DECOVAE: DECOMPOSING PREDICTABLE DYNAMICS AND UNCERTAINTY FOR EFFICIENT PROBABILISTIC TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Probabilistic forecasting of multivariate time series remains challenging because accurate prediction requires modeling both structured temporal dynamics and time-varying residual uncertainty. Existing methods may treat these sources uniformly or incur substantial computational overhead, making it difficult to jointly achieve accurate and efficient forecasting. We propose **DecoVAE**, a lightweight Trend–Seasonality–Residual variational framework that assigns distinct statistical roles to structured dynamics and residual uncertainty. Trend and seasonality determine the forecast mean, whereas the residual stream models its time-varying variance rather than predicting residual realizations directly. To reflect their different structures, DecoVAE represents trend through a smoothness-regularized latent trajectory, seasonality through a frequency-domain VAE, and residual uncertainty through a dedicated volatility stream. Across seven real-world datasets, DecoVAE achieves the lowest mean CRPS on every dataset, with average relative improvements over the strongest competing method of 10.05% for short-term and 12.28% for long-term forecasting, while remaining competitive in point accuracy (NMAE). Compared with the closest competing method, DecoVAE trains approximately $3\times$ faster and is about $7.2\times$ smaller on high-dimensional datasets. These results suggest that assigning distinct statistical roles to structured temporal dynamics and residual uncertainty enables accurate and efficient probabilistic time series forecasting.

1 INTRODUCTION

Probabilistic time series forecasting is important in energy, finance, weather, and traffic (Qiao et al., 2026; Kong et al., 2025; Zhang et al., 2024a), but jointly achieving accuracy, reliable uncertainty estimation, and computational efficiency remains challenging. Trend and seasonality exhibit different structures (Proietti & Pedregal, 2023): time-domain models may struggle with global periodicity (Zhang et al., 2025), whereas frequency-domain approaches can suffer from spectral mixing (An et al., 2026; Lu et al., 2026). Explicit decomposition therefore provides a natural way to separate heterogeneous dynamics and assign suitable inductive biases to them (Deng et al., 2024).

Existing probabilistic forecasting methods still face an accuracy–efficiency trade-off. Lightweight decomposition-based models such as DLinear (Zeng et al., 2023) provide strong point forecasts but do not explicitly model predictive uncertainty. Latent-variable approaches such as LaST (Wang et al., 2022) and CLaST (Marusov et al., 2026) learn separate trend and seasonal representations, while more expressive probabilistic models based on autoregressive, diffusion, flow-based, or Koopman–Kalman formulations (Salinas et al., 2020; Kollovieh et al., 2023; Zhang et al., 2024b; Wu et al., 2025) often introduce greater computational complexity. At the same time, ARCH (Engle, 1982), GARCH (Bollerslev, 1986), and EGARCH (Nelson, 1991) demonstrate that uncertainty itself can exhibit predictable temporal structure through time-varying volatility. Heteroscedastic neural models follow a related principle (Nix & Weigend, 1994; Kendall & Gal, 2017; Shukla & Marlin, 2021), although jointly learning mean and variance can be delicate (Seitzer et al., 2022). These observations suggest a simple principle: *structured temporal dynamics should determine the forecast mean, while residual uncertainty should model its time-varying variance*.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We introduce **DecoVAE** (Trend–Seasonality–Residual Decomposition Variational Autoencoder), a lightweight probabilistic forecasting framework that assigns distinct statistical roles to predictable dynamics and residual uncertainty. Trend and seasonality determine the forecast mean, whereas the residual stream models its time-varying variance, separating what is predictable in value from what is predictable in uncertainty. To reflect their distinct structures, trend is represented by a smooth latent trajectory in the time domain, while seasonality is modeled by a complex-valued frequency-domain VAE. Residual uncertainty is captured through a dedicated volatility stream.

Our main contributions are:

- **Role-specialized probabilistic decomposition**. Unlike decomposition-based forecasting methods that treat decomposed components primarily as predictive signals, DecoVAE assigns distinct statistical roles to structured dynamics and residual uncertainty. Trend and seasonality determine the forecast mean, whereas the residual stream models its time-varying variance rather than predicting residual realizations directly. This decouples the prediction of structured temporal dynamics from the modeling of residual uncertainty.
- **Volatility-aware residual modeling.** We assign the decomposition residual a distinct statistical role: rather than predicting residual realizations directly, DecoVAE learns a latent representation of their time-varying conditional variance, which parameterizes the forecast variance. Table 3 provides two complementary ablations supporting this design. First, adding the residual-volatility stream to the Trend+Season backbone substantially improves probabilistic forecasting and calibration. Second, controlled comparisons with the same backbone show that these gains cannot be explained merely by residual prediction or a heteroscedastic output head: the proposed variational residual-volatility formulation provides the most consistent performance across alternative residual uncertainty models.
- **Accuracy–efficiency trade-off.** Across seven real-world datasets, DecoVAE achieves the lowest mean CRPS on every benchmark, with average relative improvements of 10.05% for short-term forecasting and 12.28% for long-term forecasting over the strongest competing methods. At the same time, DecoVAE trains approximately $3\times$ faster per epoch than CLaST and K^2VAE and is about $7.2\times$ smaller on high-dimensional datasets, demonstrating a favorable balance between probabilistic forecasting accuracy and computational efficiency. Code is available at `https://anonymous.4open.science/r/DecoVAE-3387/`.

2 Related Work

2.1 Latent Modeling of Trend and Seasonality

Modern forecasting architectures increasingly employ component-specific inductive biases. Autoformer and FEDformer exploit decomposition and frequency-domain structure (Wu et al., 2021; Zhou et al., 2022), while xPatch (Stitsyuk & Choi, 2025), TimeMixer (Wang et al., 2024), and TimesNet (Wu et al., 2023) use specialized temporal representations. LaST (Wang et al., 2022) introduces variational trend and seasonal latents with structural regularization, and CLaST (Marusov et al., 2026) extends this formulation with contextual latent alignment. These approaches motivate explicit component-specific modeling, but do not assign the decomposition residual a dedicated conditional-volatility role.

2.2 Probabilistic Time Series Forecasting

Probabilistic forecasting includes autoregressive models such as DeepAR, VAE-based methods such as LaST and K^2VAE, diffusion models such as TSDiff-Cond, flow-based approaches such as TFM, and Neural-ODE-based models such as DeNOTS (Salinas et al., 2020; Wang et al., 2022; Wu et al., 2025; Kollovieh et al., 2023; Zhang et al., 2024b; Kuleshov et al., 2026). These methods provide increasingly expressive predictive distributions, often at higher computational cost. DecoVAE instead emphasizes lightweight component-specific modeling with an explicit residual-volatility stream. Additional related work on trend-seasonality decomposition techniques and loss functions is provided in Appendix A.

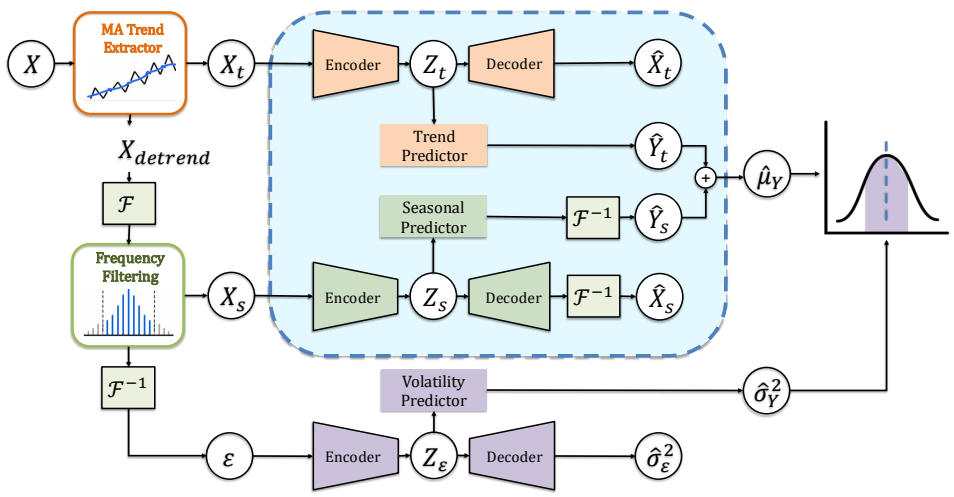


Figure 1: Overview of DecoVAE. The input series is decomposed into trend, seasonal, and stochastic residual components. Trend and seasonality model predictable temporal dynamics and jointly determine the latent-conditioned forecast mean, while the residual stream does not predict future residual realizations directly; instead, it forecasts their time-varying conditional volatility. The final probabilistic forecast therefore separates what is predictable in value from what is predictable in uncertainty.

3 METHOD

Problem statement. We consider a multivariate time series $\mathbf{X}_\tau \in \mathbb{R}^F$. Given a context $\mathbf{X}_{\tau-T+1:\tau} \in \mathbb{R}^{T\times F}$, probabilistic forecasting estimates $p(\mathbf{Y} \mid \mathbf{X}_{\tau-T+1:\tau})$ for the next L observations $\mathbf{Y} = \mathbf{X}_{\tau+1:\tau+L} \in \mathbb{R}^{L\times F}$.

Method pipeline. The main pipeline of our approach is depicted in Fig. 1. Given the context $X = X_{\tau-T+1:\tau} \in \mathbb{R}^{T\times F}$, we decompose it as $X = X_t + X_s + \varepsilon$. The Trend and Residual encoders produce time-domain latents $Z_t \sim q_t(Z_t \mid X_t)$ and $Z_\varepsilon \sim q_n(Z_\varepsilon \mid \varepsilon)$ in $\mathbb{R}^{T\times d}$, while the Seasonal encoder produces $Z_s \sim q_s(Z_s \mid X_s) \in \mathbb{C}^{K\times d}$, where $K = \lfloor T/2 \rfloor + 1$.

Given sampled latent representations, component-specific predictors map them to the forecasting horizon L. The trend and seasonal predictors produce $\widehat{Y}_t = \mathcal{P}_t(Z_t) \in \mathbb{R}^{L\times F}$ and $\widehat{Y}_s = \mathcal{F}^{-1}(\mathcal{P}_s(Z_s)) \in \mathbb{R}^{L\times F}$, whose sum defines the forecast mean, $\hat{\mu}_Y = \widehat{Y}_t + \widehat{Y}_s$. In contrast, the residual stream models uncertainty: its decoder reconstructs historical residual variance $\widehat{\boldsymbol{\sigma}}^2_\varepsilon \in \mathbb{R}^{T\times F}_+$, while $\mathcal{P}_n$ predicts future variance $\widehat{\boldsymbol{\sigma}}^2_Y = \mathcal{P}_n(Z_\varepsilon) \in \mathbb{R}^{L\times F}_+$. In our implementation, $\mathcal{P}_t$ and $\mathcal{P}_n$ are MLP-based predictors with Softplus applied to the output of $\mathcal{P}_n$ (see Appendix C.1 for detailed implementations), while $\mathcal{P}_s$ uses complex-valued linear layers for frequency-domain extrapolation.

For each future time–feature pair, $Y_{h,f} \mid Z_t, Z_s, Z_\varepsilon \sim \mathcal{N}\left(\widehat{\mu}_{Y,h,f}, \widehat{\sigma}^2_{Y,h,f}\right)$, where $h = 1, \ldots, L$ and $f = 1, \ldots, F$. We assume conditional independence across time and features given the latents, avoiding the $O((LF)^2)$ parameters and $O((LF)^3)$ likelihood cost of a full covariance model. Dependencies are instead captured through Z_t, Z_s, and Z_ε, inferred from the full multivariate context. At inference, $p(Y \mid X)$ is approximated by Monte Carlo sampling from the latent posteriors and Gaussian emissions, capturing both latent and residual uncertainty.

3.1 TREND–SEASONALITY–RESIDUAL DECOMPOSITION

We first extract the trend component X_t using a trend operator $\mathcal{D}_t$ and obtain the detrended signal as $X_{\text{detrend}} = X - X_t$. In the final model, $\mathcal{D}_t$ is a simple computationally efficient moving-average (MA) trend extractor; alternative trend models, including FEDformer's MoE and exponential moving average (EMA), are evaluated in Appendix B, with inferior performance reported. The Deco-

VAE further separates the detrended signal into seasonal and residual components in the frequency domain. Unlike classical STL (Cleveland et al., 1990), this procedure does not require a predefined seasonal period. Specifically, we compute the Fourier spectrum $S = \mathcal{F}(X_{\text{detrend}}) \in \mathbb{C}^{K\times F}$, where $K = \lfloor T/2 \rfloor + 1$. Frequency selection is performed independently for each feature f. Let $P_{k,f} = |S_{k,f}|^2$ denote the spectral power and $P_{(i),f}$ its values sorted in descending order. We select $K_{\text{opt},f} = \arg\min_k \left\{ \frac{\sum_{i=1}^{k} P_{(i),f}}{\sum_{j=1}^{K} P_{j,f}} \geq p_{\text{opt}} \right\}$, so that the retained frequency set Ω_f accounts for at least a fraction p_{opt} of the spectral energy of feature f. The seasonal component is then reconstructed by retaining the selected frequencies for each feature and applying the inverse Fourier transform: $X_s = \mathcal{F}^{-1}(S_\Omega)$, where S_Ω denotes the spectrum containing only the frequencies selected for each feature. Finally, the stochastic residual component is obtained as $\varepsilon = X - X_t - X_s \in \mathbb{R}^{T\times F}$. The resulting components $\{X_t, X_s, \varepsilon\}$ are subsequently processed by their respective VAE streams with component-specific inductive biases. The proposed spectral-cut rule minimizes the number of retained frequency bins for the prescribed spectral reconstruction tolerance, as Theorem E.1 from Appendix E presents.

3.2 Latent Modeling of Trend, Seasonality and Volatility of noise

We outline below our approach for modeling trend and seasonal components to yield their respective representations.

3.2.1 Trend and Seasonal Components

The trend and seasonal components are modeled using dedicated variational representations with domain-specific inductive biases. The trend component $X_t \in \mathbb{R}^{T\times F}$ is encoded by an MLP-based VAE into a Gaussian latent representation $Z_t \in \mathbb{R}^{T\times d}$, which is regularized to capture the smooth, low-frequency structure of the trend. In contrast, the seasonal component $X_s \in \mathbb{R}^{T\times F}$ is transformed into the frequency domain as $X_s^{(\text{freq})} = \mathcal{F}(X_s) \in \mathbb{C}^{K\times F}$, where $K = \lfloor T/2 \rfloor + 1$. A complex VAE based on the FITS-style parameterization (Xu et al., 2024) operates directly on this spectrum, preserving both amplitude and phase information and producing a frequency-aware latent representation $Z_s \in \mathbb{C}^{K\times d}$ for multi-periodic dynamics. Detailed architectures of the Trend-VAE and Seasonal-VAE are provided in Appendices C.2 and C.3, respectively.

3.2.2 Residual component

Residual values may be difficult to forecast directly, while their variance can still exhibit predictable temporal dependence. This is the key idea behind conditional-volatility models such as ARCH and GARCH. DecoVAE adopts the same principle: the trend and seasonal streams model the forecast mean, whereas the residual stream uses the observed residual history to model time-varying forecast variance.

Given the residual sequence $\varepsilon \in \mathbb{R}^{T\times F}$, the Residual-VAE encodes its temporal patterns into $Z_\varepsilon \in \mathbb{R}^{T\times d}$. Rather than reconstructing future residual realizations, the residual decoder learns the historical variance profile, while the forecasting head uses Z_ε to predict future variance. Thus, the latent representation captures information in the residual history that is informative about volatility dynamics, rather than treating the residual as unstructured noise. Further architectural details are provided in Appendix C.4.

3.3 Loss Function

The model is trained by minimizing a composite loss function that combines objectives for trend, seasonality and volatility of noise components. For notational simplicity, all loss functions below are written for an arbitrary fixed feature, with the feature index omitted. In the multivariate setting, the corresponding losses are computed over all features and averaged.

3.3.1 Trend Loss

The trend stream is trained using a combination of reconstruction error and KL divergence:

$$\mathcal{L}_{trend} = \mathcal{L}_{rec}^{trend} + \mathcal{L}_{KL}^{trend}, \tag{1}$$

where $\mathcal{L}_{rec}^{trend} = \|X_t - \widehat{X}_t\|_2^2$ is the reconstruction loss and $\mathcal{L}_{KL}^{trend} = \mathrm{KL}\big(q_\phi(Z_t) \,\|\, p(Z_t)\big)$ is the standard KL divergence for the trend latent variables.

To encourage a smooth latent trend trajectory, we define: $\mathcal{R}_{trend} = \alpha_1 \mathcal{R}_1(Z_t) + \alpha_2 \mathcal{R}_2(Z_t)$, where $\mathcal{R}_1 = \frac{1}{T-1}\sum_{\tau=1}^{T-1} \|Z_{t,\tau+1} - Z_{t,\tau}\|^2$ and $\mathcal{R}_2 = \frac{1}{T-2}\sum_{\tau=2}^{T-1} \|Z_{t,\tau+1} - 2Z_{t,\tau} + Z_{t,\tau-1}\|^2$ are the normalized squared norms of the first- and second-order temporal differences, respectively. The first-order term penalizes rapid changes in the latent trajectory, while the second-order term penalizes excessive curvature. This regularization is closely related to Hodrick–Prescott smoothing applied in the latent space.

Connection to Hodrick–Prescott smoothing. The second-order regularizer $\mathcal{R}_2$ discourages rapid changes in the curvature of the latent trend trajectory, resembling the smoothness principle of the Hodrick–Prescott (HP) filter (Hodrick & Prescott, 1997). Under a simplified linear-decoder setting, smoothness in the latent trajectory translates directly into smoothness of the decoded trend, up to a scaling determined by the decoder. This provides an intuitive connection between our latent regularization and classical HP smoothing. The formal theoretical result and proof are provided in Appendix D. We emphasize that this connection is interpretive rather than a general guarantee, since the formal result assumes a full-rank linear decoder, whereas DecoVAE uses a nonlinear MLP decoder.

3.3.2 Seasonal Loss

The seasonal stream is trained using a reconstruction term and a KL divergence term:

$$\mathcal{L}_{season} = \mathcal{L}_{rec}^{season} + \mathcal{L}_{KL}^{season}. \tag{2}$$

Unlike the trend stream, the seasonal component is reconstructed directly in the frequency domain to preserve its spectral structure. Accordingly, the reconstruction loss is defined as $\mathcal{L}_{rec}^{season} = \left\| \widehat{X}_s^{(\mathrm{freq})} - \widetilde{X}_s \right\|_2^2$, while the latent distribution is regularized by $\mathcal{L}_{KL}^{season} = \mathrm{KL}\big(q_\phi(Z_s) \,\|\, p(Z_s)\big)$, where $p(Z_s) = \mathcal{CN}(0, I)$ is an isotropic proper complex Gaussian prior.

In the final model, we do not employ an explicit spectral regularization term. Alternative regularization schemes were explored separately and are reported in Appendix F.

3.3.3 Volatility Loss

Similarly, the residual-volatility stream is trained using a reconstruction term and a KL divergence term:

$$\mathcal{L}_{vol} = \mathcal{L}_{rec}^{vol} + \mathcal{L}_{KL}^{vol}. \tag{3}$$

We distinguish between the reconstructed historical residual volatility $\widehat{\sigma}_\varepsilon^2$ and the predicted future volatility $\widehat{\sigma}_Y^2$. Assuming a zero-mean Gaussian residual distribution, the historical volatility profile is learned through the negative log-likelihood $\mathcal{L}_{rec}^{vol} = \frac{1}{2T}\sum_{\tau=1}^{T}\left(\ln \widehat{\sigma}_{\varepsilon,\tau}^2 + \frac{\varepsilon_\tau^2}{\widehat{\sigma}_{\varepsilon,\tau}^2}\right)$. The latent representation is additionally regularized by the standard KL divergence $\mathcal{L}_{KL}^{vol} = \mathrm{KL}\big(q_\phi(Z_\varepsilon) \,\|\, p(Z_\varepsilon)\big)$.

To encourage a temporally coherent historical volatility profile, we employ the smoothness regularizer $\mathcal{R}_{vol} = \lambda_{smooth}\mathcal{L}_{smooth}^{vol}$, where $\mathcal{L}_{smooth}^{vol} = \frac{1}{T-2}\sum_{\tau=2}^{T-1}\left(\widehat{\sigma}_{\varepsilon,\tau+1}^2 - 2\widehat{\sigma}_{\varepsilon,\tau}^2 + \widehat{\sigma}_{\varepsilon,\tau-1}^2\right)^2$. This term penalizes second-order temporal differences in the reconstructed historical volatility trajectory and suppresses erratic local fluctuations.

3.3.4 Total Loss

Conditioned on the sampled latent variables Z_t, Z_s, and Z_ε, the future target is modeled using a time-varying heteroscedastic Gaussian distribution with mean $\widehat{\mu}_Y$ and predicted future variance $\widehat{\sigma}_Y^2$. Accordingly, the forecasting objective is defined by the Gaussian negative log-likelihood $\mathcal{L}_{\mathrm{pred}} = \frac{1}{2L}\sum_{\tau=T+1}^{T+L}\left(\ln \hat{\sigma}_{Y,\tau}^2 + \frac{(Y_\tau - \hat{\mu}_{Y,\tau})^2}{\widehat{\sigma}_{Y,\tau}^2}\right)$.

The overall training objective is a weighted combination of the component-specific losses and the forecasting loss:

$$\mathcal{L}_{total} = \lambda_{ELBO}(\mathcal{L}_{trend} + \mathcal{L}_{season} + \mathcal{L}_{vol}) + \lambda_{pred}\mathcal{L}_{pred} + \mathcal{R}_{trend} + \mathcal{R}_{vol}, \tag{4}$$

where λ_{ELBO} and λ_{pred} control the contributions of the component-specific ELBO objectives and forecasting loss, respectively, while α_1, α_2, and λ_{smooth} control the corresponding regularization terms. The corresponding hyperparameter search ranges are provided in Appendix G.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate the prediction quality using seven real-world datasets with diverse dimensionalities, sampling frequencies, and temporal characteristics. These include four variants of the Electricity Transformer Temperature dataset (ETTh1, ETTh2, ETTm1, ETTm2) (Zhou et al., 2021), traffic flow measurements (Lai et al., 2018), household electricity consumption[1], and electricity demand data from the Electric Reliability Council of Texas (ERCOT). Complete dataset specifications are provided in Appendix H.

Evaluation Protocol. Following prior probabilistic forecasting works (Wu et al., 2025; Kollovieh et al., 2023), we primarily evaluate performance using two complementary metrics: the Continuous Ranked Probability Score (CRPS) (Gneiting & Raftery, 2007; Zamo & Naveau, 2018), which evaluates the calibration–sharpness trade-off of the marginal predictive distributions, and Normalized Mean Absolute Error (NMAE), which measures point forecast accuracy. For the controlled residual-stream ablation, we additionally report Expected Calibration Error (ECE) (Chung et al., 2021), where the Trend+Season backbone is fixed and only the residual uncertainty formulation varies. This allows calibration differences to be attributed directly to the choice of residual modeling strategy. Complete metric definitions are provided in Appendix I.

4.2 MAIN RESULTS

Short-term forecasting ($T = 96, L = 24$) As shown in Table 1, DecoVAE achieves the lowest mean CRPS on all seven datasets and the best or tied-best NMAE on five of seven datasets. The largest gains are observed on ERCOT, where DecoVAE improves over the second-best method by 30.43% in CRPS and 18.52% in NMAE. DecoVAE also achieves strong CRPS improvements on ETTh1 and ETTh2, while remaining competitive in NMAE on the ETT benchmarks. Overall, the results are statistically significant under paired Wilcoxon signed-rank tests ($p < 0.05$) compared with the closest-performing competing method.

Table 1: **Comparison of short-term probabilistic time series forecasting performance** ($T = 96$, $L = 24$). Lower CRPS and NMAE scores indicate better performance. Results are reported as mean $\pm$ standard error, computed from **three independent runs** involving model retraining and evaluation. **Boldface** highlights the best-performing method, while underlining denotes the second-best result.

Dataset	Metric	DecoVAE (ours)	CLaST	K^2VAE	LaST	DeepAR	TFM	TSDiff	DeNOTS
Electricity	CRPS	$\mathbf{0.075 \pm 0.000}$	0.089 ± 0.000	0.082 ± 0.000	0.144 ± 0.021	0.117 ± 0.001	0.150 ± 0.011	0.115 ± 0.006	0.201 ± 0.002
	NMAE	$\mathbf{0.100 \pm 0.000}$	0.108 ± 0.000	0.107 ± 0.000	0.166 ± 0.022	0.151 ± 0.002	0.201 ± 0.014	0.150 ± 0.008	0.274 ± 0.001
Traffic	CRPS	$\mathbf{0.161 \pm 0.002}$	0.203 ± 0.001	0.164 ± 0.001	0.309 ± 0.011	0.207 ± 0.001	0.321 ± 0.013	0.208 ± 0.004	0.386 ± 0.019
	NMAE	$\mathbf{0.204 \pm 0.002}$	0.241 ± 0.000	0.211 ± 0.002	0.360 ± 0.010	0.264 ± 0.002	0.426 ± 0.022	0.253 ± 0.004	0.500 ± 0.032
ETTh1	CRPS	$\mathbf{0.194 \pm 0.002}$	0.220 ± 0.002	0.222 ± 0.002	0.254 ± 0.004	0.384 ± 0.071	0.540 ± 0.018	0.373 ± 0.041	0.503 ± 0.008
	NMAE	$\mathbf{0.263 \pm 0.002}$	0.267 ± 0.003	0.283 ± 0.006	0.289 ± 0.005	0.498 ± 0.088	0.623 ± 0.026	0.492 ± 0.037	0.645 ± 0.017
ETTh2	CRPS	$\mathbf{0.294 \pm 0.001}$	0.316 ± 0.004	0.372 ± 0.005	0.460 ± 0.058	1.168 ± 0.091	1.738 ± 0.451	0.989 ± 0.213	1.920 ± 0.398
	NMAE	0.379 ± 0.002	$\mathbf{0.371 \pm 0.004}$	0.405 ± 0.005	0.546 ± 0.077	1.485 ± 0.138	1.892 ± 0.384	1.281 ± 0.245	2.207 ± 0.421
ETTm1	CRPS	$\mathbf{0.163 \pm 0.003}$	0.180 ± 0.004	0.186 ± 0.005	0.217 ± 0.004	0.359 ± 0.008	0.556 ± 0.038	0.299 ± 0.036	0.561 ± 0.048
	NMAE	$\mathbf{0.217 \pm 0.005}$	0.218 ± 0.004	0.228 ± 0.003	0.257 ± 0.008	0.473 ± 0.005	0.634 ± 0.038	0.396 ± 0.059	0.721 ± 0.050
ETTm2	CRPS	$\mathbf{0.226 \pm 0.003}$	0.229 ± 0.000	0.314 ± 0.027	0.274 ± 0.004	0.999 ± 0.182	1.784 ± 0.121	0.468 ± 0.066	1.591 ± 0.889
	NMAE	0.284 ± 0.003	$\mathbf{0.271 \pm 0.002}$	0.316 ± 0.013	0.325 ± 0.007	1.273 ± 0.201	1.904 ± 0.151	0.630 ± 0.087	1.815 ± 0.906
ERCOT	CRPS	$\mathbf{0.032 \pm 0.000}$	0.046 ± 0.000	0.051 ± 0.002	0.046 ± 0.000	0.132 ± 0.026	0.231 ± 0.035	0.069 ± 0.005	0.204 ± 0.039
	NMAE	$\mathbf{0.044 \pm 0.001}$	0.055 ± 0.001	0.055 ± 0.002	0.054 ± 0.000	0.175 ± 0.041	0.266 ± 0.048	0.086 ± 0.007	0.258 ± 0.044

Long-term forecasting ($T = 96, L = 720$) For long-term forecasting (Table 2), DecoVAE achieves the lowest mean CRPS on all seven datasets and the best NMAE on four of seven datasets.

[1] `https://archive.ics.uci.edu/dataset/321/electricityloaddiagrams20112014`, accessed in May 2026.

Relative to the second-best methods, the largest CRPS improvement is observed on ERCOT, where DecoVAE reduces CRPS from 0.080 to 0.063, corresponding to a 21.25% improvement. The largest NMAE gain is obtained on Electricity, where NMAE decreases from 0.140 to 0.127, corresponding to a 9.29% improvement. Strong CRPS gains are also observed on ETTh1 and ETTm2, with improvements of approximately 17.07% and 14.96%, respectively. On ETTh2, DecoVAE improves CRPS by approximately 11.40% over CLaST, while CLaST achieves the lowest NMAE. Overall, the improvements are statistically significant under paired Wilcoxon signed-rank tests ($p < 0.05$) relative to the closest-performing competing method, consistent with the significance observed in the short-term forecasting results.

Table 2: **Comparison of long-term probabilistic time series forecasting performance** ($T = 96$, $L = 720$). Lower CRPS and NMAE scores indicate better performance. Results are reported as mean $\pm$ standard error, computed from **three independent runs** involving model retraining and evaluation. **Boldface** highlights the best-performing method, while underlining denotes the second-best result.

Dataset	Metric	DecoVAE (ours)	CLaST	K^2VAE	LaST	DeepAR	TFM	TSDiff	DeNOTS
Electricity	CRPS	$\mathbf{0.095 \pm 0.000}$	0.126 ± 0.002	0.115 ± 0.006	0.146 ± 0.001	0.163 ± 0.009	0.210 ± 0.017	0.126 ± 0.003	0.186 ± 0.009
	NMAE	$\mathbf{0.127 \pm 0.009}$	0.146 ± 0.002	0.140 ± 0.004	0.169 ± 0.001	0.204 ± 0.011	0.292 ± 0.022	0.165 ± 0.004	0.186 ± 0.009
Traffic	CRPS	$\mathbf{0.188 \pm 0.001}$	0.247 ± 0.001	0.189 ± 0.002	0.301 ± 0.003	0.220 ± 0.003	0.473 ± 0.018	0.297 ± 0.027	0.414 ± 0.143
	NMAE	$\mathbf{0.237 \pm 0.002}$	0.288 ± 0.001	0.243 ± 0.000	0.343 ± 0.002	0.286 ± 0.006	0.637 ± 0.023	0.368 ± 0.033	0.419 ± 0.151
ETTh1	CRPS	$\mathbf{0.277 \pm 0.004}$	0.342 ± 0.005	0.334 ± 0.008	0.573 ± 0.009	0.690 ± 0.211	0.590 ± 0.060	0.575 ± 0.059	0.555 ± 0.014
	NMAE	$\mathbf{0.376 \pm 0.006}$	0.379 ± 0.004	0.428 ± 0.012	0.614 ± 0.010	0.830 ± 0.240	0.839 ± 0.088	0.701 ± 0.054	0.750 ± 0.019
ETTh2	CRPS	$\mathbf{0.482 \pm 0.010}$	0.544 ± 0.001	1.000 ± 0.235	2.531 ± 0.054	2.332 ± 0.013	2.495 ± 0.688	2.254 ± 0.106	2.082 ± 0.136
	NMAE	0.611 ± 0.009	$\mathbf{0.588 \pm 0.002}$	0.769 ± 0.083	2.602 ± 0.038	2.611 ± 0.053	3.204 ± 0.992	2.488 ± 0.104	2.356 ± 0.104
ETTm1	CRPS	$\mathbf{0.257 \pm 0.002}$	0.309 ± 0.000	0.266 ± 0.024	0.421 ± 0.013	0.466 ± 0.073	0.592 ± 0.010	0.483 ± 0.020	0.576 ± 0.009
	NMAE	0.340 ± 0.004	0.341 ± 0.001	$\mathbf{0.328 \pm 0.039}$	0.487 ± 0.015	0.623 ± 0.069	0.855 ± 0.025	0.629 ± 0.025	0.614 ± 0.010
ETTm2	CRPS	$\mathbf{0.432 \pm 0.010}$	0.508 ± 0.001	0.989 ± 0.275	1.871 ± 0.241	2.064 ± 0.646	2.899 ± 0.782	1.642 ± 0.450	1.931 ± 0.016
	NMAE	0.535 ± 0.006	$\mathbf{0.534 \pm 0.001}$	0.713 ± 0.105	2.022 ± 0.251	2.388 ± 0.633	3.783 ± 0.975	1.915 ± 0.430	2.008 ± 0.036
ERCOT	CRPS	$\mathbf{0.063 \pm 0.000}$	0.080 ± 0.001	0.084 ± 0.001	0.095 ± 0.004	0.160 ± 0.027	0.191 ± 0.007	0.111 ± 0.008	0.140 ± 0.031
	NMAE	$\mathbf{0.086 \pm 0.000}$	0.090 ± 0.002	0.103 ± 0.002	0.106 ± 0.005	0.211 ± 0.035	0.242 ± 0.009	0.137 ± 0.008	0.185 ± 0.044

4.3 Ablation Study: Separating Predictable Dynamics from Residual Uncertainty

Table 3 examines two questions: whether DecoVAE benefits from assigning distinct roles to trend, seasonality, and residual uncertainty, and whether the proposed residual-volatility formulation is preferable to simpler residual modeling strategies.

We first evaluate the contribution of the component-specific design. The **Trend** variant models structured dynamics through stochastic trend latents, while **Trend + Season** additionally captures seasonal dynamics. Adding the proposed **Res. Vol.** stream yields the full DecoVAE model, where trend and seasonality determine the forecast mean and the residual stream models its time-varying variance. Relative to the Trend+Season backbone, the full model improves CRPS, NMAE, and ECE on all four datasets, with particularly large gains in calibration.

We then isolate the choice of residual uncertainty representation by keeping the Trend+Season backbone fixed and comparing direct residual prediction (**Res. Targets**), deterministic heteroscedastic prediction (**Vol. MLP**), learned time-invariant variance (**Const. Vol.**), and the proposed variational **Res. Vol.** stream. While simpler alternatives perform well on individual datasets, their improvements are less consistent. The proposed formulation achieves the best or tied-best CRPS on all four datasets and the lowest average CRPS (0.141), NMAE (0.182), and ECE (0.026) among the residual-modeling variants, indicating that the gains are not explained by heteroscedastic prediction alone.

Additional ablations isolate the corresponding design choices: Appendix J.1 evaluates first- and second-order trend regularization, Appendix J.2 studies volatility regularization, and Appendix J.3 provides PIT histograms and empirical CDFs for calibration analysis.

4.4 Computational Efficiency and Model Analysis

Beyond predictive accuracy, we evaluate DecoVAE's computational characteristics across training speed, memory consumption, model size, and inference time.

Table 3: Ablation of component-specific modeling and residual uncertainty representations for short-term forecasting ($T = 96$, $L = 24$). The **Trend** and **Trend + Season** variants evaluate the contribution of structured predictive components, while the remaining variants share the same Trend+Season backbone and differ only in their treatment of the residual. Results are mean ± standard error over three independent runs. Lower is better.

Dataset	Metric	Trend	Trend + Season	Trend + Season + Res. Targets	Trend + Season + Vol. MLP	Trend + Season + Const. Vol.	Trend + Season + Res. Vol. (ours)
Electricity	CRPS	0.094 ± 0.001	0.088 ± 0.001	0.089 ± 0.001	$\mathbf{0.075 \pm 0.000}$	0.080 ± 0.000	$\mathbf{0.075 \pm 0.000}$
	NMAE	0.103 ± 0.001	0.104 ± 0.001	0.104 ± 0.001	$\mathbf{0.099 \pm 0.000}$	0.103 ± 0.000	0.100 ± 0.000
	ECE	0.191 ± 0.001	0.143 ± 0.005	0.158 ± 0.001	0.032 ± 0.006	$\mathbf{0.018 \pm 0.001}$	0.030 ± 0.004
Traffic	CRPS	0.201 ± 0.002	0.194 ± 0.002	0.178 ± 0.002	0.166 ± 0.002	0.181 ± 0.001	$\mathbf{0.161 \pm 0.002}$
	NMAE	0.222 ± 0.002	0.222 ± 0.001	$\mathbf{0.204 \pm 0.001}$	0.207 ± 0.001	0.216 ± 0.000	$\mathbf{0.204 \pm 0.002}$
	ECE	0.177 ± 0.006	0.140 ± 0.006	0.136 ± 0.010	0.038 ± 0.018	0.067 ± 0.003	$\mathbf{0.027 \pm 0.011}$
ETTh2	CRPS	0.338 ± 0.003	0.319 ± 0.005	0.329 ± 0.008	0.443 ± 0.031	0.325 ± 0.002	$\mathbf{0.294 \pm 0.001}$
	NMAE	0.400 ± 0.002	0.382 ± 0.007	0.389 ± 0.004	0.423 ± 0.008	0.402 ± 0.002	$\mathbf{0.379 \pm 0.002}$
	ECE	0.128 ± 0.002	0.121 ± 0.003	0.137 ± 0.015	0.076 ± 0.012	0.045 ± 0.006	$\mathbf{0.026 \pm 0.003}$
ERCOT	CRPS	0.043 ± 0.001	0.042 ± 0.000	0.040 ± 0.001	$\mathbf{0.032 \pm 0.000}$	0.034 ± 0.000	$\mathbf{0.032 \pm 0.000}$
	NMAE	0.049 ± 0.001	0.051 ± 0.001	0.046 ± 0.001	$\mathbf{0.044 \pm 0.001}$	0.046 ± 0.000	$\mathbf{0.044 \pm 0.001}$
	ECE	0.166 ± 0.007	0.140 ± 0.011	0.169 ± 0.007	$\mathbf{0.018 \pm 0.010}$	0.034 ± 0.013	0.019 ± 0.016

Time Efficiency. As shown in Figure 2, DecoVAE consistently achieves shorter training times than both K^2VAE and CLaST across all four evaluated settings. On average, DecoVAE trains approximately $3.27\times$ faster than K^2VAE and $3\times$ faster than CLaST per epoch, with maximum speedups of $4.43\times$ and $5.08\times$, respectively. Inference-time differences are more dataset-dependent: DecoVAE achieves lower inference time than CLaST in three of the four settings and outperforms K^2VAE on Electricity-L, while incurring a moderate overhead in the remaining cases due to decomposition, frequency filtering, and the tri-stream architecture.

Memory Consumption and Model Size. DecoVAE also provides substantial memory and storage savings (Fig. 3). Across the four evaluated settings, it reduces peak test-time memory consumption by approximately 54–76% relative to K^2VAE and 58–75% relative to CLaST. On Traffic-L, for example, DecoVAE requires only 37 MB of peak test memory, compared with 101 MB for K^2VAE and 148 MB for CLaST. On the larger datasets, DecoVAE has a model size of approximately 7.05 MB, compared with 70.36 MB for K^2VAE and 50.72 MB for CLaST. Thus, DecoVAE is up to $10\times$ smaller than K^2VAE and $7.2\times$ smaller than CLaST, demonstrating a favorable trade-off between forecasting quality and computational efficiency.

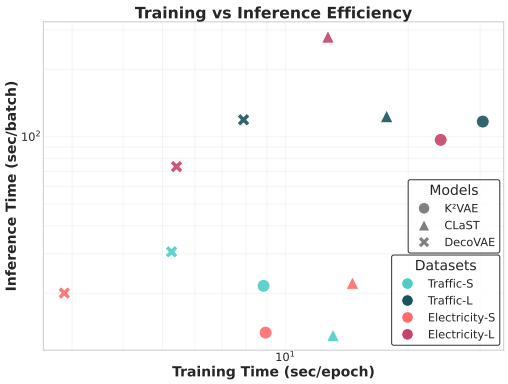


Figure 2: Comparison of training and inference time across Electricity and Traffic: **S** denotes short-term forecasting ($L = 24$), and **L** denotes long-term forecasting ($L = 720$). Lower values indicate better computational efficiency.

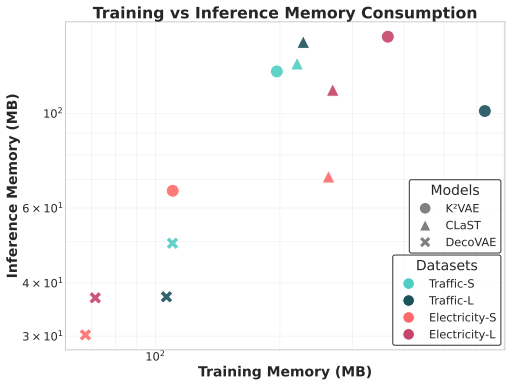


Figure 3: Comparison of training and inference memory consumption on Electricity and Traffic, where **S** denotes short-term ($L = 24$) and **L** long-term ($L = 720$) forecasting. Lower is better.

Qualitative Decomposition Analysis. Beyond computational efficiency, we qualitatively assess whether DecoVAE recovers the intended component structure (Figure 4). We first use synthetic

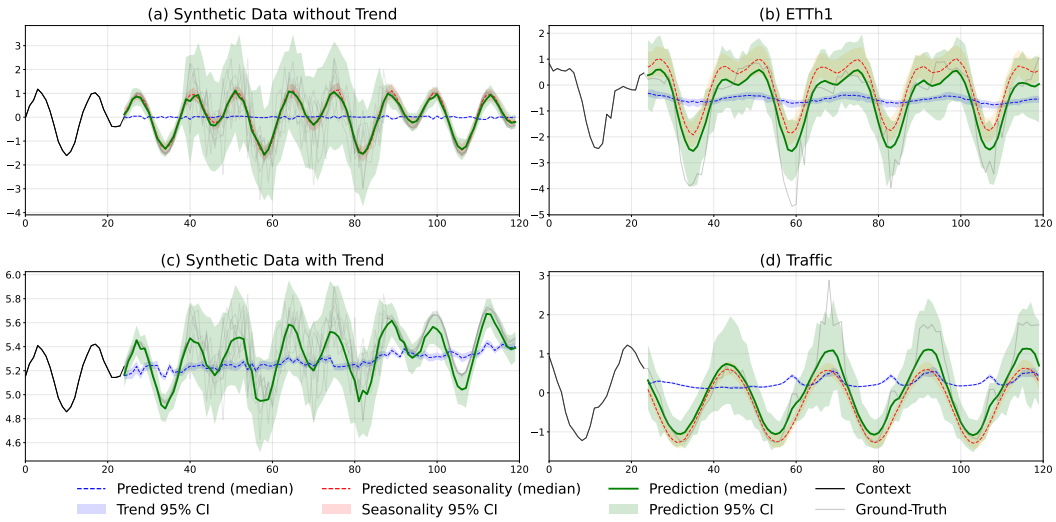


Figure 4: Probabilistic forecasts generated by DecoVAE on synthetic data without (a) and with (c) a trend, and on the real-world ETTh1 (b) and Traffic (d) datasets. Ground-truth observations are overlaid for comparison. The synthetic data exhibit changing variance to illustrate the model's ability to capture shifts in volatility regimes; multiple ground-truth realizations are shown to make these changes more apparent. For the trend variant, the seasonality component is omitted to avoid visual clutter, as it is centered around zero.

data with alternating volatility regimes, generated as specified in Appendix H. We consider both trendless and weakly trending settings. DecoVAE captures the volatility shifts and separates trend from seasonal structure in both cases, including when the true trend is zero. The estimated trend is noisier under the trending high-variance setting. On ETTh1 and Traffic, the model similarly separates slowly varying trends from periodic components and widens predictive intervals when observations depart from these patterns.

Limitations. DecoVAE models stochastic residual uncertainty through heteroscedastic Gaussian marginals with diagonal covariance. While this keeps the model lightweight and computationally efficient, it does not explicitly capture heavy-tailed or asymmetric residual distributions, nor structured cross-channel or cross-horizon dependencies, since the predictive distribution factorizes across time steps and features conditional on the latents. This is a direct consequence of prioritizing lightweight, scalable inference (Section 4.4). Extending the residual stream to a low-rank or block-structured covariance — particularly across the feature dimension, where cross-sensor correlation is physically meaningful for datasets such as Traffic and Electricity — is a natural direction for future work.

5 Conclusion

We introduce **DecoVAE**, a lightweight variational framework that assigns distinct statistical roles to predictable temporal dynamics and residual uncertainty. Trend and seasonality determine the forecast mean, whereas the residual stream models time-varying forecast variance rather than predicting residual realizations directly. Controlled ablations support this design, showing that the proposed residual-volatility formulation improves probabilistic forecasting and calibration over simpler residual uncertainty models. Across seven real-world benchmarks, DecoVAE achieves average relative CRPS improvements of 10.05% for short-term forecasting and 12.28% for long-term forecasting over the strongest competing methods. Compared with the closest competing method, DecoVAE trains approximately $3\times$ faster on average and is about $7.2\times$ smaller on high-dimensional datasets. These results demonstrate that assigning distinct statistical roles to structured temporal dynamics and residual uncertainty provides a favorable balance between probabilistic forecasting accuracy and computational efficiency.

AI USE STATEMENT

In this work, we did not use generative AI tools to generate synthetic datasets, develop theoretical or conceptual frameworks, formulate mathematical claims, provide ingredients for or assist in mathematical proofs, propose or refine hypotheses, design the research methodology or experiments, implement the proposed methods, clean or reformat datasets, or interpret experimental results.

Generative AI tools were used to assist with drafting and editing portions of the manuscript, including language editing, stylistic refinement, improving readability, and refining the presentation and organization of the text. All AI-assisted text was reviewed and revised by the authors to ensure technical accuracy, consistency with the proposed methodology, and faithful representation of the experimental results and conclusions.

The authors take full responsibility for the final content of this work, including all text, scientific claims, results, and artifacts produced with the aid of generative AI.

REFERENCES

Zhongde An, Jinhong You, Jiyanglin Li, Yiming Tang, Wen Li, Heming Du, and Shouguo Du. Fredn: Spectral disentanglement for time series forecasting via learnable frequency decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 19623–19631, 2026.

Konstantinos Barmpas, Na Lee, Yannis Panagakis, Dimitrios Adamos, Nikolaos Laskaris, and Stefanos Zafeiriou. Advancing brainwave modelling with a codebook-based foundation model. In *NeurIPS 2025 Workshop on Foundation Models for the Brain and Body*, 2025.

Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327, 1986.

Youngseog Chung, Willie Neiswanger, Ian Char, and Jeff Schneider. Beyond pinball loss: Quantile methods for calibrated uncertainty quantification. *Advances in Neural Information Processing Systems*, 34:10971–10984, 2021.

Robert B Cleveland, William S Cleveland, Jean E McRae, Irma Terpenning, et al. Stl: A seasonal-trend decomposition. *J. off. Stat*, 6(1):3–73, 1990.

Jonathan Crabbé, Nicolas Huynh, Jan Pawel Stanczuk, and Mihaela Van Der Schaar. Time series diffusion in the frequency domain. In *International Conference on Machine Learning*, pp. 9407–9438. PMLR, 2024.

Jinliang Deng, Feiyang Ye, Du Yin, Xuan Song, Ivor Tsang, and Hui Xiong. Parsimony or capability? decomposition delivers both in long-term time series forecasting. *Advances in Neural Information Processing Systems*, 37:66687–66712, 2024.

Robert F Engle. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the econometric society*, pp. 987–1007, 1982.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Robert J. Hodrick and Edward C. Prescott. Postwar u.s. business cycles: An empirical investigation. *Journal of Money, Credit and Banking*, 29(1):1–16, 1997. ISSN 00222879, 15384616. URL http://www.jstor.org/stable/2953682.

Rob J. Hyndman. *Moving Averages*, pp. 866–869. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-04898-2. doi: 10.1007/978-3-642-04898-2_380. URL https://doi.org/10.1007/978-3-642-04898-2_380.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

Marcel Kollovieh, Abdul Fatir Ansari, Michael Bohlke-Schneider, Jasper Zschiegner, Hao Wang, and Yuyang Bernie Wang. Predict, refine, synthesize: Self-guiding diffusion models for probabilistic time series forecasting. *Advances in Neural Information Processing Systems*, 36:28341–28364, 2023.

Xiangjie Kong, Zhenghao Chen, Weiyao Liu, Kaili Ning, Lechao Zhang, Syauqie Muhammad Marier, Yichen Liu, Yuhao Chen, and Feng Xia. Deep learning for time series forecasting: a survey. *International Journal of Machine Learning and Cybernetics*, 16(7):5079–5112, 2025.

Tim Kreuzer, Jelena Zdravkovic, and Panagiotis Papapetrou. Unpacking the trend: decomposition as a catalyst to enhance time series forecasting models. *Data Mining and Knowledge Discovery*, 39(5):54, 07 2025. doi: 10.1007/s10618-025-01120-8.

Ilya Kuleshov, Evgenia Romanenkova, Vladislav Andreevich Zhuzhel, Galina Boeva, Evgeni Vorsin, and Alexey Zaytsev. DeNOTS: Stable deep neural ODEs for time series. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=SFoDJZ1sSk.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pp. 95–104, 2018.

Peiyuan Liu, Beiliang Wu, Naiqi Li, Tao Dai, Fengmao Lei, Jigang Bao, Yong Jiang, and Shu-Tao Xia. Wftnet: Exploiting global and local periodicity in long-term time series forecasting. In *Icassp 2024-2024 ieee international conference on acoustics, speech and signal processing (icassp)*, pp. 5960–5964. IEEE, 2024.

Junkai Lu, Peng Chen, Chenjuan Guo, Yang Shu, Meng Wang, and Bin Yang. Towards non-stationary time series forecasting with temporal stabilization and frequency differencing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24070–24078, 2026.

Donghao Luo and Xue Wang. Decompnet: Enhancing time series forecasting models with implicit decomposition. Poster presented at the 39th Conference on Neural Information Processing Systems (NeurIPS 2025), December 2025. URL https://neurips.cc/virtual/2025/loc/san-diego/poster/116526.

Alexander Marusov, Dmitry Anikin, Petr Sokerin, Vitaliy Pozdnyakov, Ilya Kuleshov, and Alexey Zaytsev. Clast: Context-aware contrastive vae for probabilistic time series forecasting. *arXiv preprint arXiv:2608.20025*, 2026.

Daniel B Nelson. Conditional heteroskedasticity in asset returns: A new approach. *Econometrica: Journal of the econometric society*, pp. 347–370, 1991.

David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, volume 1, pp. 55–60. IEEE, 1994.

Tommaso Proietti and Diego J. Pedregal. Seasonality in high frequency time series. *Econometrics and Statistics*, 27:62–82, 2023. ISSN 2452-3062. doi: https://doi.org/10.1016/j.ecosta.2022.02.001. URL https://www.sciencedirect.com/science/article/pii/S2452306222000090.

Zhongzheng Qiao, Sheng Pan, Anni Wang, Viktoriya Zhukova, Yong Liu, Xudong Jiang, Qingsong Wen, Mingsheng Long, Ming Jin, and Chenghao Liu. It's time: Towards the next generation of time series forecasting benchmarks. *arXiv preprint arXiv:2602.12147*, 2026.

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 *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3):1181–1191, 2020.

Maximilian Seitzer, Arash Tavakoli, Dimitrije Antic, and Georg Martius. On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks. *arXiv preprint arXiv:2203.09168*, 2022.

Satya Narayan Shukla and Benjamin Marlin. Heteroscedastic temporal variational autoencoder for irregularly sampled time series. In *International Conference on Learning Representations*, 2021.

Artyom Stitsyuk and Jaesik Choi. xpatch: Dual-stream time series forecasting with exponential seasonal-trend decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 20601–20609, 2025.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and JUN ZHOU. Timemixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, 2024.

Zhiyuan Wang, Xovee Xu, Weifeng Zhang, Goce Trajcevski, Ting Zhong, and Fan Zhou. Learning latent seasonal-trend representations for time series forecasting. *Advances in Neural Information Processing Systems*, 35:38775–38787, 2022.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

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.

Xingjian Wu, Xiangfei Qiu, Hongfan Gao, Jilin Hu, Bin Yang, and Chenjuan Guo. K2vae: A koopman-kalman enhanced variational autoencoder for probabilistic time series forecasting. *arXiv preprint arXiv:2505.23017*, 2025.

Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with $10k$ parameters. In *The Twelfth International Conference on Learning Representations*, 2024.

Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6199–6203. IEEE, 2020.

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 *International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=87CYNyCGOo.

Michaël Zamo and Philippe Naveau. Estimation of the continuous ranked probability score with limited information and applications to ensemble weather forecasts. *Mathematical Geosciences*, 50(2):209–234, 2018.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

Jiawen Zhang, Xumeng Wen, Zhenwei Zhang, Shun Zheng, Jia Li, and Jiang Bian. Probts: Benchmarking point and distributional forecasting across diverse prediction horizons. *Advances in Neural Information Processing Systems*, 37:48045–48082, 2024a.

Qianru Zhang, Yuting Sun, Honggang Wen, Peng Yang, Xinzhu Li, Ming Li, Kwok-Yan Lam, Siu-Ming Yiu, and Hongzhi Yin. Time series analysis in frequency domain: A survey of open challenges, opportunities and benchmarks. *arXiv preprint arXiv:2504.07099*, 2025.

Xi Nicole Zhang, Yuan Pu, Yuki Kawamura, Andrew Loza, Yoshua Bengio, Dennis Shung, and Alexander Tong. Trajectory flow matching with applications to clinical time series modelling. *Advances in Neural Information Processing Systems*, 37:107198–107224, 2024b.

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 *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

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*, pp. 27268–27286. PMLR, 2022.

A Related Work

A.1 Trend–Seasonality Decomposition Techniques

Classical methods like STL (Cleveland et al., 1990) and moving-average decomposition (Hyndman, 2011; Kreuzer et al., 2025) separate trend (long-term direction) from seasonality (fixed-interval repetitions). Recent work enhances this: FedFormer (Zhou et al., 2022) adaptively combines multi-scale moving averages, while Learnable Decomposition (Yu et al., 2024) replaces fixed kernels with trainable convolutions. Empirical studies confirm that explicit trend-seasonality separation consistently improves forecasting performance (Luo & Wang, 2025).

A.2 Loss Function

Loss design critically affects decomposition-based forecasting. LaST (Wang et al., 2022) applies auxiliary objectives to enforce smooth trend latents and periodic seasonal latents. DBLoss (Qiu et al., 2025) decomposes both predictions and targets via exponential smoothing, computing separate trend/seasonal losses. This explicit decomposition in the objective accelerates convergence and improves accuracy over standard MSE, even when the model internally disentangles components.

B Comparison of decomposition methods

To isolate the effect of the trend–seasonality decomposition independently of residual-volatility modeling, we conduct this ablation using the Trend+Season backbone only, with the FITS-style seasonal representation fixed across all variants. The residual-volatility stream is excluded from this experiment; therefore, the absolute values are not directly comparable to the full DecoVAE results reported in Tables 1 and 2.

Table 4 compares moving average (MA), FEDformer-style mixture-of-experts (MoE), and exponential moving average (EMA) decomposition within DecoVAE. Overall, MA provides the most consistent performance across datasets, achieving the lowest mean CRPS on ETTh1, ETTh2, and ETTm1, while remaining competitive on ETTm2. MoE slightly outperforms MA in CRPS on ETTm2 and achieves the lowest NMAE on ETTh1 and ETTm1, whereas MA obtains the best NMAE on ETTm2; on ETTh2, MoE and EMA achieve nearly identical NMAE. These results indicate that no decomposition method dominates across all metrics, while the simple MA decomposition provides a strong and stable accuracy–complexity trade-off, motivating its use in the final model.

C Architecture details

C.1 Standard MLP Architecture

Throughout this paper, MLP refers to the standard Multilayer Perceptron architecture used in our implementation. It consists of an initial linear projection followed by LayerNorm, a nonlinear activation, and dropout, followed by hidden layers with the same structure and residual connections. A final linear projection maps the hidden representation to the output dimension.

Table 4: Comparison of **decomposition methods in DecoVAE** (T = 96, L = 720) on long-term forecasting task. The residual-volatility stream is excluded to isolate the effect of the decomposition strategy. Lower CRPS and NMAE scores indicate better performance. Results are reported as mean ± standard error over three independent runs. **Boldface** highlights the best-performing method, while underlining denotes the second-best result.

Dataset	Metric	MA	MoE	EMA
ETTh1	CRPS	$\mathbf{0.298 \pm 0.003}$	0.304 ± 0.009	0.356 ± 0.003
	NMAE	0.360 ± 0.004	$\mathbf{0.357 \pm 0.003}$	0.393 ± 0.002
ETTh2	CRPS	$\mathbf{0.498 \pm 0.006}$	0.510 ± 0.006	0.515 ± 0.006
	NMAE	0.585 ± 0.002	0.584 ± 0.005	$\mathbf{0.584 \pm 0.004}$
ETTm1	CRPS	$\mathbf{0.297 \pm 0.004}$	0.301 ± 0.001	0.305 ± 0.001
	NMAE	0.329 ± 0.004	$\mathbf{0.327 \pm 0.002}$	0.330 ± 0.001
ETTm2	CRPS	0.468 ± 0.005	$\mathbf{0.467 \pm 0.007}$	0.472 ± 0.001
	NMAE	$\mathbf{0.524 \pm 0.001}$	0.536 ± 0.008	0.528 ± 0.001

Unless otherwise specified, we use the default configuration of hidden dimension 128, 3 hidden layers, dropout rate 0.05, and ReLU activation. The input and output dimensions are determined by the operation in which the MLP is used.

C.2 Trend component

The trend X_t is processed by a shared encoder applied independently along the channel dimension:

$$H = \mathrm{MLP}^{enc}_{shared}(X_t), \quad H \in \mathbb{R}^{T \times d}, \tag{5}$$

where T is the context length and d is the embedding dimension. Following the standard VAE structure, we learn parameters of a Gaussian latent distribution. The mean and variance are computed from the encoder output with two separate trainable linear layers Linear_μ and Linear_σ:

$$\mu = \mathrm{Linear}_\mu(H) \in \mathbb{R}^{T \times d}, \tag{6}$$

$$\sigma^2 = \mathrm{Softplus}\big(\mathrm{Linear}_\sigma(H)\big) \in \mathbb{R}^{T \times d}. \tag{7}$$

Reparameterization and Sampling We use the standard reparameterization trick to obtain latent trend representations:

$$Z_t = \mu + \sigma \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \tag{8}$$

where $Z_t \in \mathbb{R}^{T \times d}$ is the latent trajectory, and $\odot$ denotes element-wise multiplication.

Decoding The latent trajectory is decoded to reconstruct the trend component. This is performed via an MLP operating along the feature dimension (from embedding space to the original feature space):

$$\widehat{X}_t = \mathrm{MLP}^{\mathrm{dec}}_{\mathrm{trend}}(Z_t) \in \mathbb{R}^{T \times F}. \tag{9}$$

Trend Forecasting During forecasting, the latent trend trajectory Z_t is extrapolated to the prediction horizon by a dedicated predictor $\mathcal{P}_t$, which comprises two MLPs operating along the temporal ($T \to L$) and feature ($d \to F$) dimensions. Thus, $\mathcal{P}_t = \mathrm{MLP}^{pred}_{feat,t} \circ \mathrm{MLP}^{pred}_{time,t}$, where:

$$\widehat{Z}_t = \mathrm{MLP}^{pred}_{time,t}(Z_t) \in \mathbb{R}^{L \times d} \tag{10}$$

$$\widehat{Y}_t = \mathrm{MLP}^{pred}_{feat,t}(\widehat{Z}_t) \in \mathbb{R}^{L \times F}. \tag{11}$$

Unlike the decoder which reconstructs the observed window, these prediction-specific MLPs learn temporal extrapolation from context length T to forecast horizon L, allowing the model to capture temporal dynamics that may differ between the observed and future periods.

C.3 Seasonal Component

To model seasonal dynamics, we operate directly in the frequency domain. This choice is motivated by the fact that periodic patterns correspond to localized structures in the spectrum, making them

easier to represent and manipulate compared to the time domain (Liu et al., 2024; Crabbé et al., 2024).

We evaluated three alternative parameterizations for the frequency-domain seasonal component, ultimately selecting a *complex-valued FITS-style approach* (Xu et al., 2024). By modeling the signal directly within the complex Fourier domain via complex-valued linear layers, this method preserves the Fourier transform's native structure while jointly encoding amplitude and phase into a unified latent representation. To isolate the effect of the seasonal parameterization, all variants in this ablation use the same Trend+Season backbone and exclude the residual-volatility stream. Consequently, these results are intended for within-ablation comparison and are not directly comparable to the full DecoVAE results in Table 1.

Frequency Domain Encoding Let $X_s \in \mathbb{R}^{T\times F}$ denote the seasonal component. After applying normalization, we transform the seasonal component to the frequency domain:

$$X_s^{(freq)} = \mathcal{F}(X_s) \in \mathbb{C}^{K\times F}, \tag{12}$$

where $\mathcal{F}$ denotes the FFT and $K = \lfloor T/2 \rfloor + 1$ represents the number of frequency bins. The spectrum is encoded by a shared complex-valued linear encoder applied independently across frequency bins:

$$H = \text{Linear}_{enc}(X_s^{(freq)}) \in \mathbb{C}^{K\times d}. \tag{13}$$

Complex Latent Distribution We parameterize a complex Gaussian distribution to preserve both amplitude and phase information. The mean is computed directly from the complex encoder output via a complex linear layer. The variance, which must be real and positive, is computed from the concatenated real $\Re(H)$ and imaginary $\Im(H)$ parts:

$$\mu = \text{Linear}_{\mu}(H) \in \mathbb{C}^{K\times d}, \tag{14}$$

$$\sigma^2 = \text{Softplus}\left(\text{Linear}_{\sigma}([\Re(H), \Im(H)])\right) \in \mathbb{R}^{K\times d}. \tag{15}$$

Sampling is performed via complex reparameterization:

$$Z_s = \mu + \sqrt{\frac{\sigma^2}{2}} \odot (\epsilon_r + i\,\epsilon_i) \tag{16}$$

where $Z_s \in \mathbb{C}^{K\times d}$ and $\epsilon_r, \epsilon_i \sim \mathcal{N}(0, I)$. The symbol $\odot$ denotes element-wise multiplication.

Spectral Decoding The latent representation is decoded back to the frequency domain:

$$\widehat{X}_s^{(freq)} = \text{Linear}_{dec}(Z_s) \in \mathbb{C}^{K\times F}. \tag{17}$$

Seasonal Forecasting For forecasting a future window of length L, we use a separate predictor $\mathcal{P}_s$. It consists of complex-valued linear layers that that upsamples the latent representation to $M = \lfloor L/2 \rfloor + 1$ frequency bins corresponding to the prediction horizon:

$$\widehat{Z}_s = \text{Linear}_{bins,s}^{pred}(Z_s) \in \mathbb{C}^{M\times d}. \tag{18}$$

Then, $\mathcal{P}_s$ decodes this representation to the output channels via another complex-valued linear transformation:

$$\widehat{Y}_s^{(freq)} = \text{Linear}_{feat,s}^{pred}(\widehat{Z}_s) \in \mathbb{C}^{M\times F}. \tag{19}$$

The final seasonal forecast is obtained by inverse Fourier transform and and subsequent denormalization:

$$\widehat{Y}_s = \mathcal{F}^{-1}(\widehat{Y}_s^{(freq)}) \in \mathbb{R}^{L\times F}. \tag{20}$$

Such architecture enables efficient frequency-domain interpolation for forecasting, where the learned latent representation captures the essential seasonal patterns that can be extrapolated to future time horizons through spectral upsampling.

C.4 VARIANCE PREDICTION

Direct modeling of unpredictable residual realizations provides little reconstructive signal for learning informative latent dynamics. Instead, we model its statistical properties — specifically, the time-varying variance σ_t^2 that characterizes volatility regimes.

Volatility Encoding The Residual component is processed by a shared MLP encoder applied independently along the channel dimension:

$$H_\varepsilon = \text{MLP}^{enc}_{shared}(\varepsilon) \in \mathbb{R}^{T\times d}. \tag{21}$$

Following the standard VAE structure, we parameterize a Gaussian latent distribution:

$$\mu_\varepsilon = \text{Linear}_\mu(H_\varepsilon) \in \mathbb{R}^{T\times d}, \tag{22}$$

$$\sigma_\varepsilon^2 = \text{Softplus}\left(\text{Linear}_\sigma(H_\varepsilon)\right) \in \mathbb{R}^{T\times d}. \tag{23}$$

Sampling is performed via reparameterization:

$$Z_\varepsilon = \mu_\varepsilon + \sigma_\varepsilon \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{24}$$

Variance Decoding The latent representation $Z_\varepsilon \in \mathbb{R}^{T\times d}$ is decoded to reconstruct the variance profile:

$$\hat{\sigma}_t^2 = \text{Softplus}\left(\text{MLP}^{dec}_{noise}(Z_\varepsilon)\right) \in \mathbb{R}_+^{T\times F}. \tag{25}$$

Variance Forecasting During forecasting, the latent noise representation Z_ε is extrapolated to the prediction horizon by a dedicated volatility predictor $\mathcal{P}_n$. The predictor first applies a temporal MLP to extrapolate the latent representation from the context length T to the forecast horizon L, followed by a noise decoder. Thus, $\mathcal{P}_n = \text{Softplus} \circ \text{MLP}^{dec}_{noise} \circ \text{MLP}^{pred}_{time,n}$, where:

$$\widehat{Z}_\varepsilon = \text{MLP}^{pred}_{time,n}(Z_\varepsilon) \in \mathbb{R}^{L\times d} \tag{26}$$

$$\widehat{\sigma}_Y^2 = \text{Softplus}\left(\text{MLP}^{dec}_{noise}(\widehat{Z}_\varepsilon)\right) \in \mathbb{R}_+^{L\times F}. \tag{27}$$

During inference, noise samples are obtained as:

$$\widehat{\varepsilon}_Y = \widehat{\sigma}_Y \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

D HODRICK–PRESCOTT FILTER APPROXIMATION IN LATENT SPACE

Theorem D.1 (Hodrick–Prescott Filter Approximation in Latent Space)**.** *Assume that $\mathcal{G}_t(z_{t,\tau}) = W_t z_{t,\tau}$ is a linear trend decoder, where $W_t \in \mathbb{R}^{F\times d}$ has full column rank, and let $\alpha_1 = 0$. Recall that $\mathcal{R}_2(Z_t) = \frac{1}{T-2}\sum_{\tau=2}^{T-1} \|\Delta^2 z_{t,\tau}\|^2$, where $\Delta^2 z_{t,\tau} = z_{t,\tau+1} - 2z_{t,\tau} + z_{t,\tau-1}$, and define $J[Z_t] = \sum_{\tau=1}^{T} \|X_{t,\tau} - W_t z_{t,\tau}\|^2 + \alpha_2 \mathcal{R}_2(Z_t)$. For the normalized Hodrick–Prescott objective $J_{\text{HP}}[W_t Z_t] = \sum_{\tau=1}^{T} \|X_{t,\tau} - W_t z_{t,\tau}\|^2 + \frac{\lambda_{\text{HP}}}{T-2}\sum_{\tau=2}^{T-1} \|W_t \Delta^2 z_{t,\tau}\|^2$, we have*

$$(\lambda_{\text{HP}}\sigma_{\min}^2(W_t) - \alpha_2)\mathcal{R}_2(Z_t) \le J_{\text{HP}}[W_t Z_t] - J[Z_t] \le (\lambda_{\text{HP}}\sigma_{\max}^2(W_t) - \alpha_2)\mathcal{R}_2(Z_t).$$

In particular, for $\lambda_{\text{HP}} = \alpha_2/\sigma_{\min}^2(W_t)$,

$$0 \le J_{\text{HP}}[W_t Z_t] - J[Z_t] \le \alpha_2(\kappa^2(W_t) - 1)\mathcal{R}_2(Z_t),$$

where $\kappa(W_t) = \sigma_{\max}(W_t)/\sigma_{\min}(W_t)$.

Proof. The proof follows by comparing the normalized curvature terms and applying singular value bounds.

Step 1: Linear Decoder Substitution. For the linear decoder $\mathcal{G}_t(z_{t,\tau}) = W_t z_{t,\tau}$, the normalized Hodrick–Prescott objective is

$$J_{\text{HP}}[W_t Z_t] = \sum_{\tau=1}^{T} \|X_{t,\tau} - W_t z_{t,\tau}\|^2 + \frac{\lambda_{\text{HP}}}{T-2}\sum_{\tau=2}^{T-1} \|W_t z_{t,\tau+1} - 2W_t z_{t,\tau} + W_t z_{t,\tau-1}\|^2. \tag{28}$$

By linearity,

$$W_t z_{t,\tau+1} - 2W_t z_{t,\tau} + W_t z_{t,\tau-1} = W_t \Delta^2 z_{t,\tau}, \tag{29}$$

and therefore

$$J_{\mathrm{HP}}[W_t Z_t] = \sum_{\tau=1}^{T} \|X_{t,\tau} - W_t z_{t,\tau}\|^2 + \frac{\lambda_{\mathrm{HP}}}{T-2} \sum_{\tau=2}^{T-1} \|W_t \Delta^2 z_{t,\tau}\|^2. \tag{30}$$

Step 2: Functional Difference. For $\alpha_1 = 0$, the proposed latent objective is

$$J[Z_t] = \sum_{\tau=1}^{T} \|X_{t,\tau} - W_t z_{t,\tau}\|^2 + \alpha_2 \mathcal{R}_2(Z_t), \tag{31}$$

where $\mathcal{R}_2(Z_t) = \frac{1}{T-2} \sum_{\tau=2}^{T-1} \|\Delta^2 z_{t,\tau}\|^2$. Hence,

$$J_{\mathrm{HP}}[W_t Z_t] - J[Z_t] = \frac{1}{T-2} \sum_{\tau=2}^{T-1} \left(\lambda_{\mathrm{HP}} \|W_t \Delta^2 z_{t,\tau}\|^2 - \alpha_2 \|\Delta^2 z_{t,\tau}\|^2 \right). \tag{32}$$

Step 3: Singular Value Bounds. Since W_t has full column rank, for any $v \in \mathbb{R}^d$,

$$\sigma_{\min}^2(W_t)\|v\|^2 \le \|W_t v\|^2 \le \sigma_{\max}^2(W_t)\|v\|^2. \tag{33}$$

Setting $v = \Delta^2 z_{t,\tau}$ gives

$$\big(\lambda_{\mathrm{HP}} \sigma_{\min}^2(W_t) - \alpha_2\big)\mathcal{R}_2(Z_t) \le J_{\mathrm{HP}}[W_t Z_t] - J[Z_t] \le \big(\lambda_{\mathrm{HP}} \sigma_{\max}^2(W_t) - \alpha_2\big)\mathcal{R}_2(Z_t). \tag{34}$$

Step 4: Smoothing Parameter Calibration. Choose $\lambda_{\mathrm{HP}} = \alpha_2 / \sigma_{\min}^2(W_t)$. Then the lower-bound coefficient vanishes, yielding

$$J_{\mathrm{HP}}[W_t Z_t] - J[Z_t] \ge 0. \tag{35}$$

For the upper bound,

$$\begin{aligned} J_{\mathrm{HP}}[W_t Z_t] - J[Z_t] &\le \alpha_2 \left(\frac{\sigma_{\max}^2(W_t)}{\sigma_{\min}^2(W_t)} - 1 \right) \mathcal{R}_2(Z_t) \\ &= \alpha_2 \big(\kappa^2(W_t) - 1\big)\mathcal{R}_2(Z_t), \end{aligned} \tag{36}$$

where $\kappa(W_t) = \sigma_{\max}(W_t)/\sigma_{\min}(W_t)$. Therefore,

$$0 \le J_{\mathrm{HP}}[W_t Z_t] - J[Z_t] \le \alpha_2 \big(\kappa^2(W_t) - 1\big)\mathcal{R}_2(Z_t). \tag{37}$$

Thus, under the calibrated choice of λ_{HP}, the discrepancy between the normalized Hodrick–Prescott objective applied to the decoded trend and the proposed latent objective is controlled by the latent curvature regularizer $\mathcal{R}_2(Z_t)$ and the conditioning of the decoder matrix. □

E Motivation for Spectral Selection

The following result characterizes the spectral approximation criterion used by DecoVAE. For simplicity, we consider each feature independently.

Theorem E.1 (Optimal spectral selection)**.** *Let $S = \mathcal{F}(X_{\mathrm{detrend}}) \in \mathbb{C}^{M \times F}$, where $M = \lfloor T/2 \rfloor + 1$. Fix a feature f with $\|S_{:,f}\|_2^2 > 0$. Let powers be $P_{k,f} = |S_{k,f}|^2$, and let $P_{(1),f} \ge \cdots \ge P_{(M),f}$ denote these powers sorted in descending order. For $p_{\mathrm{opt}} \in (0, 1]$, define*

$$K_{\mathrm{opt},f} = \min \left\{ k \in \{1, \ldots, M\} : \sum_{i=1}^{k} P_{(i),f} \ge p_{\mathrm{opt}} \sum_{j=1}^{M} P_{j,f} \right\}.$$

Let Ω_f contain the indices of the $K_{\mathrm{opt},f}$ largest powers, with ties broken arbitrarily, and define the masked spectrum by

$$(S_\Omega)_{k,f} = S_{k,f}\, \mathbf{1}\{k \in \Omega_f\}.$$

Then Ω_f has minimum cardinality among all frequency sets whose masked reconstructions satisfy

$$\|S_{:,f} - (S_\Omega)_{:,f}\|_2^2 \le (1 - p_{\mathrm{opt}})\|S_{:,f}\|_2^2.$$

Furthermore, for any fixed number m of retained frequencies, selecting the m largest powers minimizes the squared reconstruction error in the one-sided spectrum.

Proof. For any candidate frequency set $A \subseteq \{1, \dots, M\}$ with $|A| \le m$, the error after masking is

$$\sum_{k \notin A} P_{k,f} = \sum_{k=1}^{M} P_{k,f} - \sum_{k \in A} P_{k,f} \ge \sum_{k=1}^{M} P_{k,f} - \sum_{i=1}^{m} P_{(i),f}.$$

Equality is attained by retaining the m largest powers. By the definition of $K_{\mathrm{opt},f}$, this choice satisfies the prescribed tolerance for $m = K_{\mathrm{opt},f}$, while no smaller frequency set can satisfy it. □

White-noise interpretation. Consider the observation model $x = s + n$, where $s \in \mathbb{R}^T$ is a structured signal and $n \sim \mathcal{N}(0, \sigma^2 I_T)$. For a fixed frequency set Ω, let $\mathcal{V}_\Omega$ denote the space of real sequences supported on those Fourier frequencies. Writing $c = \mathcal{F}(x)$, orthogonality of the Fourier components gives

$$\widehat{s}_\Omega = \mathcal{F}^{-1}(c_\Omega) = \arg\min_{s \in \mathcal{V}_\Omega} \|x - s\|_2^2 = \arg\max_{s \in \mathcal{V}_\Omega} p(x \mid s).$$

The last equality follows from the squared-error form of the Gaussian log-likelihood (Williams & Rasmussen, 2006).

Thus, the selection rule provides a minimum-cardinality approximation in the one-sided spectral norm, while reconstruction on a fixed frequency set admits a Gaussian maximum-likelihood interpretation.

F Ablation Study on Spectral Parameterizations

We compare real-valued, complex-valued, and amplitude-phase factorized spectral parameterizations for the seasonal component. We compare three seasonal parameterizations and additionally evaluate parameterization-specific spectral regularizers. To isolate the effect of the seasonal parameterization, all variants in this ablation use the same Trend+Season backbone and exclude the residual-volatility stream. Consequently, these results are intended for within-ablation comparison and are not directly comparable to the full DecoVAE results in Table 1.

Complex-valued FITS-style model (Xu et al., 2024). Seasonal embeddings are derived via complex-valued linear layers directly in the frequency domain, preserving the Fourier transform's structure by jointly capturing amplitude and phase. Consequently, spectral regularization enforces explicit energy matching between the input and latent representations. Let E_x and E_z denote the average energy of the input and latent signals, respectively:

$$E_x = \mathbb{E}\left[\|X\|_2^2\right], \tag{38}$$

$$E_z = \mathbb{E}\left[\|Z_s\|_2^2\right]. \tag{39}$$

The regularization term is defined as:

$$\mathcal{L}_{spec}^{\text{FITS}} = (E_x - E_z)^2 + \lambda \cdot \mathbb{E}\left[\|Z_s\|_1\right], \tag{40}$$

which jointly enforces energy preservation while penalizing noisy latent activations.

Real-valued Frequency MLP (FreqMLP).

Decomposing the complex spectrum into real and imaginary parts enables processing via standard real-valued networks, though this sacrifices explicit preservation of the Fourier domain's complex structure. Consequently, the real-valued frequency MLP applies a simple power spectral density (PSD) penalty to the latent representation:

$$\mathrm{PSD}(Z_s) = \frac{|Z_s|^2}{d}, \qquad \mathcal{L}_{spec}^{\text{FreqMLP}} = \sum \mathrm{PSD}(Z_s), \tag{41}$$

which encourages bounded spectral energy and discourages excessive high-frequency amplification.

Amplitude-Phase factorized model (AmpPhase). The Fourier spectrum is decomposed into amplitude and phase, modeled as independent stochastic variables via separate variational encoders for disentangled spectral modeling. Reconstruction occurs separately for magnitude and phase, with the

amplitude loss integrating relative Frobenius error and log-magnitude consistency (Yamamoto et al., 2020):

$$\mathcal{L}_{amp} = \frac{\|\widehat{A} - A\|_F}{\|A\|_F + \epsilon} + \sum \left| \log(\widehat{A} + \epsilon) - \log(A + \epsilon) \right|, \tag{42}$$

while the phase loss enforces angular alignment (Barmpas et al., 2025):

$$\mathcal{L}_{phase} = \sum \left(1 - \cos(\phi - \widehat{\phi}) \right). \tag{43}$$

The total spectral loss is:

$$\mathcal{L}_{spec}^{\text{AmpPhase}} = \mathcal{L}_{amp} + \mathcal{L}_{phase}. \tag{44}$$

Further implementation details of each variant are available in the accompanying GitHub repository[2].

Table 5: Comparison of **different DecoVAE variants (without volatility stream)** on short-term forecasting task. ($T = 96$, $L = 24$). The residual-volatility stream is excluded. Lower CRPS and NMAE scores indicate better performance. Results are reported as mean ± standard error over three independent runs.

Dataset	Metric	FreqMLP (w/o Reg.)	FreqMLP (w/ Reg.)	AmpPhase	FITS (w/o Reg.)	FITS (w/ Reg.)
ETTh1	CRPS	0.229 ± 0.002	0.230 ± 0.000	0.234 ± 0.002	$\underline{0.217 \pm 0.005}$	$\mathbf{0.215 \pm 0.001}$
	NMAE	0.265 ± 0.002	0.267 ± 0.000	0.268 ± 0.002	$\underline{0.262 \pm 0.002}$	$\mathbf{0.259 \pm 0.001}$
ETTh2	CRPS	0.336 ± 0.003	0.345 ± 0.005	0.347 ± 0.001	$\mathbf{0.319 \pm 0.005}$	$\underline{0.327 \pm 0.014}$
	NMAE	$\mathbf{0.381 \pm 0.004}$	0.386 ± 0.008	0.386 ± 0.000	$\underline{0.382 \pm 0.005}$	0.383 ± 0.000
ETTm1	CRPS	$\mathbf{0.184 \pm 0.002}$	0.194 ± 0.001	0.197 ± 0.003	$\underline{0.188 \pm 0.002}$	0.190 ± 0.001
	NMAE	$\mathbf{0.217 \pm 0.001}$	0.229 ± 0.002	0.227 ± 0.003	0.225 ± 0.002	$\underline{0.225 \pm 0.001}$
ETTm2	CRPS	0.246 ± 0.007	0.247 ± 0.001	0.252 ± 0.002	$\mathbf{0.233 \pm 0.001}$	$\underline{0.234 \pm 0.001}$
	NMAE	$\mathbf{0.283 \pm 0.007}$	0.288 ± 0.002	0.286 ± 0.002	$\underline{0.286 \pm 0.001}$	0.287 ± 0.003

An extended ablation study (Table 5) compares seasonal spectral parameterizations, including real-valued FreqMLP, amplitude-phase factorized, and complex-valued FITS-style formulations, both with and without regularization. To isolate the effect of different seasonal modeling techniques on mean modeling specifically, we turn off the volatility stream in this study. The overall results support the FITS-style parameterization as a strong default.

G HYPERPARAMETER SELECTION

The first- and second-order trend regularization weights, α_1 and α_2, are both searched continuously over $\alpha_1, \alpha_2 \in [0, 20]$.

The weights for the ELBO term (λ_{ELBO}), volatility smoothness (λ_{smooth}) and prediction (λ_{pred}) terms are similarly tuned over the range $[0, 20]$.

In addition, the energy threshold used to determine the optimal number K_{opt} of seasonal frequency components is selected from the discrete set $p_{opt} \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$, and the moving-average kernel size is optimized over odd integers $k \in \{5, 7, 9, \ldots, 49\}$.

The remaining hyperparameters, including the hidden dimension (128) and learning rate (10^{-3}), are kept fixed. Models are trained using the AdamW optimizer with a maximum of 300 epochs and a batch size of 32. Training is stopped early based on validation CRPS, using a patience of 10 epochs. During evaluation, we use a test batch size of 512.

[2] https://anonymous.4open.science/r/DecoVAE-3387/

H Dataset Specifications

Our experimental study uses seven widely-adopted multivariate time series datasets with varying dimensionality, granularity, and domains (Wu et al., 2025; Zhang et al., 2024a; Zhou et al., 2021; Wu et al., 2021):

- **ETT:** Four variants (ETTh1/h2 hourly, ETTm1/m2 15-min) of transformer temperature metrics (2016–2018).
- **Electricity:** Hourly consumption from 321 clients (2012–2014).
- **ERCOT:** hourly electricity load across 8 U.S. regions, 2004–2021.
- **Traffic:** Hourly road occupancy from 862 San Francisco Bay Area sensors (2015–2016).
- **Synthetic data:** The synthetic observations are generated as

$$y_{i,t} = \mu_t + \epsilon_{i,t}, \qquad \epsilon_{i,t} \sim \mathcal{N}(0, \sigma_t^2),$$

where

$$\mu_t = \sin\left(\frac{\pi t}{12}\right) + 1.8\cos\left(\frac{\pi t}{6}\right) + \text{trend}_t, \qquad t = 0, \ldots, S-1,$$

with $S = 10000$ denoting the number of generated time steps. The noise standard deviation σ_t alternates between 0.15 and 1.5 every 48 time steps. For the trend variant, $trend_t$ is defined as:

$$\text{trend}_t = \exp\left(\frac{5t}{S-1}\right) - 1.$$

We augment each series with 11 time-dependent covariates: sinusoidal encodings for yearly to daily periodicities and a normalized time-delta feature. Following standard practice, we use chronological splitting (70% train, 10% validation, 20% test).

I Performance Metrics

We quantify forecasting quality through two established metrics that assess complementary aspects of prediction performance:

Normalized Mean Absolute Error (NMAE). This metric evaluates point forecast accuracy by normalizing absolute deviations relative to the true values:

$$\text{NMAE}(x, \hat{x}) = \frac{1}{N}\sum_{n=1}^{N} \frac{\sum_{i=1}^{F} |x_i^n - \hat{x}_i^n|}{\sum_{i=1}^{F} |x_i^n|},$$

where x_i^n represents the ground truth and $\hat{x}_i^n$ is the predicted median from 100 sampled trajectories.

Continuous Ranked Probability Score (CRPS). To assess the quality of predictive distributions, we use CRPS, which compares the cumulative distribution function $F(z)$ against the empirical distribution of observations:

$$\text{CRPS} = \int_{\mathbb{R}} \big(F(z) - \mathbb{I}\{x \le z\}\big)^2 dz.$$

We compute this integral using Monte Carlo approximation following Zamo & Naveau (2018). Both metrics are negatively oriented, with smaller values indicating superior performance.

Expected Calibration Error (ECE). To evaluate calibration of the predictive distribution, we use the Expected Calibration Error (ECE) (Chung et al., 2021), defined over predictive quantiles. Let $\widehat{q}_{i,\tau,f}(p)$ denote the predicted p-quantile for target $Y_{i,\tau,f}$. The empirical coverage at probability level $p \in (0, 1)$ is

$$\widehat{C}(p) = \frac{1}{NLF}\sum_{i=1}^{N}\sum_{\tau=1}^{L}\sum_{f=1}^{F} \mathbb{I}\left[Y_{i,\tau,f} \le \widehat{q}_{i,\tau,f}(p)\right]. \tag{45}$$

Accordingly, ECE measures the average absolute deviation between nominal and empirical coverage over all probability levels:

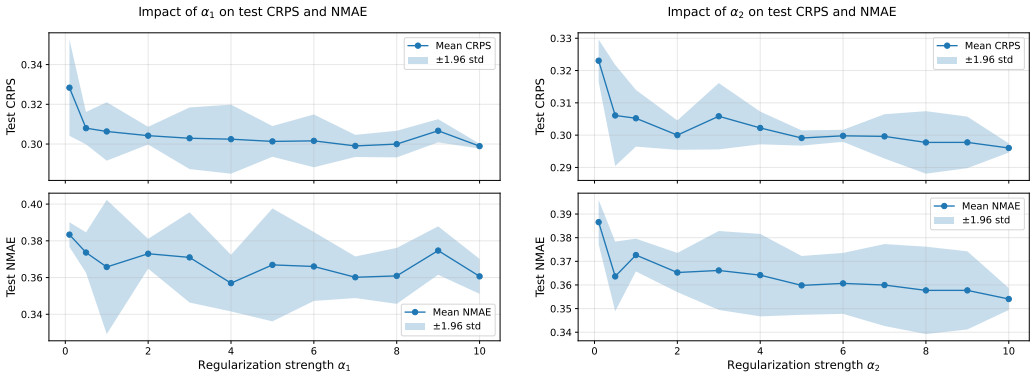
$$\mathrm{ECE} = \int_0^1 \left| \widehat{C}(p) - p \right| \, dp. \tag{46}$$

Lower ECE indicates better calibration, with zero corresponding to perfect empirical quantile calibration.

J Additional ablations

J.1 Trend Regularization Effect

To investigate the impact of our Hodrick-Prescott-inspired trend regularization, we conduct an ablation study varying the regularization strengths α_1 and α_2 on the ETTh1 dataset ($T = 96$, $L = 720$). Figure 5a and Figure 5b demonstrate that both regularization terms improve forecasting performance.



(a) Effect of α_1 (b) Effect of α_2

Figure 5: Effect of first- (a) and second-order (b) latent trend regularization on long-term forecasting performance on ETTh1 ($T = 96$, $L = 720$).

Without first-order regularization ($\alpha_1 = 0$), the model achieves CRPS of 0.33 and NMAE of 0.39. Introducing moderate regularization ($\alpha_1 \in [0.5, 7.0]$) yields improvements of approximately 9% in CRPS and 8% in NMAE, stabilizing performance at CRPS ≈ 0.30 and NMAE ≈ 0.36. Similarly, second-order regularization provides consistent gains, with performance improving from the unregularized baseline when $\alpha_2 \in [1.0, 5.0]$. These results complement Theorem D.1 by showing that the proposed differential regularization is also beneficial empirically.

J.2 Volatility Regularization Effect

Table 6 evaluates the effect of the volatility regularization on short-term forecasting performance. Overall, introducing volatility regularization improves or preserves performance across all four datasets. The largest gains are observed on Electricity, where CRPS decreases from 0.079 to 0.075 and NMAE from 0.107 to 0.100, and on Traffic, where CRPS improves from 0.165 to 0.161 and NMAE from 0.208 to 0.204. Smaller improvements are observed on ETTh2, while performance on ERCOT remains essentially unchanged within the reported precision. These results indicate that regularizing the predicted volatility profile can improve forecast quality without introducing a noticeable performance trade-off, with the strongest benefit observed on datasets exhibiting more pronounced variability.

Table 6: Comparison of forecasting performance for DecoVAE with and without Volatility regularization across different datasets ($T = 96$, $L = 24$). Lower CRPS and NMAE scores indicate better performance. Results are reported as mean $\pm$ standard error, computed from **three independent runs** involving model retraining and evaluation. **Boldface** highlights the best-performing method, while underlining denotes the second-best result.

Dataset	Metric	w/o Volatility Reg.	w/ Volatility Reg.
Electricity	CRPS	$\underline{0.079 \pm 0.001}$	$\mathbf{0.075 \pm 0.000}$
	NMAE	$\underline{0.107 \pm 0.002}$	$\mathbf{0.100 \pm 0.000}$
Traffic	CRPS	$\underline{0.165 \pm 0.001}$	$\mathbf{0.161 \pm 0.002}$
	NMAE	$\underline{0.208 \pm 0.003}$	$\mathbf{0.204 \pm 0.002}$
ETTh2	CRPS	$\underline{0.297 \pm 0.005}$	$\mathbf{0.294 \pm 0.001}$
	NMAE	$\underline{0.381 \pm 0.006}$	$\mathbf{0.379 \pm 0.002}$
ERCOT	CRPS	$\underline{0.032 \pm 0.001}$	$\mathbf{0.032 \pm 0.000}$
	NMAE	$\underline{0.044 \pm 0.002}$	$\mathbf{0.044 \pm 0.001}$

J.3 PIT HISTOGRAMS

Figure 6 shows qualitative calibration analysis based on PIT histograms and their corresponding CDFs. The Trend-only variant exhibits pronounced deviations from the ideal uniform distribution, while adding seasonality progressively improves calibration. The full DecoVAE produces PIT distributions and CDFs closest to the uniform reference across the four datasets. On ETTh2, the moderate concentration around 0.5 suggests slight overdispersion. Together, the numerical and graphical analyses consistently support the contribution of the residual-volatility stream to probabilistic calibration and forecasting quality.

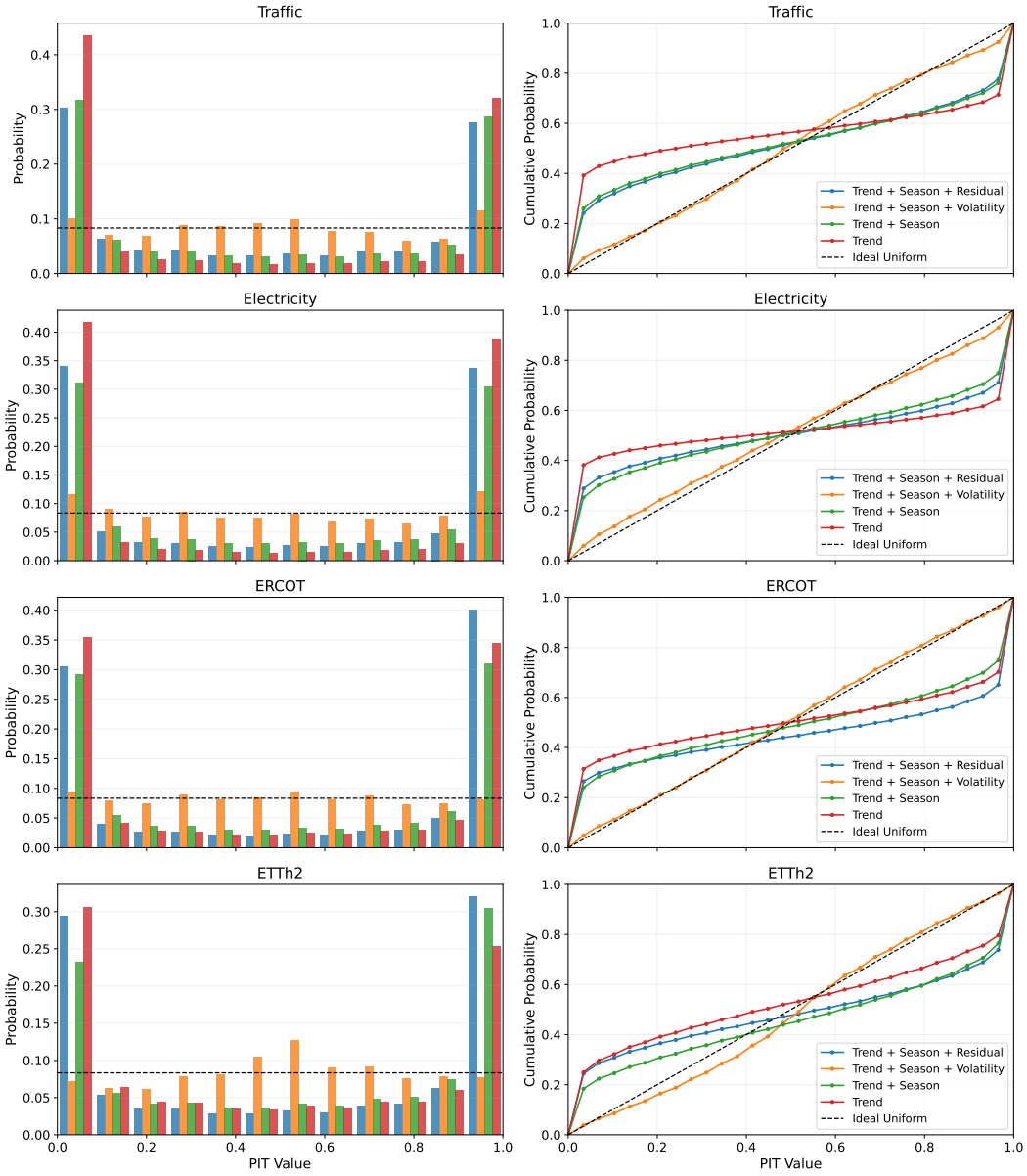


Figure 6: PIT diagnostics for different model variants across four datasets. Left: PIT histograms, with the dashed horizontal line representing the ideal uniform distribution. Right: empirical cumulative distribution functions (CDFs) of the PIT values, with the dashed diagonal indicating ideal calibration.