Multivariate Time Series Forecasting needs Cross Variable Loss

Kuiye Ding[1], **Yifan Hu**[2], **Hao Xue**[3,*], **Hanchen Wang**[1]
[1] University of Technology Sydney, [2] Tsinghua University,
[3] The Hong Kong University of Science and Technology (Guangzhou)
[*] Corresponding author.
kuiye.ding@student.uts.edu.au, huyf25@mails.tsinghua.edu.cn
haoxue@hkust-gz.edu.cn, hanchen.wang@uts.edu.au

Abstract

Multivariate time series forecasting presents unique challenges because future variables often co-evolve under shared system dynamics. While existing studies mainly focus on cross-variable dependencies in historical observations, dependencies among future values are much less explored. Specifically, modern forecasting models largely follow the Direct Forecasting (DF) paradigm, generating multi-step forecasts with point-wise objectives that do not explicitly constrain cross-variable structure. In this work, we show that the DF objective is mismatched in the presence of cross-variable and lagged dependencies, revealing an objective gap. To address this issue, we propose **C**ross-**V**ariable **Loss** (CvLoss), a plug-in structural regularizer that constrains forecast residuals on a cross-variable graph. CvLoss penalizes inconsistent edge-wise residual differences over forecast patches, encouraging consistency across both synchronous and asynchronous interactions. Our experiments show that CvLoss consistently improves competitive forecasting models, outperforms representative learning objectives, and is compatible with a variety of forecasting backbones. Code is available at https://github.com/Day333/CvLoss.

1 Introduction

Multivariate time series forecasting (MTSF) is a fundamental task across diverse domains, where the core challenge lies in accurately capturing complex synergistic relationships among variables. To model these temporal dynamics, recent pioneering works [1–6] have predominantly focused their efforts on extracting representations from historical input features, a paradigm known as input-side channel dependence modeling. In stark contrast, the intrinsic correlations among future predictions at the output-side have been largely overlooked. Current mainstream Direct Forecasting (DF) paradigms [7] typically decouple the multivariate outputs forcibly, projecting the unified historical embeddings independently into future time steps, thereby degrading MTSF into a collection of isolated scalar prediction problems. This asymmetry heavily prioritizes the input while neglecting the output. Consequently, it not only severs the physical co-evolution among future variables but also induces severe structural errors.

On one hand, blindly decoupling prediction targets directly violates the intrinsic physical and operational laws of multivariate systems. In the real world, time series are rarely generated independently; rather, they are governed by shared system dynamics and exhibit strong co-evolutionary characteristics. For instance, in quantitative financial markets [8], the returns of correlated assets often display highly synchronous surges, plunges, or hedging effects when subjected to macroeconomic shocks [9]. Similarly, in traffic flow forecasting, the traffic volumes recorded by hundreds of sensors within

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

an urban network are driven by shared meteorological conditions (e.g., sudden rainstorms causing widespread congestion), and synchronized daily commuting patterns. This results in a profound co-evolution of future traffic dynamics, as illustrated in Figure 5. If a model isolates these variables at the output stage, the predicted curves for individual variables might appear numerically well-fitted. However, when combined into a joint multivariate forecast, they frequently produce contradictory results that violate true physical constraints and market rules.

Furthermore, overlooking these synergistic relationships induces critical structural errors at the optimization level. Modern MTSF models rely heavily on point-wise loss functions, such as Mean Squared Error (MSE) [10, 11], for optimization. From the perspective of probabilistic inference, this point-wise optimization is fundamentally predicated on a highly restrictive assumption: the prediction residuals across all variables and time steps follow completely independent and isotropic Gaussian distributions (i.e., possessing a diagonal covariance matrix). However, given the aforementioned co-evolutionary phenomena, the true spatio-temporal covariance matrix of the residuals inevitably contains numerous non-zero off-diagonal elements. This creates an unbridgeable theoretical divide, termed an Objective Gap, between existing standard loss functions and the true joint spatio-temporal distribution of the data. Consequently, even if two models achieve remarkably similar point-wise MSE, their cross-variable covariance structures might differ drastically, completely failing to capture the joint geometric topology of the original data.

To overcome these dual physical and mathematical deficiencies, the optimization objective must transcend the singular pursuit of absolute point-wise accuracy and explicitly constrain the relative structural consistency among variables. This implies that we must capture not only the concurrent interactions (synchronous effects) between variables at the same time step but also the lagged propagation of system perturbations as time evolves (asynchronous effects). To elegantly unify these two phenomena, we introduce a graph-structured perspective, as graphs are inherently suited for depicting complex, non-uniform pair-wise interactions among multivariate data. Specifically, we propose a novel graph-based structural regularization scheme termed Cross-Variable Loss (CvLoss). By constructing a cross-variable graph, CvLoss connects different variables at identical time steps to capture synchronous concurrent interactions, while also linking variables across different steps to model asynchronous lagged effects. Essentially, CvLoss imposes an ℓ_1-norm Graph Total Variation (Graph TV) regularization on the prediction residual field, compelling the model to strictly adhere to the underlying relative structural consistency while minimizing absolute errors. Furthermore, as a plug-and-play regularizer, CvLoss seamlessly integrates into various mainstream forecasting architectures. Our introduced edge-sampling mechanism significantly reduces computational complexity during training, and crucially, the regularizer imposes zero overhead during inference, ensuring testing speed remains completely unaffected while maintaining extremely high predictive accuracy.

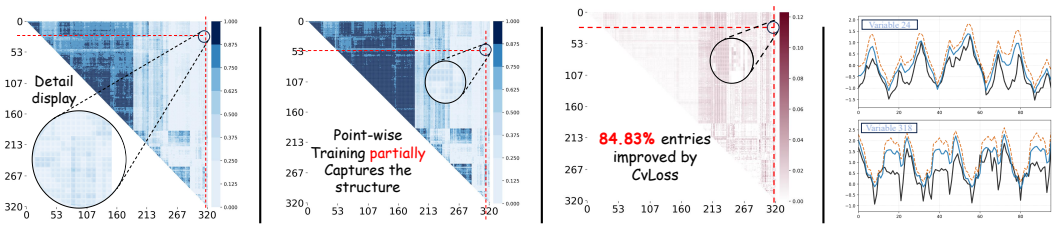
Our main contributions are summarized as follows:

- We reveal a critical limitation in current MTSF paradigms, demonstrating that standard point-wise learning objectives neglect inherent inter-channel error dependencies among future predictions.
- We propose a novel Cross-Variable Loss acting as a structural regularizer, explicitly constraining the inter-channel relationships of predicted values via a residual consistency formulation.
- Extensive experiments demonstrate our proposed objective consistently improves various state-of-the-art models, achieving superior predictive performance with affordable computational overhead.

2 Preliminaries and Related Work

2.1 Problem definition

In this paper, we focus on the MTSF problem. Throughout, italic capitals denote matrices (e.g., X, Y, Z, E), bold lowercase letters denote vectors (e.g., $\mathbf{e}$), calligraphic capitals denote sets (e.g., $\mathcal{E}$), and lowercase letters denote scalars and indices (e.g., t, d, α); the dimension constants H, T, D, P, L and N are written as capitals by convention. Given a multivariate time-series dataset with D variables, the input historical sequence is defined as $X \in \mathbb{R}^{H \times D}$, and the ground-truth label sequence is defined as $Y \in \mathbb{R}^{T \times D}$, where H is the lookback window and T is the forecast horizon. Modern time series forecasting models are primarily trained under a multitask learning manner, known as the direct forecasting (DF) paradigm. Therefore, the target is to learn a multi-output predictor $f_\theta : \mathbb{R}^{H \times D} \to \mathbb{R}^{T \times D}$, parameterised by the trainable weights θ of the forecasting backbone, that directly produces T-step forecasts for all D variables simultaneously, yielding the



(a) Ground-truth structure. (b) Baseline structure. (c) Improvement by CvLoss. (d) ECL Snapshot.

Figure 1: Motivating example on ECL with input length $H = 96$ and forecast horizon $T = 96$. (a,b) Cross-variable correlation structures of the ground-truth future series and of the series predicted by a plain iTransformer [1]. (c) Entries on which the same backbone trained with CvLoss attains a smaller structural error than the plain baseline, covering 84.83% of all cross-variable entries. (d) Forecast snapshot on a representative variable. The protocol used to compute (a)–(c) is defined in Appendix B.3.

prediction $\hat{Y} = f_\theta(X)$. Unlike current paradigms that decouple multivariate outputs into isolated scalar predictions, our formulation explicitly maintains the joint structure to capture cross-channel dependencies and shared temporal dynamics.

2.2 Learning objectives in time-series forecasting

Recent learning objectives extend point-wise MSE to capture temporal dynamics through shape alignment [12–14], likelihood maximization [15–17, 7], distribution balancing [18, 19], and intra-sequence modeling via decomposition [20] or local step alignment [21, 22]. Despite these advancements, existing approaches fundamentally treat multivariate forecasting as a collection of isolated scalar predictions. By optimizing exclusively for intra-channel temporal structures, they completely neglect explicit constraints on cross-variable dependencies. Consequently, these methods fail to capture the joint geometric structure and correlated disturbances among multivariate outputs.

2.3 Cross-variable Dependencies in time-series forecasting

Effectively modeling cross-variable dependencies is fundamental for MTSF [10, 23]. Existing paradigms typically employ Channel Dependence (CD) strategies to explicitly capture inter-channel correlations, as seen in representative architectures like iTransformer [1], TSMixer [24], Crossformer [25], and TimesNet [26]. However, CD models often overfit to spurious correlations and struggle with noise interference from irrelevant variables. To address this, recent advancements introduce Channel Partiality (CP) to selectively filter dependencies. For instance, DUET [3] proposes a Channel Clustering Module to filter noisy channels via frequency-domain soft clustering. Similarly, TimeFilter [4] constructs a patch-specific spatial-temporal graph with dynamic routing to retain essential correlations while discarding irrelevant noise. Other approaches like MTGNN [27] and MCformer [28] also explore sparse interactions to balance accuracy and efficiency. Despite these significant improvements, current models predominantly focus on extracting representations from historical observations [29–31]. The complex structural geometry and co-evolving cross-variable dependencies among future predictions remain fundamentally underexplored.

3 Proposed Method

3.1 Motivation

Modern time series forecasting models are primarily trained under the multitask learning manner, known as the direct forecasting (DF) paradigm [7]. Given an input history $X \in \mathbb{R}^{H\times D}$, a multi-output predictor $f_\theta : \mathbb{R}^{H\times D} \to \mathbb{R}^{T\times D}$ directly produces T-step forecasts for D variables: $\hat{Y} = f_\theta(X)$. The

standard training objective is the point-wise MSE:

$$\mathcal{L}_{\mathrm{df}} = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{D}\|Y_{t,:} - \hat{Y}_{t,:}\|_2^2 \tag{1}$$

This objective is simple and effective, but it only penalizes prediction errors at individual time-variable entries. Although modern forecasting architectures may implicitly capture cross-variable interactions through shared parameters, such dependencies are not explicitly reflected in the objective in Eq. (1), which does not constrain the *relative error structure* across variables. As a result, two models with similar point-wise MSE may still exhibit substantially different cross-variable correlated patterns.

Figure 1 illustrates the objective gap on ECL. With the same iTransformer backbone, CvLoss recovers the ground-truth correlation matrix more accurately on 84.83% of cross-variable entries and improves the forecast snapshot. Explicit coupling across related targets is also consistent with multi-task learning [32]. We therefore complement point-wise supervision with residual consistency across variables and examine its probabilistic motivation below.

Let $\mathbf{e} = \mathrm{vec}(\hat{Y} - Y) \in \mathbb{R}^{TD}$ denote the flattened prediction error vector over the entire forecast horizon and all variables. Suppose the residuals follow a zero-mean multivariate Gaussian distribution $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{ST})$, where $\mathbf{\Sigma}_{ST} \in \mathbb{R}^{TD\times TD}$ is the spatiotemporal covariance matrix that captures both cross-variable and cross-temporal dependencies.

Theorem 3.1 (Objective Gap with Spatiotemporal Dependencies) *Ignoring additive constants independent of the predictions, the scaled negative log-likelihood (NLL) under the true covariance $\mathbf{\Sigma}_{ST}$ is $\mathcal{L}_{\mathrm{nll}} = \frac{1}{2TD}\mathbf{e}^\top \mathbf{\Sigma}_{ST}^{-1}\mathbf{e}$. On the other hand, the standard DF objective in Eq.* (1) *can be written as $\mathcal{L}_{\mathrm{df}} = \frac{1}{TD}\mathbf{e}^\top\mathbf{e}$. Therefore, compared with the isotropic Gaussian NLL under variance σ^2, the discrepancy between the two objectives is*

$$\Delta = \mathcal{L}_{\mathrm{nll}} - \frac{1}{2\sigma^2}\mathcal{L}_{\mathrm{df}} = \frac{1}{2TD}\mathbf{e}^\top\left(\mathbf{\Sigma}_{ST}^{-1} - \frac{1}{\sigma^2}\mathbf{I}\right)\mathbf{e}, \tag{2}$$

where $\mathbf{I} \in \mathbb{R}^{TD\times TD}$ is the identity matrix. If the residual precision has non-zero off-diagonal entries, $\mathbf{\Sigma}_{ST}^{-1} \neq \sigma^{-2}\mathbf{I}$ and the two objectives are not identical as functions of the residual vector. The gap may vanish for particular residual vectors, including $\mathbf{e} = \mathbf{0}$.

Theorem 3.1 identifies the geometry omitted by a spherical Gaussian objective. Cross-variable and lagged residual dependencies motivate a relation-aware penalty without estimating the full precision matrix, which has more than 5×10^{10} entries on ECL at $T = 720$ and $D = 321$. The theorem motivates coupling residuals; Proposition 3.2 describes a structured Gaussian family, while Appendix C.4 evaluates the choice of norm.

Proposition 3.2 (MSE with a squared graph penalty is a structured Gaussian NLL) *Let $\mathcal{E}$ be a cross-variable edge set over the N patch nodes, let $\tilde{A} = A \otimes I_L$ be the lifted incidence matrix of Section 3.2, and let $\mathrm{Lap}(\mathcal{E}) = \tilde{A}^\top\tilde{A}$ be the corresponding graph Laplacian. Define the structured precision family*

$$\mathbf{\Sigma}^{-1}(\lambda) = \frac{1}{\sigma^2}\big(\mathbf{I} + \lambda\,\mathrm{Lap}(\mathcal{E})\big), \qquad \lambda \geq 0. \tag{3}$$

Then, writing $\mathbf{r} = \mathrm{vec}(E)$ for the flattened residual field, the following identity holds for every $\mathbf{r}$:

$$\mathbf{r}^\top\mathbf{\Sigma}^{-1}(\lambda)\,\mathbf{r} = \frac{1}{\sigma^2}\Big(\|\mathbf{r}\|_2^2 + \lambda\sum_{(i,j)\in\mathcal{E}}\|\mathbf{e}_i - \mathbf{e}_j\|_2^2\Big). \tag{4}$$

Proposition 3.2, proved in Appendix A.2, expresses the Gaussian NLL as point-wise MSE plus a squared graph penalty. The edge set specifies supported cross-variable interactions, and $\lambda = 0$ recovers the spherical case. Our default complete cross-variable graph includes every cross-variable pair; the implemented ℓ_1 penalty uses this graph with a robust, non-Gaussian residual model, as discussed below.

3.2 Predicting Multivariate Time Series with Cross Variable Loss Constraints

3.2.1 Definition

Patch nodes. Let $Y \in \mathbb{R}^{T\times D}$ be the target and $\hat{Y} = f_\theta(X) \in \mathbb{R}^{T\times D}$ be the prediction. We split the time axis into P non-overlapping patches of length L [33, 34], so that $T = P \times L$. Each node is defined as a pair $v = (p, d)$, where $p \in \{1, \dots, P\}$ indexes the patch and $d \in \{1, \dots, D\}$ indexes the variable. Let $\mathbf{z}_v \in \mathbb{R}^L$ and $\hat{\mathbf{z}}_v \in \mathbb{R}^L$ denote the ground-truth and predicted patch vectors at node v, respectively. Stacking all nodes gives

$$Z \in \mathbb{R}^{N\times L}, \qquad \hat{Z} \in \mathbb{R}^{N\times L}, \qquad N = P \times D. \tag{5}$$

We define the node-wise residuals as

$$\mathbf{e}_v = \hat{\mathbf{z}}_v - \mathbf{z}_v \in \mathbb{R}^L, \tag{6}$$

and stack them into the residual matrix $E \in \mathbb{R}^{N\times L}$, whose vectorisation $\mathrm{vec}(E) \in \mathbb{R}^{NL}$ is a permutation of the flattened error vector $\mathbf{e}$ of Theorem 3.1, since $NL = TD$. For $L = 1$, patch nodes recover point nodes.

Cross-variable edges. To capture the spatiotemporal dependencies across different series, we introduce an edge set $\mathcal{E}$ to define the cross-variable graph topology. Guided by Theorem 3.1, rather than explicitly estimating the intractable precision matrix, we utilize $\mathcal{E}$ as a practical surrogate structure to regularize residual discrepancies. This formulation is highly flexible and can easily accommodate different structural priors. Specifically, we design three variants for constructing $\mathcal{E}$: (1) *Synchronous cross-variable edges*, which strictly capture concurrent interactions at the same time step (i.e., connecting identical patch indices); (2) *Asynchronous cross-variable edges*, which exclusively model lagged dependencies across different time steps; and (3) *Fully connected cross-variable edges*, which include both synchronous and lagged interactions. We adopt the fully connected graph as our default setting to maximize the coverage of potential interactions without relying on domain-specific prior knowledge. We evaluate these topological designs in the ablation studies in Section 4.5.

Difference operator. Fix an arbitrary orientation for each undirected edge and let $A \in \mathbb{R}^{|\mathcal{E}|\times N}$ be the incidence matrix: for an edge $k = (i, j)$, the k-th row contains $+1$ at position i, -1 at position j, and 0 elsewhere. Since each node is an L-dimensional patch, we lift the incidence operator as

$$\tilde{A} := A \otimes I_L. \tag{7}$$

Let $\mathrm{vec}(E) \in \mathbb{R}^{NL}$ denote the vectorization of E. Then $\tilde{A}\,\mathrm{vec}(E) \in \mathbb{R}^{|\mathcal{E}|L}$ stacks the residual differences $\mathbf{e}_i - \mathbf{e}_j \in \mathbb{R}^L$ over all edges.

3.2.2 Cross-Variable Loss

Building upon the defined graph topology, we now formalize our objective. First, by reshaping the predictions into patch matrices, the standard DF loss can be equivalently rewritten as

$$\mathcal{L}_{\mathrm{df}} = \frac{1}{N \times L}\|\hat{Z} - Z\|_F^2 = \frac{1}{T \times D}\|\hat{Y} - Y\|_F^2. \tag{8}$$

To capture the relative structural geometry between variables, for each edge $(i, j) \in \mathcal{E}$, we define the predicted and true patch differences as

$$\Delta_{ij}^{\mathrm{pred}} = \hat{\mathbf{z}}_i - \hat{\mathbf{z}}_j, \qquad \Delta_{ij}^{\mathrm{true}} = \mathbf{z}_i - \mathbf{z}_j. \tag{9}$$

We then define CvLoss as the mean absolute discrepancy of these structural differences:

$$\mathcal{L}_{\mathrm{cv}} := \frac{1}{|\mathcal{E}| \times L} \sum_{(i,j)\in\mathcal{E}} \left\|\Delta_{ij}^{\mathrm{pred}} - \Delta_{ij}^{\mathrm{true}}\right\|_1. \tag{10}$$

Using the node-wise residual $\mathbf{e}_i = \hat{\mathbf{z}}_i - \mathbf{z}_i$, the structural discrepancy on each edge simplifies neatly:

$$\Delta_{ij}^{\mathrm{pred}} - \Delta_{ij}^{\mathrm{true}} = (\hat{\mathbf{z}}_i - \hat{\mathbf{z}}_j) - (\mathbf{z}_i - \mathbf{z}_j) = \mathbf{e}_i - \mathbf{e}_j. \tag{11}$$

Therefore, CvLoss admits a compact graph-theoretic form via the incidence matrix $\tilde{A}$:

$$\mathcal{L}_{\mathrm{cv}} = \frac{1}{|\mathcal{E}| \times L} \sum_{(i,j)\in\mathcal{E}} \|\mathbf{e}_i - \mathbf{e}_j\|_1 = \frac{1}{|\mathcal{E}| \times L}\|\tilde{A}\,\mathrm{vec}(E)\|_1. \tag{12}$$

This computes the total variation of prediction errors across the graph.

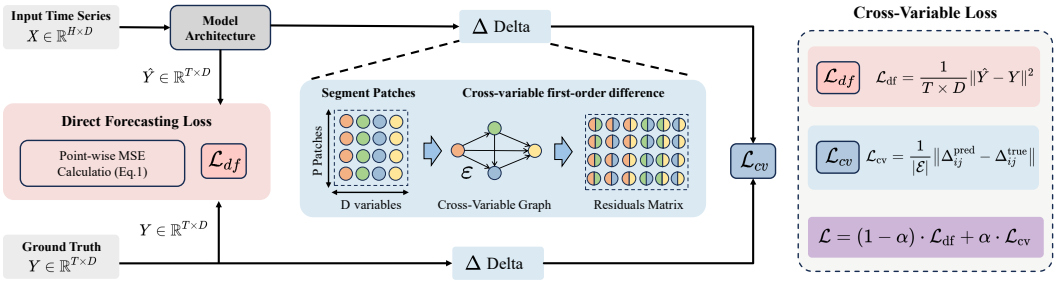


Figure 2: Overview of the workflow CvLoss.

3.2.3 Interpretation

The role of CvLoss is to explicitly regularize the cross-variable residual structure that is not directly controlled by the point-wise DF loss. In particular, $\mathcal{L}_{\text{df}}$ encourages each residual $\mathbf{e}_v$ to be small in absolute magnitude, while $\mathcal{L}_{\text{cv}}$ encourages connected nodes to have compatible residual patterns. This induces an explicit coupling across variables at the objective level.

From the graph perspective, $\mathcal{L}_{\text{cv}}$ is an ℓ_1-type graph total variation regularizer on the residual field. Unlike point-wise MSE, which evaluates each entry independently, CvLoss constrains the edge-wise discrepancies of prediction errors across variables. This encourages the model to exploit shared temporal structure and improves coordination among related variables.

We use the ℓ_1 norm because it is less sensitive to a few large edge-wise mismatches caused by localized anomalies or variable-specific shocks, while still promoting graph-level residual consistency. Replacing the squared penalty of Proposition 3.2 by ℓ_1 yields a pairwise Gibbs field over the same graph rather than a Gaussian one, so the ℓ_1 form is a task-specific empirical choice rather than a consequence of the proposition; Appendix C.4 compares the two norms directly under an identical protocol.

3.3 Model Implementation

Our framework augments vanilla DF with the CvLoss constraint, complementing point-wise supervision with cross-variable regularization.

As shown in Figure 2, the normalized history X is fed into the model to produce $\hat{Y} = f_\theta(X)$. We compute the point-wise error $\mathcal{L}_{\text{df}}$, map forecasts and labels into patch nodes, and compute the structural error $\mathcal{L}_{\text{cv}}$ over $\mathcal{E}$. Since the incidence operations are linear, $\mathcal{L}_{\text{cv}}$ is optimized end-to-end.

Finally, the independent DF error and the structural cross-variable error are fused. A weighting parameter $\alpha \in (0, 1)$ controls the relative contribution of each objective:

$$\mathcal{L}_\alpha := (1 - \alpha) \cdot \mathcal{L}_{\text{df}} + \alpha \cdot \mathcal{L}_{\text{cv}}. \tag{13}$$

Throughout the paper α is a *fixed* scalar selected on the validation split, together with the patch length L the only quantity we tune; it is never learned during training and never varies across batches, variables or edges. Appendix B.2 records the selection protocol, and Appendix D.2 reports a separate diagnostic study whose adaptive coefficients are not values of α.

By sharing statistical strength across the multivariate series, our framework complements the DF paradigm with explicit cross-variable regularization while preserving its practical advantages, including efficient inference and multi-task learning capabilities. It is worth noting that the standard $\mathcal{L}_{\text{df}}$ term serves as a mathematically necessary absolute anchor. It prevents uniform drift and trivial translation-invariant solutions (where $\mathbf{e}_i = \mathbf{e}_j = \mathbf{c} \neq \mathbf{0}$) even when the structural penalty dominates (i.e., $\alpha \to 1$). Furthermore, our proposed regularization scheme is entirely model-agnostic, seamlessly compatible with various forecasting architectures f_θ (e.g., Transformers, CNNs, and MLPs).

Table 1: Overall forecasting performance.

Models	**CvLoss (Ours)**		TimeFilter (2025)		TQNet (2025)		iTransformer (2024)		Leddam (2024)		SOFTS (2023)		PatchTST (2023)		Crossformer (2023)		TimesNet (2023)		DLinear (2023)		FEDformer (2022)	
Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	**0.372**	**0.381**	0.377	0.393	0.377	0.393	0.407	0.410	0.386	0.397	0.393	0.403	0.387	0.400	0.513	0.496	0.400	0.406	0.404	0.408	0.448	0.452
ETTm2	**0.270**	**0.317**	0.272	0.321	0.278	0.322	0.288	0.332	0.281	0.325	0.287	0.330	0.281	0.326	0.757	0.610	0.291	0.333	0.354	0.402	0.305	0.349
ETTh1	**0.419**	**0.427**	0.420	0.428	0.441	0.434	0.454	0.447	0.431	0.429	0.449	0.442	0.469	0.454	0.529	0.522	0.458	0.450	0.461	0.457	0.440	0.460
ETTh2	**0.359**	**0.391**	0.365	0.398	0.378	0.403	0.383	0.407	0.373	0.399	0.385	0.408	0.387	0.407	0.942	0.684	0.414	0.427	0.563	0.519	0.437	0.449
Weather	**0.236**	**0.260**	0.240	0.270	0.242	0.268	0.258	0.279	0.242	0.272	0.255	0.278	0.259	0.281	0.259	0.315	0.259	0.287	0.265	0.315	0.309	0.360
ECL	**0.157**	**0.252**	0.159	0.256	0.164	0.258	0.178	0.270	0.169	0.263	0.174	0.264	0.216	0.304	0.244	0.334	0.193	0.295	0.225	0.319	0.214	0.327
Traffic	**0.407**	**0.254**	0.408	0.269	0.445	0.276	0.428	0.282	0.467	0.294	0.409	0.267	0.555	0.362	0.550	0.304	0.620	0.336	0.625	0.383	0.610	0.376
Solar	0.218	**0.254**	0.228	0.262	**0.197**	0.255	0.233	0.262	0.230	0.264	0.229	0.256	0.270	0.307	0.641	0.639	0.301	0.319	0.330	0.401	0.292	0.381
PEMS03	**0.079**	**0.183**	0.084	0.191	0.080	0.186	0.096	0.204	0.101	0.210	0.087	0.192	0.151	0.265	0.138	0.253	0.119	0.271	0.219	0.295	0.167	0.291
PEMS04	**0.078**	**0.181**	0.083	0.186	0.081	0.185	0.098	0.207	0.102	0.213	0.091	0.196	0.162	0.273	0.145	0.267	0.109	0.220	0.236	0.350	0.195	0.308
PEMS07	**0.063**	**0.156**	0.071	0.170	0.065	0.161	0.088	0.190	0.087	0.192	0.075	0.173	0.166	0.270	0.181	0.272	0.106	0.208	0.241	0.343	0.133	0.282
PEMS08	**0.080**	**0.182**	0.083	0.186	0.105	0.203	0.127	0.212	0.102	0.211	0.114	0.208	0.238	0.289	0.232	0.270	0.150	0.244	0.347	0.421	0.234	0.326

Note: We fix the input length as 96 for all baselines following [1]. **Bold** and underlined denote best and second-best results, respectively. *Avg* indicates average results over multiple forecast horizons. CvLoss using TimeFilter [4] as backbone.

4 Experiments

We evaluate forecast accuracy, learning-objective comparisons, generalization across backbones, ablations and hyperparameter sensitivity. Appendix C.10 examines training cost and inference latency.

4.1 Setup

Datasets. Following [26], we evaluate on standard benchmarks: four ETT subsets, ECL, Traffic, Weather, and PEMS [1]. All datasets are chronologically split into training, validation, and test sets. Statistics are in Appendix B.1.

Baselines. We compare CvLoss to a range of competitive baselines, categorized as: (1) Transformer-based models: PatchTST [33], iTransformer [1], TQNet [35], Crossformer [25], FEDformer [36] and Leddam [37]; (2) Non-transformer models: DLinear [38], SOFTS [39], TimesNet [26] and TimeFilter [4].

Implementation. Baseline implementations closely follow the official codebase from TimeFilter [4] and TQNet [35]. To ensure fair comparison, the drop-last trick is disabled for all models, as recommended in [40]. All models are trained with the Adam optimizer [41]. When integrating CvLoss into a baseline forecast model, we retain all hyperparameters from the public benchmarks [4, 35], only tuning α and the patch size. Experiments are run on Intel(R) Xeon(R) Platinum 8470Q with 8 NVIDIA RTX 5090 GPUs. Further implementation details are provided in Appendix B.

4.2 Overall performance

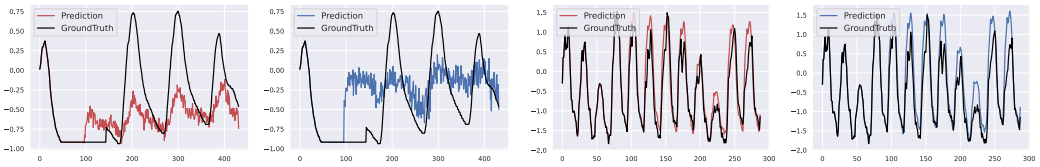
Table 1 compares long- and short-term forecasting results. Adding CvLoss consistently improves accuracy across datasets; on Traffic, for example, MAE decreases by 0.014. The gains come from explicit structural regularization over synchronous and asynchronous cross-variable dependencies.

Backbone-controlled attribution. In Table 1, CvLoss augments TimeFilter with the same data and training protocol and improves all 24 dataset-level entries (12 datasets $\times$ 2 metrics). The other columns establish the competitiveness of the backbone. Table 2 consolidates 114 fixed-backbone comparisons across seven backbones, three architecture families, twelve datasets and both metrics: CvLoss improves 111, ties 2 and loses 1, with mean relative reductions of 3.39% in MSE and 3.61% in MAE. At the horizon level, it improves all 96 comparisons in Table 10 and 31 of 32 in Table 4, with one tie.

Table 2: **Backbone-controlled summary of CvLoss.** Every comparison contrasts a forecasting backbone trained with its standard objective against the *identical* backbone, data split, and training protocol trained with CvLoss, so the loss term is the only variable. A *cell* is one (backbone, dataset, metric) triple evaluated at the dataset-level average over forecast horizons; W/T/L counts cells on which CvLoss is better, equal, or worse, and the last two columns give the mean relative error reduction over the cells of that block.

Evidence	Backbone(s) held fixed	Datasets	Cells	W / T / L	Mean reduction (%)	
					MSE	MAE
Table 1	TimeFilter	12	24	**24** / 0 / 0	3.20	3.12
Table 3	TQNet, PDF	4	16	**15** / 0 / 1	2.50	2.06
Table 4	TQNet	4	8	**8** / 0 / 0	1.16	1.66
Table 10	TQNet, TimeFilter, CFPT, iTransformer	3	24	**24** / 0 / 0	1.61	2.17
Table 19	iTransformer, DLinear, PatchTST	7	42	**40** / 2 / 0	5.29	5.66
Total	**7 distinct backbones**	**12**	**114**	**111 / 2 / 1**	**3.39**	**3.61**

Note: Counts summarize overlapping evaluation blocks under their respective protocols and are descriptive rather than a pooled statistical test. The only loss is TQNet/ETTh1 MAE in Table 3 ($0.434 \rightarrow 0.438$); the two ties are DLinear/Weather MSE and PatchTST/ECL MSE in Table 19, equal at three-decimal precision.



(a) ETTm2 snapshot. (b) ECL snapshot.

Figure 3: The forecast sequence of DF (in blue) and CvLoss (in red), with historical length $\mathrm{H} = 96$.

These blocks retain their respective evaluation protocols and overlap in backbones and datasets; the aggregate summarizes evidence rather than a pooled significance test. Appendix C.9 reports paired effect sizes and confidence intervals. Gains depend on the cross-variable structure left unmodeled by each backbone: Traffic with iTransformer improves MSE by 17.7%, PEMS07 with TimeFilter by 11.3%, and ETTh1 with TimeFilter by 0.24%. Section 4.6 and Appendix B.3 examine this dependence.

Examples. A qualitative comparison between forecasts generated by DF versus CvLoss is presented in Figure 3. While the model trained with DF captures general periodic trends, it often fails to model accurate amplitudes and intricate temporal dynamics. For example, on the ETTm2 dataset, it suffers from severe amplitude attenuation, completely missing the prominent ground-truth peaks. Similarly, on ECL, it overestimates the high periodic peaks, such as those observed around the 1750th and 250th steps. In contrast, CvLoss accurately captures these large peak magnitudes and complex fluctuations.

4.3 Learning objective comparison

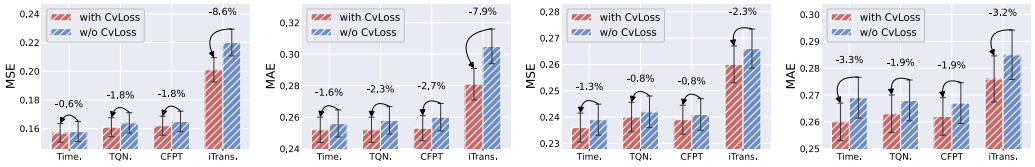
CvLoss is compared with 11 learning objectives from four categories: shape alignment, likelihood maximization, distribution balancing, and decomposition. Table 3 summarizes average performance on four datasets, with full horizon-level results in Appendix Table 14.

Shape-alignment losses (e.g., Soft-DTW [14], Koopman [17]) and likelihood objectives (e.g., Time-o1 [16], FreDF [7], QDF [15]) improve over MSE, but still treat multivariate outputs as scalar predictions. They do not explicitly enforce the cross-variable residual consistency motivated by Theorem 3.1. By penalizing edge-wise residual discrepancies, CvLoss ranks first in 20 settings (Appendix Table 14) and generalizes across TQNet [35] and PDF [42] (Appendix Table 13).

Table 3: Comparable results with other learning objectives for time-series forecasting.

Loss		**Ours**		QDF		Time-o1		FreDF		Koopman		Soft-DTW		DF	
Metrics		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
TQNet	ETTm1	0.373	**0.386**	**0.371**	0.389	0.372	0.390	0.375	0.390	0.499	0.595	0.394	0.387	0.377	0.393
	ETTh1	**0.430**	0.438	0.431	**0.431**	0.432	0.437	0.432	0.432	0.442	0.451	0.453	0.438	0.441	0.434
	ECL	**0.162**	**0.253**	0.165	0.257	0.167	0.257	0.168	0.257	0.166	0.258	0.524	0.623	0.165	0.259
	Weather	**0.241**	**0.264**	0.242	0.268	0.245	0.269	0.244	0.268	0.306	0.282	0.276	0.255	0.242	0.268
PDF	ETTm1	**0.378**	**0.383**	0.381	0.394	0.386	0.399	0.387	0.400	0.587	0.485	0.396	0.404	0.387	0.396
	ETTh1	**0.429**	0.432	0.436	**0.429**	0.438	0.438	0.437	0.435	0.497	0.472	0.447	0.447	0.452	0.440
	ECL	**0.189**	**0.269**	0.194	0.277	0.195	0.276	0.194	0.274	0.196	0.281	0.695	0.548	0.198	0.281
	Weather	**0.259**	**0.276**	**0.259**	0.281	0.264	0.284	0.268	0.287	0.268	0.290	1.296	0.452	0.265	0.283

Note: **Bold** and underlined denote best and second-best results, respectively. Follow the settings of QDF [15].



(a) ECL with MSE (b) ECL with MAE (c) Weather with MSE (d) Weather with MAE

Figure 4: Improvement of CvLoss applied to different forecast models, shown with colored bars for means over forecast lengths (96, 192, 336, 720) and error bars for 50% confidence intervals.

4.4 Generalization studies

We test CvLoss on TimeFilter, TQNet, CFPT [43], and iTransformer. As shown in Figure 4, CvLoss improves different architectures and datasets. On ECL, all four backbones improve, with iTransformer reducing MSE by 8.6% and MAE by 7.9%. On Weather, MSE drops by 1.3%–2.3% and MAE by 1.9%–3.3%, showing that CvLoss is broadly plug-and-play.

4.5 Ablation studies

We ablate the edge set $\mathcal{E}$: CvLoss† includes synchronous cross-variable interactions at the same patch index; CvLoss‡ includes lagged cross-variable interactions at different indices; and default CvLoss includes both. Table 4 shows that combining synchronous and lagged interactions gives the best overall forecasting performance.

Table 4: Ablation study results.

Model	Sync.	Async.	Data	T=96		T=192		T=336		T=720		Avg	
				MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
DF	✗	✗	ETTm1	0.311	0.353	0.357	0.378	0.390	0.401	0.449	0.439	0.377	0.393
			ETTh1	0.371	0.393	0.428	0.426	0.475	0.446	0.488	0.470	0.441	0.434
			ECL	0.134	0.229	0.154	0.247	0.170	0.265	0.200	0.293	0.165	0.259
			Weather	0.156	0.200	0.205	0.245	0.262	0.287	0.343	0.342	0.242	0.269
CvLoss†	✓	✗	ETTm1	0.306	0.344	0.357	**0.372**	0.388	**0.399**	0.450	0.439	0.375	0.389
			ETTh1	0.369	0.387	0.426	0.422	0.475	0.444	0.505	0.485	0.444	0.435
			ECL	0.134	0.228	0.152	0.243	0.169	0.261	0.199	0.289	0.164	0.255
			Weather	0.159	0.202	0.208	0.246	0.265	0.287	0.344	0.341	0.244	0.269
CvLoss‡	✗	✓	ETTm1	0.305	0.343	0.353	0.373	0.390	0.400	0.448	0.431	0.374	0.387
			ETTh1	0.369	0.387	0.428	0.422	0.473	0.444	0.505	0.485	0.444	0.435
			ECL	0.134	0.228	0.153	0.244	0.170	0.262	0.200	0.290	0.164	0.256
			Weather	0.156	0.196	0.203	0.241	0.263	0.286	0.343	0.340	0.241	0.266
CvLoss	✓	✓	ETTm1	**0.305**	**0.343**	**0.352**	0.373	**0.388**	0.400	**0.444**	**0.430**	**0.372**	**0.387**
			ETTh1	**0.368**	**0.385**	**0.426**	**0.422**	**0.471**	**0.443**	**0.486**	**0.469**	**0.438**	**0.430**
			ECL	**0.133**	**0.225**	**0.151**	**0.241**	**0.166**	**0.258**	**0.196**	**0.287**	**0.162**	**0.253**
			Weather	**0.154**	**0.192**	**0.203**	**0.238**	**0.262**	**0.285**	**0.342**	**0.339**	**0.240**	**0.264**

Note: **Bold** and underlined denote best and second-best results, respectively. "Sync." and "Async." are abbreviations for synchronous and asynchronous.

Table 5: Varying α results of TQNet

α	ETTh2		ECL		Weather	
	MSE	MAE	MSE	MAE	MSE	MAE
0	0.379	0.403	0.164	0.258	0.242	0.269
0.001	0.379	0.403	0.164	0.259	0.242	0.269
0.002	0.379	0.403	0.164	0.258	0.242	0.269
0.005	0.379	0.403	0.164	0.258	0.242	0.268
0.01	0.379	0.403	0.164	0.258	0.242	0.268
0.02	0.379	0.403	0.164	0.258	0.242	0.268
0.05	0.378	0.403	0.164	0.257	0.242	0.268
0.1	0.378	0.402	0.164	0.257	0.242	0.268
0.2	0.377	0.402	0.162	0.255	0.242	0.267
0.5	0.372	0.398	**0.162**	**0.252**	0.241	0.265
1	**0.365**	**0.391**	0.176	0.264	**0.241**	**0.263**

Note: **Bold** and underlined denote the best and second-best results.

Table 6: Varying α results of TimeFilter.

α	ETTh2		ECL		Weather	
	MSE	MAE	MSE	MAE	MSE	MAE
0	0.367	0.399	0.159	0.256	0.241	0.270
0.001	0.367	0.399	0.159	0.256	0.240	0.270
0.002	0.367	0.399	0.159	0.256	0.241	0.270
0.005	0.367	0.399	0.160	0.257	0.241	0.270
0.01	0.367	0.399	0.159	0.256	0.240	0.270
0.02	0.367	0.399	0.163	0.259	0.240	0.270
0.05	0.366	0.399	0.159	0.255	0.240	0.269
0.1	0.366	0.398	0.159	0.255	0.240	0.269
0.2	0.364	0.397	0.159	0.254	0.239	0.267
0.5	**0.361**	0.393	**0.157**	**0.251**	0.237	0.264
1	0.362	**0.390**	0.168	0.261	**0.236**	**0.260**

Note: **Bold** and underlined denote the best and second-best results.

4.6 Hyperparameter sensitivity

Tables 5 and 6 report the sensitivity of CvLoss to the loss weight α on TQNet and TimeFilter. Increasing α from 0 yields consistent improvements, indicating the benefit of incorporating cross-variable structural dependencies into the learning objective across different backbones and datasets. For example, CvLoss reduces the MSE of TQNet from 0.379 to 0.365 on ETTh2, and that of TimeFilter from 0.159 to 0.157 on ECL. However, using the structural penalty ($\alpha = 1$) is not always optimal; on ECL, both models perform best around $\alpha = 0.5$ and degrade when α reaches 1. This suggests that CvLoss is most effective when combined with point-wise MSE, which anchors forecasting accuracy and prevents translation-invariant solutions under structural regularization.

5 Conclusion

This paper studies the overlooked output-side dependencies in multivariate time series forecasting. We show that the standard point-wise DF objective is mismatched with real-world spatiotemporal correlations, and introduce CvLoss to explicitly regularize cross-variable residual consistency. CvLoss is implemented as a graph total variation penalty over forecast patches, allowing both synchronous and lagged interactions to be modeled without changing the inference procedure. Extensive experiments show that CvLoss consistently improves SOTA forecasting backbones, outperforms competitive learning objectives, and stays efficient in practice. These results suggest that modeling the structural geometry of future variables is a simple but effective direction for improving multivariate forecasting.

6 Limitations

CvLoss uses a fixed patching scheme and a uniform cross-variable graph. The complete graph includes both synchronous and lagged interactions but does not estimate heterogeneous edge strengths; adaptive graph weights and patch sizes offer directions for greater efficiency and interpretability. Random edge sampling provides a scalability option, while the reported accuracy results use the complete graph. Our evaluation concerns deterministic forecasting on regular benchmarks; probabilistic forecasting, irregular sampling [44], and missing-value settings remain future directions.

The Gaussian analysis motivates residual coupling, while Proposition 3.2 relates the squared graph penalty to a structured precision matrix. The implemented ℓ_1 penalty instead defines a non-Gaussian pairwise Gibbs field. Appendix C.4 compares both norms and supports the empirical choice of ℓ_1. Top-K selection in Appendix D and the supports studied in Section 4.5 examine graph variants, without learning a full weighted dependency structure.

Acknowledgments and Disclosure of Funding

Funding. Hao Xue acknowledges support from the HKUST(GZ) Startup Fund (Grant No. G0101000338).

Competing interests. The authors declare no competing interests.

References

[1] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024.

[2] Lifan Zhao and Yanyan Shen. Rethinking channel dependence for multivariate time series forecasting: Learning from leading indicators. In *The Twelfth International Conference on Learning Representations*, 2024.

[3] Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. Duet: Dual clustering enhanced multivariate time series forecasting. In *SIGKDD*, pages 1185–1196, 2025.

[4] Yifan Hu, Guibin Zhang, Peiyuan Liu, Disen Lan, Naiqi Li, Dawei Cheng, Tao Dai, Shu-Tao Xia, and Shirui Pan. Timefilter: Patch-specific spatial-temporal graph filtration for time series forecasting. In *Forty-second International Conference on Machine Learning*, 2025.

[5] Kuiye Ding, Fanda Fan, Yao Wang, Ruijie Jian, Xiaorui Wang, Luqi Gong, Yishan Jiang, Chunjie Luo, and Jianfeng Zhan. Dualsg: A dual-stream explicit semantic-guided multivariate time series forecasting framework. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 508–517, 2025.

[6] Chunyi Hou, Yongchuan Yu, Jinquan Ji, Siyao Zhang, Xumeng Shen, and Jianzhuo Yan. Graph-patchformer: Patch interaction transformer with adaptive graph learning for multivariate time series forecasting. *Neural Networks*, 194:108140, 2026. doi: 10.1016/j.neunet.2025.108140.

[7] Hao Wang, Licheng Pan, Zhichao Chen, Degui Yang, Sen Zhang, Yifei Yang, Xinggao Liu, Haoxuan Li, and Dacheng Tao. Fredf: Learning to forecast in the frequency domain. In *ICLR*, 2025.

[8] Yifan Hu, Yuante Li, Peiyuan Liu, Yuxia Zhu, Naiqi Li, Tao Dai, Shu tao Xia, Dawei Cheng, and Changjun Jiang. Fintsb: A comprehensive and practical benchmark for financial time series forecasting. *arXiv preprint arXiv:2502.18834*, 2025.

[9] Yifan Hu, Peiyuan Liu, Yuante Li, Dawei Cheng, Naiqi Li, Tao Dai, Jigang Bao, and Xia Shu-Tao. Finmamba: Market-aware graph enhanced multi-level mamba for stock movement prediction. *arXiv preprint arXiv:2502.06707*, 2025.

[10] Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Mingsheng Long, and Jianmin Wang. Deep time series models: A comprehensive survey and benchmark. 2024.

[11] Xiangfei Qiu, Hanyin Cheng, Xingjian Wu, Junkai Lu, Jilin Hu, Chenjuan Guo, Christian S. Jensen, and Bin Yang. A comprehensive survey of deep learning for multivariate time series forecasting: A channel strategy perspective, 2026.

[12] Xiang Liu, Naiqi Li, and Shu-Tao Xia. Gdtw: A novel differentiable dtw loss for time series tasks. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pages 2860–2864. IEEE, 2021.

[13] Vincent Le Guen and Nicolas Thome. Shape and time distortion loss for training deep time series forecasting models. *Proc. Adv. Neural Inf. Process. Syst.*, 32, 2019.

[14] Marco Cuturi and Mathieu Blondel. Soft-dtw: a differentiable loss function for time-series. In *Proc. Int. Conf. Mach. Learn.*, pages 894–903. PMLR, 2017.

[15] Hao Wang, Licheng Pan, Yuan Lu, Zhichao Chen, Tianqiao Liu, Shuting He, Zhixuan Chu, Qingsong Wen, Haoxuan Li, and Zhouchen Lin. Quadratic direct forecast for training multi-step time-series forecast models. In *ICLR*, pages 1–9, 2026.

[16] Hao Wang, Licheng Pan, Zhichao Chen, Xu Chen, Qingyang Dai, Lei Wang, Haoxuan Li, and Zhouchen Lin. Time-o1: Time-series forecasting needs transformed label alignment. *Proc. Adv. Neural Inf. Process. Syst.*, 2025.

[17] Henning Lange, Steven L Brunton, and J Nathan Kutz. From fourier to koopman: Spectral methods for long-term time series prediction. *Journal of Machine Learning Research*, 22(41): 1–38, 2021.

[18] Licheng Pan, Hao Wang, Haocheng Yang, Yuqi Li, Qingsong Wen, Xiaoxi Li, Zhichao Chen, Haoxuan Li, Zhixuan Chu, and Yuan Lu. Deep time-series forecasting needs kernelized moment balancing, 2026.

[19] Hao Wang, Licheng Pan, Yuan Lu, Zhixuan Chu, Xiaoxi Li, Shuting He, Zhichao Chen, Haoxuan Li, Qingsong Wen, and Zhouchen Lin. Distdf: Time-series forecasting needs joint-distribution wasserstein alignment. In *Proc. Int. Conf. Learn. Represent.*, pages 1–9, 2026.

[20] Xiangfei Qiu, Xingjian Wu, Hanyin Cheng, Xvyuan Liu, Chenjuan Guo, Jilin Hu, and Bin Yang. Dbloss: Decomposition-based loss function for time series forecasting. In *NeurIPS*, 2025.

[21] Qi Xiong, Kai Tang, Minbo Ma, Ji Zhang, Jie Xu, and Tianrui Li. Modeling temporal dependencies within the target for long-term time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 37(12):7300–7314, 2025. doi: 10.1109/TKDE.2025.3609415.

[22] Dilfira Kudrat, Zongxia Xie, Yanru Sun, Tianyu Jia, and Qinghua Hu. Patch-wise structural loss for time series forecasting. In *Forty-second International Conference on Machine Learning*, 2025.

[23] Yuxuan Liang, Haomin Wen, Yuqi Nie, Yushan Jiang, Ming Jin, Dongjin Song, Shirui Pan, and Qingsong Wen. Foundation models for time series analysis: A tutorial and survey. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024.

[24] Si-An Chen, Chun-Liang Li, Sercan O Arik, Nathanael Christian Yoder, and Tomas Pfister. TSMixer: An all-MLP architecture for time series forecast-ing. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

[25] Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *International Conference on Learning Representations*, 2023.

[26] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

[27] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020.

[28] Wenyong Han, Tao Zhu, Liming Chen, Huansheng Ning, Yang Luo, and Yaping Wan. Mcformer: Multivariate time series forecasting with mixed-channels transformer. *IEEE Internet of Things Journal*, 11(17):28320–28329, 2024.

[29] TingYu Zhao, Bo Peng, Yuan Sun, DaiPeng Yang, ZhenGuang Zhang, and Xi Wu. Rethinking superpixel segmentation from biologically inspired mechanisms. *Applied Soft Computing*, 156: 111467, 2024.

[30] Jiawen Wei, Lan Jiang, Pengbo Wei, Ziwen Ye, Teng Song, Chen Chen, and Guangrui Ma. TARFVAE: Efficient one-step generative time series forecasting via TARFLOW based VAE. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

[31] Runze Li, Hanchen Wang, Wenjie Zhang, Binghao Li, Yu Zhang, Xuemin Lin, and Ying Zhang. Fadti: Fourier and attention driven diffusion for multivariate time series imputation. In *Proceedings of the 2026 IEEE International Conference on Data Mining (ICDM)*, 2026.

[32] Theodoros Evgeniou, Charles A. Micchelli, and Massimiliano Pontil. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6(21):615–637, 2005.

[33] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

[34] Kuiye Ding, Fanda Fan, Chunyi Hou, Zheya Wang, Lei Wang, Zhengxin Yang, and Jianfeng Zhan. Timemosaic: Temporal heterogeneity guided time series forecasting via adaptive granularity patch and segment-wise decoding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(25):20790–20798, Mar. 2026. doi: 10.1609/aaai.v40i25.39218. URL `https://ojs.aaai.org/index.php/AAAI/article/view/39218`.

[35] Shengsheng Lin, Haojun Chen, Haijie Wu, Chunyun Qiu, and Weiwei Lin. Temporal query network for efficient multivariate time series forecasting. In *Forty-second International Conference on Machine Learning*, 2025.

[36] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, pages 27268–27286. PMLR, 2022.

[37] Guoqi Yu, Jing Zou, Xiaowei Hu, Angelica I Aviles-Rivero, Jing Qin, and Shujun Wang. Revitalizing multivariate time series forecasting: Learnable decomposition with inter-series dependencies and intra-series variations modeling. In *Forty-first International Conference on Machine Learning*, 2024.

[38] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proc. AAAI Conf. Artif. Intell.*, 2023.

[39] Lu Han, Xu-Yang Chen, Han-Jia Ye, and De-Chuan Zhan. SOFTS: Efficient multivariate time series forecasting with series-core fusion. In *Advances in Neural Information Processing Systems*, 2024.

[40] Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. Tfb: Towards comprehensive and fair benchmarking of time series forecasting methods. In *VLDB*, pages 2363–2377, 2024.

[41] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, pages 1–9, 2015.

[42] Tao Dai, Beiliang Wu, Peiyuan Liu, Naiqi Li, Jigang Bao, Yong Jiang, and Shu-Tao Xia. Periodicity decoupling framework for long-term series forecasting. *International Conference on Learning Representations*, 2024.

[43] Feifei Kou, Jiahao Wang, Lei Shi, Yuhan Yao, Yawen Li, Suguo Zhu, Zhongbao Zhang, and Junping Du. Cfpt: Empowering time series forecasting through cross-frequency interaction and periodic-aware timestamp modeling. In *Forty-second International Conference on Machine Learning*, 2025.

[44] Jinquan Ji, Chunyi Hou, Yu Cao, Yukun Ma, and Jianzhuo Yan. Fine-grained dynamic graph learning for irregular multivariate time series. *Information Processing & Management*, 63(6): 104740, 2026. doi: 10.1016/j.ipm.2026.104740.

[45] 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 *The Thirty-Fifth AAAI Conference on Artificial Intelligence*, 2021.

[46] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with Auto-Correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, 2021.

[47] 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 *Advances in Neural Information Processing Systems*, 2022.

[48] Peiyuan Liu, Beiliang Wu, Yifan Hu, Naiqi Li, Tao Dai, Jigang Bao, and Shu-Tao Xia. Timebridge: Non-stationarity matters for long-term time series forecasting. *International Conference on Machine Learning*, 2025.

A Theoretical Justification

In this section, we provide the detailed derivation of Theorem 3.1, which reveals the suboptimality of the standard DF objective under real-world spatiotemporal correlations, followed by the proof of Proposition 3.2.

A.1 Proof of Theorem 3.1

Proof. From a probabilistic perspective, finding the optimal parameters θ for the forecasting model f_θ is equivalent to maximizing the likelihood of the ground-truth target Y given the input history X. Let

$$\mathbf{e} = \mathrm{vec}(\hat{Y} - Y) \in \mathbb{R}^{TD}$$

be the flattened forecast error vector.

Assume that the ground-truth residuals follow a zero-mean multivariate Gaussian distribution, i.e.,

$$\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{ST}),$$

where $\mathbf{\Sigma}_{ST} \in \mathbb{R}^{TD \times TD}$ is the true spatiotemporal covariance matrix. The probability density function of the residuals is given by:

$$p(\mathbf{e}) = \frac{1}{\sqrt{(2\pi)^{TD} \det(\mathbf{\Sigma}_{ST})}} \exp\left(-\frac{1}{2}\mathbf{e}^\top \mathbf{\Sigma}_{ST}^{-1}\mathbf{e}\right)$$

Minimizing the NLL of this distribution yields the optimal objective function. Taking the negative logarithm, we have:

$$-\log p(\mathbf{e}) = \frac{TD}{2}\log(2\pi) + \frac{1}{2}\log\det(\mathbf{\Sigma}_{ST}) + \frac{1}{2}\mathbf{e}^\top \mathbf{\Sigma}_{ST}^{-1}\mathbf{e}$$

Assuming the covariance matrix $\mathbf{\Sigma}_{ST}$ represents the intrinsic data distribution and is independent of the model parameters θ, the first two terms are constants with respect to the optimization process. By dropping these constants and scaling the objective by $\frac{1}{TD}$ to obtain the average loss per prediction entry, we define the practical NLL objective as:

$$\mathcal{L}_{\mathrm{nll}} = \frac{1}{2TD}\mathbf{e}^\top \mathbf{\Sigma}_{ST}^{-1}\mathbf{e}$$

Now, consider the standard point-wise MSE objective used in the DF paradigm, defined as:

$$\mathcal{L}_{\mathrm{df}} = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{D}\|Y_{t,:} - \hat{Y}_{t,:}\|_2^2 = \frac{1}{TD}\mathbf{e}^\top\mathbf{e}$$

Notice that $\mathcal{L}_{\mathrm{df}}$ can be probabilistically interpreted as the negative log-likelihood under a strictly naive assumption: the residuals of all variables across all time steps are entirely independent and share a uniform variance σ^2. Under this assumption, the covariance matrix reduces to a spherical structure $\mathbf{\Sigma}_{ST} = \sigma^2\mathbf{I}$, where $\mathbf{I} \in \mathbb{R}^{TD \times TD}$ is the identity matrix. In this special case, the NLL naturally degrades to:

$$\mathcal{L}_{\mathrm{nll_naive}} = \frac{1}{2TD}\mathbf{e}^\top(\sigma^2\mathbf{I})^{-1}\mathbf{e} = \frac{1}{2\sigma^2}\left(\frac{1}{TD}\mathbf{e}^\top\mathbf{e}\right) = \frac{1}{2\sigma^2}\mathcal{L}_{\mathrm{df}}$$

To formally quantify the structural mismatch between the standard DF paradigm and the optimal estimation under true spatiotemporal correlations, we define the structural gap Δ as the difference between the true optimal objective $\mathcal{L}_{\mathrm{nll}}$ and the scaled DF objective $\frac{1}{2\sigma^2}\mathcal{L}_{\mathrm{df}}$:

$$\Delta = \mathcal{L}_{\mathrm{nll}} - \frac{1}{2\sigma^2}\mathcal{L}_{\mathrm{df}} = \frac{1}{2TD}\mathbf{e}^\top\mathbf{\Sigma}_{ST}^{-1}\mathbf{e} - \frac{1}{2TD\sigma^2}\mathbf{e}^\top\mathbf{I}\mathbf{e}$$

Factoring out the common terms, we obtain:

$$\Delta = \frac{1}{2TD}\mathbf{e}^\top\left(\mathbf{\Sigma}_{ST}^{-1} - \frac{1}{\sigma^2}\mathbf{I}\right)\mathbf{e}$$

If cross-variable or lagged residual dependencies yield non-zero off-diagonal entries in $\mathbf{\Sigma}_{ST}^{-1}$, the symmetric matrix $\mathbf{\Sigma}_{ST}^{-1} - \sigma^{-2}\mathbf{I}$ is non-zero. Its quadratic form is therefore not identically zero, so point-wise MSE and the true Gaussian NLL are different objectives. The gap can still be zero for particular residual vectors; the result concerns equality of the objective functions rather than a non-zero difference at every point. This concludes the proof.

A.2 Proof of Proposition 3.2

Theorem 3.1 shows that MSE is the correct objective only when the residual precision matrix is a multiple of the identity, but it does not say which non-diagonal precision matrix should replace it, and estimating $\mathbf{\Sigma}_{ST}^{-1}$ from data is not feasible at the scales considered here. Proposition 3.2 closes this gap from the opposite direction: instead of estimating a precision matrix, we fix a one-parameter family of precision matrices whose off-diagonal support is a cross-variable graph, and show that the negative log-likelihood of that family is exactly MSE plus a squared graph total-variation penalty.

Proof. Recall from Section 3.2 that $A \in \mathbb{R}^{|\mathcal{E}|\times N}$ is the incidence matrix of the cross-variable graph under an arbitrary edge orientation, that $\tilde{A} = A \otimes I_L$ is its lifted counterpart, and that $\mathbf{r} = \mathrm{vec}(E) \in \mathbb{R}^{NL}$ collects the node-wise residuals $\mathbf{e}_1, \ldots, \mathbf{e}_N \in \mathbb{R}^L$. By construction, the k-th block of $\tilde{A}\mathbf{r}$ associated with the edge $k = (i, j)$ equals $\mathbf{e}_i - \mathbf{e}_j$, so

$$\|\tilde{A}\mathbf{r}\|_2^2 = \sum_{(i,j)\in\mathcal{E}} \|\mathbf{e}_i - \mathbf{e}_j\|_2^2.$$

Writing $\mathrm{Lap}(\mathcal{E}) = \tilde{A}^\top\tilde{A}$ for the lifted graph Laplacian, the same quantity is the associated quadratic form,

$$\mathbf{r}^\top \mathrm{Lap}(\mathcal{E})\,\mathbf{r} = \mathbf{r}^\top\tilde{A}^\top\tilde{A}\,\mathbf{r} = \|\tilde{A}\mathbf{r}\|_2^2 = \sum_{(i,j)\in\mathcal{E}} \|\mathbf{e}_i - \mathbf{e}_j\|_2^2.$$

Substituting the structured precision family of Eq. (3) and expanding by linearity gives

$$\mathbf{r}^\top\mathbf{\Sigma}^{-1}(\lambda)\,\mathbf{r} = \frac{1}{\sigma^2}\Big(\mathbf{r}^\top\mathbf{r} + \lambda\,\mathbf{r}^\top\mathrm{Lap}(\mathcal{E})\,\mathbf{r}\Big) = \frac{1}{\sigma^2}\Big(\|\mathbf{r}\|_2^2 + \lambda\sum_{(i,j)\in\mathcal{E}}\|\mathbf{e}_i - \mathbf{e}_j\|_2^2\Big),$$

which is Eq. (4). This concludes the proof.

Remarks. Four consequences are worth recording, since they determine how the theory should and should not be read.

- **The family is a valid precision family.** $\mathrm{Lap}(\mathcal{E})$ is positive semi-definite, so $\mathbf{I} + \lambda\,\mathrm{Lap}(\mathcal{E}) \succ 0$ for every $\lambda \geq 0$ and $\mathbf{\Sigma}^{-1}(\lambda)$ is a genuine precision matrix. Its off-diagonal entries are $-\lambda/\sigma^2$ exactly on the node pairs in $\mathcal{E}$ and zero elsewhere, so $\mathcal{E}$ is the conditional-independence structure of the corresponding Gaussian field rather than an analogy.
- **It contains the spherical case.** At $\lambda = 0$ we recover $\mathbf{\Sigma}^{-1} = \sigma^{-2}\mathbf{I}$, which is precisely the situation in which the objective gap Δ of Theorem 3.1 vanishes. The theorem and the loss are therefore two ends of one family rather than unrelated statements.
- **It matches the deployed weighting.** Dividing the objective $\mathcal{L}_\alpha = (1-\alpha)\mathcal{L}_{\mathrm{df}} + \alpha\mathcal{L}_{\mathrm{cv}}$ by $(1-\alpha)$ leaves the minimiser unchanged and yields $\mathcal{L}_{\mathrm{df}} + \lambda\mathcal{L}_{\mathrm{cv}}$ with $\lambda = \alpha/(1-\alpha)$. Values of α close to 1 therefore correspond to a large λ within the family above, not to a weight outside the stated range $(0, 1)$.
- **The complete graph is the weakest member.** Since $\mathcal{E}$ is the support of the off-diagonal entries, taking $\mathcal{E}$ to be the complete cross-variable graph imposes *no* zero constraint on any cross-variable pair; sparser choices impose strictly more. The empty graph, i.e. plain MSE, is the member excluded by Theorem 3.1 whenever cross-variable dependencies are present.

Proposition 3.2 concerns the squared penalty. The objective we deploy in Eq. (12) uses the ℓ_1 norm on the same edge differences, which yields the pairwise field

$$p_1(\mathbf{r} \mid \mathcal{E}) \;\propto\; \exp\Big[-a\sum_{v=1}^{N}\|\mathbf{e}_v\|_2^2 - b\sum_{(i,j)\in\mathcal{E}}\|\mathbf{e}_i - \mathbf{e}_j\|_1\Big], \qquad a = \frac{1-\alpha}{NL} > 0, \quad b = \frac{\alpha}{|\mathcal{E}|L} > 0.$$

Table 7: Dataset description.

Dataset	D	Forecast length	Train / validation / test	Frequency	Domain
ETTh1	7	96, 192, 336, 720	8545/2881/2881	Hourly	Health
ETTh2	7	96, 192, 336, 720	8545/2881/2881	Hourly	Health
ETTm1	7	96, 192, 336, 720	34465/11521/11521	15min	Health
ETTm2	7	96, 192, 336, 720	34465/11521/11521	15min	Health
Weather	21	96, 192, 336, 720	36792/5271/10540	10min	Weather
ECL	321	96, 192, 336, 720	18317/2633/5261	Hourly	Electricity
Traffic	862	96, 192, 336, 720	12185/1757/3509	Hourly	Transportation
Solar-Energy	137	96, 192, 336, 720	36601/5161/10417	10min	Energy
PEMS03	358	12, 24, 36, 48	15617/5135/5135	5min	Transportation
PEMS04	307	12, 24, 48, 96	10172/3375/281	5min	Transportation
PEMS07	883	12, 24, 48, 96	16911/5622/468	5min	Transportation
PEMS08	170	12, 24, 36, 48	10690/3548/265	5min	Transportation

Note: *D* denotes the number of variates. *Frequency* denotes the sampling interval of time points. *Train, Validation, Test* denotes the number of samples employed in each split. The taxonomy aligns with [26].

This factorises into node and edge potentials and is therefore a pairwise Gibbs Markov random field with respect to the same graph, but it is not Gaussian, and we consequently restrict the precision-matrix reading to the ℓ_2 case. We also do not claim that $\mathcal{E}$ recovers a sparse precision support: the default topology is complete, with random edge sampling used only for scalability. The ℓ_1 form is adopted because squared penalties can be dominated by a small number of large edge-wise discrepancies arising from localized anomalies or variable-specific shocks. This is a task-specific empirical hypothesis about time-series residuals rather than a universal claim. A comparable separation between the norm used for analysis and the norm used in practice appears in FreDF [7], whose analysis is ℓ_2/Gaussian while its deployed frequency-domain loss is ℓ_1. Appendix C.4 tests the two norms directly.

B Reproduction Details

B.1 Dataset descriptions

Our empirical evaluation is conducted on a diverse collection of widely-used time series forecasting benchmarks. Each dataset presents distinct characteristics in terms of dimensionality and temporal resolution. A summary is provided in Table 7.

- **ETT** [45]: This dataset includes seven electricity transformer indicators logged between July 2016 and July 2018. It is partitioned into four subsets according to sampling rates: ETTh1 and ETTh2 for hourly data, and ETTm1 and ETTm2 for 15-minute intervals.
- **Weather** [46]: Encompasses 21 meteorological factors collected every 10 minutes throughout the year 2020 from the Max Planck Institute for Biogeochemistry.
- **ECL** [46]: Records the hourly power usage patterns of 321 individual customers.
- **Traffic** [46]: Monitors hourly road occupancy via 862 sensors located on freeways in the San Francisco Bay Area from 2015 to 2016.
- **Solar-Energy**[1]: 10-minute solar power production records from 137 photovoltaic plants throughout 2006.
- **PEMS** [47]: Derived from public California highway traffic records, aggregated at 5-minute steps. We adopt two standard subsets, PEMS03 and PEMS08, for our study.

[1] `https://www.nrel.gov/grid/solar-power-data.html`

Why these datasets. The twelve benchmarks above are the standard suite of [26, 1] and were selected before any results were obtained; we record the reasoning here because the choice interacts with what CvLoss is designed to do. Three considerations apply. First, we require coverage of the variable-count range, since CvLoss regularizes interactions between variables and its cost and its benefit both depend on D: the suite spans $D = 7$ (the four ETT subsets) to $D = 862$ (Traffic), with Weather (21), Solar (137), PEMS08 (170), PEMS04 (307), ECL (321), PEMS03 (358) and PEMS07 (883) in between. Second, we require coverage of the cross-variable correlation range, which Theorem 3.1 predicts should govern the size of the effect; the suite spans weakly correlated channels (ETT, average CCC 0.14–0.18) to strongly correlated ones (Solar, 0.91; PEMS07, 0.78), as reported in Table 20. Third, both the long-horizon regime ($T \in \{96, 192, 336, 720\}$) and the short-horizon regime used for PEMS ($T \in \{12, 24, 36, 48\}$) are represented, so the conclusions are not specific to one horizon scale. All four ETT subsets are included, at both the hourly (ETTh1, ETTh2) and the 15-minute (ETTm1, ETTm2) sampling rates.

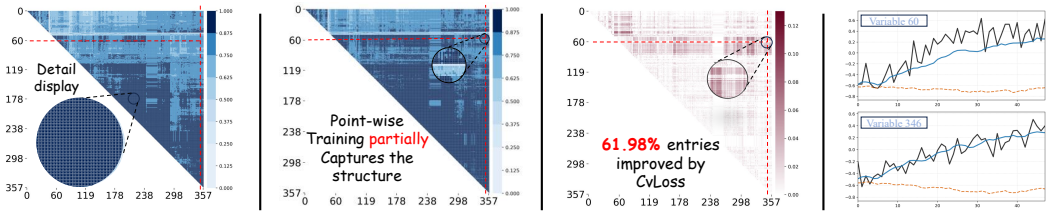
We deliberately exclude the Exchange dataset, following the protocol of [4]. Exchange is close to a driftless random walk, a property documented since [47], and on such a series the trivial predictor that repeats the last observed value is competitive with deep forecasters. A benchmark on which the degenerate baseline is already near-optimal cannot discriminate between learning objectives, which is the comparison this paper makes. The point is compounded for our specific claim: Exchange consists of eight loosely coupled national exchange rates, so it carries little of the cross-variable structure that CvLoss is designed to exploit, and including it would neither support nor challenge the mechanism under test. We note the exclusion explicitly rather than silently, since the omission is a choice about what the evidence can show.

Following established protocols [40, 1], all datasets are segmented chronologically into training, validation, and testing portions. Regarding the ETT, Weather, ECL, Solar and Traffic datasets, we maintain a constant input window of 96 time steps and assess model performance across four distinct forecasting horizons: 96, 192, 336, and 720. For the PEMS datasets, while the historical look-back period remains 96, the evaluation focuses on shorter prediction intervals of 12, 24, 36, and 48 steps. Throughout our experimental phase, we disable the *dropping-last trick* during the final test set assessment to ensure that every data point in the terminal batch is fully utilized.

B.2 Implementation details of model training

To ensure a strictly fair comparison, we build all experiments on top of publicly released benchmark implementations and retain the original model architectures and hyperparameter configurations. When integrating CvLoss, we do not modify the forecasting backbone, data preprocessing, optimizer, training schedule, or evaluation protocol; only the proposed objective term and its associated weight are added. We also disable the drop-last trick for all methods following [40], so that every test sample is evaluated consistently. To establish a fair comparison, we reproduced all baseline models using their official, publicly available implementations, primarily sourcing from the TQNet [35], TimeFilter [4], iTransformer [1], TimeBridge [48] and Time-Series-Library [10] repositories. The reproducibility of these baseline results was verified prior to our experiments. All models were trained to minimize the MSE loss function using the Adam optimizer [41]. All baselines apply to the publicly available parameters in their library without any changes. To prevent overfitting, we employed an early stopping mechanism that terminates training if the validation loss fails to improve for three consecutive epochs.

What is tuned, and on which split. Two quantities are tuned for CvLoss: the loss weight α and the patch length L. Nothing else is. Both are selected on the *validation* split and never on the test split, and all methods, baselines included, use early stopping on the validation loss with a patience of three epochs. Every number reported in the main text and in the appendix uses the fixed convex objective $\mathcal{L}_\alpha = (1-\alpha)\mathcal{L}_{\mathrm{df}} + \alpha\mathcal{L}_{\mathrm{cv}}$ with a single scalar $\alpha \in (0, 1)$ chosen this way; α is never learned during training, and there is no configuration in which it varies across batches, variables or edges. The sensitivity tables report the full sweep over α, whereas the main tables report the validation-selected setting; the two are consistent by construction, since the $\alpha = 0$ rows of Tables 11 and 12 reproduce the unregularised baselines exactly. The separate quantities in Table 20 are a diagnostic and are *not* values of α; this is stated with that table.



(a) Ground-truth structure. (b) Baseline structure. (c) Improvement by CvLoss. (d) PEMS03 Snapshot.

Figure 5: Motivating example on PEMS03 with input length $H = 96$ and forecast horizon $T = 48$. (a,b) Cross-variable correlation structures of the ground truth and the plain iTransformer [1] prediction. (c) Entries where CvLoss reduces structural error over the baseline, covering 61.98% of cross-variable pairs. (d) Forecast snapshot: orange denotes the baseline prediction, black denotes the ground truth, and blue denotes ours (CvLoss). See details in Appendix B.3.

Baselines are tuned at least as favourably as our method. This deserves a precise statement, because a plug-in objective can appear to win simply by receiving more tuning effort than the models it is added to. The comparison here is arranged so that the opposite holds. Baselines are the authors' own public implementations, run with the hyperparameters published in their libraries and verified to reproduce beforehand; we change none of them. When CvLoss is added, the backbone, data preprocessing, normalisation, optimizer, learning-rate schedule, batch size, early-stopping rule and evaluation protocol are all left exactly as the baseline defines them, and only the additional objective term and its weight are introduced. Our method therefore has strictly less freedom than the models it is compared against: two scalars against a fully tuned configuration. The drop-last trick is disabled for every model, ours included, following [40], so no method benefits from a truncated final test batch.

No ground-truth future statistics enter the objective. In every configuration used for the reported results, the edge set $\mathcal{E}$ is fixed a priori, since it is the complete cross-variable graph of Section 3.2, and therefore depends on no property of the future window. The quantity CvLoss consumes is the residual $E = \hat{Z} - Z$, which is the same supervision signal that the standard DF objective already consumes in Eq. (1); no additional statistic of Y is used. The one variant that inspects the ground-truth correlation matrix $\mathbf{C}(Y)$ is the top-K edge selection of Appendix D, which is used *during training only* to concentrate computation on the worst-aligned variable pairs, and which produced none of the numbers in this paper. In all configurations the CvLoss term leaves the computation graph at inference: it contributes no parameters, no buffers and no forward-pass computation to the deployed model, which is why the inference latency in Figure 11 sits at parity with the unmodified backbone.

B.3 Additional Visualization of Cross-Variable Correlation Structures

To further examine how CvLoss affects the recovery of cross-variable dependencies, we provide additional visualizations on the ETTh2 and Weather datasets. These cases follow the same qualitative protocol as Figure 1, while using input length $H = 96$ and forecast horizon $T = 96$.

Let $Y \in \mathbb{R}^{T\times D}$ denote the ground-truth future series and let $\hat{Y}^{\text{base}}, \hat{Y}^{\text{cv}} \in \mathbb{R}^{T\times D}$ denote the future series predicted by the baseline iTransformer and by iTransformer trained with CvLoss, respectively. For any future series $S \in \mathbb{R}^{T\times D}$, we define its cross-variable correlation structure as

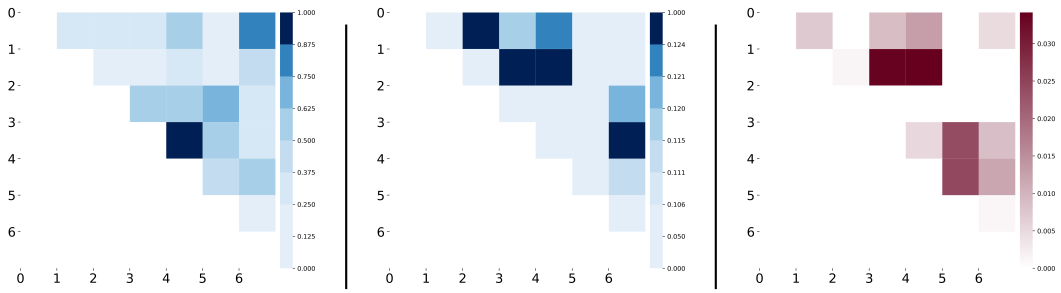
$$C(S) = \big[\,\big|\mathrm{Corr}(S_{:,i}, S_{:,j})\big|\,\big]_{i,j=1}^{D} \in \mathbb{R}^{D\times D},$$

where $\mathrm{Corr}(\cdot,\cdot)$ denotes the Pearson correlation coefficient. Accordingly, we visualize

$$C^{\text{gt}} = C(Y), \qquad C^{\text{base}} = C(\hat{Y}^{\text{base}}).$$

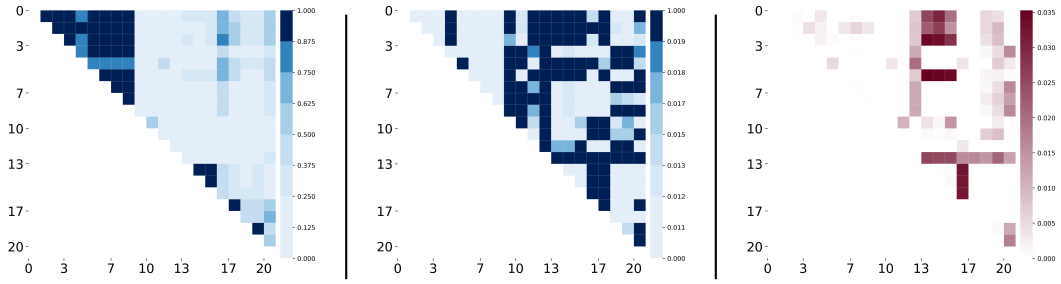
To quantify structural deviation from the ground-truth pattern, we define the structural error matrices as

$$\mathrm{SE}^{\text{base}} = \big|C^{\text{base}} - C^{\text{gt}}\big|, \qquad \mathrm{SE}^{\text{cv}} = \big|C(\hat{Y}^{\text{cv}}) - C^{\text{gt}}\big|,$$



(a) Ground-truth structure. (b) Baseline structure. (c) Improvement by CvLoss.

Figure 6: The cross-variable correlation structures of future series given X on the ETTh2 dataset, with input length $H = 96$ and forecast horizon $T = 96$. The correlation matrices are computed from the ground-truth future series (a) and the future series predicted by iTransformer [1] (b), while (c) shows the entries on which CvLoss yields smaller structural errors than iTransformer (62%).



(a) Ground-truth structure. (b) Baseline structure. (c) Improvement by CvLoss.

Figure 7: The cross-variable correlation structures of future series given X on the Weather dataset, with input length $H = 96$ and forecast horizon $T = 96$. The correlation matrices are computed from the ground-truth future series (a) and the future series predicted by iTransformer [1] (b), while (c) shows the entries on which CvLoss yields smaller structural errors than iTransformer(44%).

where the absolute value is taken element-wise. The improvement induced by CvLoss is then measured by

$$\Delta = \mathrm{SE}^{\mathrm{base}} - \mathrm{SE}^{\mathrm{cv}}.$$

A positive entry $\Delta_{ij} > 0$ means that CvLoss produces a smaller structural error than the baseline on the (i, j)-th cross-variable entry, i.e., the recovered correlation is closer to the ground-truth structure. Therefore, panel (c) visualizes the positive part of Δ, highlighting where CvLoss improves over the baseline in recovering cross-variable dependencies.

These supplementary cases complement the ECL example of Figure 1 in the main text and the PEMS03 example of Figure 5, and verify that the structural benefit of CvLoss is not restricted to a single dataset. As shown in Figures 6 and 7, CvLoss consistently improves the recovery of cross-variable correlation structures across different data regimes, including both low-dimensional and higher-dimensional multivariate forecasting benchmarks. This observation is consistent with our main claim that CvLoss acts as a structural regularizer on multivariate outputs and improves the preservation of cross-variable dependencies beyond point-wise supervision alone.

C More Experimental Results

C.1 Overall performance

Additional overall results are reported in Tables 8 and 9, where we provide performance under different forecast horizons.

Table 8: The comprehensive results on the long-term forecasting task.

Models		CvLoss (Ours)		TimeFilter (2025)		TQNet (2025)		iTransformer (2024)		Leddam (2024)		SOFTS (2023)		PatchTST (2023)		Crossformer (2023)		TimesNet (2023)		DLinear (2023)		FEDformer (2022)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.305**	**0.339**	0.313	0.354	0.311	0.353	0.334	0.368	0.319	0.359	0.325	0.361	0.329	0.367	0.404	0.426	0.338	0.375	0.346	0.374	0.379	0.419
	192	**0.353**	**0.367**	0.356	0.380	0.357	0.378	0.377	0.391	0.369	0.383	0.375	0.389	0.367	0.385	0.450	0.451	0.374	0.387	0.382	0.391	0.426	0.441
	336	**0.381**	**0.388**	0.386	0.403	0.390	0.401	0.426	0.420	0.394	0.402	0.405	0.412	0.399	0.410	0.532	0.515	0.410	0.411	0.415	0.415	0.445	0.459
	720	0.450	**0.430**	0.452	0.437	**0.449**	0.439	0.491	0.459	0.460	0.442	0.466	0.447	0.454	0.439	0.666	0.589	0.478	0.450	0.473	0.451	0.543	0.490
	Avg	**0.372**	**0.381**	0.377	0.394	0.377	0.393	0.407	0.410	0.386	0.397	0.393	0.403	0.387	0.400	0.513	0.496	0.400	0.406	0.404	0.408	0.448	0.452
ETTm2	96	**0.167**	**0.251**	0.169	0.255	0.173	0.255	0.180	0.264	0.176	0.257	0.180	0.261	0.175	0.259	0.287	0.366	0.187	0.267	0.193	0.293	0.203	0.287
	192	**0.232**	**0.294**	0.235	0.299	0.243	0.300	0.250	0.309	0.243	0.303	0.246	0.306	0.241	0.302	0.414	0.492	0.249	0.309	0.284	0.361	0.269	0.328
	336	**0.290**	**0.332**	0.293	0.336	0.301	0.340	0.311	0.348	0.303	0.341	0.319	0.352	0.305	0.343	0.597	0.542	0.321	0.351	0.382	0.429	0.325	0.366
	720	**0.389**	**0.391**	0.390	0.393	0.395	0.394	0.412	0.407	0.400	0.398	0.405	0.401	0.402	0.400	1.730	1.042	0.408	0.403	0.558	0.525	0.421	0.415
	Avg	**0.270**	**0.317**	0.272	0.321	0.278	0.322	0.288	0.332	0.281	0.325	0.287	0.330	0.281	0.326	0.757	0.610	0.291	0.333	0.354	0.402	0.305	0.349
ETTh1	96	**0.369**	**0.391**	0.370	0.394	0.371	0.393	0.386	0.405	0.377	0.394	0.381	0.399	0.414	0.419	0.423	0.448	0.384	0.402	0.397	0.412	0.376	0.419
	192	**0.412**	**0.420**	0.413	**0.420**	0.428	0.426	0.441	0.436	0.424	0.422	0.435	0.431	0.460	0.445	0.471	0.474	0.436	0.429	0.446	0.441	0.420	0.448
	336	**0.447**	**0.439**	0.450	0.440	0.475	0.446	0.487	0.458	0.459	0.442	0.480	0.452	0.501	0.466	0.570	0.546	0.491	0.469	0.489	0.467	0.459	0.465
	720	**0.447**	**0.457**	0.448	**0.457**	0.488	0.470	0.503	0.491	0.463	0.459	0.499	0.488	0.500	0.488	0.653	0.621	0.521	0.500	0.513	0.510	0.506	0.507
	Avg	**0.419**	**0.427**	0.420	0.428	0.441	0.434	0.454	0.447	0.431	0.429	0.449	0.442	0.469	0.454	0.529	0.522	0.458	0.450	0.461	0.457	0.440	0.460
ETTh2	96	**0.282**	**0.334**	0.283	0.338	0.294	0.344	0.297	0.349	0.292	0.343	0.297	0.347	0.302	0.348	0.745	0.584	0.340	0.374	0.340	0.394	0.358	0.397
	192	**0.356**	**0.383**	0.362	0.392	0.368	0.393	0.380	0.400	0.367	0.389	0.373	0.394	0.388	0.400	0.877	0.656	0.402	0.414	0.482	0.479	0.429	0.439
	336	**0.401**	**0.419**	0.409	0.431	0.417	0.428	0.428	0.432	0.412	0.424	0.410	0.426	0.426	0.433	1.043	0.731	0.452	0.452	0.591	0.541	0.496	0.487
	720	**0.397**	**0.426**	0.408	0.434	0.434	0.446	0.427	0.445	0.419	0.438	0.411	0.433	0.431	0.446	1.104	0.763	0.462	0.468	0.839	0.661	0.463	0.474
	Avg	**0.359**	**0.391**	0.366	0.399	0.378	0.403	0.383	0.407	0.373	0.399	0.385	0.408	0.387	0.407	0.942	0.684	0.414	0.427	0.563	0.519	0.437	0.449
Weather	96	**0.149**	**0.188**	0.153	0.199	0.156	0.200	0.174	0.214	0.156	0.202	0.166	0.208	0.177	0.218	0.158	0.230	0.172	0.220	0.195	0.252	0.217	0.296
	192	**0.198**	**0.236**	0.202	0.246	0.205	0.245	0.221	0.254	0.207	0.250	0.217	0.253	0.225	0.259	0.206	0.277	0.219	0.261	0.237	0.295	0.276	0.336
	336	**0.257**	**0.283**	0.260	0.289	0.262	0.287	0.278	0.296	0.262	0.291	0.282	0.300	0.278	0.297	0.272	0.335	0.280	0.306	0.282	0.331	0.339	0.380
	720	**0.340**	**0.334**	0.345	0.344	0.343	0.342	0.358	0.349	0.343	0.343	0.356	0.351	0.354	0.348	0.398	0.418	0.365	0.359	0.345	0.382	0.403	0.428
	Avg	**0.236**	**0.260**	0.240	0.270	0.242	0.268	0.258	0.279	0.242	0.272	0.255	0.278	0.259	0.281	0.259	0.315	0.259	0.287	0.265	0.315	0.309	0.360
Electricity	96	**0.131**	**0.225**	0.133	0.230	0.134	0.229	0.148	0.240	0.141	0.235	0.143	0.233	0.195	0.285	0.219	0.314	0.168	0.272	0.210	0.302	0.193	0.308
	192	**0.152**	**0.243**	0.154	0.248	0.154	0.247	0.162	0.253	0.159	0.252	0.158	0.248	0.199	0.289	0.231	0.322	0.184	0.289	0.210	0.305	0.201	0.315
	336	**0.162**	**0.257**	0.164	0.262	0.170	0.265	0.178	0.269	0.173	0.268	0.178	0.269	0.215	0.305	0.246	0.337	0.198	0.300	0.223	0.319	0.214	0.329
	720	**0.182**	**0.282**	0.184	0.284	0.200	0.293	0.225	0.317	0.201	0.295	0.218	0.305	0.256	0.337	0.280	0.363	0.220	0.320	0.258	0.350	0.246	0.355
	Avg	**0.157**	**0.252**	0.159	0.256	0.164	0.258	0.178	0.270	0.169	0.263	0.174	0.264	0.216	0.304	0.244	0.334	0.193	0.295	0.225	0.319	0.214	0.327
Traffic	96	**0.372**	**0.238**	0.375	0.251	0.413	0.261	0.395	0.268	0.426	0.276	0.376	0.251	0.544	0.359	0.522	0.290	0.593	0.321	0.650	0.396	0.587	0.366
	192	**0.395**	**0.248**	0.396	0.262	0.432	0.271	0.417	0.276	0.458	0.289	0.398	0.261	0.540	0.354	0.530	0.293	0.617	0.336	0.598	0.370	0.604	0.373
	336	**0.413**	**0.256**	0.416	0.271	0.450	0.277	0.433	0.283	0.486	0.297	0.415	0.269	0.551	0.358	0.558	0.305	0.629	0.336	0.605	0.373	0.621	0.383
	720	**0.446**	**0.275**	**0.446**	0.290	0.486	0.295	0.467	0.302	0.498	0.313	0.447	0.287	0.586	0.375	0.589	0.328	0.640	0.350	0.645	0.394	0.626	0.382
	Avg	**0.407**	**0.254**	0.408	0.269	0.445	0.276	0.428	0.282	0.467	0.294	0.409	0.267	0.555	0.362	0.550	0.304	0.620	0.336	0.625	0.383	0.610	0.376
Solar-Energy	96	0.183	**0.229**	0.199	0.239	**0.175**	0.234	0.203	0.237	0.197	0.241	0.200	0.230	0.234	0.286	0.310	0.331	0.250	0.292	0.290	0.378	0.242	0.342
	192	0.215	**0.250**	0.227	0.261	**0.198**	0.255	0.233	0.261	0.231	0.263	0.229	0.253	0.267	0.310	0.734	0.725	0.296	0.318	0.320	0.398	0.285	0.380
	336	0.232	0.267	0.240	0.272	**0.205**	**0.262**	0.248	0.273	0.241	0.268	0.243	0.269	0.290	0.315	0.750	0.735	0.319	0.330	0.353	0.415	0.282	0.376
	720	0.241	**0.270**	0.247	0.277	**0.210**	0.270	0.249	0.275	0.250	0.281	0.245	0.272	0.289	0.317	0.769	0.765	0.338	0.337	0.357	0.413	0.357	0.427
	Avg	0.218	**0.254**	0.228	0.262	**0.197**	0.255	0.233	0.262	0.230	0.264	0.229	0.256	0.270	0.307	0.641	0.639	0.301	0.319	0.330	0.401	0.292	0.381
1st Count		**34**	**39**	1	2	6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: We fix the input length as 96 following [1]. **Bold** typeface highlights the top performance for each metric, while underlined text denotes the second-best results. *Avg* indicates the results averaged over forecasting lengths: T=96, 192, 336 and 720.

C.2 Patch length sensitivity

Additional results for the patch length hyperparameter are reported in Figure 8, covering the ETTh2 and ETTm2 datasets under the forecast horizons $T = 192$ and $T = 336$.

C.3 Comparison with different learning objectives

Baselines. Since this paper focuses on the design of learning objectives, we select competitive learning objectives tailored for training forecast models as baselines. For the comprehensive evaluation in Table 14, we compare against 11 objectives, which can be categorized as follows: ❶ **shape-alignment objectives:** GDTW [12], Dilate [13], and Soft-DTW [14]; ❷ **likelihood maximization objectives:** QDF [15], Time-o1 [16], Koopman [17], FreDF [7], and MSE (also denoted as DF); ❸ **distribution balancing objectives:** KMB-DF [18], and DistDF [19]; and ❹ **decomposition-based objectives:** DBLoss [20]. For the cross-model evaluation in Table 13, we select a representative subset of 7 objectives: QDF, Time-o1, FreDF, Koopman, Soft-DTW, DF, and DBLoss. The implementation of baselines follows the official codebase from [18].

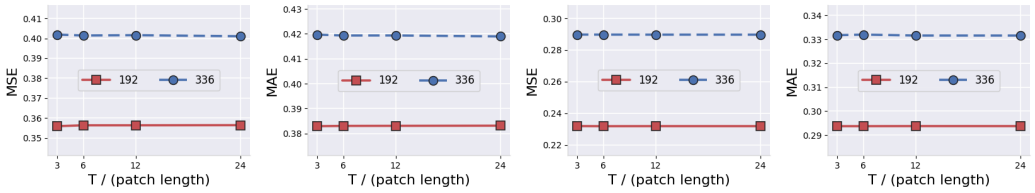
Table 9: Full results of short-term forecasting.

Models		CvLoss (Ours)		TimeFilter (2025)		TQNet (2025)		iTransformer (2024)		Leddam (2024)		SOFTS (2023)		PatchTST (2023)		Crossformer (2023)		TimesNet (2023)		DLinear (2023)		FEDformer (2022)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
PEMS03	12	**0.060**	**0.160**	0.063	0.165	0.060	0.161	0.071	0.174	0.068	0.174	0.064	0.165	0.099	0.216	0.090	0.203	0.085	0.192	0.122	0.243	0.126	0.251
	24	**0.075**	**0.180**	0.079	0.185	0.077	0.183	0.093	0.201	0.094	0.202	0.083	0.188	0.142	0.259	0.121	0.240	0.118	0.223	0.201	0.317	0.149	0.275
	48	**0.101**	**0.210**	0.110	0.222	0.103	0.213	0.125	0.236	0.140	0.254	0.114	0.223	0.211	0.319	0.202	0.317	0.155	0.260	0.333	0.425	0.227	0.348
	Avg	**0.079**	**0.183**	0.084	0.191	0.080	0.186	0.096	0.204	0.101	0.210	0.087	0.192	0.151	0.265	0.138	0.253	0.119	0.271	0.219	0.295	0.167	0.291
PEMS04	12	**0.066**	**0.164**	0.068	0.167	0.067	0.166	0.078	0.183	0.076	0.182	0.074	0.176	0.105	0.224	0.098	0.218	0.087	0.195	0.148	0.272	0.138	0.262
	24	**0.075**	**0.178**	0.080	0.183	0.077	0.181	0.095	0.205	0.097	0.209	0.088	0.194	0.153	0.257	0.131	0.256	0.103	0.215	0.224	0.340	0.177	0.293
	48	**0.094**	**0.201**	0.101	0.209	0.098	0.207	0.120	0.233	0.132	0.249	0.110	0.219	0.229	0.339	0.205	0.326	0.136	0.250	0.335	0.437	0.270	0.368
	Avg	**0.078**	**0.181**	0.083	0.186	0.081	0.185	0.098	0.207	0.102	0.213	0.091	0.196	0.162	0.273	0.145	0.267	0.109	0.220	0.236	0.350	0.195	0.308
PEMS07	12	**0.051**	**0.140**	0.055	0.150	0.052	0.144	0.067	0.165	0.066	0.164	0.057	0.152	0.095	0.207	0.094	0.200	0.082	0.181	0.115	0.242	0.109	0.225
	24	**0.061**	**0.155**	0.068	0.166	0.063	0.159	0.088	0.190	0.079	0.185	0.073	0.173	0.150	0.262	0.139	0.247	0.101	0.204	0.210	0.329	0.125	0.244
	48	**0.077**	**0.174**	0.089	0.193	0.080	0.180	0.110	0.215	0.115	0.228	0.096	0.195	0.253	0.340	0.311	0.369	0.134	0.238	0.398	0.458	0.165	0.288
	Avg	**0.063**	**0.156**	0.071	0.170	0.065	0.161	0.088	0.190	0.087	0.192	0.075	0.173	0.166	0.270	0.181	0.272	0.106	0.208	0.241	0.343	0.133	0.282
PEMS08	12	**0.062**	**0.159**	0.064	0.162	0.071	0.170	0.079	0.182	0.070	0.173	0.074	0.171	0.168	0.232	0.165	0.214	0.112	0.212	0.227	0.343	0.173	0.273
	24	**0.076**	**0.178**	0.079	0.182	0.095	0.195	0.115	0.219	0.091	0.200	0.104	0.201	0.224	0.281	0.215	0.260	0.141	0.238	0.318	0.409	0.210	0.310
	48	**0.101**	**0.210**	0.105	0.214	0.149	0.243	0.186	0.235	0.145	0.261	0.164	0.253	0.321	0.354	0.315	0.335	0.198	0.283	0.497	0.510	0.320	0.394
	Avg	**0.080**	**0.182**	0.083	0.186	0.105	0.203	0.127	0.212	0.102	0.211	0.114	0.208	0.238	0.289	0.232	0.270	0.150	0.244	0.347	0.421	0.234	0.326

Note: The input sequence length is set to 96 for all baselines. All results are averaged across four different forecasting horizons: $T \in \{12, 24, 48\}$. The best and second-best results are highlighted in **bold** and underlined, respectively.

Table 10: Backbone-controlled comparison on four forecasting backbones: each pair contrasts the unmodified model with the identical model trained with CvLoss. The *Wins* row counts, for each backbone, the horizon–dataset–metric cells on which CvLoss is better, out of $3 \times 4 \times 2 = 24$. These counts are produced by the evaluation script from the *unrounded* outputs, not by comparing the three-decimal values printed here: seven cells are equal at the displayed precision but are improvements at the fourth decimal, so the totals of $24/24$ per backbone and $96/96$ overall are exact. We keep three decimals for consistency with the other tables rather than widening the display.

Method		TQNet		**+ Ours**		TimeFilter		**+ Ours**		CFPT		**+ Ours**		iTransformer		**+ Ours**	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh2	96	0.294	0.344	**0.281**	**0.332**	0.283	0.337	**0.282**	**0.334**	0.287	0.337	**0.283**	**0.333**	0.297	0.349	**0.296**	**0.342**
	192	0.368	0.393	**0.358**	**0.380**	0.362	0.392	**0.356**	**0.383**	0.368	0.390	**0.361**	**0.383**	0.380	0.400	**0.374**	**0.391**
	336	0.417	0.428	**0.404**	**0.419**	0.404	0.424	**0.401**	**0.419**	0.416	0.428	**0.409**	**0.420**	0.428	0.432	**0.415**	**0.427**
	720	0.434	0.446	**0.413**	**0.433**	0.407	0.433	**0.397**	**0.426**	0.415	0.436	**0.392**	**0.418**	0.427	0.445	**0.421**	**0.438**
	Avg	0.378	0.403	**0.364**	**0.391**	0.364	0.397	**0.359**	**0.391**	0.372	0.398	**0.361**	**0.389**	0.383	0.407	**0.377**	**0.399**
Weather	96	0.156	0.200	**0.154**	**0.192**	0.153	0.199	**0.149**	**0.188**	0.156	0.201	**0.152**	**0.192**	0.174	0.214	**0.168**	**0.207**
	192	0.205	0.245	**0.203**	**0.238**	0.202	0.246	**0.198**	**0.236**	0.205	0.245	**0.203**	**0.240**	0.221	0.254	**0.221**	**0.254**
	336	0.262	0.287	**0.262**	**0.285**	0.260	0.289	**0.257**	**0.283**	0.261	0.286	**0.260**	**0.281**	0.278	0.296	**0.278**	**0.296**
	720	0.343	0.342	**0.343**	**0.339**	0.342	0.341	**0.340**	**0.334**	0.343	0.339	**0.340**	**0.336**	0.358	0.349	**0.357**	**0.347**
	Avg	0.242	0.268	**0.241**	**0.264**	0.239	0.269	**0.236**	**0.260**	0.241	0.267	**0.239**	**0.262**	0.258	0.279	**0.256**	**0.276**
ECL	96	0.134	0.229	**0.133**	**0.225**	0.133	0.230	**0.131**	**0.225**	0.137	0.232	**0.136**	**0.227**	0.148	0.240	**0.146**	**0.235**
	192	0.154	0.247	**0.151**	**0.241**	0.154	0.248	**0.152**	**0.243**	0.153	0.247	**0.153**	**0.242**	0.162	0.253	**0.161**	**0.249**
	336	0.170	0.265	**0.166**	**0.258**	0.164	0.261	**0.162**	**0.257**	0.168	0.266	**0.166**	**0.259**	0.178	0.269	**0.174**	**0.264**
	720	0.200	0.293	**0.196**	**0.287**	0.184	0.284	**0.182**	**0.282**	0.199	0.294	**0.196**	**0.286**	0.225	0.317	**0.205**	**0.292**
	Avg	0.164	0.258	**0.162**	**0.253**	0.158	0.256	**0.157**	**0.252**	0.165	0.260	**0.163**	**0.254**	0.178	0.270	**0.172**	**0.260**
Wins		0		**24**		0		**24**		0		**24**		0		**24**	



(a) ETTh2 with MSE (b) ETTh2 with MAE (c) ETTm2 with MSE (d) ETTm2 with MAE

Figure 8: Sensitivity analysis of the patch length hyperparameter. The plots illustrate the forecasting performance in terms of MSE and MAE on the ETTh2 (a, b) and ETTm2 (c, d) datasets. The x-axis represents the ratio of the forecast horizon to the patch length (T / patch length), while the lines denote different forecast horizons ($T = 192$ and $T = 336$). The nearly horizontal curves demonstrate that the predictive accuracy remains consistent regardless of the specific patch length chosen, indicating that CvLoss is highly robust to this hyperparameter.

Table 11: Varying α results of iTransformer

α	ETTh2		ECL		Weather	
	MSE	MAE	MSE	MAE	MSE	MAE
0	0.384	0.407	0.176	0.267	0.259	0.280
0.001	0.384	0.407	0.176	0.267	0.260	0.281
0.002	0.384	0.407	0.176	0.267	0.260	0.281
0.005	0.384	0.407	0.176	0.267	0.259	0.280
0.01	0.384	0.407	0.176	0.267	0.262	0.282
0.02	0.384	0.407	0.176	0.267	0.261	0.281
0.05	0.384	0.407	0.176	0.267	0.262	0.282
0.1	0.383	0.406	0.175	0.266	0.259	0.280
0.2	0.383	0.406	0.174	0.264	0.259	0.279
0.5	0.379	0.403	**0.172**	**0.260**	**0.257**	0.276
1	**0.378**	**0.399**	0.179	0.265	0.258	**0.276**

Note: **Bold** and underlined denote the best and second-best results.

Table 12: Varying α results of TimeBridge.

α	ETTh2		ECL		Weather	
	MSE	MAE	MSE	MAE	MSE	MAE
0	0.366	0.408	0.159	0.258	0.221	0.261
0.001	0.366	0.408	0.161	0.259	0.221	0.261
0.002	0.366	0.408	0.156	0.255	0.221	0.261
0.005	0.366	0.408	0.159	0.258	0.221	0.261
0.01	0.366	0.408	0.157	0.257	0.221	0.261
0.02	0.366	0.408	0.157	0.257	0.221	0.261
0.05	0.365	0.407	0.157	0.255	0.221	0.261
0.1	0.363	0.406	0.156	0.254	0.220	0.260
0.2	0.360	0.403	0.158	0.255	0.220	0.258
0.5	0.351	0.396	**0.153**	0.250	**0.220**	0.255
1	**0.342**	**0.386**	0.155	**0.248**	0.220	**0.251**

Note: **Bold** and underlined denote the best and second-best results.

Implementation. To thoroughly assess the effectiveness of our proposed objective, we evaluate it under two distinct experimental settings. In the first setting (Table 14), we employ CFPT [43] as the default forecast model to compare all 10 learning objectives across six standard benchmarks (ETTh1, ETTh2, ETTm1, ETTm2, Weather, and ECL). In the second setting (Table 13), we adopt TQNet and PDF as the backbone models to verify the generalizability of our method across different architectures. This evaluation is conducted on four datasets (ETTh1, ETTm1, ECL, and Weather) against the aforementioned subset of 6 learning objectives. To ensure fair comparison, the drop-last trick is disabled for all baselines, as recommended in [40]. All objectives are trained with the Adam optimizer [41]. When integrating CvLoss to train a forecast model, we retain all hyperparameters from the public benchmarks [43], only tuning α and the segment patch length.

On the weakest competing objectives. Two of the eleven objectives, Koopman [17] and Soft-DTW [14], diverge sharply on some datasets. Koopman reaches 0.499 MSE on ETTm1 with TQNet, and Soft-DTW reaches 4.502 in Table 14, so a reader may reasonably suspect a reproduction failure that would inflate our win counts. It is not one. Our figures reproduce the published numbers of the objective-comparison protocol we adopt: on the PDF backbone, Table 2 of QDF [15] reports Koopman at $0.587/0.485$ on ETTm1 and Soft-DTW at $0.695/0.548$ on ECL and $1.296/0.452$ on Weather, matching ours. These older shape-alignment objectives simply degrade on some multivariate benchmarks, as their own authors' evaluations show. None of our conclusions rests on them: neither ranks first in any setting, so removing both leaves the first-place count of CvLoss in Table 14 unchanged. The objectives that actually compete with CvLoss are QDF, KMB-DF, DBLoss, Time-o1 and FreDF, and they differ from it by a few thousandths.

C.4 Choice of the edge-wise norm: ℓ_1 versus ℓ_2

Proposition 3.2 shows that MSE plus a *squared* graph total-variation penalty is the negative log-likelihood of a Gaussian with a structured precision matrix, whereas the objective we deploy in Eq. (12) uses the ℓ_1 norm on the same edge differences. The theory therefore backs the ℓ_2 form, and the ℓ_1 form is an empirical preference. That asymmetry could hide a substantive weakness. If ℓ_2 were as good, the theory-backed variant would be available at no cost; if ℓ_2 were far worse, the mechanism would not be what the theory describes. We therefore test the two norms directly under an identical protocol, varying nothing but the norm.

Table 15 reports the outcome. Relative to the unregularised base, the ℓ_2 variant improves five of the eight averaged cells, matches the base in two, and slightly degrades Weather/TQNet MSE. Relation-aware coupling is therefore beneficial in its own right, which is the prediction Proposition 3.2 actually makes, and the benefit is not an artefact of the particular norm. The ℓ_1 variant is nonetheless better than ℓ_2 in all eight cells. We read this as consistent with the motivation given in Section 3.2: a squared edge penalty is dominated by the few pairs with the largest structural mismatch, which in these series are typically localized anomalies rather than the systematic co-evolution the term is meant to enforce. The practical conclusion is that the theory identifies the right family and the norm is chosen within it on evidence, not that ℓ_1 is derived.

Table 13: Comparable results with different learning objectives.

Loss		**CvLoss**		**QDF**		Time-o1		FreDF		Koopman		Soft-DTW		DF	
Metrics		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Forecast model:TQNet															
ETTm1	96	**0.305**	**0.343**	0.307	0.349	0.309	0.351	0.314	0.355	0.806	0.578	0.315	0.353	0.311	0.353
	192	**0.352**	**0.373**	**0.352**	0.376	0.353	0.375	0.359	0.378	0.619	0.515	0.360	0.377	0.357	0.378
	336	0.388	0.400	0.383	0.398	0.383	0.398	**0.382**	**0.396**	0.507	0.468	0.398	0.402	0.390	0.401
	720	0.444	**0.430**	**0.441**	0.434	0.444	0.436	0.444	0.432	0.450	0.437	0.476	0.446	0.449	0.439
	Avg	0.373	**0.386**	**0.371**	0.389	0.372	0.390	0.375	0.390	0.595	0.499	0.387	0.394	0.377	0.393
ETTh1	96	0.368	**0.385**	**0.365**	0.389	0.381	0.395	0.369	0.391	0.415	0.425	0.379	0.390	0.371	0.393
	192	0.426	0.422	0.427	**0.421**	0.427	0.424	**0.425**	0.422	0.430	0.422	0.437	0.424	0.428	0.426
	336	0.471	**0.443**	**0.466**	0.449	0.471	0.444	0.467	0.445	0.474	0.445	0.488	0.453	0.475	0.446
	720	0.486	0.469	**0.466**	0.467	0.469	**0.466**	0.468	0.469	0.483	0.474	0.510	0.487	0.488	0.470
	Avg	0.438	**0.430**	**0.431**	0.431	0.437	0.432	0.432	0.432	0.451	0.442	0.453	0.438	0.441	0.434
ECL	96	**0.133**	**0.225**	0.135	0.229	0.136	0.228	0.136	0.228	0.137	0.231	0.162	0.258	0.134	0.229
	192	**0.151**	**0.241**	0.153	0.245	0.154	0.245	0.155	0.245	0.154	0.247	0.446	0.449	0.154	0.247
	336	**0.166**	**0.258**	0.169	0.262	0.171	0.262	0.172	0.263	0.171	0.264	0.912	0.675	0.170	0.265
	720	**0.196**	**0.287**	0.202	0.290	0.208	0.293	0.209	0.293	0.204	0.292	0.971	0.715	0.200	0.293
	Avg	**0.162**	**0.253**	0.165	0.257	0.167	0.257	0.168	0.257	0.166	0.258	0.623	0.524	0.165	0.259
Weather	96	**0.154**	**0.192**	0.158	0.201	0.159	0.201	0.158	0.199	0.223	0.268	0.161	0.202	0.156	0.200
	192	**0.203**	**0.238**	0.207	0.245	0.209	0.246	0.209	0.246	0.269	0.304	0.212	0.247	0.205	0.245
	336	**0.262**	**0.285**	0.263	0.286	0.268	0.290	0.266	0.288	0.291	0.309	0.270	0.289	**0.262**	0.287
	720	**0.342**	**0.339**	**0.342**	**0.339**	0.344	0.341	0.344	0.341	0.346	0.343	0.378	0.365	0.343	0.342
	Avg	**0.241**	**0.264**	0.242	0.268	0.245	0.269	0.244	0.268	0.282	0.306	0.255	0.276	0.242	0.268
Forecast model:PDF															
ETTm1	96	**0.310**	**0.342**	0.320	0.358	0.326	0.361	0.325	0.362	1.051	0.663	0.323	0.362	0.326	0.363
	192	**0.360**	**0.369**	0.361	0.380	0.371	0.386	0.372	0.388	0.420	0.414	0.371	0.388	0.365	0.381
	336	**0.390**	**0.392**	**0.390**	0.401	0.401	0.409	0.399	0.409	0.421	0.415	0.408	0.413	0.397	0.402
	720	**0.451**	**0.430**	**0.451**	0.437	0.448	0.439	0.453	0.443	0.456	0.448	0.480	0.454	0.458	0.437
	Avg	**0.378**	**0.383**	0.381	0.394	0.386	0.399	0.387	0.400	0.587	0.485	0.396	0.404	0.387	0.396
ETTh1	96	**0.370**	**0.390**	0.375	0.391	0.380	0.403	0.373	0.393	0.632	0.533	0.383	0.405	0.388	0.400
	192	**0.419**	**0.419**	0.423	**0.419**	0.422	0.425	0.423	0.426	0.424	0.429	0.430	0.432	0.440	0.428
	336	**0.460**	**0.439**	0.461	**0.439**	0.463	0.441	0.477	0.446	0.456	0.450	0.462	0.453	0.483	0.449
	720	0.479	0.474	0.484	**0.468**	0.485	0.483	**0.475**	0.476	0.476	0.478	0.511	0.496	0.495	0.482
	Avg	**0.432**	**0.429**	0.436	**0.429**	0.438	0.438	0.437	0.435	0.497	0.472	0.447	0.447	0.452	0.440
ECL	96	0.165	**0.246**	0.171	0.257	0.173	0.253	**0.163**	**0.246**	0.194	0.278	0.164	0.250	0.175	0.259
	192	0.174	**0.255**	0.177	0.261	0.181	0.262	0.179	0.261	**0.173**	0.260	0.387	0.410	0.182	0.266
	336	**0.187**	**0.269**	0.192	0.277	0.196	0.282	0.196	0.278	0.189	0.276	0.966	0.698	0.197	0.282
	720	0.230	**0.307**	0.234	0.312	0.229	**0.307**	0.237	0.312	**0.228**	0.310	1.263	0.834	0.237	0.315
	Avg	**0.189**	**0.269**	0.194	0.277	0.195	0.276	0.194	0.274	0.196	0.281	0.695	0.548	0.198	0.281
Weather	96	0.174	**0.211**	0.176	0.218	0.178	0.219	**0.173**	0.216	0.202	0.242	0.178	0.219	0.181	0.221
	192	**0.224**	**0.254**	0.225	0.260	0.236	0.267	0.235	0.268	0.225	0.258	0.232	0.262	0.234	0.263
	336	**0.279**	**0.294**	0.280	0.299	0.284	0.304	0.274	0.295	0.280	0.302	0.281	0.296	0.285	0.300
	720	0.358	**0.345**	0.357	0.347	0.357	0.348	0.356	0.350	**0.353**	0.347	4.502	1.036	0.360	0.348
	Avg	**0.259**	**0.276**	**0.259**	0.281	0.264	0.284	0.268	0.287	0.268	0.290	1.296	0.452	0.265	0.283

C.5 Generalization studies

Additional experimental results of varying forecast models are available in Table 10, Table 13 and Fig 9.

C.6 Hyperparameter Sensitivity

We evaluate the stability of `CvLoss` regarding the loss weight α and patch length L. Loss Weight α: Tables 11 and 12 show accuracy consistently improving as α increases. Optimal performance typically occurs at $\alpha \approx 0.5$ or 1, where structural regularization complements point-wise accuracy. Patch Length L: Figure 8 demonstrates that forecasting performance remains stable across various T/L ratios. The horizontal curves confirm that `CvLoss` is highly robust to patch size, requiring minimal tuning to achieve significant predictive gains.

Table 14: Full results on the multi-step forecasting task. The length of history window is set to 96 for all baselines. Avg indicates the results averaged over forecasting lengths: T=96, 192, 336 and 720 for ETT, ECL, and Weather. Follow the settings of QDF [18]

Loss		**CvLoss (Ours)**		KMB-DF (2026)		DBLoss (2025)		QDF (2025)		DistDF (2025)		Time-o1 (2025)		FreDF (2025)		Koopman (2021)		GDTW (2021)		Dilate (2019)		Soft-DTW (2017)		MSE (2002)	
Metrics		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.309**	**0.342**	0.315	0.352	0.310	0.345	0.318	0.359	0.318	0.355	0.321	0.353	0.320	0.354	0.326	0.363	0.324	0.362	0.323	0.361	0.322	0.359	0.321	0.358
	192	**0.350**	0.369	0.351	0.379	0.352	**0.368**	0.355	0.379	0.356	0.380	0.358	0.380	0.356	0.379	0.359	0.383	0.368	0.392	0.363	0.385	0.366	0.385	0.357	0.382
	336	0.385	**0.392**	**0.381**	0.400	0.391	**0.392**	0.384	0.401	0.383	0.401	0.386	0.403	0.386	0.401	0.383	0.401	0.405	0.421	0.384	0.401	0.399	0.409	0.387	0.401
	720	**0.442**	0.428	**0.442**	0.433	0.447	**0.427**	0.443	0.435	0.444	0.434	0.447	0.436	0.444	0.434	0.446	0.436	0.496	0.473	0.446	0.437	0.490	0.461	0.446	0.435
	Avg	**0.372**	**0.383**	**0.372**	0.391	0.375	**0.383**	0.375	0.393	0.375	0.393	0.378	0.393	0.376	0.392	0.378	0.395	0.398	0.412	0.379	0.396	0.394	0.403	0.378	0.394
ETTm2	96	**0.165**	0.245	0.166	0.249	0.166	**0.244**	0.168	0.250	0.168	0.251	0.173	0.251	0.174	0.251	0.182	0.262	0.176	0.257	0.172	0.253	0.176	0.256	0.175	0.256
	192	0.230	**0.288**	**0.229**	0.291	0.231	**0.288**	0.232	0.292	0.232	0.293	0.233	0.291	0.233	0.290	0.235	0.294	0.243	0.305	0.236	0.296	0.243	0.299	0.234	0.295
	336	**0.287**	**0.326**	0.289	0.330	0.291	0.327	0.288	0.329	0.289	0.330	0.290	0.329	0.290	0.327	0.291	0.331	0.303	0.344	0.290	0.331	0.306	0.342	0.289	0.330
	720	**0.383**	0.385	0.384	0.388	0.386	**0.384**	0.386	0.387	0.385	0.387	0.386	0.385	0.386	0.386	0.388	0.390	0.434	0.419	0.393	0.390	0.438	0.422	0.385	0.389
	Avg	**0.266**	0.311	0.267	0.315	0.268	**0.310**	0.268	0.315	0.269	0.315	0.271	0.314	0.271	0.313	0.274	0.319	0.289	0.332	0.273	0.317	0.291	0.330	0.271	0.317
ETTh1	96	0.371	**0.387**	0.372	0.389	0.370	0.388	0.373	0.392	0.373	0.391	**0.369**	0.392	0.373	0.389	0.374	0.393	0.377	0.395	0.376	0.394	0.377	0.394	0.373	0.391
	192	0.424	**0.418**	**0.423**	0.420	0.424	**0.418**	0.426	0.421	0.426	0.420	0.424	0.424	0.424	**0.418**	0.426	0.424	0.434	0.427	0.430	0.423	0.436	0.427	0.427	0.421
	336	0.467	0.441	**0.460**	**0.438**	0.466	0.439	0.464	0.441	0.468	0.441	0.464	0.439	0.465	0.439	0.466	0.440	0.486	0.457	0.470	0.442	0.479	0.451	0.466	0.441
	720	0.466	0.461	**0.447**	**0.455**	0.468	0.458	0.473	0.464	0.467	0.462	0.462	0.460	0.472	0.468	0.483	0.469	0.512	0.497	0.489	0.478	0.551	0.510	0.477	0.468
	Avg	0.432	0.427	**0.426**	**0.426**	0.432	**0.426**	0.434	0.429	0.434	0.428	0.430	0.429	0.434	0.428	0.437	0.431	0.452	0.444	0.441	0.434	0.461	0.445	0.436	0.430
ETTh2	96	**0.283**	**0.333**	0.285	0.337	0.288	**0.333**	0.288	0.337	0.287	0.337	0.286	0.336	0.289	0.337	0.287	0.338	0.289	0.340	0.287	0.338	0.292	0.342	0.287	0.337
	192	**0.361**	**0.381**	0.363	0.387	0.365	**0.381**	0.367	0.390	0.364	0.391	0.367	0.384	0.365	0.382	0.365	0.390	0.377	0.399	0.368	0.389	0.380	0.398	0.368	0.390
	336	**0.409**	**0.420**	0.411	0.425	0.419	0.426	0.412	0.427	0.412	0.426	0.414	0.427	0.414	0.427	0.413	0.429	0.422	0.432	0.413	0.427	0.455	0.443	0.416	0.428
	720	**0.392**	**0.418**	0.396	0.426	**0.392**	0.419	0.401	0.428	0.398	0.427	0.402	0.428	0.406	0.430	0.405	0.430	0.457	0.463	0.415	0.437	0.445	0.453	0.415	0.436
	Avg	**0.361**	**0.388**	0.364	0.394	0.366	0.390	0.367	0.396	0.365	0.395	0.367	0.394	0.368	0.394	0.368	0.397	0.386	0.409	0.371	0.398	0.393	0.409	0.372	0.398
ECL	96	**0.135**	**0.227**	0.136	0.231	**0.135**	0.228	0.137	0.232	0.137	0.232	0.141	0.235	0.139	0.232	0.138	0.232	0.882	0.777	0.137	0.232	0.136	0.232	0.137	0.232
	192	**0.152**	**0.242**	**0.152**	0.245	**0.152**	0.243	0.153	0.246	**0.152**	0.246	0.154	0.246	0.156	0.246	0.153	0.247	0.853	0.762	0.154	0.247	0.154	0.247	0.153	0.247
	336	0.166	**0.259**	0.168	0.265	**0.165**	0.260	0.168	0.264	0.168	0.265	0.168	0.263	0.169	0.263	0.169	0.267	0.809	0.711	0.169	0.267	0.169	0.267	0.168	0.266
	720	0.195	**0.285**	0.198	0.293	**0.193**	**0.285**	0.199	0.293	0.201	0.294	0.198	0.291	0.199	0.290	0.201	0.296	0.900	0.769	0.202	0.297	0.203	0.298	0.199	0.294
	Avg	**0.162**	**0.253**	0.163	0.258	**0.162**	0.254	0.164	0.259	0.164	0.259	0.165	0.259	0.166	0.258	0.165	0.260	0.861	0.755	0.165	0.260	0.165	0.261	0.165	0.260
Weather	96	**0.152**	**0.192**	**0.152**	0.196	0.154	0.193	0.155	0.200	0.155	0.200	0.153	0.197	0.154	0.197	0.173	0.215	0.158	0.206	0.173	0.215	0.177	0.220	0.156	0.201
	192	**0.203**	**0.238**	0.204	0.242	0.205	0.239	0.205	0.244	0.205	0.245	0.205	0.243	0.204	0.242	0.205	0.245	0.209	0.250	0.205	0.246	0.209	0.245	0.205	0.245
	336	**0.260**	**0.280**	**0.260**	0.284	0.261	**0.280**	0.261	0.285	0.261	0.286	0.261	0.284	0.261	0.284	0.262	0.286	0.270	0.296	0.262	0.287	0.278	0.296	0.261	0.286
	720	0.340	0.336	**0.339**	0.337	**0.339**	**0.333**	0.343	0.338	0.343	0.341	0.343	0.340	0.341	0.338	0.345	0.342	0.354	0.352	0.345	0.342	0.385	0.363	0.343	0.339
	Avg	**0.239**	0.262	**0.239**	0.265	0.240	**0.261**	0.241	0.267	0.241	0.268	0.240	0.266	0.240	0.265	0.246	0.272	0.248	0.276	0.246	0.273	0.262	0.281	0.241	0.267
1st Count		**20**	**20**	13	3	7	16	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0

Table 15: Effect of the edge-wise norm. All settings are identical apart from the penalty: *Base* is the unmodified backbone, $+\ell_2$ replaces $\|\cdot\|_1$ in Eq. (12) by the squared ℓ_2 norm of Proposition 3.2, and $+\ell_1$ is CvLoss as deployed. Values are averaged over $T \in \{96, 192, 336, 720\}$; the best entry in each row and metric is in **bold**.

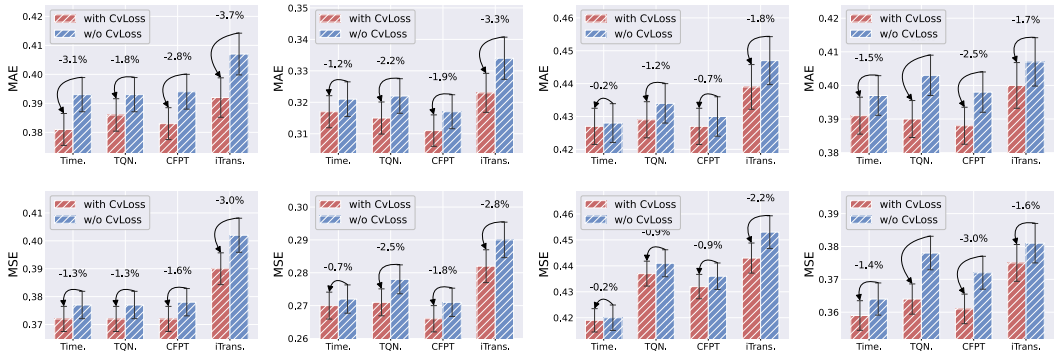
Dataset	Backbone	MSE			MAE		
		Base	$+\ell_2$	$+\ell_1$	Base	$+\ell_2$	$+\ell_1$
ECL	TimeFilter	0.158	0.157	**0.156**	0.256	0.254	**0.252**
ECL	TQNet	0.165	0.164	**0.162**	0.259	0.257	**0.254**
Weather	TimeFilter	0.241	0.241	**0.238**	0.271	0.270	**0.262**
Weather	TQNet	0.242	0.244	**0.241**	0.269	0.269	**0.264**

C.7 Case study with TQNet and TimeFilter of varying historical lengths

Additional experimental results of varying historical lengths are available in Table 16, complementing the fixed length of 96 used in the main text. The forecast models selected include TimeFilter [4], which is the recent state-of-the-art forecast model, and TQNet [35]. The results demonstrate that CvLoss consistently improves both forecast models across different historical sequence lengths.

C.8 Random Seed Sensitivity

Additional experimental results of random seed sensitivity are available in Table 17, where we report the mean and standard deviation of results obtained from experiments conducted with seven different random seeds (2020–2026). The results indicate minimal sensitivity of the proposed method to random initialization, as all standard deviations remain below 0.004.



(a) ETTm1 (b) ETTm2 (c) ETTh1 (d) ETTh2

Figure 9: Performance of different forecast models with and without CvLossThe forecast errors are averaged over forecast lengths and the error bars represent 50% confidence intervals.

Table 16: Varying input sequence length results on the Weather dataset.

Models			CvLoss		TimeFilter		CvLoss		TQNet	
Metrics			MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Historical sequence length	96	96	0.149	0.188	0.153	0.199	0.154	0.191	0.156	0.200
		192	0.198	0.236	0.202	0.246	0.205	0.240	0.205	0.245
		336	0.257	0.283	0.259	0.289	0.263	0.286	0.262	0.287
		720	0.340	0.334	0.345	0.344	0.343	0.339	0.343	0.342
		Avg	0.236	0.260	0.240	0.269	0.241	0.264	0.242	0.268
	192	96	0.142	0.180	0.146	0.195	0.154	0.194	0.153	0.200
		192	0.191	0.230	0.194	0.243	0.198	0.238	0.200	0.244
		336	0.250	0.278	0.256	0.289	0.254	0.283	0.257	0.287
		720	0.333	0.331	0.334	0.341	0.327	0.333	0.326	0.335
		Avg	0.229	0.255	0.232	0.267	0.233	0.262	0.234	0.266
	336	96	0.153	0.192	0.153	0.202	0.148	0.191	0.154	0.205
		192	0.209	0.241	0.215	0.255	0.199	0.242	0.199	0.246
		336	0.269	0.287	0.288	0.302	0.247	0.281	0.249	0.285
		720	0.363	0.347	0.372	0.360	0.319	0.330	0.325	0.337
		Avg	0.248	0.267	0.257	0.280	0.228	0.261	0.232	0.268
	720	96	0.151	0.195	0.159	0.213	0.151	0.198	0.160	0.214
		192	0.216	0.254	0.219	0.266	0.203	0.245	0.210	0.258
		336	0.278	0.301	0.290	0.312	0.262	0.294	0.260	0.299
		720	0.357	0.349	0.392	0.377	0.318	0.332	0.328	0.343
		Avg	0.250	0.275	0.265	0.292	0.233	0.267	0.240	0.278

C.9 Paired effect sizes

The standard deviations in Table 17 are *marginal*: each summarises the spread of one arm across seeds. Comparing two methods by asking whether their marginal intervals overlap is the wrong instrument here, and it is a conservative one, because the two arms are not independent samples. Both are trained with the same seed indices, the same data ordering and the same initialisation scheme, so a large part of the seed-to-seed variance is shared and cancels in the difference. This subsection therefore reports the paired quantity directly.

For each cell and each seed $s \in \{2020, \ldots, 2026\}$ we form the paired difference

$$d_s = \mathcal{M}_{\text{base},s} - \mathcal{M}_{CvLoss,s}, \tag{14}$$

where $\mathcal{M}$ is the reported error metric, so that $d_s > 0$ means CvLoss reduces the error on that seed. Because both arms use the same seed indices, the mean paired difference coincides with the difference of the two reported seven-seed means, $\bar{d} = \frac{1}{7}\sum_s d_s = \bar{\mathcal{M}}_{\text{base}} - \bar{\mathcal{M}}_{CvLoss}$. We summarise each cell by $\bar{d}$, by the relative reduction $\bar{d}/\bar{\mathcal{M}}_{\text{base}}$, and by the paired 95% interval $\bar{d} \pm t_{0.975,6}\,\mathrm{SD}(d_s)/\sqrt{7}$ with $t_{0.975,6} = 2.447$, computed from the unrounded per-seed outputs.

Table 17: CvLoss ablation on ECL and Weather. Results are reported as mean$_{\pm \mathrm{std}}$ over seeds 2020–2026. Lower is better.

Hor.	Met.	ECL				Weather			
		TimeFilter	+CvLoss	TQNet	+CvLoss	TimeFilter	+CvLoss	TQNet	+CvLoss
96	MSE	$0.133_{\pm 0.001}$	$\mathbf{0.131}_{\pm 0.000}$	$0.134_{\pm 0.000}$	$\mathbf{0.133}_{\pm 0.000}$	$0.155_{\pm 0.002}$	$\mathbf{0.150}_{\pm 0.001}$	$0.157_{\pm 0.001}$	$\mathbf{0.156}_{\pm 0.002}$
	MAE	$0.230_{\pm 0.001}$	$\mathbf{0.225}_{\pm 0.000}$	$0.230_{\pm 0.000}$	$\mathbf{0.225}_{\pm 0.000}$	$0.202_{\pm 0.002}$	$\mathbf{0.189}_{\pm 0.001}$	$0.200_{\pm 0.001}$	$\mathbf{0.194}_{\pm 0.003}$
192	MSE	$0.154_{\pm 0.002}$	$\mathbf{0.152}_{\pm 0.000}$	$0.153_{\pm 0.000}$	$\mathbf{0.151}_{\pm 0.000}$	$0.204_{\pm 0.002}$	$\mathbf{0.201}_{\pm 0.002}$	$0.205_{\pm 0.000}$	$\mathbf{0.204}_{\pm 0.001}$
	MAE	$0.248_{\pm 0.001}$	$\mathbf{0.243}_{\pm 0.000}$	$0.246_{\pm 0.000}$	$\mathbf{0.241}_{\pm 0.000}$	$0.248_{\pm 0.002}$	$\mathbf{0.239}_{\pm 0.002}$	$0.245_{\pm 0.001}$	$\mathbf{0.239}_{\pm 0.000}$
336	MSE	$0.164_{\pm 0.002}$	$\mathbf{0.161}_{\pm 0.001}$	$0.169_{\pm 0.000}$	$\mathbf{0.166}_{\pm 0.000}$	$0.261_{\pm 0.002}$	$\mathbf{0.259}_{\pm 0.002}$	$0.263_{\pm 0.001}$	$\mathbf{0.262}_{\pm 0.001}$
	MAE	$0.261_{\pm 0.002}$	$\mathbf{0.257}_{\pm 0.000}$	$0.264_{\pm 0.000}$	$\mathbf{0.257}_{\pm 0.000}$	$0.290_{\pm 0.001}$	$\mathbf{0.285}_{\pm 0.002}$	$0.288_{\pm 0.001}$	$\mathbf{0.285}_{\pm 0.001}$
720	MSE	$0.184_{\pm 0.003}$	$\mathbf{0.181}_{\pm 0.001}$	$0.204_{\pm 0.004}$	$\mathbf{0.200}_{\pm 0.003}$	$0.345_{\pm 0.001}$	$\mathbf{0.341}_{\pm 0.002}$	$0.343_{\pm 0.001}$	$0.343_{\pm 0.000}$
	MAE	$0.284_{\pm 0.002}$	$\mathbf{0.282}_{\pm 0.002}$	$0.297_{\pm 0.003}$	$\mathbf{0.292}_{\pm 0.003}$	$0.344_{\pm 0.001}$	$\mathbf{0.336}_{\pm 0.001}$	$0.342_{\pm 0.000}$	$\mathbf{0.339}_{\pm 0.000}$
Avg	MSE	$0.158_{\pm 0.002}$	$\mathbf{0.156}_{\pm 0.001}$	$0.165_{\pm 0.001}$	$\mathbf{0.162}_{\pm 0.001}$	$0.241_{\pm 0.001}$	$\mathbf{0.238}_{\pm 0.001}$	$0.242_{\pm 0.000}$	$\mathbf{0.241}_{\pm 0.001}$
	MAE	$0.256_{\pm 0.001}$	$\mathbf{0.252}_{\pm 0.001}$	$0.259_{\pm 0.001}$	$\mathbf{0.254}_{\pm 0.001}$	$0.271_{\pm 0.001}$	$\mathbf{0.262}_{\pm 0.001}$	$0.269_{\pm 0.000}$	$\mathbf{0.264}_{\pm 0.001}$

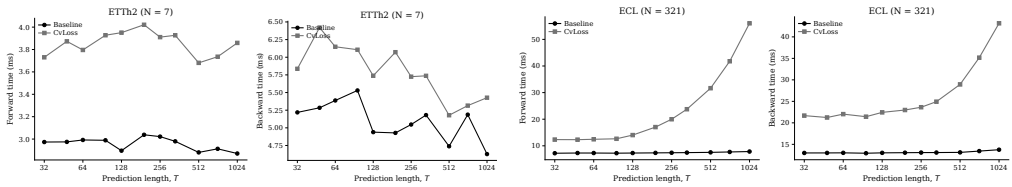
Of the 40 cells covered by Table 17, 27 are already separated by their marginal intervals. Table 18 reports the paired analysis for the remaining 13, which are exactly the cases the marginal criterion leaves unresolved.

Table 18: Paired seven-seed effects for the 13 cells whose marginal intervals overlap in Table 17. $\bar{d}$ is the mean paired reduction in the stated metric, *Rel.* the corresponding relative reduction, and the interval is the paired 95% confidence interval for $\bar{d}$. Positive values favour CvLoss. No additional training was performed: these are the same seven completed runs re-analysed in paired form.

Dataset	Backbone	T	Metric	$\bar{d}$	Rel.	95% CI
Weather	TQNet	96	MSE	0.0011	0.70%	[0.0002, 0.0020]
ECL	TimeFilter	192	MSE	0.0018	1.17%	[0.0003, 0.0033]
Weather	TimeFilter	192	MSE	0.0024	1.18%	[0.0004, 0.0044]
ECL	TimeFilter	336	MSE	0.0026	1.59%	[0.0004, 0.0048]
Weather	TimeFilter	336	MSE	0.0019	0.73%	[0.0001, 0.0037]
Weather	TQNet	336	MSE	0.0008	0.30%	[−0.0001, 0.0017]
ECL	TimeFilter	720	MSE	0.0027	1.47%	[0.0003, 0.0051]
ECL	TQNet	720	MSE	0.0041	2.01%	[0.0005, 0.0077]
Weather	TQNet	720	MSE	0.0004	0.12%	[−0.0001, 0.0009]
ECL	TimeFilter	720	MAE	0.0021	0.74%	[0.0003, 0.0039]
ECL	TQNet	720	MAE	0.0048	1.62%	[0.0004, 0.0092]
ECL	TimeFilter	Avg	MSE	0.0020	1.27%	[0.0008, 0.0032]
Weather	TQNet	Avg	MSE	0.0009	0.37%	[0.0001, 0.0017]

All 13 paired means favour CvLoss, and 11 of the 13 paired intervals exclude zero. The two that do not, Weather/TQNet at $T = 336$ and $T = 720$, both MSE, are positive but small, and we report them as inconclusive rather than as wins; they are the two settings in which the effect of CvLoss is genuinely at the resolution of the experiment. Across all 40 cells the median relative reduction is 1.66% with an interquartile range of 1.13 percentage points, which we regard as the honest summary of the size of the effect on these two datasets.

Two remarks on how these numbers should not be used. First, we do not report a pooled significance test over the full grid of metric–horizon–dataset cells. Such a test would treat the cells as independent observations, which they are not: MSE and MAE are computed from the same runs, and horizons within a dataset share the data, the split and the configuration. A rank-based test over the same cells would inherit the same dependence, so substituting one statistic for another would not repair the assumption. Second, none of the paper's claims requires it. The evidence is the direction and consistency of the controlled comparisons themselves, summarised in Table 2, together with the paired intervals above, and these are descriptive statements about the evaluated settings rather than inferences to a population of tasks.



(a) ETTh2: forward (left) and backward (right).

(b) ECL: forward (left) and backward (right).

Figure 10: Training latency on ETTh2 and ECL. CvLoss adds little overhead on ETTh2 and a larger, mainly backward-pass overhead on ECL.

C.10 Complexity

We evaluate the computational characteristics of CvLoss from two perspectives: (i) the training and inference overhead introduced by the original formulation, and (ii) the scalability improvement brought by random edge sampling.

We first measure the runtime overhead of CvLoss on ETTh2 and ECL with the batch size fixed to 32. As shown in Fig. 10, CvLoss introduces additional computation only during training, since it regularizes cross-variable dependence through an extra loss term. On the small-scale ETTh2 benchmark, this overhead is marginal in both the forward and backward passes. On the larger ECL benchmark, the overhead becomes more visible, especially in backpropagation, which is expected because modeling cross-variable interactions is more expensive when the number of variables is large. Importantly, although the prediction length T varies from 32 to 1024, the runtime remains in a similar range rather than increasing sharply, indicating that the added cost is well controlled across different horizons.

At inference time, CvLoss is removed from the computation graph and therefore does not participate in prediction. As shown in Fig. 11, the inference latency of the model trained with CvLoss remains highly comparable to that of the baseline iTransformer on both datasets. The ratio plots further show that the latency fluctuates around parity, confirming that CvLoss introduces no practical deployment-time overhead.

While the above results demonstrate that CvLoss is inexpensive at inference and manageable during training, the original formulation still builds a fully connected cross-variable graph. When the number of patches is P and the variable dimension is D, the graph contains $P \times D$ nodes, and the number of pairwise interactions scales as $O(P^2D^2)$. This quadratic growth becomes a scalability bottleneck on high-dimensional datasets such as ECL. To address this issue, we further introduce *random edge sampling*, which limits the number of evaluated interactions to at most 1000 per batch. In practice, this truncates the cost of CvLoss from rapidly increasing quadratic complexity to an effectively constant level. We emphasise that random edge sampling is a scalability device studied in this subsection alone: it produced none of the accuracy numbers reported elsewhere in the paper, all of which use the complete cross-variable graph without sampling or selection.

Fig. 12 compares the training runtime before and after sampling. On ETTh2, where the variable dimension is small, the full CvLoss remains tractable, but the sampled version still makes the runtime flatter as the number of patches increases. On ECL, the benefit is much more pronounced: the runtime of the full CvLoss grows rapidly with the number of patches, whereas the sampled version remains almost unchanged. This confirms that random edge sampling effectively removes the main scalability bottleneck of the original formulation.

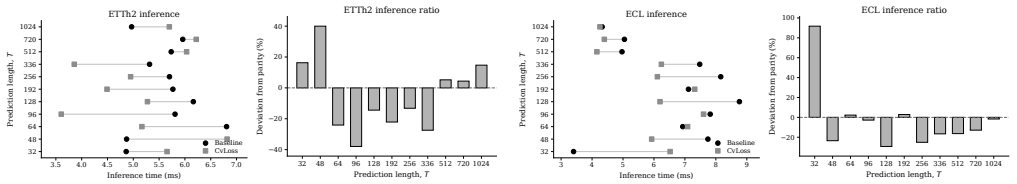
Fig. 13 further shows that random edge sampling substantially improves training efficiency without sacrificing predictive performance. As the number of patches increases, the speedup grows rapidly, while the relative MSE change remains close to zero.

Fig. 12 summarizes this observation from an efficiency–accuracy perspective. The results show that large computational gains can be achieved with negligible forecasting degradation, suggesting that a small number of randomly sampled interactions is sufficient to preserve the regularization effect of CvLoss in practice.

Table 19: Comprehensive long-term forecasting results grouped by backbone models in Benchmark TS-Library [26].

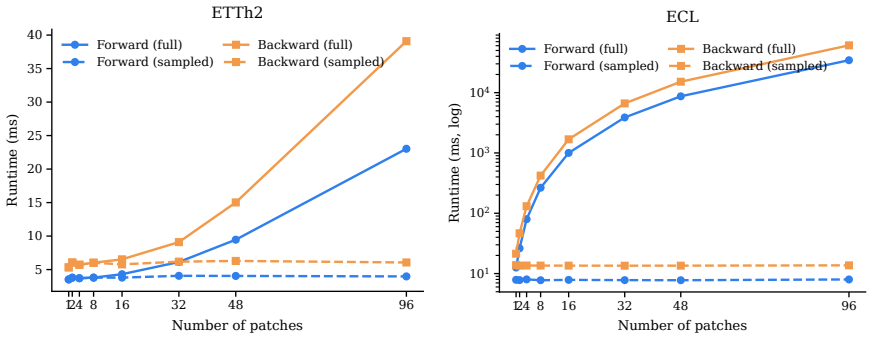
Models		iTransformer						DLinear						PatchTST					
		Original (2024)		DBLoss (2025)		CvLoss (Ours)		Original (2023)		DBLoss (2025)		CvLoss (Ours)		Original (2023)		DBLoss (2025)		CvLoss (Ours)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	0.347	0.378	0.333	0.356	**0.317**	**0.349**	0.346	0.374	0.337	0.359	**0.330**	**0.355**	0.324	0.365	0.347	0.368	**0.311**	**0.347**
	192	0.375	0.389	0.380	0.381	**0.369**	**0.379**	0.382	0.391	0.378	0.380	**0.374**	**0.378**	0.372	0.392	0.397	0.394	**0.364**	**0.377**
	336	0.409	0.412	0.415	0.404	**0.398**	**0.397**	0.415	0.415	0.408	**0.401**	**0.407**	0.402	0.398	0.409	0.434	0.415	**0.392**	**0.397**
	720	0.477	0.450	0.481	**0.441**	**0.474**	0.442	0.473	0.451	**0.467**	**0.437**	0.468	0.440	0.457	0.444	0.484	0.446	**0.455**	**0.438**
	Avg	0.402	0.407	0.402	0.395	**0.390**	**0.392**	0.404	0.408	0.397	0.394	**0.395**	**0.394**	0.388	0.402	0.415	0.406	**0.381**	**0.390**
ETTm2	96	0.185	0.271	0.181	0.259	**0.176**	**0.256**	0.193	0.293	0.183	**0.262**	**0.180**	0.262	0.186	0.268	0.180	0.257	**0.178**	**0.255**
	192	0.252	0.312	0.246	0.302	**0.241**	**0.298**	0.285	0.361	0.257	0.317	**0.246**	**0.313**	0.246	0.305	0.248	0.302	**0.240**	**0.296**
	336	0.317	0.353	0.307	0.340	**0.302**	**0.337**	0.385	0.429	0.308	0.352	**0.305**	**0.346**	0.311	0.348	0.308	0.343	**0.300**	**0.337**
	720	**0.406**	0.402	0.409	**0.398**	0.407	0.400	0.556	0.523	**0.413**	**0.422**	0.414	0.425	0.418	0.414	0.407	0.397	**0.402**	**0.397**
	Avg	0.290	0.334	0.286	0.325	**0.282**	**0.323**	0.355	0.402	0.290	0.338	**0.286**	**0.337**	0.290	0.334	0.286	0.325	**0.280**	**0.321**
ETTh1	96	0.388	0.403	0.389	0.401	**0.372**	**0.394**	0.396	0.411	0.391	0.404	**0.380**	**0.389**	0.377	0.397	0.383	0.396	**0.376**	**0.394**
	192	0.440	0.434	0.439	0.431	**0.433**	**0.427**	0.445	0.440	0.440	0.433	**0.433**	**0.425**	0.426	0.432	0.429	**0.426**	**0.424**	0.427
	336	0.483	0.458	0.479	0.453	**0.474**	**0.451**	0.487	0.465	**0.478**	**0.454**	0.485	0.458	0.469	0.457	**0.462**	**0.443**	0.468	0.453
	720	0.500	0.491	**0.473**	**0.471**	0.491	0.484	0.513	0.510	**0.489**	0.490	0.512	**0.509**	0.519	0.504	**0.452**	**0.459**	0.517	0.499
	Avg	0.453	0.447	0.445	**0.439**	**0.443**	**0.439**	0.460	0.457	**0.449**	**0.445**	0.453	**0.445**	0.448	0.448	**0.431**	**0.431**	0.446	0.443
ETTh2	96	0.297	0.348	0.292	**0.341**	**0.289**	**0.341**	0.341	0.395	0.300	0.351	**0.293**	**0.346**	0.309	0.359	**0.282**	**0.333**	0.287	0.339
	192	0.379	0.400	0.378	0.394	**0.375**	**0.392**	0.482	0.479	0.389	0.408	**0.377**	**0.399**	0.381	0.407	**0.364**	**0.386**	0.367	0.389
	336	0.422	0.433	0.424	0.430	**0.415**	**0.428**	0.593	0.542	0.447	0.453	**0.434**	**0.444**	0.412	0.429	**0.404**	0.419	0.404	**0.418**
	720	0.428	0.449	0.429	0.444	**0.419**	**0.440**	0.840	0.661	**0.590**	**0.539**	0.600	0.549	0.436	0.456	**0.411**	**0.433**	0.421	0.441
	Avg	0.381	0.407	0.381	0.402	**0.375**	**0.400**	0.564	0.519	0.431	0.438	**0.426**	**0.434**	0.384	0.413	**0.365**	**0.393**	0.370	0.396
Weather	96	0.184	0.224	0.191	0.224	**0.176**	**0.211**	0.196	0.256	0.201	**0.245**	**0.195**	0.246	0.176	0.218	0.179	0.213	**0.173**	**0.211**
	192	0.231	0.262	0.242	0.268	**0.223**	**0.253**	0.239	0.299	0.236	**0.278**	**0.235**	0.288	0.221	0.256	0.224	0.253	**0.219**	**0.251**
	336	0.286	0.302	0.288	0.298	**0.279**	**0.293**	0.281	0.331	**0.280**	**0.312**	0.285	0.329	0.280	0.298	0.278	**0.291**	**0.277**	0.293
	720	0.363	0.352	0.364	0.349	**0.360**	**0.347**	0.345	0.382	**0.339**	**0.359**	0.345	0.379	0.356	0.349	0.355	**0.342**	**0.355**	0.343
	Avg	0.266	0.285	0.271	0.285	**0.260**	**0.276**	0.265	0.317	**0.264**	**0.299**	0.265	0.310	0.258	0.280	0.259	0.275	**0.256**	**0.274**
Electricity	96	0.192	0.277	0.200	0.280	**0.164**	**0.248**	0.210	0.302	0.211	0.296	**0.195**	**0.275**	0.180	0.273	0.185	0.274	**0.180**	**0.270**
	192	0.201	0.289	0.206	0.290	**0.183**	**0.264**	0.210	0.305	0.210	0.299	**0.194**	**0.278**	0.187	0.280	0.192	0.281	**0.187**	**0.276**
	336	0.221	0.308	0.225	0.309	**0.192**	**0.276**	0.223	0.319	0.223	0.313	**0.206**	**0.292**	0.204	0.296	0.209	0.297	**0.204**	**0.293**
	720	0.267	0.344	0.271	0.345	**0.263**	**0.337**	0.258	0.350	0.257	0.342	**0.244**	**0.329**	0.246	0.328	0.252	0.331	**0.243**	**0.325**
	Avg	0.220	0.305	0.226	0.306	**0.201**	**0.281**	0.225	0.319	0.222	0.310	**0.210**	**0.293**	0.204	0.294	0.209	0.296	**0.204**	**0.291**
Traffic	96	0.545	0.371	0.597	0.386	**0.434**	**0.285**	0.697	0.429	0.690	0.421	**0.642**	**0.379**	0.459	0.298	0.461	0.292	**0.455**	**0.290**
	192	0.560	0.380	0.595	0.386	**0.469**	**0.309**	0.647	0.407	0.639	0.399	**0.595**	**0.361**	0.469	0.301	0.470	0.296	**0.465**	**0.295**
	336	0.591	0.395	0.620	0.400	**0.465**	**0.297**	0.653	0.410	0.645	0.400	**0.601**	**0.358**	0.483	0.307	0.483	0.301	**0.478**	**0.298**
	720	0.657	0.427	0.686	0.433	**0.568**	**0.370**	0.694	0.429	0.682	0.417	**0.647**	**0.397**	0.517	0.326	0.518	0.319	**0.512**	**0.315**
	Avg	0.588	0.393	0.624	0.401	**0.484**	**0.315**	0.673	0.419	0.664	0.409	**0.621**	**0.374**	0.482	0.308	0.483	0.302	**0.478**	**0.300**

Note: Average results (*Avg*) are calculated over $T \in \{96, 192, 336, 720\}$. Evaluation for the **best** and second-best results are independently conducted within the **iTransformer**, the **DLinear**, and the **PatchTST**. "Ours" refers to different enhancements applied to the respective base models.



(a) ETTh2: inference latency (left) and latency ratio (right). (b) ECL: inference latency (left) and latency ratio (right).

Figure 11: Inference latency on ETTh2 and ECL. CvLoss remains comparable to the baseline, with ratios close to parity.



(a) ETTh2 runtime. (b) ECL runtime.

Figure 12: Training runtime versus the number of patches. Dashed lines denote random edge sampling with at most 1000 edges.

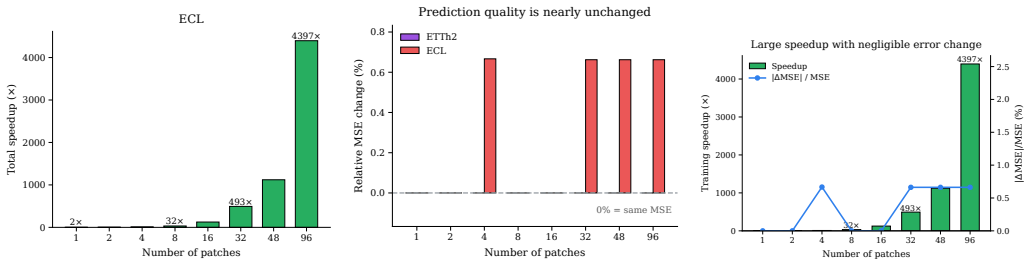
D Complexity Analysis and Edge Selection via Structural Discrepancy

In the main text, we introduced CvLoss as a lightweight plug-in loss function that regularizes the structural consistency among variables. While a fully connected cross-variable graph effectively captures comprehensive spatiotemporal interactions, calculating the loss over all possible variable pairs yields a quadratic computational complexity of $\mathcal{O}(P^2D^2)$. This poses a significant scalability challenge for high-dimensional multivariate systems. To address this bottleneck, we introduce a targeted edge selection mechanism based on structural discrepancy to significantly reduce the computational complexity.

Status of this mechanism. Three edge constructions appear in this paper, and it matters which produced which result. The complete cross-variable graph of Section 3.2, with no sampling and no selection, produced *every* accuracy number in the main text and in the appendix. The random edge sampling of Appendix C.10 and the top-K selection described below are scalability devices for the $O(P^2D^2)$ edge count, and are evaluated only in the efficiency studies of Figures 12 and 13. The synchronous and asynchronous supports compared in Section 4.5 are ablations of the topology, not the default. The distinction is important for one further reason: the top-K rule below is the only construction in the paper that reads the ground-truth correlation matrix $\mathbf{C}(Y)$, and it does so during training only, on the training split. No configuration uses any statistic of the future window at test time.

D.1 Targeting Maximum Structural Discrepancy

As established in our theoretical analysis, the standard MSE fundamentally ignores latent cross-variable relationships in its design. Consequently, models trained purely with MSE often produce predictions with significant structural discrepancies. This phenomenon is visually evident in the cross-variable correlation heatmaps presented in our motivating examples (e.g., Figure 1).



(a) ECL speedup. (b) MSE change. (c) Trade-off.

Figure 13: Optimization benefit of random edge sampling. The speedup grows rapidly with the number of patches, while the relative MSE change remains close to zero. The trade-off plot summarizes that large speedups are achieved with negligible performance degradation.

The heatmaps intuitively represent the structural relationships between variables, where each point corresponds to the correlation between a specific variable pair. By comparing the predicted correlation heatmap $\mathbf{C}(\hat{Y})$ with the ground-truth correlation heatmap $\mathbf{C}(Y)$, we observe clear differences indicating where the model fails to capture the true co-evolutionary dynamics.

To optimize computational efficiency, we do not need to compute the first-order difference loss over all points in the heatmap. Instead, we can selectively compute the loss exclusively for those variable pairs that exhibit the largest discrepancy between the prediction and the ground truth. Specifically, we measure the structural error for each variable pair (i, j) as $\Delta_{ij} = |\mathbf{C}(\hat{Y})_{ij} - \mathbf{C}(Y)_{ij}|$. We then select the top-K pairs with the maximum Δ_{ij} to construct our edge set $\mathcal{E}$.

By calculating the CvLoss strictly on these most severely misaligned pairs, we force the model to correct its most critical structural failures. This targeted approach dramatically reduces the computational complexity from $\mathcal{O}(D^2)$ to a manageable $\mathcal{O}(K)$, demonstrating that CvLoss can be efficiently scaled to high-dimensional multivariate time series forecasting without suffering from quadratic computational overhead.

D.2 Diagnostic study: does the objective track cross-variable structure?

This subsection is a diagnostic and does not describe the deployed method. Everywhere else in this paper, the two loss terms are combined by the fixed convex weight $\alpha \in (0, 1)$ of Eq. (13), selected on validation as described in Appendix B.2. We therefore ask a separate question: if the relative importance of the structural term were free to adapt, would it track how much cross-variable structure a dataset actually contains? To answer it, we run an auxiliary experiment in which the two terms carry *unnormalised* coefficients learned during training. The quantities in Table 20 are those auxiliary coefficients. They are not values of α, they are not constrained to $(0, 1)$, and they produced no result reported anywhere else in the paper; we report them only for the correspondence described below.

With that caveat, Table 20 places the converged coefficients alongside the statistical properties of each dataset, specifically the average Concordance Correlation Coefficient (CCC) and Pearson Correlation Coefficient.

We observe a clear positive correspondence between the channel-wise correlation (CCC/Pearson), the dimensionality (D), and the converged coefficient on the structural term. For datasets with fewer variables and lower cross-channel correlation, such as the ETT series ($D = 7$, CCC ≈ 0.14–0.18), the optimization converges to relatively small and comparable coefficients for both MSE and the structural penalty, indicating a balanced reliance.

Conversely, for high-dimensional spatial-temporal graphs exhibiting strong internal consistency, such as the PEMS traffic datasets ($D \geq 170$, CCC > 0.77) and Solar ($D = 137$, CCC > 0.91), the converged coefficient on the structural term grows substantially. This is the behaviour Theorem 3.1 predicts: the objective gap it identifies is larger precisely when the residual precision matrix is further from spherical, so a dataset with a rich, highly synchronized cross-variable geometry has more for the structural term to correct. The same pattern is visible in the deployed method, where the largest

accuracy gains occur on the most correlated datasets (PEMS07, CCC 0.78, -11.3% MSE; Solar, CCC 0.91, -4.4%) and the smallest on the least correlated (the ETT series, CCC 0.14–0.18). The practical implication is that channel correlation can be measured before training to anticipate whether the term will help.

Table 20: **Diagnostic study only — these are not values of α.** Converged coefficients of an auxiliary experiment in which the two loss terms carry *unnormalised* learned weights, reported alongside the channel-wise correlation of each dataset. They are unconstrained in magnitude, are not the convex weight $\alpha \in (0, 1)$ of Eq. (13), and produced none of the accuracy results in this paper; see Appendix D.2. The datasets are ordered by their number of variables (D).

Dataset	**Channels (D)**	**Avg CCC**	**Avg Pearson**	**Weight for MSE**	**Weight for Add Loss**
ETTh1	7	0.1765	0.2516	1.2706	1.2657
ETTh2	7	0.1449	0.4001	1.2080	1.2002
ETTm1	7	0.1804	0.2601	2.3572	1.9831
ETTm2	7	0.1436	0.3952	2.1759	2.0209
Weather	21	0.2079	0.3520	2.4887	2.2864
Solar	137	0.9115	0.9157	6.9919	3.4617
PEMS08	170	0.7799	0.8161	17.1380	4.2828
PEMS04	307	0.7739	0.7858	18.0127	4.3956
ECL	321	0.4619	0.4965	8.1102	3.0385
PEMS03	358	0.8206	0.8431	21.7118	4.8293
Traffic	862	0.5289	0.5643	4.6804	3.0653
PEMS07	883	0.7810	0.8066	22.8313	4.9570

E Broader Impacts

CvLoss is a general training objective for multivariate time series forecasting. Its potential positive impact is to improve forecasting reliability in applications such as energy management, traffic monitoring, weather analysis, and other operational systems where variables evolve jointly. Better forecasts may support more efficient resource allocation and planning. We do not identify direct negative societal impacts specific to CvLoss, since the method does not introduce new sensitive data, human-subject data, generative capabilities, or deployment-specific decision rules. Any societal risk would mainly come from downstream misuse or misinterpretation of forecasting models in application domains, rather than from the proposed loss itself.

F Statement on the Use of Large Language Models (LLMs)

In accordance with the conference guidelines, we disclose our use of Large Language Models (LLMs) in the preparation of this paper as follows:

We used LLMs (specifically, OpenAI GPT-5.2, GPT-5.4 and Google Gemini 3) *solely for checking grammar errors and improving the readability of the manuscript*. The LLMs *were not involved in research ideation, the development of research contributions, experiment design, data analysis, or interpretation of results*. All substantive content and scientific claims were created entirely by the authors. The authors have reviewed all LLM-assisted text to ensure accuracy and originality, and take full responsibility for the contents of the paper. The LLMs are not listed as an author.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 describe the output-side objective gap and graph residual regularizer; Sections 3–4 provide the corresponding analysis and benchmark evidence.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6 discusses the graph and patching assumptions, the Gaussian motivation versus the implemented ℓ_1 penalty, and the scope of the forecasting evaluation.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The assumptions are stated with Theorem 3.1 and Proposition 3.2; Appendix A provides their derivations and proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Sections 3–4 specify the loss and evaluation protocol; Appendix B details datasets, preprocessing, validation-based selection and training, with additional results in Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The public repository at `https://github.com/Day333/CvLoss` provides the objective implementation, dependencies and experiment scripts; Appendix B describes the public benchmarks and reproduction protocol.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 and Appendix B describe dataset splits, optimizer, early stopping and validation selection of the loss weight and patch length.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Figure 4 defines its horizon-based intervals; Appendix C reports means and standard deviations over seven seeds and paired confidence intervals, with their computation described.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Section 4.1 identifies the CPU and GPU hardware, and Appendix C.10 reports training and inference timing measurements. A complete accounting of memory usage and total compute for every experiment is not provided.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work evaluates a training objective on public forecasting benchmarks and involves no crowdsourcing or direct interaction with human participants. Dataset sources, reproducibility information and broader impacts are described in the appendix.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: The Broader Impacts section discusses applications in energy, traffic and weather forecasting and the risks of downstream misuse or misinterpretation.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The work does not release high-risk generative models or newly collected sensitive datasets; it releases a general-purpose forecasting loss implementation.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [No]

Justification: The original datasets, forecasting models and objective implementations are cited in Section 4 and Appendix B. The manuscript does not enumerate the license and terms of use of every third-party asset.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new asset is the CvLoss code release at `https://github.com/Day333/CvLoss`; its README documents the loss, dependencies and example usage. No new dataset is introduced.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The research does not involve crowdsourcing or direct research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The research does not recruit or interact with human participants; it evaluates existing public forecasting benchmarks.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not a component of the forecasting method or experiments. The appendix contains a separate statement on their use for manuscript preparation.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.