

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/390592222>

# Probabilistic Traffic Forecasting with Dynamic Regression

Article in *Transportation Science* · April 2025

DOI: 10.1287/trsc.2024.0560

CITATIONS

0

READS

239

3 authors:



Vincent Zheng

McGill University

12 PUBLICATIONS 127 CITATIONS

SEE PROFILE



Seongjin Choi

University of Minnesota

41 PUBLICATIONS 413 CITATIONS

SEE PROFILE



Lijun Sun

McGill University

182 PUBLICATIONS 5,579 CITATIONS

SEE PROFILE

# Probabilistic Traffic Forecasting with Dynamic Regression

**Vincent Zhihao Zheng**

ZHHAO.ZHENG@MAIL.MCGILL.CA

*Department of Civil Engineering*

*McGill University*

*817 Sherbrooke Street West, Montréal, Québec H3A 0C3, Canada*

**Seongjin Choi**

CHOIS@UMN.EDU

*Department of Civil, Environmental, and Geo- Engineering*

*University of Minnesota*

*500 Pillsbury Dr. SE, Minneapolis, MN 55455, USA*

**Lijun Sun\***

LIJUN.SUN@MCGILL.CA

*Department of Civil Engineering*

*McGill University*

*817 Sherbrooke Street West, Montréal, Québec H3A 0C3, Canada*

## Abstract

This paper proposes a dynamic regression (DR) framework that enhances existing deep spatiotemporal models by incorporating structured learning for the error process in traffic forecasting. The framework relaxes the assumption of time independence by modeling the error series of the base model (i.e., a well-established traffic forecasting model) using a matrix-variate autoregressive (AR) model. The AR model is integrated into training by redesigning the loss function. The newly designed loss function is based on the likelihood of a non-isotropic error term, enabling the model to generate probabilistic forecasts while preserving the original outputs of the base model. Importantly, the additional parameters introduced by the DR framework can be jointly optimized alongside the base model. Evaluation on state-of-the-art (SOTA) traffic forecasting models using speed and flow datasets demonstrates improved performance, with interpretable AR coefficients and spatiotemporal covariance matrices enhancing the understanding of the model.

**Keywords:** Probabilistic traffic forecasting, dynamic regression, error correlation

## 1 Introduction

Traffic forecasting is a multivariate and multistep time series forecasting challenge that is a fundamental task within intelligent transportation systems (ITS). It has numerous applications, including trip planning, travel time estimation, and traffic flow management, among others (Vlahogianni et al., 2014). Imagine a traffic network equipped with  $N$  sensors, collecting traffic data within a matrix of dimensions  $N \times T$  over a total observation span of  $T$ . The ultimate objective of traffic forecasting is to anticipate the traffic conditions for  $Q$  future intervals based on the most recent  $P$  historical intervals.

---

\*. Corresponding author.

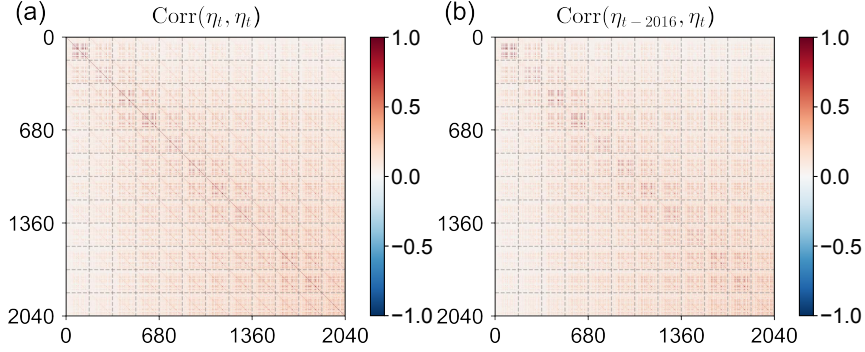


Fig. 1: Correlation matrices of residuals obtained from the original ASTGCN model on the PEMS08 dataset, where  $\boldsymbol{\eta}_t = \text{vec}(\mathbf{R}_t) \in \mathbb{R}^{NQ}$  is the vectorized residual ( $N = 170$  and  $Q = 12$ ). (a) Contemporaneous correlations. (b) Cross-correlations at lag 2016 (i.e., one week).

Deep learning (DL) models are increasingly popular in traffic forecasting due to their ability to capture complex nonlinearity and spatiotemporal structures present in traffic data. Noteworthy deep spatiotemporal forecasting models, including STGCN (Yu et al., 2018), DCRNN (Li et al., 2018), Graph Wavenet (Wu et al., 2019), and STSGCN (Song et al., 2020), have demonstrated strong performance. DL models typically use mean absolute error (MAE) or mean squared error (MSE) for training, assuming that errors are temporally independent and follow an isotropic distribution. However, these assumptions do not align with real-world datasets. First, the omission of certain features often gives rise to time-correlated residuals. For example, in traffic speed forecasting, time-varying factors, such as weather conditions and vehicle flow rates, are often overlooked, leading to temporally correlated errors. Second, spatiotemporal forecasting suggests the potential existence of contemporaneous correlations within the errors, which contradicts the assumption of isotropic errors. For instance, it is intuitive that the variance of predicted values would increase from the 5-minute-ahead prediction to the 60-minute-ahead prediction. Neglecting these factors could detrimentally impact model performance.

As demonstrated in Figure 1, the contemporaneous correlations,  $\text{Corr}(\boldsymbol{\eta}_t, \boldsymbol{\eta}_t)$ , and the lag-2016 cross-correlation,  $\text{Corr}(\boldsymbol{\eta}_{t-2016}, \boldsymbol{\eta}_t)$ , calculated based on the residuals of ASTGCN (Song et al., 2020) trained on the PEMS08 traffic flow dataset, exhibit distinct correlation structures. The strong off-diagonal elements in  $\text{Corr}(\boldsymbol{\eta}_t, \boldsymbol{\eta}_t)$  suggest that the errors are unlikely isotropic (Figure 1 (a)). The cross-correlation  $\text{Corr}(\boldsymbol{\eta}_{t-2016}, \boldsymbol{\eta}_t)$  includes both the autocorrelation of errors  $\text{Corr}(\eta_{i,t-2016}, \eta_{i,t})$  and the cross-lag correlation  $\text{Corr}(\eta_{i,t-2016}, \eta_{j,t})$  between all pairs of components in the multivariate series. Figure 1 (b) indicates significant cross-correlations in  $\text{Corr}(\boldsymbol{\eta}_{t-2016}, \boldsymbol{\eta}_t)$ , likely resulting from omitted influential covariates, such as traffic congestion information, that significantly affect observed flow (Cheng et al., 2021). While there are many modern techniques to identify and incorporate these covariates into deep learning models, our method offers a model-agnostic alternative. By leveraging the statistical attributes of the error process, our approach enhances model performance without relying on extensive features (some of which may be impractical to obtain) or perfectly tuned deep learning models.

This study presents an efficient method for adjusting and modeling correlated errors using a dynamic regression framework. The method is designed for seamless integration into any deep learning model applied to spatiotemporal traffic forecasting (i.e., multivariate sequence-to-sequence (Seq2Seq) forecasting). The approach assumes that the error of the base model follows a matrix-variate AR process, which accounts for the error cross-correlation problem and can be easily integrated into the model training process. In addition to gaining insights into the coefficients of the AR process, the method learns a non-isotropic covariance matrix effectively for the error term, which is assumed to follow a multivariate normal distribution. Notably, the parameters of the error AR module and the error covariance matrix can be jointly optimized with the parameters of the base model. The primary contributions of this study are outlined as follows:

- We propose utilizing a bi-linear AR structure for the matrix-valued error process to effectively address cross-correlation. By incorporating seasonal lags into the AR structure, we capture the characteristic seasonal patterns present in traffic data, thereby improving model accuracy and reliability.
- We model the error term in the DR framework using a multivariate normal distribution with a non-isotropic covariance matrix. The negative log-likelihood of this distribution serves as the loss function for model optimization. This approach yields an interpretable covariance matrix, which can be used for probabilistic forecasting with uncertainty quantification in the Seq2Seq problem.
- The proposed method is model-agnostic, enhancing existing DL models by leveraging the statistical attributes of the error process, which is applicable to any spatiotemporal traffic forecasting problem. We evaluate our method using several SOTA deep traffic forecasting models and consistently observe improvements across various metrics.

The subsequent sections of this paper are structured as follows. In Section 2, we conduct a comprehensive review of related works, covering deep traffic forecasting models, probabilistic forecasting methods, and the challenges associated with error correlations. Section 3 provides a detailed account of the traffic forecasting problem and introduces our proposed dynamic regression framework. In Section 4, we present the experimental setup and report the obtained results. Section 5 discusses some conceptual questions related to the proposed method. Section 6 summarizes key findings and concludes the paper.

## 2 Related Works

### 2.1 Deep Learning for Traffic Forecasting

In this section, we review key deep learning frameworks for traffic forecasting, a spatiotemporal forecasting (STF) problem. Traffic data is characterized by complex spatial dependencies, such as interactions between road segments or sensors, and temporal patterns, including daily or weekly cycles. To effectively model these intertwined dependencies, most existing works leverage Graph Neural Networks (GNNs), which excel at capturing the spatial relationships in traffic networks. These models are often combined with RNNs or Temporal Convolutional Networks (TCNs) to model temporal dynamics.

Graph-based approaches for traffic forecasting can be broadly categorized into spectral graph convolution and diffusion graph convolution methods (Ye et al., 2020). Spectral graph convolution operates in the frequency domain by leveraging the graph Laplacian to capture spatial dependencies through spectral decomposition. For instance, STGCN (Yu et al., 2018) combines Graph Convolutional Networks (GCNs) to model spatial correlations with Convolutional Neural Networks (CNNs) to capture temporal dependencies. GCNs are particularly effective for incorporating graph structures into spatial modeling, while CNNs enable faster training through parallel computation, avoiding the sequential nature of RNNs. Building on STGCN, ASTGCN (Guo et al., 2019) introduces a spatiotemporal attention mechanism to preprocess traffic data, enabling the model to better identify and focus on relevant temporal and spatial features before passing the data to the convolutional layers.

In contrast, diffusion graph convolution operates directly in the vertex domain, modeling spatial dependencies through localized diffusion processes. DCRNN (Li et al., 2018) exemplifies this approach by integrating a diffusion convolution operation to model spatial dependencies, combined with Gated Recurrent Units (GRUs) for temporal modeling. Some models, such as STGCN and ASTGCN, rely on a pre-defined adjacency matrix that remain fixed during training, limiting flexibility. To address this, Graph WaveNet (Wu et al., 2019) employs an adaptive adjacency matrix that learns the graph structure by treating the matrix entries as trainable parameters. This model integrates dilated causal convolution as Temporal Convolutional Networks (TCNs) for temporal dependency modeling and GCNs for spatial dependency. Earlier GCN-based models treated spatial and temporal dependencies separately, limiting their ability to capture spatiotemporal interactions. To overcome this limitation, STSGCN (Song et al., 2020) connects individual spatial graphs of adjacent time steps into a unified graph, enabling simultaneous learning of spatial and temporal information. This approach has demonstrated superior performance compared to previous GCN-based frameworks by fully leveraging spatiotemporal interactions. In D<sup>2</sup>STGNN (Shao et al., 2022), the authors decompose traffic signals into diffusion signals and inherent signals, processed separately using specialized modules. The diffusion module utilizes spatial-temporal localized convolutional layers to capture dynamic spatial dependencies, while the inherent module incorporates GRUs and self-attention layers to model temporal patterns intrinsic to the data. Additionally, the framework constructs adjacency matrices at different time steps to effectively capture the dynamic correlations among nodes.

Other SOTA models include GMAN (Zheng et al., 2020), AGCRN (Bai et al., 2020), and ASTGNN (Guo et al., 2021), among others. For a comprehensive overview of STF models, we refer readers to the recent survey by Shao et al. (2024).

## 2.2 Probabilistic Forecasting Methods

Probabilistic forecasting is central to time series analysis. One effective approach is to design models that output a set of quantiles rather than a single point forecast. For instance, Wen et al. (2017) introduced MQ-RNN, a Seq2Seq RNN framework that directly forecasts quantiles. The model parameters are optimized using the quantile loss function. Similarly, Toubau et al. (2018) applied this methodology for probabilistic forecasting in power markets.

An alternative approach to probabilistic forecasting involves estimating the probability density functions (PDF) of target variables. This method typically assumes a simple distri-

bution form for the target variables and uses neural networks to output the distribution parameters. A prominent example is DeepAR (Salinas et al., 2020), which employs RNNs to model the transition of hidden states. These hidden states condition the generation of parameters for a Gaussian distribution, from which prediction samples are drawn. Similarly, GPVar (Salinas et al., 2019) uses a multivariate Gaussian distribution to model the joint distribution of multivariate time series. The authors utilize a Gaussian copula to transform the original time series observations into Gaussian random variables, with RNNs parameterizing the output distribution.

Neural networks can also be designed to learn probabilistic model parameters rather than assuming a fixed distribution. Rangapuram et al. (2018) proposed a deep state space model (SSM), utilizing RNNs to output SSM parameters, which allows for drawing prediction samples from the observation model of the SSM. Similarly, de Bézenac et al. (2020) introduced the Normalizing Kalman Filter (NKF) model, enhancing the classical linear Gaussian state space model (LGM) with normalizing flows (NFs). These NFs model non-linear dynamics on top of the observation model, allowing PDF estimation of observations. The parameters of the NKF model are generated by RNNs. Additionally, Wang et al. (2019) proposed a probabilistic forecasting model based on deep factors. This model incorporates both a fixed global component and a random component, with the latter encompassing classical probabilistic time series models. The combination of these components forms the latent function, which conditions a parametric distribution for generating observations.

Some approaches focus on enhancing expressive conditioning for probabilistic forecasting. One such method involves using Transformers instead of RNNs to model latent dynamics, thereby relaxing the Markovian assumption in RNNs (Tang and Matteson, 2021). Another avenue for improvement involves adopting more flexible distribution forms, such as normalizing flows (Rasul et al., 2020), diffusion models (Rasul et al., 2021), and Copula-based models (Drouin et al., 2022; Ashok et al., 2023).

Our approach stands out from previously mentioned probabilistic models in two key aspects. Firstly, it extends probabilistic forecasting to multivariate Seq2Seq traffic forecasting, where the target outcome is a matrix rather than a scalar or vector. Secondly, it can seamlessly integrate with existing deterministic traffic forecasting models without altering their outputs.

### 2.3 Adjusting for Correlated Errors

Correlated errors in time series data have been extensively studied in econometrics through models with precise forms (Durbin and Watson, 1950; Ljung and Box, 1978; Breusch, 1978; Godfrey, 1978; Cochrane and Orcutt, 1949; Prais and Winsten, 1954; Beach and MacKinnon, 1978; Prado et al., 2021; Hyndman and Athanasopoulos, 2018; Hamilton, 2020). Recent advances in DL models have increased interest in learning and adapting the error process. Two primary statistical approaches are commonly employed for modeling the error process: (i) capturing autocorrelation (or cross-correlation in multivariate data), and (ii) learning contemporaneous correlation in errors.

To address autocorrelation or cross-correlation, Sun et al. (2021) proposed a reparametrization strategy for the input and output of a neural network used in time series forecasting. This reparametrization inherently employs a first-order error AR process through a linear regression framework. The method effectively enhances performance for various DL-based

one-step-ahead forecasting models across a diverse range of time series datasets, enabling joint parameter optimization of both base and error regressors. Additionally, Kim et al. (2022) introduced a lightweight DL module designed to calibrate cross-correlation within predictions from pre-trained traffic forecasting models. This calibration module uses recent observed residuals and predictions to anticipate future residuals, ultimately enhancing the performance of numerous traffic forecasting models, especially on traffic speed datasets. Zheng et al. (2024) introduced a probabilistic forecasting model trained using a Generalized Least Squares (GLS) loss, which explicitly accounts for the time-varying autocorrelation of batched error terms. This approach was later extended to the multivariate scenario, accounting for error cross-correlation through the use of an efficient parameterization of the covariance matrix (Zheng and Sun, 2024).

Regarding learning contemporaneous correlation, Jia and Benson (2020) emphasized that assuming independence in the residuals of a node regression problem is unwarranted. They advocated for modeling contemporaneous correlation using a multivariate Gaussian distribution. This method, termed residual propagation in GNNs, adjusts predictions of unknown nodes based on known node labels. Similarly, Huang et al. (2020) introduced a correct-and-smooth approach, serving as a post-processing scheme to rectify correlated residuals in GNNs, focusing on a classification task. Additionally, Choi et al. (2022) proposed a dynamic mixture of matrix normal Gaussian as a regularization technique to address contemporaneous correlation within the errors of Seq2Seq models.

Applying existing methods directly to train a multivariate Seq2Seq model is challenging. A key challenge is that the error term becomes an  $N \times Q$  matrix-variate time series. The cohesive learning of this process and the base model constitutes a primary challenge. Our work differs from Sun et al. (2021) and Kim et al. (2022) in several important ways. Firstly, we extend the scope of Sun et al. (2021) to encompass multivariate Seq2Seq forecasting, while avoiding the reparametrization of input and output. Secondly, our approach allows for the concurrent optimization of both the DR parameters and the parameters of the base model, unlike the post-hoc adjustment in Kim et al. (2022). Lastly, our method is constructed based on a probabilistic formulation with the capability for uncertainty quantification, setting our work apart from the aforementioned studies. The works of Zheng et al. (2024) and Zheng and Sun (2024) model the correlated errors in probabilistic forecasting but do not directly apply to the Seq2Seq task.

### 3 Methodology

This section formulates the multivariate Seq2Seq traffic forecasting problem and outlines the dynamic regression framework for adjusting error cross-correlation and capturing contemporaneous correlations. Figure 2 presents the overview of the proposed DR framework.

#### 3.1 Traffic Forecasting

A traffic network can be defined as a directed graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ , where  $\mathcal{V}$  with  $|\mathcal{V}| = N$  is a set of nodes representing traffic sensors;  $\mathcal{E}$  is a set of links connecting these nodes;  $\mathbf{W} \in \mathbb{R}^{N \times N}$  is a weighted adjacency matrix characterizing the proximity of nodes. Let  $\mathbf{z}_t \in \mathbb{R}^N$  denote the vector of observed traffic states at time  $t$ . The traffic forecasting problem



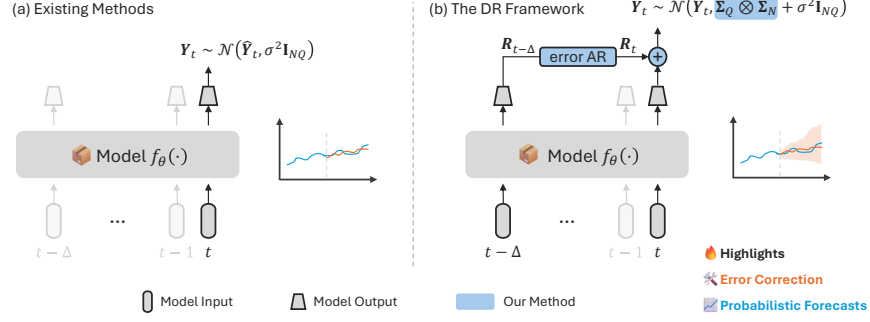


Fig. 2: Overview of the proposed DR framework, highlighting its structure and comparison with conventional methods.

involves learning a function that maps data from a context horizon of  $P$  past time steps to a prediction horizon of  $Q$  future time steps.

Let  $\mathbf{X}_t = [z_{t-P+1}, \dots, z_t] \in \mathbb{R}^{N \times P}$  and  $\mathbf{Y}_t = [z_{t+1}, \dots, z_{t+Q}] \in \mathbb{R}^{N \times Q}$ , we have

$$\mathbf{Y}_t = f_\theta(\mathbf{X}_t, \mathbf{W}) + \mathbf{R}_t, \quad (1)$$

where  $f_\theta(\mathbf{X}_t, \mathbf{W})$  is a DL model that generates the predicted mean and  $\mathbf{R}_t \in \mathbb{R}^{N \times Q}$  is the error term. For simplicity, we will omit  $\mathbf{W}$  for the remainder of the paper. In many cases,  $f_\theta(\cdot)$  is trained with MAE/MSE as the loss function to obtain an optimal parameter set  $\theta^*$ , e.g., a MSE loss can be written as:

$$\mathcal{L} = \sum_t \|\mathbf{R}_t\|_F^2 = \sum_t \|\mathbf{Y}_t - f_\theta(\mathbf{X}_t)\|_F^2. \quad (2)$$

This loss function simply assumes: (i)  $\mathbf{R}_t$  is temporally independent, i.e., there is no correlation between  $\mathbf{R}_s$  and  $\mathbf{R}_t$  when  $s \neq t$ ; and (ii) entries in  $\mathbf{R}_t$  follow a zero-mean isotropic Gaussian distribution with no concurrent correlations, i.e.,  $\boldsymbol{\eta}_t = \text{vec}(\mathbf{R}_t) \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_{NQ})$ . Likewise, using MAE as the loss function corresponds to assuming entries in  $\mathbf{R}_t$  follow a Laplacian distribution.

### 3.2 Modeling Error with Matrix-valued Autoregression

Following the idea of dynamic regression (Hyndman and Athanasopoulos, 2018), we assume that the relationship between the input  $\mathbf{X}_t$  and the output  $\mathbf{Y}_t$  is not fully captured by  $f_\theta(\cdot)$ , and the error  $\mathbf{R}_t$  is governed by a temporal process. For example, for a one-step-ahead prediction model (i.e.,  $\boldsymbol{\eta}_t = \mathbf{R}_t$  as  $Q = 1$ ), it is straightforward to model  $\boldsymbol{\eta}_t$  using a  $p$ -th order vector autoregressive model:

$$\boldsymbol{\eta}_t = \mathbf{C}_1 \boldsymbol{\eta}_{t-1} + \dots + \mathbf{C}_p \boldsymbol{\eta}_{t-p} + \boldsymbol{\epsilon}_t, \quad (3)$$

where  $\mathbf{C}_l \in \mathbb{R}^{N \times N}$  ( $l = 1, \dots, p$ ) are the regression coefficient, and  $\boldsymbol{\epsilon}_t$  is a white-noise process.

However, for a Seq2Seq model with  $Q > 1$ ,  $\mathbf{R}_t$  cannot be directly modeled using Eq. (3), as  $\{\mathbf{R}_{t-l} \mid l = 1, \dots, Q-1\}$  will not be entirely accessible due to overlapping. To address



this issue, we model the relation between  $\mathbf{R}_t$  and its accessible lagged values, i.e.,  $\mathbf{R}_{t-\Delta_l}$  for  $l = 1, \dots, p$  as long as  $\Delta_l \geq Q$ . Therefore, we have the model for the vector  $\boldsymbol{\eta}_t$  as:

$$\boldsymbol{\eta}_t = \mathbf{C}_1 \boldsymbol{\eta}_{t-\Delta_1} + \dots + \mathbf{C}_p \boldsymbol{\eta}_{t-\Delta_p} + \boldsymbol{\epsilon}_t, \quad (4)$$

where  $\mathbf{C}_l \in \mathbb{R}^{NQ \times NQ}$  and the white-noise covariance matrix is also of size  $NQ \times NQ$ . As traffic data often exhibit strong daily and weekly patterns, for simplicity we only introduce a single lagged residual component  $\mathbf{R}_{t-\Delta}$ , where  $\Delta$  is a predetermined lag (e.g., one day or one week) exhibiting strong correlations with the present time.

**Assumption 1** *The error  $\mathbf{R}_t$  of the base deep learning model follows a first-order bi-linear autoregressive process (Eq. (5)).*

A notable limitation of the aforementioned formulation is the introduction of an excessive number of parameters within  $\mathbf{C}$ . For improved scalability of this approach, we assume  $\mathbf{R}_t$  follows a first-order bi-linear AR model (Chen et al., 2021; Hsu et al., 2021):

$$\mathbf{R}_t = \mathbf{A} \mathbf{R}_{t-\Delta} \mathbf{B} + \mathbf{E}_t, \quad (5)$$

where  $\mathbf{A} \in \mathbb{R}^{N \times N}$  and  $\mathbf{B} \in \mathbb{R}^{Q \times Q}$  are regression coefficients. With Assumption 1, there are two error terms:  $\mathbf{R}_t$ , the error from the base DL model, and  $\mathbf{E}_t$ , the error from the AR model. In the DR framework, only the AR model error  $\mathbf{E}_t$  is assumed to be time-independent. The vectorized version of Eq. (5) becomes

$$\boldsymbol{\eta}_t = (\mathbf{B}^\top \otimes \mathbf{A}) \boldsymbol{\eta}_{t-\Delta} + \boldsymbol{\epsilon}_t, \quad (6)$$

and we can see that the bi-linear formulation is equivalent to imposing a Kronecker product structure on the coefficient matrix  $\mathbf{C} = \mathbf{B}^\top \otimes \mathbf{A}$  in Eq. (4), leading to a significant reduction in parameters. Combining Eqs. (1) and (5), we can obtain a new model that seamlessly integrates with the autocorrelated errors:

$$\begin{aligned} \mathbf{Y}_t &= f_\theta(\mathbf{X}_t) + \mathbf{A} \mathbf{R}_{t-\Delta} \mathbf{B} + \mathbf{E}_t \\ &= f_\theta(\mathbf{X}_t) + \mathbf{A} (\mathbf{Y}_{t-\Delta} - f_\theta(\mathbf{X}_{t-\Delta})) \mathbf{B} + \mathbf{E}_t, \end{aligned} \quad (7)$$

where  $\mathbf{A}$  and  $\mathbf{B}$ , as trainable parameters, can be jointly updated with the base model  $f_\theta(\cdot)$ . When  $\mathbf{R}_t$  and  $\mathbf{R}_{t-\Delta}$  are uncorrelated (e.g., when  $\mathbf{A} = \mathbf{0}$ ), The model reduces to its original implementation. If both  $\mathbf{A}$  and  $\mathbf{B}$  are identity matrices, the model implies a seasonal random walk for the error process. To enforce sparsity in  $\mathbf{A}$  and  $\mathbf{B}$ , we introduce an  $\ell_1$  regularization term into the loss function:

$$\mathcal{L}_{\text{reg}} = \frac{1}{N^2} \|\mathbf{A}\|_1 + \frac{1}{Q^2} \|\mathbf{B}\|_1. \quad (8)$$

Once  $f_\theta(\cdot)$  and coefficients  $\mathbf{A}$  and  $\mathbf{B}$  are learned, prediction at time  $t$  can be made by:

$$\hat{\mathbf{Y}}_t = f_\theta(\mathbf{X}_t) + \mathbf{A} (\mathbf{Y}_{t-\Delta} - f_\theta(\mathbf{X}_{t-\Delta})) \mathbf{B}. \quad (9)$$

In this paper, we refer to “residuals” as the difference between the observed values and the values predicted by the model, i.e.,  $\mathbf{Y}_t - \hat{\mathbf{Y}}_t$ . Here,  $\hat{\mathbf{Y}}_t$  is generated by a trained model  $g_{\Theta^*}(\cdot)$ . On the other hand, we refer to “errors” as the deviation of the observed values from the true underlying values, i.e.,  $\mathbf{Y}_t - g_{\Theta}(\cdot)$ , where  $g_{\Theta}(\cdot)$  is the true underlying model. Therefore, “errors” are theoretical quantities because the true underlying model is usually unknown in real-world applications. “Residuals”, however, are observable quantities because they are based on the actual data and the fitted model.

### 3.3 Spatiotemporal Covariance Structure for the Matrix Error Term

Following [Stegle et al. \(2011\)](#) and [Luttinen and Ilin \(2012\)](#), we assume that the error  $\epsilon_t$  in Eq. (6) follows a zero-mean multivariate Gaussian distribution  $\epsilon_t = \text{vec}(\mathbf{E}_t) \sim \mathcal{N}(\mathbf{0}_{NQ}, \mathbf{\Sigma})$ , where  $\mathbf{\Sigma} = \mathbf{\Sigma}_Q \otimes \mathbf{\Sigma}_N + \sigma^2 \mathbf{I}_{NQ}$  is a non-isotropic covariance matrix (Assumption 2). Here,  $\mathbf{\Sigma}_N$  and  $\mathbf{\Sigma}_Q$  represent the contemporaneous covariance among the rows and columns of  $\mathbf{E}_t$ . The Kronecker structure significantly reduces the number of parameters in  $\mathbf{\Sigma}$ . The negative log-likelihood of this distribution is used as the new loss function:

$$\begin{aligned} \mathcal{L}_{\text{nl}} &= -\ln p(\mathbf{E}_t | \mathbf{\Sigma}_Q, \mathbf{\Sigma}_N, \sigma^2) \\ &= \frac{1}{2} \left[ \ln |\mathbf{\Sigma}| + \epsilon_t^\top \mathbf{\Sigma}^{-1} \epsilon_t + NQ \ln(2\pi) \right]. \end{aligned} \quad (10)$$

**Assumption 2** *The error  $\epsilon_t = \text{vec}(\mathbf{E}_t)$  in the dynamic regression framework follows a zero-mean multivariate Gaussian distribution with a covariance matrix  $\mathbf{\Sigma}$  parameterized as a Kronecker product plus a diagonal  $\mathbf{\Sigma}_Q \otimes \mathbf{\Sigma}_N + \sigma^2 \mathbf{I}_{NQ}$  (Eq. (11)).*

The large size ( $NQ \times NQ$ ) of  $\mathbf{\Sigma}$  makes computing its inverse and determinant increasingly challenging as ( $N$ ) grows. To efficiently optimize the log-likelihood, we apply the Sherman–Morrison–Woodbury identity and the matrix determinant lemma. This is facilitated by leveraging the low-rank decomposition of the covariance matrices  $\mathbf{\Sigma}_N = \mathbf{L}_N \mathbf{L}_N^\top$  and  $\mathbf{\Sigma}_Q = \mathbf{L}_Q \mathbf{L}_Q^\top$ , where  $\mathbf{L}_N \in \mathbb{R}^{N \times R_n}$  and  $\mathbf{L}_Q \in \mathbb{R}^{Q \times R_q}$ , with  $R_n$  and  $R_q$  being the rank parameters. Consequently,

$$\begin{aligned} \mathbf{\Sigma} &= \mathbf{\Sigma}_Q \otimes \mathbf{\Sigma}_N + \sigma^2 \mathbf{I}_{NQ} \\ &= (\mathbf{L}_Q \mathbf{L}_Q^\top) \otimes (\mathbf{L}_N \mathbf{L}_N^\top) + \sigma^2 \mathbf{I}_{NQ} \\ &= (\mathbf{L}_Q \otimes \mathbf{L}_N) (\mathbf{L}_Q \otimes \mathbf{L}_N)^\top + \sigma^2 \mathbf{I}_{NQ} \\ &= \mathbf{S} \mathbf{S}^\top + \mathbf{D}, \end{aligned} \quad (11)$$

where  $\mathbf{S} = \mathbf{L}_Q \otimes \mathbf{L}_N$  and  $\mathbf{D} = \sigma^2 \mathbf{I}_{NQ}$ . The matrix inversion and determinant can be efficiently computed as

$$\mathbf{\Sigma}^{-1} = (\mathbf{D} + \mathbf{S} \mathbf{S}^\top)^{-1} = \mathbf{D}^{-1} - \mathbf{D}^{-1} \mathbf{S} \left( \mathbf{I}_R + \mathbf{S}^\top \mathbf{D}^{-1} \mathbf{S} \right)^{-1} \mathbf{S}^\top \mathbf{D}^{-1}, \quad (12)$$

$$|\mathbf{\Sigma}| = \det(\mathbf{D} + \mathbf{S} \mathbf{S}^\top) = \det(\mathbf{I}_R + \mathbf{S}^\top \mathbf{D}^{-1} \mathbf{S}) \det(\mathbf{D}), \quad (13)$$

where  $R = R_n R_q$ . Eq. (12) and Eq. (13) simplify the computation of the inverse and determinant of  $\mathbf{\Sigma}$  to calculating the inverse and determinant of an  $R \times R$  capacitance matrix  $(\mathbf{I}_R^{-1} + \mathbf{S}^\top \mathbf{D}^{-1} \mathbf{S})$ . This can be efficiently achieved using its Cholesky factor  $\mathbf{L}_C$ , where  $(\mathbf{I}_R^{-1} + \mathbf{S}^\top \mathbf{D}^{-1} \mathbf{S}) = \mathbf{L}_C \mathbf{L}_C^\top$ , and  $\mathbf{L}_C$  is a lower triangular matrix with positive diagonal entries. Specifically, the Mahalanobis term in Eq. (10) can be simplified as

$$\begin{aligned} &\epsilon_t^\top \mathbf{\Sigma}^{-1} \epsilon_t \\ &= \epsilon_t^\top \mathbf{D}^{-1} \epsilon_t - \epsilon_t^\top \mathbf{D}^{-1} \mathbf{S} \left( \mathbf{L}_C \mathbf{L}_C^\top \right)^{-1} \mathbf{S}^\top \mathbf{D}^{-1} \epsilon_t \\ &= \epsilon_t^\top \mathbf{D}^{-1} \epsilon_t - \left( \mathbf{L}_C^{-1} \mathbf{S}^\top \mathbf{D}^{-1} \epsilon_t \right)^\top \left( \mathbf{L}_C^{-1} \mathbf{S}^\top \mathbf{D}^{-1} \epsilon_t \right) \\ &= \epsilon_t^\top \mathbf{D}^{-1} \epsilon_t - \mathbf{k}_t^\top \mathbf{k}_t, \end{aligned} \quad (14)$$

Table 1: Dataset description.

Datasets	Type	#Nodes	#Time Steps	Resolution
PEMSD7 (M)	Speed	228	12,672	5 min
PEMS08	Flow	170	17,856	5 min

where  $\mathbf{k}_t = \mathbf{L}_C^{-1} \mathbf{S}^\top \mathbf{D}^{-1} \boldsymbol{\epsilon}_t$ . The computation of  $\mathbf{k}_t$  can be efficiently performed by solving a triangular system of linear equations,  $\mathbf{L}_C \mathbf{k}_t = \mathbf{S}^\top \mathbf{D}^{-1} \boldsymbol{\epsilon}_t$ . Since  $\mathbf{D}$  is a diagonal, this method avoids direct matrix inversion. Additionally, the calculation of the log-determinant is greatly simplified as follows:

$$\begin{aligned}
\ln|\boldsymbol{\Sigma}| &= \ln|\mathbf{D} + \mathbf{S}\mathbf{S}^\top| \\
&= \ln|\mathbf{I}_R + \mathbf{S}^\top \mathbf{D}^{-1} \mathbf{S}| + \ln|\mathbf{D}| \\
&= 2 \sum_i^R \ln[\mathbf{L}_C]_{i,i} + 2NQ \ln(\sigma).
\end{aligned} \tag{15}$$

The new loss function enables probabilistic forecasting by first drawing samples  $\tilde{\mathbf{E}}_t$  from  $\mathcal{N}(\mathbf{0}_{NQ}, \boldsymbol{\Sigma})$  and subsequently transforming these samples into the target variable through the following process:

$$\tilde{\mathbf{Y}}_t = \hat{\mathbf{Y}}_t + \tilde{\mathbf{E}}_t, \tag{16}$$

where  $\hat{\mathbf{Y}}_t$  is the predicted mean as defined in Eq. (9). Since the covariance matrix  $\boldsymbol{\Sigma}$  captures the contemporaneous correlations in  $\mathbf{E}_t$ , we expect this approach to improve uncertainty quantification in probabilistic forecasting.

## 4 Experiment

To evaluate the effectiveness of our method, we conduct experiments on two traffic datasets: PEMS07 (M) and PEMS08. PEMS07 (M) is a highway traffic speed dataset initially employed in STGCN (Yu et al., 2018). PEMS08 is a highway traffic flow dataset originally utilized in ASTGCN (Guo et al., 2019). We follow the identical data processing procedures outlined in the original studies. For PEMS08, we allocate 60% of the data for training, 20% for validation, and the remaining 20% for testing. For PEMS07 (M), we use a 7:1:2 split. Across all datasets, we normalize data using z-score transformation based on training set statistics. Missing values are excluded from both training and testing. Table 1 summarizes dataset details.

### 4.1 Base Models

In addition to the models originally designed for the selected datasets, we include three additional models as base models  $f_\theta(\cdot)$  to diversify the range of architectures evaluated in our approach. Most existing STF frameworks combine GNNs with sequential models to effectively model traffic data. The choice of sequential models encompasses RNN-based methods (e.g., AGCRN (Bai et al., 2020)), TCN-based methods (e.g., STGCN (Yu et al., 2018) and Graph WaveNet (Wu et al., 2019)), and attention-based methods (e.g., ASTGCN (Guo et al., 2019)).

Furthermore, we incorporate a SOTA method, D<sup>2</sup>STGNN (Shao et al., 2022), which is highlighted in the LargeST benchmark (Liu et al., 2024). The implementation of the proposed methods is publicly available at: <https://github.com/rottenivy/dynamic-regression>.

## 4.2 Experimental Setup

Our experimental setup involves a computer environment with an Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20 GHz and four NVIDIA Tesla V100 GPUs. Base models were implemented using either the original source code or their PyTorch versions. All models were configured to employ 12 historical observation steps ( $P = 12$ ) for predicting 12 future steps ( $Q = 12$ ). During training, the Adam optimizer was utilized with an initial learning rate of 0.001 and a weight decay of 0.0001. To prevent over-fitting, early stopping was employed when the validation loss showed consistent increase over 15 epochs. All metrics are averaged over three independent runs.

For the error AR process, we need to determine the lag length  $\Delta$ . Since traffic data exhibits strong local correlation and seasonality, we mainly consider the correlation between the current residual and its 1) most recent available predecessor ( $\Delta = Q$ ); 2) counterpart one day apart ( $\Delta = 1$  day); and 3) counterpart one week apart ( $\Delta = 1$  week). Although the use of seasonal lags is a common practice in time series modeling across various domains, such as in Seasonal ARIMA models (Hyndman and Athanasopoulos, 2018), the explicit incorporation of seasonal error correlations is novel in the context of prior related works addressing correlated errors in traffic data (Sun et al., 2021; Kim et al., 2022). We determine the optimal  $\Delta$  based on validation loss for each model and dataset.

The final loss function of our method is composed of two parts:

$$\mathcal{L} = \mathcal{L}_{\text{nl}} + \mathcal{L}_{\text{reg}}, \quad (17)$$

where  $\mathcal{L}_{\text{reg}}$  is the  $\ell_1$  regularization in Eq. (8) and  $\mathcal{L}_{\text{nl}}$  is the non-isotropic loss function in Eq. (10). For comparative analyses, we also train the base model without our DR framework, utilizing an isotropic loss function and excluding the error AR process, i.e.,  $\mathcal{L} \propto \boldsymbol{\eta}_t^\top \boldsymbol{\eta}_t$  by assuming  $\boldsymbol{\eta}_t = \text{vec}(\mathbf{R}_t) \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_{NQ})$ . This is equivalent to using MSE as the loss function. In comparison to the base model trained without our method, we introduce four additional trainable parameters, i.e., regression coefficients  $\mathbf{A} \in \mathbb{R}^{N \times N}$  and  $\mathbf{B} \in \mathbb{R}^{Q \times Q}$  for the error AR process, and two low-rank matrices  $\mathbf{L}_N \in \mathbb{R}^{N \times R_n}$  and  $\mathbf{L}_Q \in \mathbb{R}^{Q \times R_q}$  for the error covariance learning. We use  $R_n = N$  and  $R_q = Q$  for the initial comparison; however, sensitivity analysis will be conducted to assess the impact of these rank parameters. However, the additional parameters are negligible compared to the overall size of the base model.

We use several evaluation metrics for validating the effectiveness of our method, namely, root relative mean squared error (RRMSE) (Bai et al., 2020; Lai et al., 2018; Song et al., 2021; Shih et al., 2019), empirical Continuous Ranked Probability Score (CRPS) (Gneiting and Raftery, 2007), and quantile loss ( $\rho$ -risk) (Salinas et al., 2020).

RRMSE is mainly used to assess the accuracy of point forecasts of the models, as it directly evaluates the predicted mean  $\hat{\mathbf{Y}}_t$ :

$$\text{RRMSE} = \frac{\sqrt{\sum_t \|\mathbf{Y}_t - \hat{\mathbf{Y}}_t\|_F^2}}{\sqrt{\sum_t \|\mathbf{Y}_t - \bar{\mathbf{Y}}_t\|_F^2}}, \quad (18)$$

where  $\bar{Y}_t$  is the mean value of the testing dataset. Lower RRMSE values indicate better accuracy. As it solely evaluates the predicted mean, RRMSE primarily reflects the effectiveness of the error AR module.

CRPS is a widely used quantitative measure to evaluate the quality of probabilistic forecasts when the observation is a scalar (Salinas et al., 2019; Rasul et al., 2020, 2021; Drouin et al., 2022; Ashok et al., 2023). In this work, we use the kernel representation form of the CRPS (Gneiting and Raftery, 2007):

$$\text{CRPS}(F, z) = E_F|Z - z| - \frac{1}{2} E_F|Z - Z'|, \quad (19)$$

where  $z$  is the observation,  $F$  is the CDF of the predicted variable,  $Z$  and  $Z'$  are independent copies of a set of prediction samples associated with the distribution  $F$ . Lower CRPS values indicate better agreement between predicted and observed distributions. It can be shown that CRPS inherently accounts for the calibration and sharpness of probabilistic forecasts:

- Overconfident (Too Sharp): Forecasts that are too narrow will have larger squared differences where they fail to encompass the observed value, leading to a higher CRPS (the first term in Eq. (19)).
- Underconfident (Too Wide): Forecasts that are too wide will have persistent deviations over a broad range of values, also leading to a higher CRPS (the second term in Eq. (19)).

We compute CRPS at each spatial location and each forecast step by drawing samples from Eq. (16). The scores are summarized by first summing across the entire testing horizon for all time series and then normalizing by the sum of the corresponding ground truth values.

The quantile loss  $L_\rho(z, \hat{Z}^\rho)$  is used to quantify the accuracy of a given quantile, denoted by  $\rho$ , from the predictive distribution:

$$L_\rho(z, \hat{Z}^\rho) = 2(\hat{Z}^\rho - z) \left( (1 - \rho) \mathbf{I}_{\hat{Z}^\rho > z} - \rho \mathbf{I}_{\hat{Z}^\rho \leq z} \right), \quad (20)$$

where  $\mathbf{I}$  is a binary indicator function that equals 1 when the condition is met,  $\hat{Z}^\rho$  represents the predicted  $\rho$ -quantile, and  $z$  represents the ground truth value. We summarize the quantile losses for the testing set across all time series segments by evaluating a normalized summation of these losses:  $\left( \sum_t L_\rho(z_t, \hat{Z}_t^\rho) \right) / (\sum_t z_t)$ . In this paper, we evaluate the 0.5-risk, 0.75-risk, and the 0.9-risk, calculated from the corresponding quantiles of the empirical predictive distribution. We drew 100 prediction samples to form the empirical predictive distribution used to calculate the probabilistic scores, specifically the CRPS and the  $\rho$ -risks.

### 4.3 Experimental Results

We first analyze cross-correlation and contemporaneous correlation in residuals from a base model without our method. We calculate two types of correlation:  $\text{Corr}(\boldsymbol{\eta}_t, \boldsymbol{\eta}_t)$  and  $\text{Corr}(\boldsymbol{\eta}_{t-\Delta}, \boldsymbol{\eta}_t)$ .  $\text{Corr}(\boldsymbol{\eta}_t, \boldsymbol{\eta}_t)$  is the contemporaneous correlation of the variables in  $\boldsymbol{\eta}_t$ , while  $\text{Corr}(\boldsymbol{\eta}_{t-\Delta}, \boldsymbol{\eta}_t)$  is the cross-correlation at lag  $\Delta$ . If the error independently follows an identical isotropic distribution,  $\text{Corr}(\boldsymbol{\eta}_t, \boldsymbol{\eta}_t)$  should be an identity matrix and  $\text{Corr}(\boldsymbol{\eta}_{t-\Delta}, \boldsymbol{\eta}_t)$  should

be a zero matrix. In Figure 3, we present the residual correlation matrices of PEMS08 using the results from Graph WaveNet. We observe that  $\text{Corr}(\boldsymbol{\eta}_t, \boldsymbol{\eta}_t)$  is not diagonal, suggesting there exists spatial and across-step correlations in the residuals. In terms of  $\text{Corr}(\boldsymbol{\eta}_{t-\Delta}, \boldsymbol{\eta}_t)$ , we examine different values including  $\Delta = 12$  (1 hour),  $\Delta = 288$  (1 day), and  $\Delta = 2016$  (1 week). Interestingly, we find that  $\Delta = 2016$  exhibits the strongest cross-correlation patterns, while correlations with  $\Delta = 12$  and  $\Delta = 288$  are weak. We believe this is mainly due to the fact that traffic flow is heavily determined by travel demand, which often exhibits prominent weekly periodicity. We selected the optimal value of  $\Delta$  for each dataset based on the validation loss. Specifically, in our study, we employed  $\Delta = 12$  for PEMS07 (M) and  $\Delta = 2016$  for PEMS08.

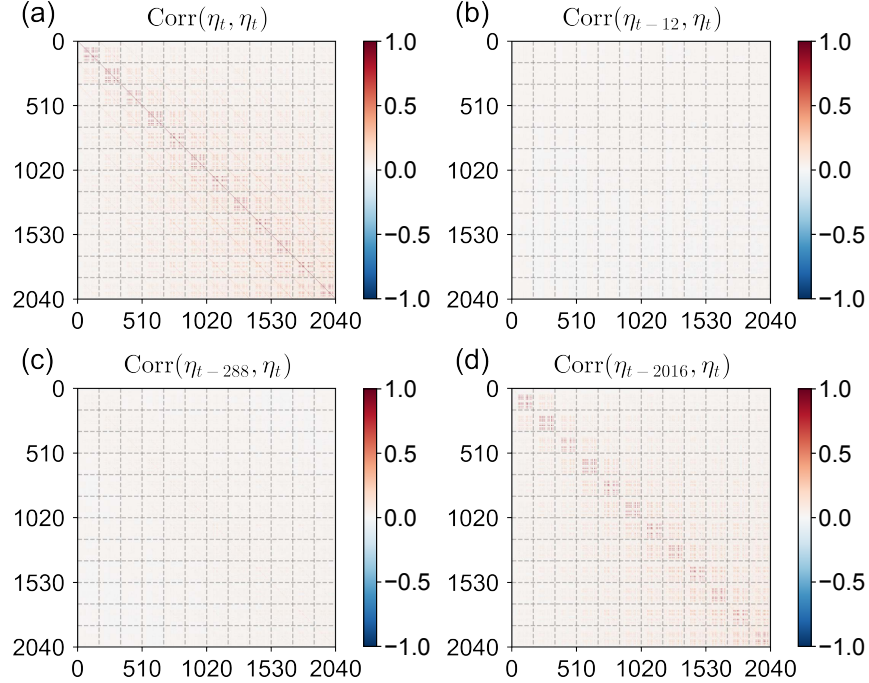


Fig. 3: Correlation matrices of residuals obtained from the original Graph WaveNet model on the PEMS08 dataset. From (a) to (d): Contemporaneous correlations, Cross-correlations at lag 12 (i.e., most recent), lag 288 (i.e., 1 day), lag 2016 (i.e., 1 week).

#### 4.3.1 OVERALL COMPARISON

Table 2 offers a comprehensive overview of our approach’s performance, encompassing various deep learning models and datasets assessed through diverse metrics. The chosen metrics shed light on multiple facets of model effectiveness. Notably, our method demonstrates improvements of 1.67% and 3.38% concerning RRMSE for PEMS07 (M) and PEMS08, respectively. Since RRMSE evaluates only the mean prediction  $\hat{\mathbf{Y}}_t$ , it directly measures the effectiveness of the error AR module. The comparatively modest improvement in RRMSE for PEMS07 (M) compared to PEMS08 implies that the base models already exhibit commendable performance on this specific dataset.

Table 2: Impact of the proposed method on various models and datasets.

Data	Model	RRMSE	CRPS	0.5-risk	0.75-risk	0.9-risk
PEMSD7 (M)	STGCN	0.4801	0.0566	0.0699	0.0591	0.0385
	+ DR	<b>0.4501</b>	<b>0.0544</b>	<b>0.0691</b>	<b>0.0575</b>	<b>0.0363</b>
	Graph WaveNet	0.3676	0.0427	0.05	0.0461	0.0308
	+ DR	<b>0.3622</b>	<b>0.0389</b>	<b>0.0478</b>	<b>0.0409</b>	<b>0.0268</b>
	AGCRN	<b>0.375</b>	0.0435	0.0509	0.0471	0.0316
	+ DR	0.3838	<b>0.0406</b>	<b>0.0499</b>	<b>0.0427</b>	<b>0.0289</b>
	D <sup>2</sup> STGNN	0.3639	0.0416	0.0489	0.0452	0.0305
	+ DR	<b>0.3592</b>	<b>0.0391</b>	<b>0.0481</b>	<b>0.0407</b>	<b>0.0271</b>
Avg. rel. impr.		1.67%	6.37%	2.29%	8.32%	9.60%
PEMS08	ASTGCN	0.1955	0.0649	0.0841	0.0756	0.0476
	+ DR	<b>0.1838</b>	<b>0.0566</b>	<b>0.0758</b>	<b>0.0638</b>	<b>0.0397</b>
	Graph WaveNet	0.1707	0.0544	0.0698	0.0614	0.0389
	+ DR	<b>0.1639</b>	<b>0.0465</b>	<b>0.0619</b>	<b>0.0517</b>	<b>0.0321</b>
	AGCRN	0.1789	0.0579	0.0746	0.0665	0.0422
	+ DR	<b>0.1739</b>	<b>0.0526</b>	<b>0.0698</b>	<b>0.0602</b>	<b>0.0392</b>
	D <sup>2</sup> STGNN	0.1678	0.0537	0.0683	0.0615	0.0388
	+ DR	<b>0.1665</b>	<b>0.0481</b>	<b>0.0645</b>	<b>0.0545</b>	<b>0.0331</b>
Avg. rel. impr.		3.38%	11.72%	8.30%	13.07%	13.97%

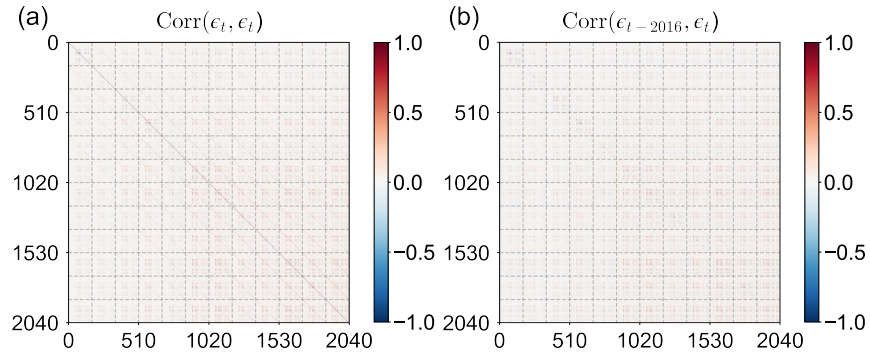


Fig. 4: Correlation matrices of residuals obtained from the Graph WaveNet model on the PEMS08 dataset after applying our method. (a) Contemporaneous correlations; (b) Cross-correlations at lag 2016 (i.e., one week).



The CRPS and  $\rho$ -risk are probabilistic scores employed to assess the efficacy of probabilistic forecasting. Notably, our method consistently demonstrates superior performance across nearly all scenarios, as reflected in these probabilistic scores. Specifically, CRPS evaluates the overall accuracy of the predictive distribution. Our approach yields average improvements of 6.37% and 11.72% on PEMS07 (M) and PEMS08, respectively. The  $\rho$ -risk is assessed at different quantiles ( $\rho = 0.5, 0.75, 0.9$ ) in this study. In accordance with Eq. (20), the 0.5-risk essentially equates to the MAE between the actual value and the 0.5-quantile of the predictive distribution. Conversely, the 0.75-risk and 0.9-risk apply less penalty when the associated quantiles surpass the actual value. For instance, the 0.9-quantile will be heavily penalized if it falls below the actual value. Our method significantly improves the 0.9-risk, where improvements amount to 9.60% and 13.97% on PEMS07 (M) and PEMS08, respectively.

Figure 4 visually demonstrates a significant reduction in both contemporaneous correlation and cross-correlation, as compared to Figure 3. It is essential to note that the reduction in both contemporaneous correlation and cross-correlation can be attributed to the error AR module, as more accurate predictions guide the error  $\epsilon_t$  towards isotropy. Simultaneously, spatiotemporal covariance learning has the effect of flexibly regularizing the error by allowing the existence of contemporaneous correlation.

#### 4.3.2 ABLATION STUDIES

In this section, we conduct ablation studies and sensitivity analyses to assess the contribution and effectiveness of each component within the DR framework.

**Component Ablations in the DR Framework.** Table 3 shows that the error AR process primarily enhances the mean prediction, as indicated by the significant improvement in RRMSE without  $\mathcal{L}_{\text{nil}}$ . On the other hand, the non-isotropic loss alone (w/o AR) provides a mild enhancement to the model in terms of RRMSE since it only regularizes the loss with a more flexible assumption. However, the full covariance learning appears to be beneficial when performing probabilistic forecasting, as indicated by the improvement seen in the 0.75-risk and 0.9-risk.

**Sensitivity Analysis of Rank Parameters.** Since  $Q = 12$  is already a small value, we fix  $R_q = Q = 12$  in this study and perform ablations on  $R_n$ . Table 4 shows that reducing  $R_n$  degrades probabilistic performance, as seen in CRPS and  $\rho$ -risk metrics. This decline is attributed to the reduced ability to effectively model the covariance. Due to the probabilistic nature of this component, its impact on RRMSE appears minimal and random.

**Performance Limits of the DR Framework.** To understand the performance limits of the proposed framework, we also investigate the predictive power of using only a bi-linear AR model similar to Eq. (5), with  $\mathbf{R}_t$  and  $\mathbf{R}_{t-\Delta}$  replaced by  $\mathbf{Y}_t$  and  $\mathbf{Y}_{t-\Delta}$ . Table 5 shows that the DL model alone outperforms the AR model regardless of the loss function used. This is because the AR model is too simple to effectively capture the traffic patterns. Furthermore, the performance of the AR model improves significantly when using a higher order AR process, highlighting its dependency on richer temporal dynamics. Lastly, the non-isotropic loss consistently enhances performance when applied to the same model, demonstrating its effectiveness in refining predictions.

Neither the DL model nor the AR model alone is able to outperform our proposed method. Although the DL model can theoretically capture all spatiotemporal correlations, making residuals i.i.d., achieving this in practice is difficult. It requires careful curation of spatiotemporal features and substantial effort in model design and optimization. In contrast, our method provides a model-agnostic alternative that enhances prediction performance by leveraging the statistical properties of the error term, offering a more practical and efficient solution.

Table 3: Ablation studies on DR components using Graph WaveNet and the PEMS08 dataset.

Model	RRMSE	CRPS	0.5-risk	0.75-risk	0.9-risk
Graph WaveNet	0.1707	0.0544	0.0698	0.0614	0.0389
+ DR (w/o AR)	0.1702	0.0543	0.0698	0.0605	0.0384
+ DR (w/o $\mathcal{L}_{\text{null}}$ )	<b>0.1628</b>	0.0524	0.0650	0.0606	0.0384
+ DR	0.1639	<b>0.0465</b>	<b>0.0619</b>	<b>0.0517</b>	<b>0.0321</b>

Table 4: Sensitivity analysis on the rank parameters  $R_n$  using Graph WaveNet and the PEMS08 dataset.

Model	RRMSE	CRPS	0.5-risk	0.75-risk	0.9-risk
Graph WaveNet	0.1707	0.0544	0.0698	0.0614	0.0389
+ DR ( $R_n = N$ )	0.1639	0.0465	0.0619	<b>0.0517</b>	<b>0.0321</b>
+ DR ( $R_n = 120$ )	<b>0.1612</b>	<b>0.0464</b>	<b>0.0611</b>	0.0525	0.0322
+ DR ( $R_n = 80$ )	0.1626	0.0482	0.0628	0.0551	0.0338
+ DR ( $R_n = 40$ )	0.1619	0.0492	0.0633	0.0565	0.0349
+ DR ( $R_n = 10$ )	0.1623	0.0506	0.064	0.0581	0.0366

Table 5: Empirical performance limits of our method using Graph WaveNet and the PEMS08 dataset.

Loss	Model	RRMSE	CRPS	0.5-risk	0.75-risk	0.9-risk
isotropic	DL only	0.1707	0.0544	0.0698	0.0614	0.0389
	AR ( $p = 1$ ) only	0.2838	0.0908	0.1102	0.1081	0.0705
	AR ( $p = 2$ ) only	0.2666	0.0863	0.106	0.1007	0.0646
	AR ( $p = 3$ ) only	0.2054	0.0677	0.0853	0.0794	0.0505
non-isotropic	DL only	0.1702	0.0543	0.0698	0.0605	0.0384
	AR ( $p = 1$ ) only	0.2599	0.0753	0.0984	0.0834	0.0546
	AR ( $p = 2$ ) only	0.2439	0.0709	0.093	0.0802	0.0532
	AR ( $p = 3$ ) only	0.1948	0.0596	0.0778	0.0688	0.0445
non-isotropic	DR	<b>0.1639</b>	<b>0.0465</b>	<b>0.0619</b>	<b>0.0517</b>	<b>0.0321</b>

### 4.3.3 MODEL INTERPRETATION

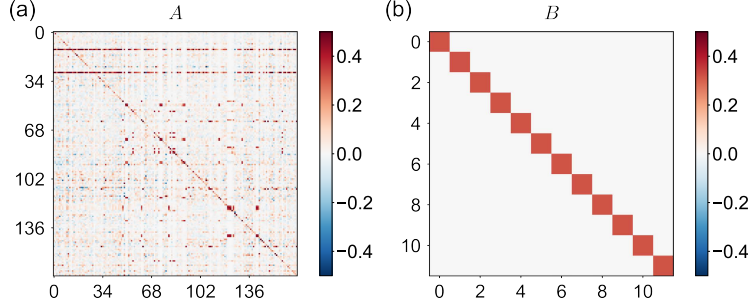


Fig. 5: Learned AR coefficient matrices  $\mathbf{A}$  and  $\mathbf{B}$  for the Graph WaveNet model on the PEMS08 dataset.

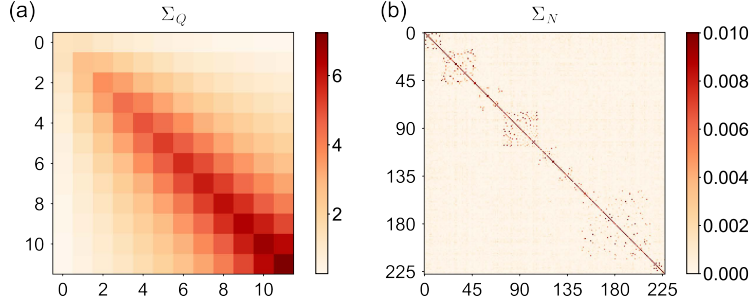


Fig. 6: Learned covariance matrices  $\Sigma_Q$  and  $\Sigma_N$  for the Graph WaveNet model on the PEMSD7(M) dataset.

We analyze coefficient matrices ( $\mathbf{A}$  and  $\mathbf{B}$ ) and covariance matrices ( $\Sigma_N$  and  $\Sigma_Q$ ) to interpret the model. Figure 5 showcases the coefficient matrices of the error AR module in Graph WaveNet on PEMS08. In Figure 5 (a), the correlations between residuals from different spatial locations are depicted. The prominent diagonal reveals that most spatial locations exhibit strong self-correlations, indicating that the residuals of the base DL model are autocorrelated. Intriguingly, certain locations display strong correlations with residuals from other locations, indicating that the spatial dependency has not been fully explored by the DL model. Figure 5 (b) further highlights a pronounced diagonal in matrix  $\mathbf{B}$ . Given that  $\mathbf{B}$  signifies the column effect of the past residual  $\mathbf{R}_{t-\Delta}$  on  $\mathbf{R}_t$ , the current residual showcases the highest correlation with the past residual at the corresponding forecasting step.

Figure 6 presents the learned error covariance matrices obtained for Graph WaveNet on PEMSD7 (M). The temporal covariance matrix  $\Sigma_Q$  (Figure 6 (a)) encapsulates the covariance across the prediction horizon. We observe that the diagonal elements of  $\Sigma_Q$  progressively expand with each prediction step, an intuitively rational behavior for multistep prediction tasks, where the uncertainty of prediction increases as we predict further into the future. This knowledge can be translated into probabilistic forecasting, where the prediction interval increases along the prediction horizon, providing additional information for the downstream

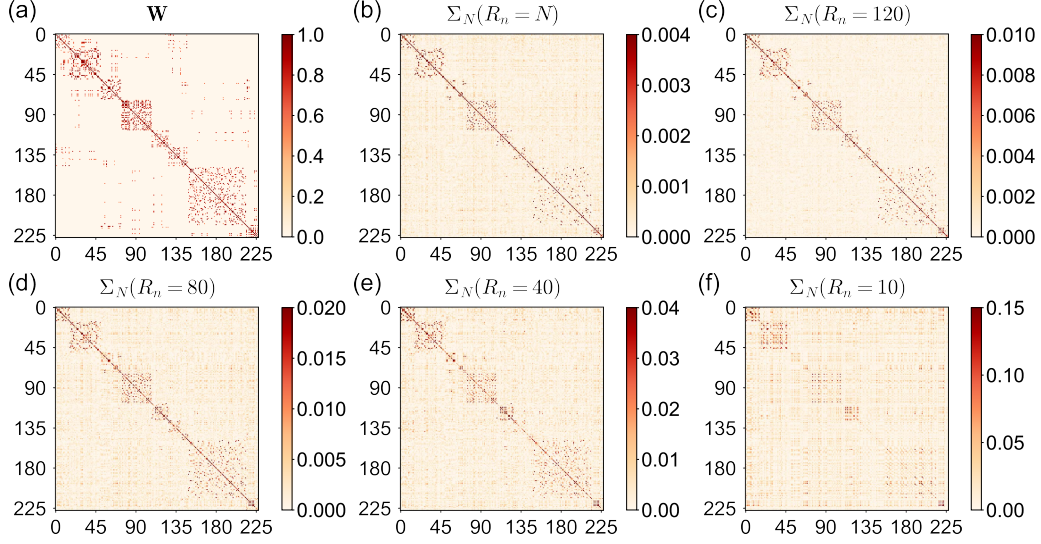


Fig. 7: Comparison of the adjacency matrix and learned covariance matrices  $\Sigma_N$  with different rank parameters  $R_n$  using Graph WaveNet and the PEMS7(M) dataset. Values in  $\Sigma_N$  are clipped to a specific range to enhance the visibility of the patterns.

decision-making process. Furthermore, the visualization reveals a discernible propagation of errors in the extended prediction period, evidenced by the elevated covariance between consecutive prediction steps. This suggests DL models could be improved to mitigate error propagation in multistep predictions. For example, one could leverage the observed residuals to calibrate next-step predictions in an autoregressive manner (Zheng et al., 2024; Zheng and Sun, 2024).

Figure 6 (b) illustrates the spatial covariance matrix  $\Sigma_N$ , which captures the statistical dependencies among the residuals across different spatial locations. Although  $\Sigma_N$  may exhibit some similarities to the adjacency matrix commonly used in GNNs (Figure 7 (a)), its role and interpretation are fundamentally distinct. Whereas the adjacency matrix encodes predefined or learned spatial dependencies, the covariance matrix captures error correlations that GNNs miss. These correlations may stem from factors not explicitly represented in the adjacency matrix, such as correlated noise or external influences.

The low-rank parameterization of  $\Sigma_Q$  and  $\Sigma_N$  also offers computational benefits. For example, as  $R_n$  decreases, the model demonstrates robustness by retaining its ability to recover covariance patterns represented by the full-rank parameterization (i.e.,  $R_n = N$ ), as shown in Figure 7 (b-e). This indicates the model can capture essential spatial dependencies even with reduced parameterization. However, when  $R_n$  is reduced to a very small value of 10 (Figure 7 (f)), the model begins to lose its capacity to recover intricate covariance patterns. Moreover, this parameterization represents each spatial location and forecasting step through  $R_n$  and  $R_q$  latent factors, respectively. The covariance structures induced by these factors are naturally expressed as  $\mathbf{L}_N \mathbf{L}_N^\top$  for spatial correlations and  $\mathbf{L}_Q \mathbf{L}_Q^\top$  for temporal correlations.

The fully learnable design of the proposed method serves as a robust enhancement to existing DL models. In cases of model misspecification, our method can theoretically reduce

to the original implementation. As shown in Eq. (7) and Eq. (10), the model becomes identical to the original implementation with an isotropic loss when  $\mathbf{A}/\mathbf{B}$  and  $\mathbf{\Sigma}_Q/\mathbf{\Sigma}_N$  are zero matrices. Thus, our method generalizes the conventional approach, allowing the model to dynamically adapt to the error generation and propagation process during training. Furthermore, Figure 8 indicates that the non-isotropic loss is more robust than the isotropic loss when there is misspecification in the error AR process.

#### 4.3.4 VALIDATION OF ASSUMPTIONS

In this section, we present experimental results that validate the two assumptions (Assumption 1 and Assumption 2) stated in Section 3.

To validate the first assumption, we test higher orders of the error AR process as shown in Figure 8. Using more lags in the error AR process does not necessarily improve performance and may degrade it with inappropriate lags (i.e.,  $\Delta = 12$  for the PEMS08 dataset). This is evident in Figure 3, where only the lag-2016 residuals are found to be correlated with the current residuals. Interestingly, the proposed non-isotropic loss demonstrates greater robustness to misspecification in the error AR process. As shown in Figure 8, when trained with non-isotropic loss, the inappropriate lag  $\Delta = 12$  does not negatively influence the model as it does with isotropic loss. This is likely attributed to the greater flexibility that the non-isotropic loss provides during training.

To validate the second assumption, we compare the learned covariance structure with the empirical covariance of the residuals. We collect all residuals  $\mathbf{E}_t$  in the testing set and calculate the row covariance and column covariance by:

$$\begin{aligned}\mathbf{E}_{\text{row}} &= [\mathbf{E}_1, \mathbf{E}_2, \dots, \mathbf{E}_T] \in \mathbb{R}^{N \times (T \times Q)} \\ \mathbf{\Sigma}_R &= \frac{1}{TQ - 1} \mathbf{E}_{\text{row}} \mathbf{E}_{\text{row}}^\top\end{aligned}\tag{21}$$

$$\begin{aligned}\mathbf{E}_{\text{col}} &= [\mathbf{E}_1^\top, \mathbf{E}_2^\top, \dots, \mathbf{E}_T^\top] \in \mathbb{R}^{Q \times (T \times N)} \\ \mathbf{\Sigma}_C &= \frac{1}{TN - 1} \mathbf{E}_{\text{col}} \mathbf{E}_{\text{col}}^\top\end{aligned}\tag{22}$$

where  $T$  is the number of observations in the testing set. The results are presented in Figure 9. We observe that the empirical covariance matrices exhibit a very similar structure to the learned covariance matrices shown in Figure 6. The Kronecker product with a diagonal structure simplifies log-likelihood evaluation in Eq. (10) using the Sherman–Morrison–Woodbury identity (Eq. (12) and Eq. (13)). Without this parameterization, computing the log-likelihood involves evaluating the inverse and the determinant of  $\mathbf{\Sigma}$  with size  $NQ \times NQ$ , for which a naïve implementation has a prohibitive time complexity of  $\mathcal{O}(N^3Q^3)$ . This is computationally prohibitive in most cases.

#### 4.3.5 VISUALIZATION OF PREDICTIONS

In this section, we evaluate our model by visualizing both deterministic and probabilistic forecast results. The deterministic forecast given by Eq. (9) combines predictions from the DL model  $f_\theta$  and the error AR process  $\mathbf{A}\mathbf{R}_{t-\Delta}\mathbf{B}$ . Figure 10 illustrates the breakdown of the forecasts made by our method, highlighting the complementary effects of the two

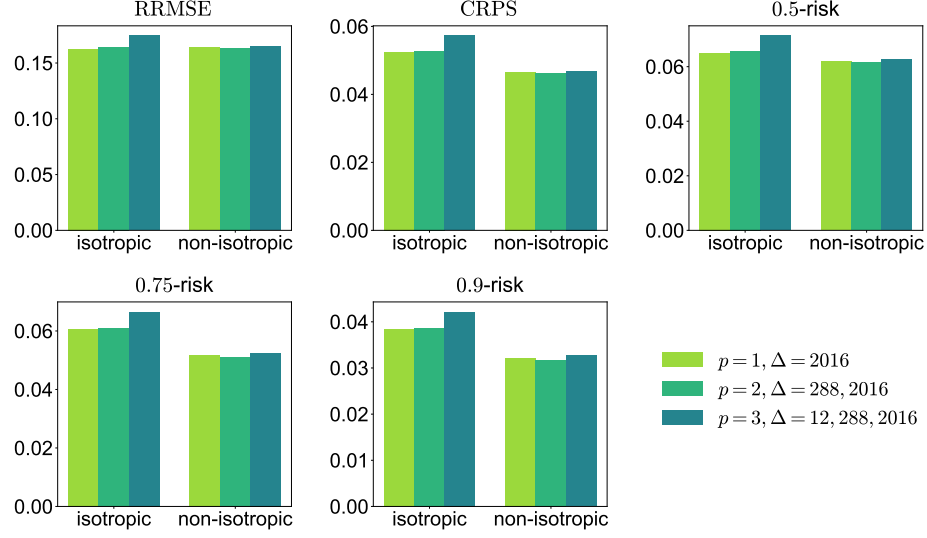


Fig. 8: Ablation studies on the order of the error AR process using the Graph WaveNet model on the PEMS08 dataset.

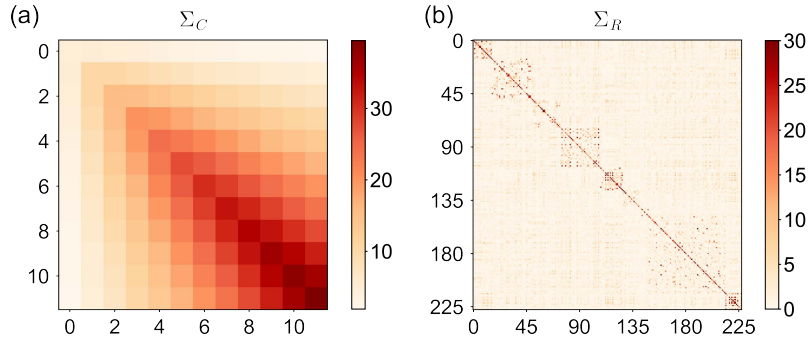


Fig. 9: Empirical covariance matrices  $\Sigma_R$  and  $\Sigma_C$  calculated using the residuals of the Graph WaveNet model on the PEMS07(M) dataset. Note that the value scales differ from those in Figure 6 because the residuals are presented in the original data scales.

components. The error AR process primarily focuses on seasonal characteristics, such as traffic flow during congested regimes, while the DL model captures short-range, rapidly changing patterns.

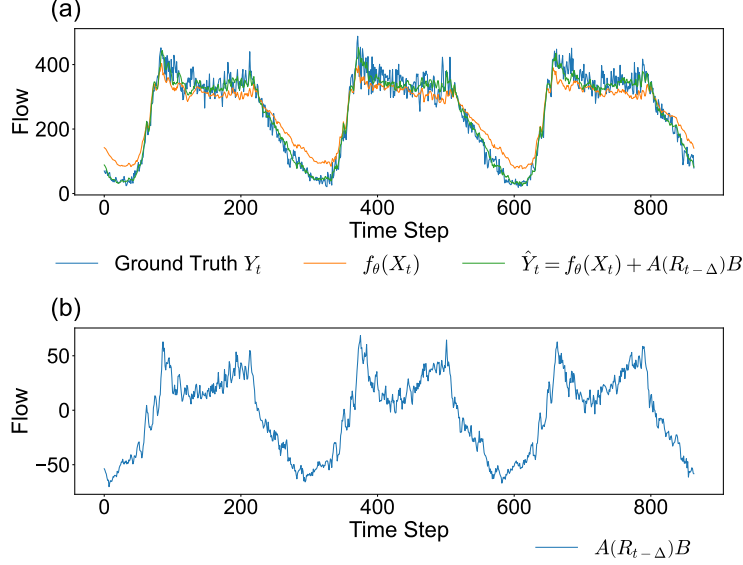


Fig. 10: An example of the 6-step-ahead prediction results using the Graph WaveNet with the DR framework for the last three days of the PEMS08 dataset for one traffic sensor.

The probabilistic forecast is given by the sampling process outlined in Eq. (16). The predicted mean  $\hat{\mathbf{Y}}_t$  serves as the location parameter in the predictive distribution for  $\mathbf{Y}_t$ , while  $\tilde{\mathbf{E}}_t$  represents a sample of the error term, capturing uncertainties in the forecasts by incorporating contemporaneous correlations learned in Section 3.3. Notably, the Kronecker structure within the covariance matrix explicitly characterizes both the covariance over the prediction horizon and the covariance over the spatial locations. We compare predictions from four randomly selected sensors at a fixed time using the original model and our DR framework.

Figure 11 and Figure 12 offer a visual comparison of the probabilistic forecasting performance of Graph WaveNet on PEMS07(M) with and without the incorporation of our method. In Figure 11, the original model assumes isotropic errors, with  $\boldsymbol{\eta}_t$  following a Gaussian distribution  $\mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_{NQ})$ . This results in a probabilistic forecasting characterized by constant variance across the prediction horizon. However, such an approach proves unrealistic as it fails to account for the escalating uncertainty when making predictions into the future. This condition holds true for the spatial dimension as well, where all sensors exhibit the same variance, and no across-sensor correlations are considered.

In contrast, Figure 12 provides a visual representation of probabilistic forecasts when our DR method is implemented, incorporating both the calibration from error AR module and the learning of contemporaneous correlations. The impact of the error AR module is discernible in the relative position of the predictive distribution with respect to the observations. A notable instance is the comparison of time series C in Figure 11 and



Figure 12, where the predictive distribution in Figure 12 exhibit a better alignment with the ground truth observations.

Furthermore, the explicit effects of modeling non-isotropic errors are evident in the widened uncertainty over the prediction horizon, as well as the diverse uncertainties observed for different time series at the same prediction step. Although the modeling of spatial correlations may not be directly observable, these factors are statistically accounted for during the sampling process. The noticeable variance disparity between the first and last prediction steps underscores the enhanced representation of uncertainty over time achieved through our proposed method.

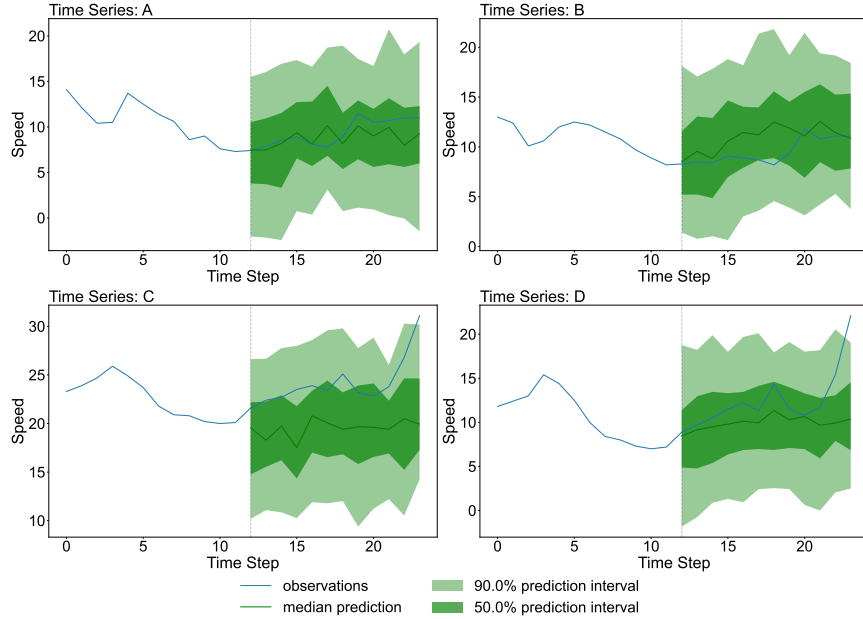


Fig. 11: Probabilistic forecasting examples using Graph WaveNet on PEMS7(M) without the integration of our proposed method. The dashed line splits the input time horizon ( $P$ ) and the output time horizon ( $Q$ )

## 5 Discussion

STF models, as they advance, can theoretically approximate independent errors. However, our framework highlights two alternative paths for STF models:

**Modeling Error Dynamics Separately.** The intentional separation between the base STF model and the error dynamics in the proposed DR framework explicitly divides the tasks into: 1) learning the primary spatiotemporal dependencies using the base model; and 2) capturing the error dynamics, including error correlations across time, space, and the prediction horizon, using the DR framework. This modular design acknowledges that, while an STF model may approximate independent errors, achieving this in practice remains challenging due to the complexity of spatiotemporal feature engineering and model design. Our framework provides a plug-and-play enhancement for any STF model, requiring no major architectural changes. A more detailed discussion is provided in Appendix A.

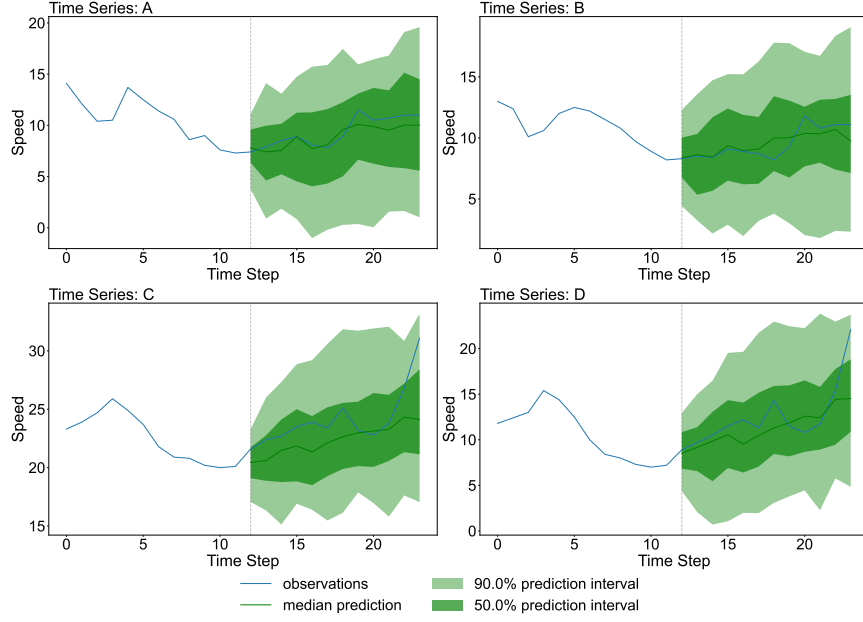


Fig. 12: Probabilistic forecasting examples using Graph WaveNet on PEMS7(M) with the integration of our proposed method. The dashed line splits the input time horizon ( $P$ ) and the output time horizon ( $Q$ )

**Advancing Probabilistic Forecasts.** Deterministic frameworks, constrained by isotropic assumptions, fail to capture two key phenomena in spatiotemporal traffic forecasting: 1) the growing uncertainty of predictions over the prediction horizon, represented by  $\Sigma_Q$  in our framework; and 2) the varying levels of variance across different spatial locations, represented by the diagonal elements of  $\Sigma_N$ . In contrast, our framework serves as a model-agnostic probabilistic enhancement that can be seamlessly integrated into any STF model, regardless of its deterministic performance. A more detailed discussion is provided in Appendix B.

To summarize, the conceptual contribution of our work lies in explicitly recognizing and efficiently modeling error correlations, rather than relying on the implicit assumption that a single STF model can fully capture them. This modular approach improves interpretability by revealing key error structures, such as increasing uncertainty over time and spatial error correlations, which remain obscured in end-to-end deep learning models.

## 6 Conclusion

This paper presents a dynamic regression framework that enhances deep spatiotemporal models for probabilistic traffic forecasting by relaxing assumptions of time-independent and isotropic errors. We adjust the loss function to account for correlated errors, enabling seamless integration with existing models. Specifically, we represent the error series using a first-order matrix-variate autoregressive process with a non-isotropic error term. The new parameters introduced by our method can be jointly learned with the base forecasting model. Extensive experiments with state-of-the-art traffic forecasting models on real-world

datasets validate the effectiveness of our DR framework. The acquired covariance matrix also facilitates probabilistic forecasting with enhanced uncertainty quantification for the multivariate Seq2Seq forecasting task. Additionally, the learned parameters offer clear physical and statistical interpretations.

This study has two primary limitations. First, learning the enriched error covariance structure increases training costs. For instance, applying our method to Graph WaveNet on PEMS08 increased training time by nearly 20%. Second, scalability challenges may persist when applying our method to datasets with very large spatial dimensions, despite optimization efforts. Future work can focus on scaling our method for higher-dimensional datasets and incorporating time-varying error covariance matrices.

This work is a first step in addressing spatiotemporal error correlation in multivariate Seq2Seq deep learning models for traffic forecasting, contributing to the field. While developed for traffic forecasting, our method can be adapted to broader spatiotemporal forecasting tasks, such as daily weather and climatology prediction.

## Acknowledgements

The research is supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada (Discovery Grant RGPIN 2019-05950). Vincent Zhihao Zheng also acknowledges the support received from the FRQNT B2X Doctoral Scholarship Program. The authors are grateful to the Associate Editor and three anonymous referees for valuable comments on an earlier version of the paper.

## References

- Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Nicolas Chapados, and Alexandre Drouin. Tactis-2: Better, faster, simpler attentional copulas for multivariate time series. *arXiv preprint arXiv:2310.01327*, 2023.
- Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in Neural Information Processing Systems*, 33:17804–17815, 2020.
- Charles M Beach and James G MacKinnon. A maximum likelihood procedure for regression with autocorrelated errors. *Econometrica: journal of the Econometric Society*, pages 51–58, 1978.
- Trevor S Breusch. Testing for autocorrelation in dynamic linear models. *Australian economic papers*, 17(31):334–355, 1978.
- Rong Chen, Han Xiao, and Dan Yang. Autoregressive models for matrix-valued time series. *Journal of Econometrics*, 222(1):539–560, 2021.
- Zhanhong Cheng, Martin Trépanier, and Lijun Sun. Incorporating travel behavior regularity into passenger flow forecasting. *Transportation Research Part C: Emerging Technologies*, 128:103200, 2021.

- Seongjin Choi, Nicolas Saunier, Martin Trepanier, and Lijun Sun. Spatiotemporal residual regularization with dynamic mixtures for traffic forecasting. *arXiv preprint arXiv:2212.06653*, 2022.
- Donald Cochran and Guy H Orcutt. Application of least squares regression to relationships containing auto-correlated error terms. *Journal of the American statistical association*, 44(245):32–61, 1949.
- Emmanuel de Bézenac, Syama Sundar Rangapuram, Konstantinos Benidis, Michael Bohlke-Schneider, Richard Kurle, Lorenzo Stella, Hilaf Hasson, Patrick Gallinari, and Tim Januschowski. Normalizing kalman filters for multivariate time series analysis. *Advances in Neural Information Processing Systems*, 33:2995–3007, 2020.
- Alexandre Drouin, Étienne Marcotte, and Nicolas Chapados. Tactis: Transformer-attentional copulas for time series. In *International Conference on Machine Learning*, pages 5447–5493. PMLR, 2022.
- James Durbin and Geoffrey S Watson. Testing for serial correlation in least squares regression: I. *Biometrika*, 37(3/4):409–428, 1950.
- Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.
- Leslie G Godfrey. Testing against general autoregressive and moving average error models when the regressors include lagged dependent variables. *Econometrica: Journal of the Econometric Society*, pages 1293–1301, 1978.
- Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 922–929, 2019.
- Shengnan Guo, Youfang Lin, Huaiyu Wan, Xiucheng Li, and Gao Cong. Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 34(11):5415–5428, 2021.
- James Douglas Hamilton. *Time Series Analysis*. Princeton University Press, 2020.
- Nan-Jung Hsu, Hsin-Cheng Huang, and Ruey S Tsay. Matrix autoregressive spatio-temporal models. *Journal of Computational and Graphical Statistics*, 30(4):1143–1155, 2021.
- Qian Huang, Horace He, Abhay Singh, Ser-Nam Lim, and Austin R Benson. Combining label propagation and simple models out-performs graph neural networks. *arXiv preprint arXiv:2010.13993*, 2020.
- Rob J Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, 2018.
- Junteng Jia and Austin R Benson. Residual correlation in graph neural network regression. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 588–598, 2020.

- Daejin Kim, Youngin Cho, Dongmin Kim, Cheonbok Park, and Jaegul Choo. Residual correction in real-time traffic forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 962–971, 2022.
- 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*, pages 95–104, 2018.
- Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International conference on learning representations*, 2018.
- Xu Liu, Yutong Xia, Yuxuan Liang, Junfeng Hu, Yiwei Wang, Lei Bai, Chao Huang, Zhenguang Liu, Bryan Hooi, and Roger Zimmermann. Largest: a benchmark dataset for large-scale traffic forecasting. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS ’23*, Red Hook, NY, USA, 2024. Curran Associates Inc.
- Greta M Ljung and George EP Box. On a measure of lack of fit in time series models. *Biometrika*, 65(2):297–303, 1978.
- Jaakko Luttinen and Alexander Ilin. Efficient gaussian process inference for short-scale spatio-temporal modeling. In *Artificial Intelligence and Statistics*, pages 741–750. PMLR, 2012.
- Raquel Prado, Marco AR Ferreira, and Mike West. *Time Series: Modeling, Computation, and Inference*. CRC Press, 2021.
- Sigbert J Prais and Christopher B Winsten. Trend estimators and serial correlation. Technical report, Cowles Commission discussion paper Chicago, 1954.
- Syama Sundar Rangapuram, Matthias W Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. Deep state space models for time series forecasting. *Advances in neural information processing systems*, 31, 2018.
- Kashif Rasul, Abdul-Saboor Sheikh, Ingmar Schuster, Urs Bergmann, and Roland Vollgraf. Multivariate probabilistic time series forecasting via conditioned normalizing flows. *arXiv preprint arXiv:2002.06103*, 2020.
- Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning*, pages 8857–8868. PMLR, 2021.
- David Salinas, Michael Bohlke-Schneider, Laurent Callot, Roberto Medico, and Jan Gasthaus. High-dimensional multivariate forecasting with low-rank gaussian copula processes. *Advances in neural information processing systems*, 32, 2019.
- 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.

- ZeZhi Shao, Zhao Zhang, Wei Wei, Fei Wang, Yongjun Xu, Xin Cao, and Christian S. Jensen. Decoupled dynamic spatial-temporal graph neural network for traffic forecasting. *Proc. VLDB Endow.*, 15(11):2733–2746, July 2022. ISSN 2150-8097. doi: 10.14778/3551793.3551827. URL <https://doi.org/10.14778/3551793.3551827>.
- ZeZhi Shao, Fei Wang, Yongjun Xu, Wei Wei, Chengqing Yu, Zhao Zhang, Di Yao, Tao Sun, Guangyin Jin, Xin Cao, et al. Exploring progress in multivariate time series forecasting: Comprehensive benchmarking and heterogeneity analysis. *IEEE Transactions on Knowledge and Data Engineering*, 2024.
- Shun-Yao Shih, Fan-Keng Sun, and Hung-yi Lee. Temporal pattern attention for multivariate time series forecasting. *Machine Learning*, 108:1421–1441, 2019.
- Chao Song, Youfang Lin, Shengnan Guo, and Huaiyu Wan. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *AAAI Conference on Artificial Intelligence*, volume 34, pages 914–921, 2020.
- Dongjin Song, Haifeng Chen, Guofei Jiang, and Yao Qin. Dual stage attention based recurrent neural network for time series prediction, February 23 2021. US Patent 10,929,674.
- Oliver Stegle, Christoph Lippert, Joris M Mooij, Neil Lawrence, and Karsten Borgwardt. Efficient inference in matrix-variate gaussian models with iid observation noise. *Advances in neural information processing systems*, 24, 2011.
- Fan-Keng Sun, Chris Lang, and Duane Boning. Adjusting for autocorrelated errors in neural networks for time series. *Advances in Neural Information Processing Systems*, 34: 29806–29819, 2021.
- Binh Tang and David S Matteson. Probabilistic transformer for time series analysis. *Advances in Neural Information Processing Systems*, 34:23592–23608, 2021.
- Jean-François Toubreau, Jérémie Bottieau, François Vallée, and Zacharie De Grève. Deep learning-based multivariate probabilistic forecasting for short-term scheduling in power markets. *IEEE Transactions on Power Systems*, 34(2):1203–1215, 2018.
- Eleni I Vlahogianni, Matthew G Karlaftis, and John C Golias. Short-term traffic forecasting: Where we are and where we’re going. *Transportation Research Part C: Emerging Technologies*, 43:3–19, 2014.
- Yuyang Wang, Alex Smola, Danielle Maddix, Jan Gasthaus, Dean Foster, and Tim Januschowski. Deep factors for forecasting. In *International conference on machine learning*, pages 6607–6617. PMLR, 2019.
- Ruofeng Wen, Kari Torkkola, Balakrishnan Narayanaswamy, and Dhruv Madeka. A multi-horizon quantile recurrent forecaster. *arXiv preprint arXiv:1711.11053*, 2017.
- Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. page 1907–1913, 2019.

- Jiexia Ye, Juanjuan Zhao, Kejiang Ye, and Chengzhong Xu. How to build a graph-based deep learning architecture in traffic domain: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(5):3904–3924, 2020.
- Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, page 3634–3640, 2018.
- Chuanpan Zheng, Xiaoliang Fan, Cheng Wang, and Jianzhong Qi. Gman: A graph multi-attention network for traffic prediction. In *AAAI Conference on Artificial Intelligence*, volume 34, pages 1234–1241, 2020.
- Vincent Zhihao Zheng and Lijun Sun. Multivariate probabilistic time series forecasting with correlated errors. *arXiv preprint arXiv:2402.01000*, 2024.
- Zhihao Zheng, Seongjin Choi, and Lijun Sun. Better batch for deep probabilistic time series forecasting. In *International Conference on Artificial Intelligence and Statistics*, pages 91–99. PMLR, 2024.



## Appendix A. Benefits of Separately Modeling the Error Dynamics

This section discusses the benefits of the intentional separation between the base GNN model and the error dynamics in the proposed DR framework. This separation explicitly divides the tasks into:

1. Learning the primary spatiotemporal dependencies (with the base GNN model).
2. Capturing the error dynamics, including error correlations across time, space, and prediction horizon (with the DR framework).

This modular design recognizes that while GNN models may approximate independent errors, achieving this in practice is challenging due to the complexity of spatiotemporal feature engineering and model design. Our framework serves as a plug-and-play enhancement for any GNN model, improving performance without extensive redesign. We provide an example in Table 6 using a SOTA GNN model, D<sup>2</sup>STGNN (Shao et al., 2022), as shown in the LargeST benchmark (Liu et al., 2024). We observe that D<sup>2</sup>STGNN, a more recent model developed three years after Graph WaveNet, achieves only a marginal improvement in performance (1.01%  $\sim$  1.69% in RRMSE, and 1.28%  $\sim$  2.57% in CRPS) despite a significant increase (around 27.7%) in parameter size and the introduction of a new model design. In comparison, when enhanced with our framework, Graph WaveNet achieves a comparable improvement in RRMSE, the deterministic score, with a smaller parameter increase (16.8%  $\sim$  18.8%) and without the need for rigorous model redesign. Crucially, our framework significantly improves probabilistic forecasts, as reflected in the CRPS.

Table 6: Efficiency analysis of the DR framework. The DR framework can enhance an older model to achieve the performance of a SOTA model in RRMSE while significantly improving probabilistic performance in CRPS. The relative changes are calculated based on Graph WaveNet.

Data	Model	Num. Params.	RRMSE	CRPS
PEMSD7	Graph WaveNet (2019)	309,757	0.3676	0.0427
	D <sup>2</sup> STGNN (2022)	395,709 ( $\uparrow$ 27.7%)	0.3639 ( $\downarrow$ 1.01%)	0.0416 ( $\downarrow$ 2.57%)
	Graph WaveNet + DR	361,885 ( $\uparrow$ 16.8%)	0.3622 ( $\downarrow$ 1.46%)	0.0389 ( $\downarrow$ 8.89%)
PEMS08	Graph WaveNet (2019)	308,597	0.1707	0.0544
	D <sup>2</sup> STGNN (2022)	394,317 ( $\uparrow$ 27.7%)	0.1678 ( $\downarrow$ 1.69%)	0.0537 ( $\downarrow$ 1.28%)
	Graph WaveNet + DR	366,685 ( $\uparrow$ 18.8%)	0.1639 ( $\downarrow$ 3.98%)	0.0465 ( $\downarrow$ 14.5%)

## Appendix B. Limitations of the Deterministic Framework

The common deterministic framework, constrained by its isotropic assumption, fails to capture two key phenomena in spatiotemporal traffic forecasting: increasing uncertainty over the prediction horizon ( $\Sigma_Q$ ) and spatially varying variance ( $\Sigma_N$ ). In contrast, our framework can serve as a model-agnostic probabilistic enhancement to any SOTA GNN, irrespective of

its deterministic performance. As shown in Table 7, our method can still achieve significant improvements in probabilistic scores when applied to the SOTA D<sup>2</sup>STGNN.

Table 7: Performance comparison of D<sup>2</sup>STGNN with and without the DR framework. The best results are highlighted in **bold**.

Data	Model	RRMSE	CRPS	0.5-risk	0.75-risk	0.9-risk
PEMSD7	D <sup>2</sup> STGNN	0.3639	0.0416	0.0489	0.0452	0.0305
	+ DR	<b>0.3592</b>	<b>0.0391</b>	<b>0.0481</b>	<b>0.0407</b>	<b>0.0271</b>
	rel. impr.	1.29%	6.01%	1.64%	9.96%	11.15%
PEMS08	D <sup>2</sup> STGNN	0.1678	0.0537	0.0683	0.0615	0.0388
	+ DR	<b>0.1665</b>	<b>0.0481</b>	<b>0.0645</b>	<b>0.0545</b>	<b>0.0331</b>
	rel. impr.	0.77%	10.43%	5.56%	11.38%	14.69%

To summarize, the conceptual contribution of our work lies in explicitly acknowledging and parsimoniously modeling error correlations, rather than relying on the implicit assumption that a single deterministic model (GNN alone) can fully capture them. This modular approach significantly enhances interpretability, as the DR framework provides valuable insights into the error structure (e.g., growing uncertainty over time and spatial error correlations) that would otherwise remain hidden in a purely end-to-end GNN model. Moreover, our framework remains consistent with deterministic assumptions, as its parameters adapt to different models and datasets. It reduces to the original model when  $\mathbf{A} = \mathbf{0}$  and  $\Sigma_Q = \Sigma_N = \mathbf{0}$ .

### Appendix C. Consistency with Traffic Flow Theory

To validate consistency with traffic flow theory, we compare the speed-flow relationship from PEMS08 ground truth data with predictions from Graph WaveNet trained with our DR framework. We analyze three traffic scenarios: workday non-rush hours (Figure 13), workday rush hours (Figure 14), and holidays (Figure 15). Results indicate strong alignment between predicted and ground truth speed-flow relationships across scenarios.

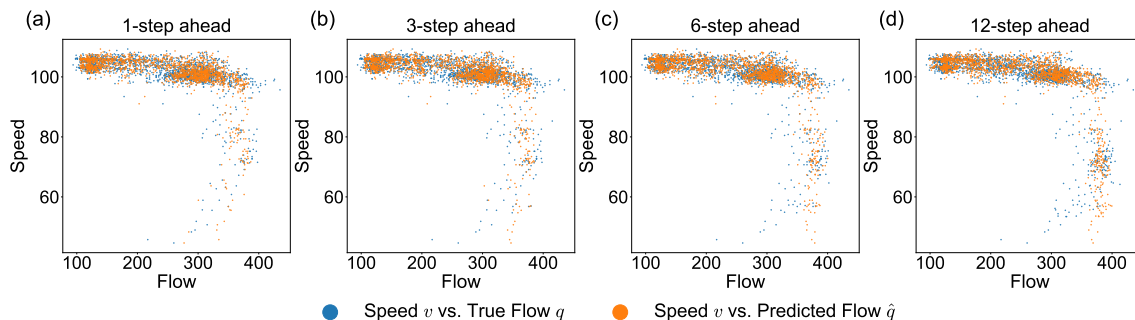


Fig. 13: Comparison of the speed-flow relationship using ground truth data and predicted values during workday non-rush hours. Results are obtained using the Graph WaveNet model on the PEMS08 dataset.

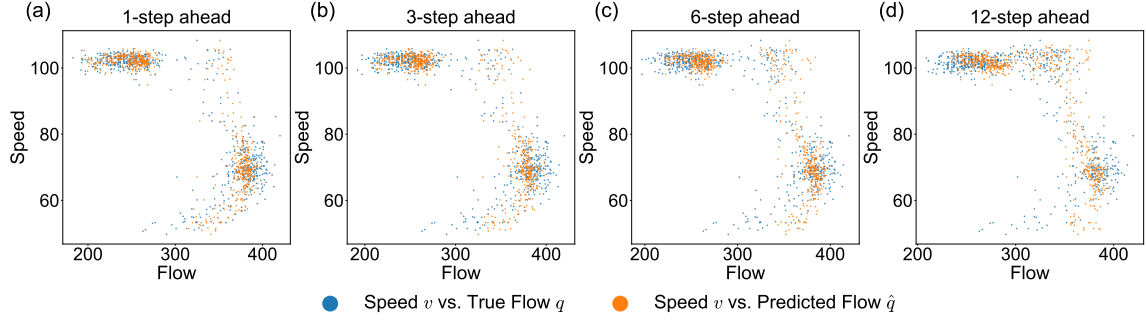


Fig. 14: Comparison of the speed-flow relationship using ground truth data and predicted values during weekday rush hours. Results are obtained using the Graph WaveNet model on the PEMS08 dataset.

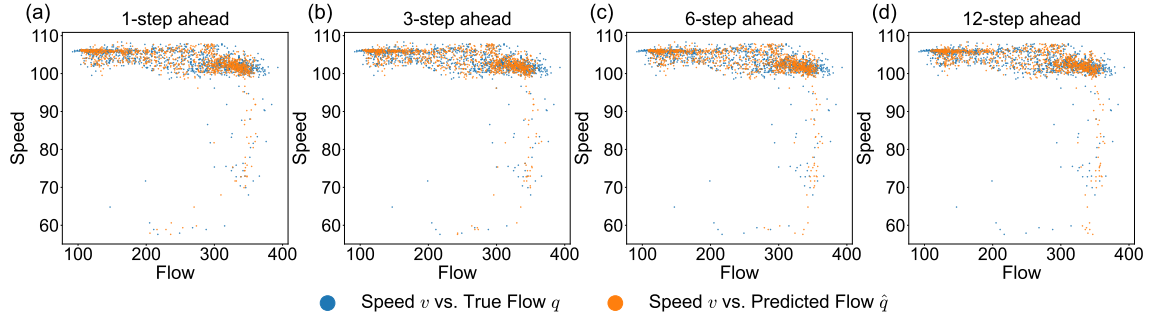


Fig. 15: Comparison of the speed-flow relationship using ground truth data and predicted values during holiday. Results are obtained using the Graph WaveNet model on the PEMS08 dataset.

## Appendix D. Additional Plots

We present an additional comparison for Graph WaveNet on PEMS08 in Figure 16 and Figure 17. Notably, our model demonstrates improved alignment between the predictive distribution and observations. Specifically, in Figure 17, the 50% prediction intervals consistently encompass the ground truth observations across various forecasting steps, unlike Figure 16. The variance increase is less pronounced in flow data than in speed data, as evident in the comparison between Figure 12 and Figure 17. This observation may suggest that predicting future values in the flow dataset involves less uncertainty than in the speed dataset, possibly due to the stronger seasonality effect in the flow data providing more information for longer-term forecasts. This also aligns with the indication of the optimal  $\Delta = 2016$  for PEMS08 in our error AR module.

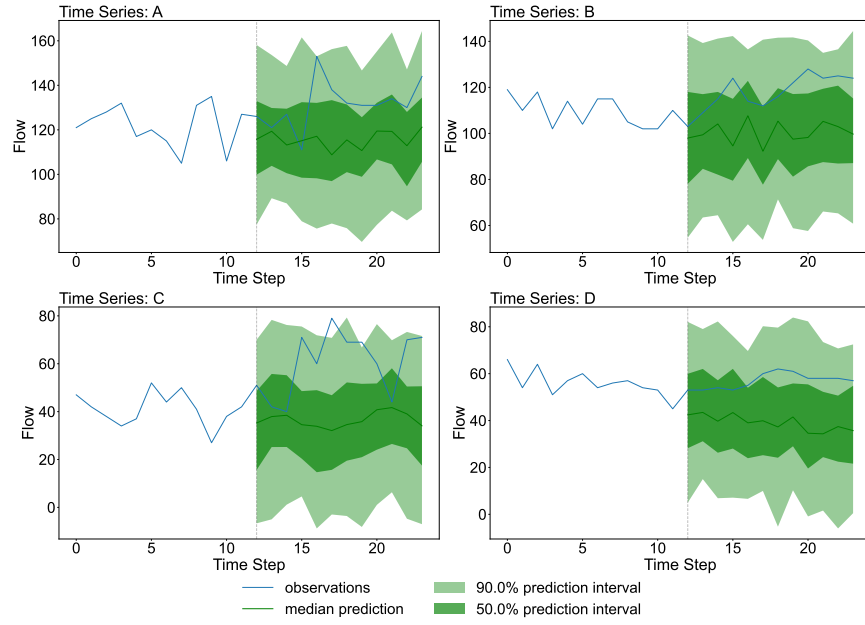


Fig. 16: Probabilistic forecasting examples using Graph WaveNet on PEMS08 without the integration of our proposed method. The dashed line splits the input time horizon ( $P$ ) and the output time horizon ( $Q$ )

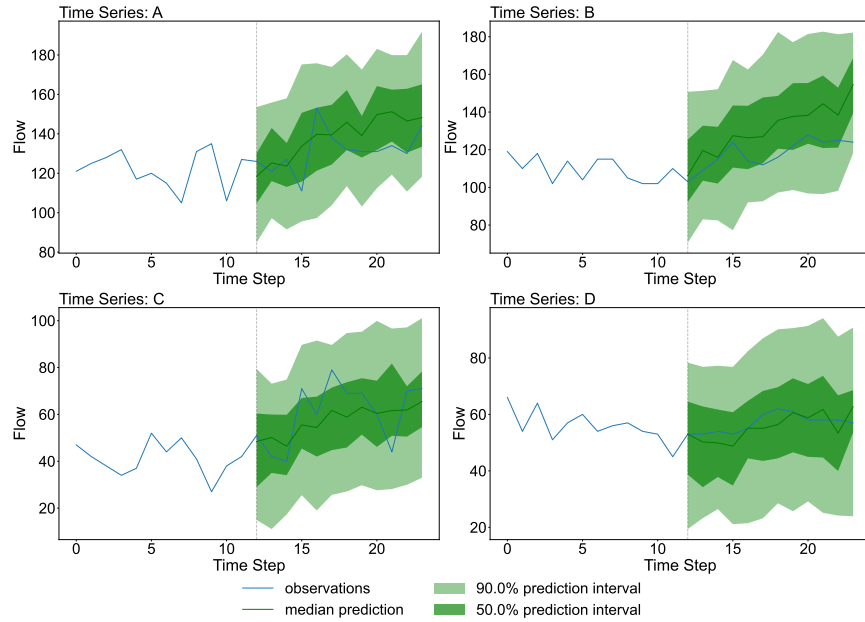


Fig. 17: Probabilistic forecasting examples using Graph WaveNet on PEMS08 with the integration of our proposed method. The dashed line splits the input time horizon ( $P$ ) and the output time horizon ( $Q$ )