cosine. OT has the smallest corrected distance at 0.08, and its direction cosine rises to 0.97.

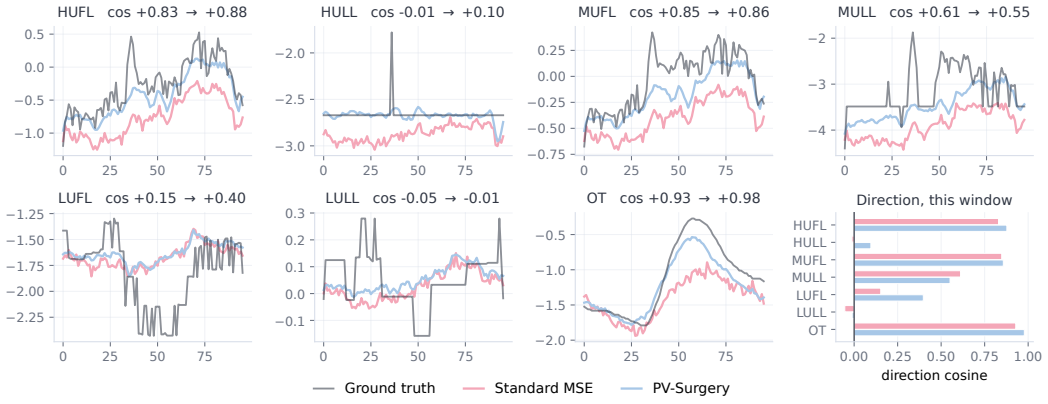


Figure 19: iTransformer on ETTm2. Panel titles report the direction cosine for standard MSE training and PV-Surgery. The final panel collects the same values.

iTransformer on ETTm2. On the same dataset, iTransformer obtains a smaller MSE reduction of 4.61%. Window-level MSE improves in 15,752 of 23,409 windows. Negative direction cosines become less frequent in 5,020 windows and more frequent in 4,671. Figure 19 again shows a substantial level correction. The panel distance falls from 0.28 to 0.11 for HUFL, from 0.32 to 0.14 for MUFL, and from 0.17 to 0.03 for HULL. The improvement is small for LUFL and LULL. MULL provides a useful counterexample. Its distance decreases from 0.31 to 0.16, while its direction cosine decreases from 0.61 to 0.55. A forecast can therefore move closer in level without matching the temporal shape more closely.

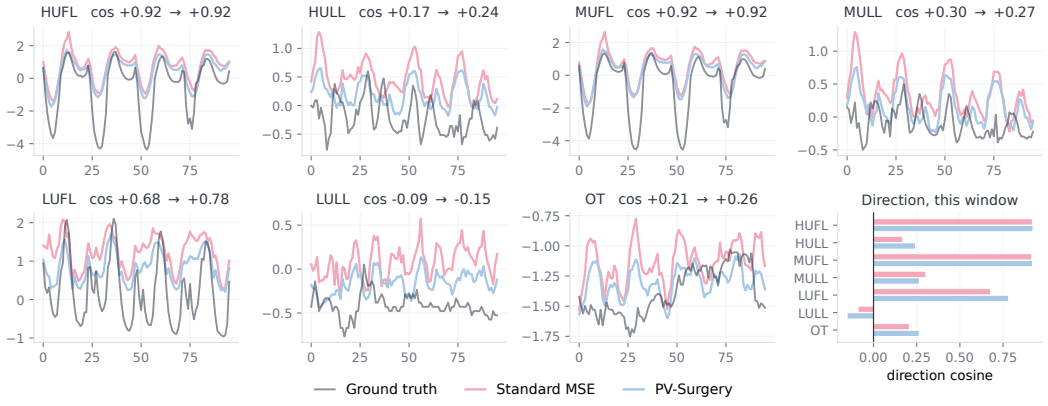


Figure 20: MICN on ETTh1. Panel titles report the direction cosine for standard MSE training and PV-Surgery. The final panel collects the same values.

MICN on ETTh1. PV-Surgery lowers MSE by 10.85% for MICN on ETTh1. It improves window-level MSE in 4,188 of 5,709 windows. The number of negative direction cosines decreases in 1,156 windows and increases in 221. In Figure 20, the baseline tends to overshoot the target and

PV-Surgery moves the forecasts downward. The largest distance reductions occur for LULL, from 0.37 to 0.21, and LUFL, from 0.35 to 0.24. HUFL and MUFL retain direction cosines near 0.92. Direction does not improve for every variable. MULL falls from 0.30 to 0.27, and LULL becomes more negative, even though both panel distances decrease.

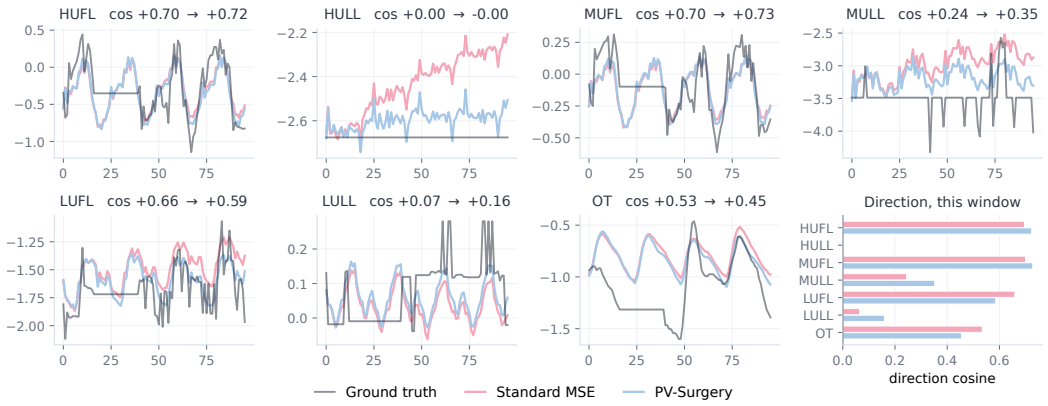


Figure 21: DLinear on ETTh2. Panel titles report the direction cosine for standard MSE training and PV-Surgery. The final panel collects the same values.

DLinear on ETTh2. PV-Surgery lowers MSE by 4.55% in the DLinear setting. Window-level MSE decreases in 4,174 of 5,709 windows. Negative direction cosines become less frequent in 1,073 windows and more frequent in 264. The clearest change in Figure 21 occurs for HULL. The baseline drifts away from an almost flat target, while the PV-Surgery forecast remains closer throughout the horizon. Its panel distance falls from 0.45 to 0.15. MULL shows a smaller reduction from 0.28 to 0.18. LUFL and OT again show that the two diagnostics need not agree. Their panel distances decrease, but their direction cosines fall from 0.66 to 0.59 and from 0.53 to 0.45.

Across these four settings, the mean direction cosine changes by 0.088, 0.034, 0.041, and 0.028. The selected windows show larger changes in level and amplitude than in temporal shape. They also show why the mechanism results in Appendix J should not be interpreted as a guarantee for every variable. PV-Surgery can reduce aggregate MSE and pairwise conflict while the direction cosine of an individual variable remains unchanged or becomes worse. These examples add a window-level view of the aggregate results and show that lower error does not always come with a higher direction cosine.

L Computational Cost and Training Time

PV-Surgery leaves the forecasting architecture unchanged but requires additional computation at each training step. We describe this additional computation and measure its wall-clock cost over the main experimental grid.

Analytical cost. Like standard MSE training, PV-Surgery uses one backward pass for each optimizer step. Its additional cost comes from reconstructing and processing the variable-wise gradient rows. Let $N_{\mathcal{H}}$ denote the number of parameters in the hooked layers and $N_{\mathcal{S}}$ the number in the selected layers. Dense zero-extended proxy storage and later row-wise processing scale as $\mathcal{O}(DN_{\mathcal{H}})$. Computing the observed weight outer products costs $\mathcal{O}(\sum_{\ell \in \mathcal{H}_{\text{run}}} BDM_\ell F_{\text{in},\ell} F_{\text{out},\ell})$, where $F_{\text{in},\ell}$ and $F_{\text{out},\ell}$ are the layer widths and the bias term is lower order. Pairwise cosine similarities on the selected subspace cost $\mathcal{O}(D^2 N_{\mathcal{S}})$, and evaluating the unpooled candidate plus a pooled candidate with P pools costs $\mathcal{O}((D+P)N_{\mathcal{S}})$.

Layer selection cannot reduce the cache-construction term because the cache is needed to score the layers. It does reduce the pairwise and candidate computations because $N_{\mathcal{S}}$ is no larger than $N_{\mathcal{H}}$. Candidate selection adds another direction evaluation when pooling is active. These operations affect training only. The forward pass used at inference is identical to that of the original backbone.

Protocol. We pair each PV-Surgery run with the standard MSE run from the same backbone, benchmark, and prediction length. All 140 pairs use seed 42, the same batch size, and the same GPU type. Early stopping gives different training lengths, so total time alone mixes computational overhead with the number of completed epochs. We divide each run's logged training time by its completed epoch count and form the PV-Surgery to MSE ratio within each pair. The baseline completes 22.1 epochs on average and PV-Surgery completes 23.3. These timings cover complete training epochs, including data loading, model computation, optimizer steps, and logging. They therefore measure practical training time rather than the isolated cost of the PV-Surgery operator.

Table 20: Training cost of PV-Surgery against standard MSE training over the 140 matched settings of Table 1. Epochs and seconds per epoch are means over the 28 settings of each backbone. The ratio columns are computed per setting and then aggregated, so they are not the quotient of the two preceding columns.

Backbone	Epochs		Seconds / epoch		Per-epoch ratio		Slower
	MSE	+PV	MSE	+PV	median	mean	
DLinear	27.2	23.4	6.2	5.5	0.88	1.15	6/28
MICN	17.6	17.8	46.1	62.1	1.35	1.37	28/28
SCINet	28.5	27.7	18.7	28.5	1.52	1.51	28/28
iTransformer	15.6	25.4	6.4	15.3	2.09	2.29	28/28
TimeXer	21.8	22.3	6.6	14.8	2.22	2.24	28/28
All	22.1	23.3	16.8	25.3	**1.54**	**1.71**	**118/140**

Measured overhead. Table 20 reports the paired results. Across all settings, the per-epoch ratio has a median of 1.54 and a mean of 1.71. PV-Surgery is slower in 118 of the 140 pairs. Summing the logged time over the grid gives 12.84 hours for standard MSE training and 20.56 hours for PV-Surgery, a total-time ratio of 1.60. On the six benchmarks with prediction lengths 96, 192, 336, and 720, the median per-epoch ratio remains between 1.51 and 1.54. We do not observe a systematic increase with forecast length in this grid.

The overhead varies more across backbones. The median ratio is 2.09 for iTransformer and 2.22 for TimeXer, both of which expose several cache-compatible linear layers. MICN has a lower median ratio of 1.35, although its baseline epoch is much slower in absolute time. DLinear has a median ratio below one at 0.88, but its mean ratio is 1.15 and six of its twenty-eight settings are slower with PV-Surgery. The DLinear result should therefore be read as low overhead with substantial timing variation, not as evidence that surgery generally accelerates training.

Implementation considerations. The measured ratios depend on the hardware and current implementation and should not be treated as architecture-independent constants. Caching a separate variable axis and, on some execution paths, materializing dense $D \times N_{\mathcal{H}}$ proxy rows increase memory traffic, while evaluating both unpooled and pooled candidates adds computation. Compact slice-wise assembly and reuse of intermediate quantities could reduce this overhead. Reducing the number of cached layers offers another cost-coverage trade-off. PV-Surgery changes only training-time gradient construction, so inference remains unchanged.