Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)

Reconstructing Missing Variables for Multivariate Time Series Forecasting via Conditional Generative Flows

Xuanming Hu[1] , **Wei Fan**[2] , **Haifeng Chen**[3] , **Pengyang Wang**[4*] , **Yanjie Fu**[1*]
[1]School of Computing and Augmented Intelligence, Arizona State University
[2]Medical Sciences Division, University of Oxford
[3]NEC Laboratories America Inc
[4]Department of CIS, SKL-IOTSC, University of Macau
{solomonhxm, yanjie.fu}@asu.edu, wei.fan@wrh.ox.ac.uk, haifeng@nec-labs.com,
pywang@um.edu.mo

Abstract

The *Variable Subset Forecasting* (VSF) problem, where the majority of variables are unavailable in the inference stage of multivariate forecasting, has been an important but under-explored task with broad impacts in many real-world applications. Missing values, absent inter-correlation, and the impracticality of retraining largely hinder the ability of multivariate forecasting models to capture inherent relationships among variables, impacting their performance However, existing approaches towards these issues either heavily rely on local temporal correlation or face limitations in fully recovering missing information from the unavailable subset, accompanied by notable computational expenses. To address these problems, we propose a novel density estimation solution to recover the information of missing variables via flows-based generative framework. In particular, a novel generative network for time series, namely Time-series Reconstruction Flows (TRF), is proposed to estimate and reconstruct the missing variable subset. In addition, a novel meta-training framework, Variable-Agnostic Meta Learning, has been developed to enhance the generalization ability of TRF, enabling it to adapt to diverse missing variables situations. Finally, extensive experiments are conducted to demonstrate the superiority of our proposed method compared with baseline methods.

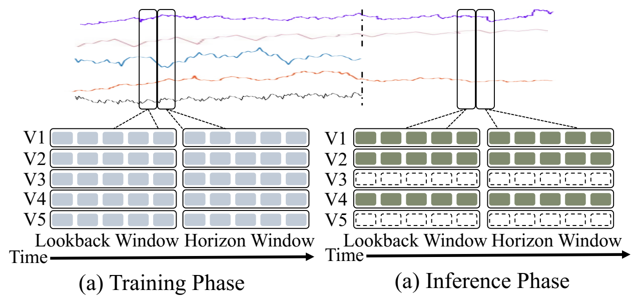


Figure 1: Setting of Variable Subset Time-series Forecast: During training, Variables V1 through V5 are given whereas during the inference phase, only V1, V2, V4 are accessible.

1 Introduction

Multivariate time series forecasting is critical in many real world applications, such as traffic prediction [Bai *et al.*, 2020], tourism demand analysis [Du Preez and Witt, 2003] and air quality estimation [Du *et al.*, 2019]. However, real world multivariate time series data are usually imperfect and incomplete. For instance, in wind speed forecasting, when there are malfuncitoning sensors, long term data unavailability arises; when there are dynamic fluctuations in resource availability, resource domain shifts arise [Chauhan *et al.*, 2022]. This can be generalized as the task of Variable Subset time-series Forecast (VSF). Figure 1 shows the VSF task has two steps: 1) in the training step, a multivariate time series forecasting model can access to all variables for training; 2) in the inference step, only a subset of variables are available and the other variables are missing. The objective is to forecast the future values of the available variable subset.

The VSF task introduce two unique issues for existing multivariate time series forecasting literature: 1) ***Issue 1: incomplete inter-variable correlation awareness***: Due to the unavailability of variables, existing multivariate time series forecasting models is incapable of modeling the interdependency between missing variables and available variables. The incomplete correlation awareness hinder forecasting performance [Guo *et al.*, 2019; Bai *et al.*, 2018]; 2) ***Issue 2: variable subset agnostic generalization***: In a dynamic open environment, it is usually unlikely to know which variables will be absent during inference/testing in advance. Once available variable subset is changed, we have to retrain forecasting models, which is inefficient and untimely. Moreover, training many subset-specific forecasting models over all possible variable combinations is computationally costly and impractical in real world deployment.

To tackle the two issues of VSF, an intuitive idea is to impute missing data. Prior imputation techniques capture both inter-variable and intra-variable correlation to impute missing values [Luo *et al.*, 2018; Tang *et al.*, 2020]. However,

* Corresponding authors

the complete absence (not partial) of variables impedes their ability to capture inter-variable information (**Issue 1**), making such techniques unsuitable for the VSF setting. VSF remained understudied until the work in [Chauhan *et al.*, 2022] introduced a wrapper approach, i.e., Forecast Distance Weighting (FDW), to recover missing variables from the nearest neighbors selected by ensemble weighting. However, top-K neighbors based recovery only consider partial information and could suffer if any top-K neighbor is biased, and the hyper parameter K is not generalizable. Moreover, FDW needs to re-search neighbors whenever the missing variable subset changes (**Issue 1 & 2**).

To address the issues, we develop deep time series generative reconstruction flow for VSF. Our approach builds upon two major contributions:

First, Time-series Reconstruction Flows (TRF) for recovering missing subset. To address the first issue, we propose to reformulate the missing variable reconstruction problem as deep collective generative task that model the dynamic temporal pattern of time series data. This generative reformulation inspires a density estimation perspective to generate the missing variables. Specifically, based on the available variable subset, we intend to estimate the unknown conditional density of unavailable variables via generative networks and rebuild the missing variables by leveraging the estimated distribution. Inspired by human reasoning [Byrne, 2019; Hu *et al.*, 2024], which seeks to convey complex ideas through the use of familiar and simple objects, we can express the intricate density of inaccessible variables by transforming a simple distribution via a set of mapping function. To implement the mapping process, we introduce ***invertible Time-series Reconstruction Flows*** (TRF), a conditional flows-based generative framework, to reconstruct the missing variables for multivariate forecasting. The fully invertible flow structure of TRF can ensure accurate reconstruction by mapping any distribution of missing variable subset to the Gaussian distribution through forward transformation and rebuilding it with the inverse transformation. In addition, we incorporate the available subset as the conditional input, enabling us to effectively constrain the generation of missing variables by leveraging the inter-variable correlation between the missing and available variable subset. To update TRF, we utilize a log-likelihood training schema. This proves advantageous for our framework as it facilitates learning the generation process without introducing extensive constraints or losses, making our framework easy to train.

Second, Variable-Agnostic learning paradigm for TRF. Retraining forecasting models for a dynamic available variable subset and iterating through all possible variable subset combinations are both impractical and costly. In response, we propose a meta-learning based training framework, ***Variable-Agnostic Meta Learning***, to help TRF to quickly adapt to new available variables subsets. Specifically, we regard diverse variable subsets as different meta-learning tasks. For each epoch, we sample a batch of variable subsets and update the parameters of TRF for each variable subset. We aggregate the subset-specific parameters and update the TRF by this aggregated parameters. Hence, the parameters of TRF can be more generalized for all possible variable subset. For a new missing variable subset, a small number of gradient steps will yield good generalization performance, allowing TRF to quickly adapt to this new scenario and tackle **Issue 2**.

In summary, we reformulate the VSF problem from a novel density estimation perspective. We propose a novel flow-based framework TRF to recover the information of missing variable subset. To improve the generalization ability of TRF, we design a training framework, namely Variable-Agnostic Meta Learning. Ultimately, we conduct extensive experiments to indicate the effectiveness of our framework.

2 Related Works

Variable Subset Time-series Forecasting. Multivariate time series forecasting has been an abiding research topic since last century [Yi *et al.*, 2024b; Yi *et al.*, 2024a; Hu *et al.*, 2023; Fan *et al.*, 2023; Ren *et al.*, 2022; Fan *et al.*, 2022]. To model temporal patterns and pairwise dependencies among variables, researchers has proposed several GNN-based models, such as DCRNN [Li *et al.*, 2017], MTGNN [Wu *et al.*, 2020], and ASTGCN [Guo *et al.*, 2019]. However, Variable Subset Time-series Forecast is underexplored. Missing several variables in the inference phase will considerably hinder the ability of GNN-based models to capture the inter-dependencies among variables. To tackle this problem, [Chauhan *et al.*, 2022] propose their wrapper technique by borrowing data from training set. However, they fail to consider the distribution shift of time series and make an imprecise estimation of the missing variables. In this work, we utilize a deep learning method to estimate the density of missing variables and the empirical results demonstrate the effectiveness and superiority of our proposed solution.

Normalizing Flows. Normalizing flows are a set of bijective mapping functions which are designed to model an unknown distribution based on the given data [Dinh *et al.*, 2014; Dinh *et al.*, 2016]. For example, based on the coupling layer proposed by [Dinh *et al.*, 2016], Glow [Kingma and Dhariwal, 2018] is constructed with more complex structure and achieves superior performance. MAF, proposed by [Papamakarios *et al.*, 2017], utilizes the autoregressive mechanism for better density estimation.

Meta-learning. Dissimilar to traditional machine learning methods, meta-learning algorithms aim to acquire generalizable information and transfer the knowledge to new tasks [Vanschoren, 2018]. For instance, MAML algorithm proposed by [Finn *et al.*, 2017] can prompt faster training in new tasks by a small number of gradient steps with a small amount of training data. Reptile [Nichol and Schulman, 2018] initials the parameters of a neural network via stochastic gradient descent on each task.

3 Problem Formulations

Let $\mathbf{X}_t^{(N)} \in \mathbb{R}^N$ denote the recording of N distinct variables at time-step t, we define historical L values as the lookback window $\mathbf{X}_{t-L:t}^{(N)} = \{\mathbf{X}_{t-L+1}^{(N)}, \cdots, \mathbf{X}_t^{(N)}\}$, and define the future H values as the horizon window $\mathbf{X}_{t:t+H}^{(N)} = \{\mathbf{X}_{t+1}^{(N)}, \cdots, \mathbf{X}_{t+H}^{(N)}\}$. For the classical setting of multivariate

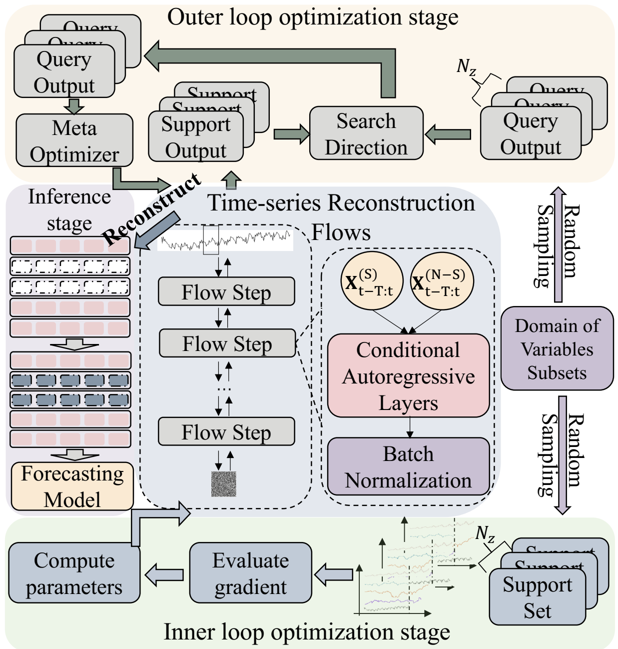


Figure 2: An overview of our proposed methods, consisting of our Time-series Reconstruction Flows and the training framework of Variable-Agnostic Meta Learning.

time series forecasting, we intend to find a mapping function $\mathcal{F}_\theta$ to project lookback window into horizon window, which can be formulated as: $\mathbf{X}_{t:t+H}^{(N)} = \mathcal{F}_\theta(\mathbf{X}_{t-L:t}^{(N)})$.

However, in real life, the data collection for time series can meet a common issue: due to the long-term unavailability of data [Yick *et al.*, 2008] and resource domain shift in time series [Hamid and Wibowo, 2018], chances are that only partial variables are available in the future [Chauhan *et al.*, 2022]. Accordingly, this problem of time series forecasting with partial variables during inference is formulated as ***Variable Subset Forecasting*** (VSF) problem. Under such settings, we denote the available variable subset as $\mathcal{S}$ and the universe of variables as $\mathcal{N}$, i.e., $\mathcal{S} \subset \mathcal{N}$ and $|\mathcal{S}| \ll |\mathcal{N}|$, where $\mathcal{N}$ is denoted as the set of N integers, $[[1 \ldots N]]$. In the training phase, each input $\mathbf{X}_t^{(N)}$ is a vector in $\mathbb{R}^{|\mathcal{N}| \times D}$ while in the inference phase, each data point can be represented as $\mathbf{X}_t^{(S)} \in \mathbb{R}^{|\mathcal{S}| \times D}$. In other words, though the models can take all variables in training, only a small subset of variables is considered in inference; formally, the VSF problem can be written as:

$$\begin{aligned} \mathbf{X}_{t:t+H}^{(S)} &= \mathcal{F}_\theta(\mathbf{X}_{t-L:t}^{(S)}),\ t \in \mathcal{T}_{test}, \\ \text{while}\ \ \mathbf{X}_{t:t+H}^{(N)} &= \mathcal{F}_\theta(\mathbf{X}_{t-L:t}^{(N)}),\ t \in \mathcal{T}_{train} \end{aligned} \tag{1}$$

where $\mathcal{T}_{train}$ and $\mathcal{T}_{test}$ are sets of the time-step of training data and test data.

4 Methodology

We introduce our proposed framework for VSF problem. First, we formulate VSF problem from a novel generative density estimation perspective in Section 4.1. We reconstruct the missing variables via density estimation and generation. Then, we concretely propose a novel generative network, ***Time-series Reconstruction Flows*** (TRF) to recover the information of absent variables and elaborate how to integrate our framework and forecasting models in Section 4.2. To further improve the generalization ability of the model, we propose meta training framework, ***Variable-Agnostic Meta Learning***, to TRF in Section 4.3.

4.1 Density Estimation and Generation for Variable Subset of Time Series

In VSF problem, the unavailability of the majority variables (i.e., $|\mathcal{N}| - |\mathcal{S}|$) would lead to the missing values and absent inter-dependencies between variables in $\mathcal{S}$ and $\mathcal{N} \backslash \mathcal{S}$ during the inference, which largely hinder their forecasting performances. For this problem, previous VSF work [Chauhan *et al.*, 2022] used fixed assumption of similarity and retrieved neighbors of available variables for imputation. However, it encounters constraints in completely recovering information that is missing from the unavailable subset, along with significant computational costs of measuring pair-wise similarity. In this paper, we reconsider the VSF problem from a novel density estimation perspective. We can fully recover the missing information by leveraging generative models to simultaneously reconstruct all the unavailable variables, without introducing additional computational cost.

Formally, given the fully observable training data, we try to estimate the unknown conditional density $p_{\mathcal{X}}(X^{\mathcal{N} \backslash \mathcal{S}} | X^{\mathcal{S}})$ of the missing variables $X^{\mathcal{N} \backslash \mathcal{S}}$ and available subset data $X^{\mathcal{S}}$, where $\mathcal{X}$ is the input space. Inspired by the human's logic of explaining intricate things via understandable objects ,we express the density of missing variables via Gaussian distribution $p_{\mathcal{Z}}$. As a result, we utilize a set of bijective functions f_θ to learn the dependencies of time series data and transform the original density into $p_{\mathcal{Z}}$, i.e., $f_\theta : \mathcal{X} \rightarrow \mathcal{Z}$. Hence, we can express the conditional density $p_{\mathcal{X}}(X^{\mathcal{N} \backslash \mathcal{S}} | X^{\mathcal{S}})$ by the change of variables formula [Dinh *et al.*, 2014]: $p_{\mathcal{X}}(X^{\mathcal{N} \backslash \mathcal{S}} | X^{\mathcal{S}}) = p_{\mathcal{Z}}(f_\theta(X^{\mathcal{N} \backslash \mathcal{S}}; X^{\mathcal{S}})) \left| \det \frac{\partial f_\theta}{\partial X^{\mathcal{N} \backslash \mathcal{S}}}(X^{\mathcal{N} \backslash \mathcal{S}}; X^{\mathcal{S}}) \right|$.

4.2 Time-series Reconstruction Flows

To estimate the intricate density of high dimensional data, there are a lot of generative networks such as GAN [Goodfellow *et al.*, 2014], VAE [Kingma and Welling, 2013]. However, GAN-based models suffer from model collapse and VAE-based model cannot estimate temporal density accurately by maximizing the lower bound of likelihood, so they are not suitable for time series data with complex temporal dependencies. Inspired by current normalizing flows [Dinh *et al.*, 2016; Papamakarios *et al.*, 2017], we propose ***Time-series Reconstruction Flows (TRF)***, a conditional flows-based generative model to estimate the density of missing variables of time series. Specifically, along the conditional generation formulation as proposed in Section 4.1, we consider observed variables as conditional subset embedding to reconstruct the missing variables, so that our generation framework can better access the information of available subset.

Algorithm 1 Variable-Agnostic Meta Learning

Input: $\mathcal{D}_{variables}$: Domain of the variables subsets, α, β: step size hyperparameters
1: Randomly initialize the parameter θ of the TRFlows f_θ
2: **while** not done **do**
3: Sample batch of variables subset $S_i \sim \mathcal{D}_{variables}$
4: **for all** S_i **do**
5: Evaluate $\nabla_\theta \mathcal{L}_{S_i}(f_\theta)$ with respect to N variables.
6: Compute adapted parameters with the gradient descent: $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}_{S_i}(f_\theta)$
7: **end for**
8: Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{S_i \sim \mathcal{D}_{variables}} \mathcal{L}_{S_i}(f_{\theta_i'})$
9: **end while**

Conditional Subset Embedding

To model the complex density of time series, the generative network needs to capture the temporal dependency among data. While directly modeling the dependencies of the whole long time series is unrealistic, we cooperate generation with the forecasting models, and design window based subset embedding for the variables. Formally, we utilize linear layers to obtain the embedding conditional vector $\mathbf{H}^S$ as: $\mathbf{H}^S = \mathrm{Projection}(\mathbf{X}_{t-L:t}^{(S)})$, where $H^S \in \mathbb{R}^{T \times S \times D}$ and $\mathbf{X}_{t-L:t}^{(S)}$ is lookback window of available subset $\mathcal{S}$.

Conditional TRF

We construct our window-based TRF consisting of conditional autoregressive layers and batch normalization [Ioffe and Szegedy, 2015]. Conditional autoregressive layers aim to estimate the distribution $p_\mathcal{X}$ of missing variables via a couple of invertible transformation functions. Formally, we input the horizon window of missing variables $\mathbf{X}_{t-L:t}^{(N-S)}$ and accessible variables $\mathbf{X}_{t-L:t}^{(S)}$, where $\mathbf{X}_{t-L:t}^{(N-S)} \in \mathcal{N} \backslash \mathcal{S}$, and $\mathbf{X}_{t-L:t}^{(S)} \in \mathcal{S}$. We can rewrite the expression the original density with horizon windows as:

$$p_\mathcal{X}(\mathbf{X}_{t-L:t}^{(N-S)} | \mathbf{X}_{t-L:t}^{(S)}) = p_\mathcal{Z}(f_\theta(\mathbf{X}_{t-L:t}^{(N-S)} | \mathbf{X}_{t-L:t}^{(S)}) \left| det \frac{\partial f_\theta}{\partial \mathbf{X}_{t-L:t}^{(N-S)}} (\mathbf{X}_{t-L:t}^{(N-S)}; \mathbf{X}_{t-L:t}^{(S)}) \right| \tag{2}$$

We can further express the conditional probability by the chain rule of probability as $p_\mathcal{X}(\mathbf{X}_{t-L:t}^{(N-S)} | \mathbf{X}_{t-L:t}^{(S)}, \theta) = \prod_i p_{\mathcal{X}_i}(\mathbf{X}_i^{(N-S)} | \mathbf{X}_{1:i-1}^{(N-S)}, \mathbf{X}_{t-L:t}^{(S)}, \theta)$. Due to the dynamic pattern of time series data, the transformation process is difficult and requires a couple of layers, i.e., $f_\theta = f_\theta^0 \circ f_\theta^1 \circ \cdots \circ f_\theta^N$. So for the forward calculation, we define the input of the first flows layer as $\mathbf{h}^0 := X_{t-L:t}^{(N-S)}$, and the output of the last layer as $\mathbf{h}^N$. We can utilize the following recursion to construct the n^{th} layer:

$$\mathbf{h}_i^{n+1} = \mathbf{h}_i^n \cdot exp(f_{\theta,s_i}^n(\mathbf{h}_{1:i-1}^n; \mathbf{H}^S)) + f_{\theta,b_i}^n(\mathbf{h}_{1:i-1}^n; \mathbf{H}^S) \tag{3}$$

where, f_{θ,s_i}^n and f_{θ,b_i}^n are the unconstrained functions to obtain scale and bias. The log determinant can be calculated as $log \left| det \frac{\partial f_\theta^n}{\partial \mathbf{h}^n}(\mathbf{h}^n; \mathbf{H}^S) \right| = -\sum_i f_{\theta,s_i}(\mathbf{h}_{1:i-1}^n; \mathbf{H}^S)$.

To implement the function f_{θ,s_i}^n and f_{θ,b_i}^n, we utilize the masking technique of MADE [Germain *et al.*, 2015]. For the n^{th}, the general form of masking matrix can be formulated as $M_{k',k}^{W^n} = 1_{m^n(k') \geq m^{n-1}(k)}$, where k and k' are the hidden unit index, and the values of $m^n(k)$ for each hidden layer $n \in \{1, \cdots, N\}$ are sampled uniformly. And for the last layer, the masking rule is $M_{k,d}^V = 1_{d > m^N(k)}$. With the help of MADE, we can estimate the density without the sequential loop, and accelerate the training process of TRFlows.

We utilize batch normalization to further improve the propagation of signal. With batch normalization, we can not only stabilize the training process, but also train a deeper stack of coupling layers to model more complex conditional distribution. The procedure of normalization of n^{th} layer can be written as $\mathbf{h}^n \mapsto \frac{\mathbf{h}^n - \widetilde{\mu}}{\sqrt{\widetilde{\sigma}^2 + \epsilon}}$. Here, $\widetilde{\mu}$ and $\widetilde{\sigma}^2$ are the estimated batch mean and variance. We can calculate the Jacobian determinant as $\left(\prod_i \left(\widetilde{\sigma}_i^2 + \epsilon\right)\right)^{-\frac{1}{2}}$. To update TRF, we can derive the negative log-likelihood as the loss function by applying chain rules to Equation 2:

$$\mathcal{L}(f_\theta) = -logp_\mathcal{Z}(f_\theta(\mathbf{X}_{t-L:t}^{(N-S)}; \mathbf{X}_{t-L:t}^{(S)}) - \sum_{n=0}^{N-1} log \left| det \frac{\partial f_\theta^n}{\partial \mathbf{h}^n}(\mathbf{h}^n; \mathbf{X}_{t-L:t}^{(S)}) \right| \tag{4}$$

Forecasting Process

In the inference stage of VSF, we randomly sample noisy vectors from Gaussian distribution and employ the inverse function of equation 3 to recursively reconstruct missing variables, which can be written as: $\mathbf{h}_i^n = exp(-f_{\theta,s_i}^n(\mathbf{h}_{1:i-1}^{n+1}; \mathbf{H}^S)) \cdot (\mathbf{h}_i^{n+1} - f_{\theta,b_i}^n(\mathbf{h}_{1:i-1}^{n+1}; \mathbf{H}^S))$. We denote the final reconstructed missing variables as $\hat{\mathbf{X}}_{t-L:t}^{(N-S)}$. We concatenate the available variables and reconstructed missing variables in the inference stage of the multivariate time series forecasting models for prediction, which can be formulated as: $\mathbf{X}_{t:t+H}^{(N)} = \mathcal{F}_\theta(\mathrm{CONCAT}(\mathbf{X}_{\mathrm{t-L:t}}^{(\mathrm{S})}, \hat{\mathbf{X}}_{\mathrm{t-L:t}}^{(\mathrm{N-S})}))$. It should be noted that our proposed method is applicable to any forecasting models that leverage inter-correlation.

4.3 Variable-Agnostic Meta Learning

Reconstructing missing variables set $\mathcal{N} \backslash \mathcal{S}$ is challenging since the subset $\mathcal{S}$ is randomly chosen, and training one specific generative model for every possible combination of variable subsets is unrealistic. As a result, we need to improve the generalization ability of TRFlows f_θ to make a precise estimation of all possible combination of missing variables. Inspired by the meta-learning technique [Finn *et al.*, 2017], we design ***Variable-Agnostic Meta Learning*** as the training framework to learn the common information and transfer the knowledge to new variable subset.

Formally, we define the domain of the variables subsets as $\mathcal{D}_{variables} = \{S_1, ..., S_z\}$, where z is the number of variable subsets, and each variable subset S_i contains N variables with different missing variables combinations. It should be noted that the selection of variable subset should be consistent with that of inference phase. We sample a batch of variable subsets S_i, and when adapting to different subsets, we update the parameters θ of flows f_θ to θ_i': $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}_{S_i}(f_\theta)$ where α is

Backbone		MTGNN				ASTGCN				T-GCN			
Setting		Partial		Reconstruct		Partial		Reconstruct		Partial		Reconstruct	
VSF Method		N/A		**TRF**(Ours)		N/A		**TRF**(Ours)		N/A		**TRF**(Ours)	
Metric		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
METR-LA	0.05	5.209	10.33	**3.843**	**7.837**	5.807	11.26	**5.185**	**10.13**	9.154	15.89	**8.546**	**14.52**
	0.15	4.597	9.391	**3.739**	**7.672**	5.593	10.66	**5.135**	**9.858**	8.885	15.16	**8.343**	**14.21**
	0.25	4.441	9.077	**3.685**	**7.567**	5.276	10.24	**5.101**	**9.752**	8.531	14.56	**8.092**	**13.90**
SOLAR	0.05	5.495	7.526	**3.997**	**5.927**	6.257	8.412	**4.382**	**6.254**	9.319	12.50	**5.533**	**7.750**
	0.15	4.286	6.146	**3.703**	**5.718**	5.786	8.058	**4.279**	**6.223**	8.909	12.25	**5.487**	**7.967**
	0.25	3.825	5.650	**3.540**	**5.480**	5.374	7.692	**4.205**	**6.168**	8.386	11.70	**5.460**	**8.038**
TRAFFIC	0.05	22.21	43.17	**11.58**	**28.05**	23.04	43.39	**18.84**	**39.03**	47.69	74.80	**38.30**	**59.55**
	0.15	18.60	38.12	**11.57**	**27.98**	22.43	43.09	**18.90**	**39.22**	45.53	70.95	**38.43**	**59.86**
	0.25	16.05	34.69	**11.52**	**27.93**	21.73	42.54	**18.86**	**39.24**	42.92	66.40	**38.39**	**59.86**
ECG5000	0.05	4.220	6.832	**3.596**	**6.344**	3.712	6.248	**3.477**	**5.903**	6.825	11.28	**6.298**	**9.977**
	0.15	4.082	6.592	**3.546**	**6.147**	3.604	6.053	**3.476**	**5.675**	6.704	10.68	**6.218**	**9.962**
	0.25	3.819	6.327	**3.457**	**6.039**	3.565	5.966	**3.473**	**5.921**	6.555	9.892	**6.148**	**9.770**

Table 1: Variable Subset Time-series Forecasting performance of backbones and TRF (Ours) under Partial and Reconstruct setting.

the learning rate of the inner loop optimization, and $\mathcal{L}_{S_i}(f_\theta)$ is the loss following equation 4. To train the model parameters, we optimize the performance of the adapted generation model $f_{\theta_i'}$ via diverse tasks sampled from $\mathcal{D}_{variables}$. We formulate the meta-objective as:

$$\min_\theta \sum_{S_i \sim \mathcal{D}_{variables}} \mathcal{L}_{S_i}(f_{\theta_i'}) = \sum_{S_i \sim \mathcal{D}_{variables}} \mathcal{L}_{S_i}(f_{\theta - \alpha \nabla_\theta \mathcal{L}_{S_i}(f_\theta)}) \tag{5}$$

We target to update the model parameters in such a manner that optimal performance can be achieved via minimum gradient steps on a new task. So we employ the stochastic gradient descent (SGD) and update the parameters θ by computing the gradient of $\mathcal{L}_{S_i}(f_{\theta_i'})$, which can be written as: $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{S_i \sim \mathcal{D}_{variables}} \mathcal{L}_{S_i}(f_{\theta_i'})$, where β is the meta learning rate of the outer loop optimization. The training process of the Meta-flows is shown in Algorithm 1.

5 Experimental Results

5.1 Experimental Setup

Dataset. We conducted our experiments on four real world datasets: (1) *METR-LA* [Li *et al.*, 2017] includes statistics on traffic speed gathered from 207 loop detectors on county highways during Mar 1st 2012 and Jun 30th 2012 in Los Angeles. (2) *SOLAR* contains the solar power output of the 137 plants in Alabama in 2007. (3) *TRAFFIC* describes road occupancy rates collected by 862 sensors in the San Francisco Bay area from 2015 to 2016. (4) *ECG5000* contains 140 electrocardiograms (ECG) with a length of 5000 each.

Multivariate Time Series Forecasting Model Backbone Setting. After missing variable reconstruction, we employed the widely-used models as backbones for our downstream multivariate time series forecasting task: (1) T-GCN [Zhao *et al.*, 2019]: utilizes graph convolutional network and gated recurrent unit to model spatial-temporal correlation. (2) ASTGCN [Guo *et al.*, 2019] exploits spatial and temporal attentions to model correlations among dynamic time series. (3) MTGNN [Wu *et al.*, 2020] utilizes a graph learning to describe uni-directed relations among variables.

To evaluate the average performance of our method, we randomly set the number of available variables $|\mathcal{S}|$ for 100 times, in order to indicate various missing percentages,. During training, all the time series backbone models were trained with all variables available in lookback windows and horizon windows. During inference (testing), we created three settings: 1) ***Partial***: partial variables were observed and available. We set missing variables to 0. This setting aims to show missing variables will lower time series forcasting performance. 2) ***Reconstruct***: Not just partial observed variables were available, but also the other missing variables were reconstructed by our method in both in lookback and horizon windows; in other words, our reconstruction method plays a role. This setting aims to show reconstruction can advance forecasting. By default, we adopted the Reconstruct setting during inference (testing), unless specified otherwise. 3) ***Oracle***: All variables were observed and there are no missing variables. Yet, we evaluate the forecasting performance only on the partially available variable subset ($\mathcal{S}$) in Setting 2). This setting aims show effective reconstruction of missing variables can approach the performances of all variables available. The lengths of horizon windows and lookback windows were set to 12.

Evaluation Metrics. Since the Oracle setting use data with fully observed and available variables, the performance of the Oracle setting can be seen as the upper bound of forecasting. Thereafter, we measure the performance difference between Reconstruct and the Oracle settings with respect to MAE and RMSE, which is formulated as $\Delta_{\text{MAE}} = \frac{E_{MAE}^{reconstruct} - E_{MAE}^{oracle}}{E_{MAE}^{oracle}} \times 100\%$, and $\Delta_{\text{RMSE}} = \frac{E_{RMSE}^{reconstruct} - E_{RMSE}^{oracle}}{E_{RMSE}^{oracle}} \times 100\%$, where, $E_{MAE}^{reconstruct}$ and $E_{RMSE}^{reconstruct}$ represent the average MAE and RMSE of the Reconstruct setting, and E_{MAE}^{oracle} and E_{RMSE}^{oracle} are the av-

Backbone		MTGNN				ASTGCN				T-GCN			
Setting		Reconstruct		Reconstruct		Reconstruct		Reconstruct		Reconstruct		Reconstruct	
VSF Method		**FDW**		**TRF**(Ours)		**FDW**		**TRF**(Ours)		**FDW**		**TRF**(Ours)	
Metric		Δ_{MAE}	Δ_{RMSE}	Δ_{MAE}	Δ_{RMSE}	Δ_{MAE}	Δ_{RMSE}	Δ_{MAE}	Δ_{RMSE}	Δ_{MAE}	Δ_{RMSE}	Δ_{MAE}	Δ_{RMSE}
METR-LA	0.05	9.16%	7.16%	**8.67%**	**6.82%**	2.17%	3.06%	**1.96%**	**2.72%**	7.02%	5.8%	**6.65%**	**5.31%**
	0.15	6.49%	4.68%	**6.08%**	**4.42%**	1.89%	2.72%	**1.65%**	**2.48%**	5.03%	5.43%	**4.74%**	**5.10%**
	0.25	4.97%	3.45%	**4.62%**	**3.29%**	1.53%	2.34%	**1.38%**	**2.11%**	3.83%	4.50%	**3.59%**	**4.22%**
SOLAR	0.05	39.1%	31.9%	**36.9%**	**30.1%**	11.5%	10.4%	**10.8%**	**9.78%**	9.43%	7.45%	**7.95%**	**6.23%**
	0.15	28.7%	24.4%	**26.2%**	**22.9%**	7.95%	7.29%	**7.48%**	**6.87%**	7.08%	4.62%	**6.32%**	**4.19%**
	0.25	22.8%	19.8%	**20.7%**	**17.2%**	6.15%	5.68%	**5.79%**	**5.24%**	5.70%	3.49%	**5.41%**	**3.28%**
TRAFFIC	0.05	2.83%	4.52%	**2.67%**	**4.27%**	0.23%	0.49%	**0.21%**	**0.46%**	0.65%	0.42%	**0.61%**	**0.38%**
	0.15	1.98%	2.79%	**1.86%**	**2.59%**	0.19%	0.45%	**0.17%**	**0.41%**	0.68%	0.42%	**0.65%**	**0.39%**
	0.25	1.76%	2.00%	**1.61%**	**1.86%**	0.16%	0.41%	**0.15%**	**0.38%**	0.64%	0.38%	**0.63%**	**0.36%**
ECG5000	0.05	5.24%	6.29%	**4.84%**	**5.92%**	0.28%	1.06%	**0.26%**	**0.99%**	1.79%	2.79%	**1.70%**	**2.62%**
	0.15	2.16%	3.14%	**2.04%**	**2.98%**	0.26%	1.03%	**0.23%**	**0.94%**	2.31%	3.11%	**2.18%**	**2.91%**
	0.25	1.17%	2.40%	**1.07%**	**2.27%**	0.23%	0.98%	**0.21%**	**0.93%**	1.66%	1.98%	**1.54%**	**1.85%**

Table 2: Performance comparisons with the state-of-the-art method in variable subset time-series forecasting under the Reconstruct setting (all of the metrics are measured by Δ_{MAE} and Δ_{RMSE}).

erage MAE and RMSE of the oracle setting.

5.2 Overall Performance

We evaluated the performance under the reconstruct setting (i.e., partial variables available with missing variable reconstruction) with respect to different available variable subset sizes ($|\mathcal{S}|$), i.e., 5%, 15%, 25% of total variables. Table 1 presents a comparative analysis of three time series forecasting backbone models under the partial setting (without reconstruction) and the reconstruct setting (with our TRF reconstruction). The results show that our TRF reconstruction of missing variables advance the three backbone forecasting models. In the Appendix D.1, Table 5 details the results under an oracle setting. Notably, when operating under a reconstruct setting with a subset of variables, there is a marked decrease in the performance of the backbone models. However, the integration of our reconstruction method leads to significant enhancements across all backbone models. For instance, in the TRAFFIC dataset, with only 5% variables available, integrating our TRF method can boost the performance of MTGNN by 47.86% (i.e., improving the Mean Absolute Error (MAE) from 22.21 to 11.58). Similarly, for the SOLAR dataset with 5% variables available, ASTGCN's performance is elevated by 29.97% (reducing the MAE from 6.257 to 4.382). These results support the effectiveness of our generative flow as missing variable reconstruction strategy.

5.3 Comparison with Variable Subset Time-Series Forecasting under the Reconstruct Setting

Under the reconstruct setting (i.e., reconstruct missing variables then forecast), we compared our method with the state-of-the-art VSF model, namely the Forecast Distance Weighting (FDW) ensemble method [Chauhan *et al.*, 2022] Table 2 shows that our method outperform FDW, achieving an average performance improvement of over 5%. One possible explanation is that our deep generative flow learning method is capable to model temporal patterns to estimate time series density distribution of missing variables for recovery. This feature is particularly significant as non-parametric methods often fail to effectively recover missing values.

Methods	METR-LA	SOLAR	TRAFFIC	ECG5000
TRF	**6.08 / 4.42**	**26.2** / 22.9	**1.86 / 2.59**	**2.04 / 2.98**
KNNE	8.21 / 9.68	31.7 / 27.4	2.07 / 2.62	2.24 / 3.05
MICE	32.3 / 22.7	34.1 / **19.1**	37.3 / 25.2	41.0 / 26.8
TRMF	72.4 / 54.7	57.2 / 25.4	174 / 92.1	37.5 / 22.8
US-GAN	23.9 / 25.2	36.0 / 24.4	20.0 / 24.1	5.32 / 8.18
SAITS	6.24 / 4.56	32.8 / 28.7	2.22 / 3.23	2.17 / 3.02
CSDI	6.67 / 4.94	28.5 / 24.0	2.38 / 3.15	2.42 / 3.53

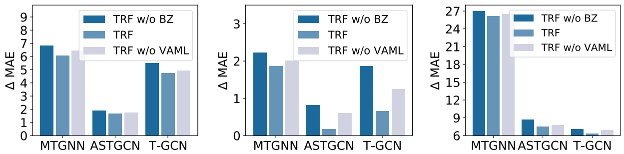
Table 3: Performance comparisons with Multivariate Time Series Imputation techniques on MTGNN across four datasets (measured by Δ_{MAE} / Δ_{RMSE}).

5.4 Comparison with Imputation Methods

Imputation can complete missing data as well. We compared our method with six popular imputation baseline methods, i.e., CSDI [Tashiro *et al.*, 2021], kNNE [Domeniconi and Yan, 2004], MICE [Van Buuren and Groothuis-Oudshoorn, 2011], TRMF [Yu *et al.*, 2016], SAITS [Du *et al.*, 2023] and US-GAN [Miao *et al.*, 2021] with respect to various forecasting backbone models under the reconstruct setting. Due to page limits, we only present the results for MTGNN while other results are detailed in the Appendix D.2. Table 3 shows that our framework can outperform all of the imputation baselines, except the RMSE of traffic dataset. This can be explained as imputation methods are insufficient in modeling inter-variable correlation under time series dynamics.

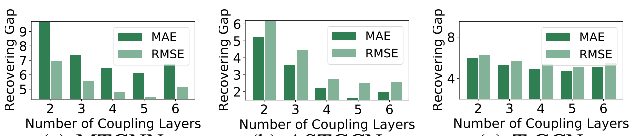
5.5 Ablation Study

We evaluated the effectiveness of each component of our method on four datasets. We denoted **TRF w/o VAML** as the variant of our method without Variables-Agnostic Meta Learning. **TRF w/o BZ** is another variant without batch nor-



(a) METR-LA. (b) TRAFFIC. (c) SOLAR.

Figure 3: Ablation study of different baselines on datasets (all performances are measured in Δ_{MAE}).



(a) MTGNN. (b) ASTGCN. (c) T-GCN.

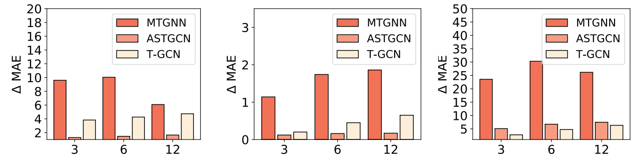
Figure 4: Hyperparameter analysis of the number of coupling layers on METR-LA dataset, where all performances are measured in Δ_{MAE} and Δ_{RMSE}.

malization. Figure 3 shows that removing different components reduces forecasting accuracy for diverse backbones. This illustrates that all components of our method works as one for estimating the unknown density of missing variables and reconstruct the missing relationship. Moreover, we observed that **TRF w/o VAML** generally perform better than **TRF w/o BZ**. A potential explanation is that stabilizing the training process is more crucial for recovering missing information compared to emphasizing generalization ability.

5.6 Model Analysis

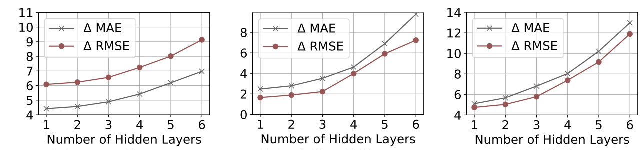
Horizon Analysis. We evaluated the influence of different horizon lengths on model performance. Figure 5 shows that when dealing with a smaller horizon window, the performance difference between the reconstruct setting and the oracle setting is smaller, indicating a better forecasting accuracy. On the contrary, we observed that under the oracle setting, forecasting accuracy is not influenced by the horizon window size. This is because, all variables are observed and available under the oracle setting.

Analysis of the Number of Coupling Layers. We evaluated the impact of the number of coupling layers. Figure 4 shows that the performance of our proposed framework exhibit limited correlation with the number of coupling layers under T-GCN. In contrast, the performance is sensitive to the number of coupling layers under ASTGCN and MTGNN. For instance, Δ_{MAE} and Δ_{RMSE} decrease when the number of coupling layers increases, while Δ_{MAE} and Δ_{RMSE} become larger when the number of coupling layers increases. This shows there exist an optimal coupling layer number. The potential reason for this trend is that increasing the number of coupling layers in a certain level can improve the performance of framework since flows without deep stack cannot model intricate density, but when the number of coupling layers gets large enough, it adds additional training challenges to impede model convergence.



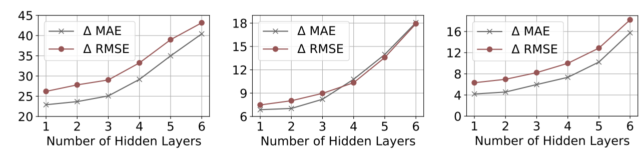
(a) METR-LA. (b) TRAFFIC. (c) SOLAR.

Figure 5: Horizon analysis of different baselines on datasets (all performances are measured in Δ_{MAE}).



(a) MTGNN. (b) ASTGCN. (c) T-GCN.

Figure 6: Hyperparameter analysis of number of hidden layers in each MADE on METR-LA dataset, where we measure performance in Δ_{MAE} and Δ_{RMSE}.



(a) MTGNN. (b) ASTGCN. (c) T-GCN.

Figure 7: Hyperparameter analysis of number of hidden layers in each MADE on SOLAR dataset (all performances are measured in Δ_{MAE} and Δ_{RMSE}).

Analysis of the Number of Hidden Layers. Figure 6 and Figure 7 show how the performance of our method changes over the number of hidden layers in each MADE. We observed that the lager the number of hidden layers is, the poorer the forecasting accuracy is. It is highly possible that more hidden layers will bring more training obstacles and more complex masking transformation for the TRF so that the performances of TRF are greatly hindered.

6 Conclusion

In this work, we reformulate the VSF problem from the perspective of density estimation. To assist the forecasting model in reconstructing the missing values and capturing inter-dependencies among variables, we first estimate the unknown density of inaccessible variable subset, and then reconstruct the missing variables by generation. Specifically, we propose Time-series Reconstruction Flows (TRF), a conditional flows-based generative framework consisting of conditional autoregressive layers based on subset embedding, to estimate the density and rebuild missing variables. To further improve the generalization ability, we design the Variables-Agnostic Meta Learning as the training framework such that TRF can adapt to estimate the density new variables subset more accurately. Finally, we conduct experiments on real-world datasets to indicate our proposed method can recover the most of the performance of forecasting models.

Acknowledgments

This research is funded by the Science and Technology Development Fund (FDCT), Macau SAR (file no. 0123/2023/RIA2, 001/2024/SKL), the Start-up Research Grant of University of Macau (File no. SRG2021-00017-IOTSC), and National Science Foundation (NSF) via the grant numbers: 2040950, 2006889, 2045567.

Contribution Statement

In this work, Xuanming Hu and Wei Fan contributed equally by leading the project, offering theoretical support, and handling the overall model design, code implementation, experimental design, and paper writing. Haifeng Chen, Pengyang Wang, and Yanjie Fu gave valuable feedback on the paper drafts. All authors reviewed and approved the manuscript.

References

[Bai *et al.*, 2018] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[Bai *et al.*, 2020] 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.

[Bartholomew, 1971] David J Bartholomew. Time series analysis forecasting and control., 1971.

[Byrne, 2019] Ruth MJ Byrne. Counterfactuals in explainable artificial intelligence (xai): Evidence from human reasoning. In *IJCAI*, pages 6276–6282, 2019.

[Chauhan *et al.*, 2022] Jatin Chauhan, Aravindan Raghuveer, Rishi Saket, Jay Nandy, and Balaraman Ravindran. Multi-variate time series forecasting on variable subsets. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 76–86, 2022.

[Dinh *et al.*, 2014] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[Dinh *et al.*, 2016] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[Domeniconi and Yan, 2004] Carlotta Domeniconi and Bojun Yan. Nearest neighbor ensemble. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 1, pages 228–231. IEEE, 2004.

[Du *et al.*, 2019] Shengdong Du, Tianrui Li, Yan Yang, and Shi-Jinn Horng. Deep air quality forecasting using hybrid deep learning framework. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2412–2424, 2019.

[Du *et al.*, 2023] Wenjie Du, David Côté, and Yan Liu. Saits: Self-attention-based imputation for time series. *Expert Systems with Applications*, 219:119619, 2023.

[Du Preez and Witt, 2003] Johann Du Preez and Stephen F Witt. Univariate versus multivariate time series forecasting: an application to international tourism demand. *International Journal of Forecasting*, 19(3):435–451, 2003.

[Fan *et al.*, 2022] Wei Fan, Shun Zheng, Xiaohan Yi, Wei Cao, Yanjie Fu, Jiang Bian, and Tie-Yan Liu. Depts: Deep expansion learning for periodic time series forecasting. *arXiv preprint arXiv:2203.07681*, 2022.

[Fan *et al.*, 2023] Wei Fan, Pengyang Wang, Dongkun Wang, Dongjie Wang, Yuanchun Zhou, and Yanjie Fu. Dish-ts: a general paradigm for alleviating distribution shift in time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 7522–7529, 2023.

[Finn *et al.*, 2017] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[Germain *et al.*, 2015] Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation. In *International conference on machine learning*, pages 881–889. PMLR, 2015.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[Guo *et al.*, 2019] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 922–929, 2019.

[Hamid and Wibowo, 2018] Nur Hamid and Wahyu Catur Wibowo. Wind speed forecasting using multivariate time-series radial basis function neural network. In *2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 423–428. IEEE, 2018.

[Hamzaçebi, 2008] Coşkun Hamzaçebi. Improving artificial neural networks' performance in seasonal time series forecasting. *Information Sciences*, 178(23):4550–4559, 2008.

[Hu *et al.*, 2023] Xuanming Hu, Wei Fan, Kun Yi, Pengfei Wang, Yuanbo Xu, Yanjie Fu, and Pengyang Wang. Boosting urban prediction via addressing spatial-temporal distribution shift. In *2023 IEEE International Conference on Data Mining (ICDM)*, pages 160–169. IEEE, 2023.

[Hu *et al.*, 2024] Xuanming Hu, Wei Fan, Dongjie Wang, Pengyang Wang, Yong Li, and Yanjie Fu. Dual-stage flows-based generative modeling for traceable urban planning. In *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*, pages 370–378. SIAM, 2024.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In

International conference on machine learning, pages 448–456. pmlr, 2015.

[Kingma and Dhariwal, 2018] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Li *et al.*, 2017] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

[Luo *et al.*, 2018] Yonghong Luo, Xiangrui Cai, Ying Zhang, Jun Xu, et al. Multivariate time series imputation with generative adversarial networks. *Advances in neural information processing systems*, 31, 2018.

[Miao *et al.*, 2021] Xiaoye Miao, Yangyang Wu, Jun Wang, Yunjun Gao, Xudong Mao, and Jianwei Yin. Generative semi-supervised learning for multivariate time series imputation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 8983–8991, 2021.

[Nichol and Schulman, 2018] Alex Nichol and John Schulman. Reptile: a scalable metalearning algorithm. *arXiv preprint arXiv:1803.02999*, 2(3):4, 2018.

[Oreshkin *et al.*, 2019] Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *arXiv preprint arXiv:1905.10437*, 2019.

[Papamakarios *et al.*, 2017] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. *Advances in neural information processing systems*, 30, 2017.

[Ren *et al.*, 2022] Weijieying Ren, Pengyang Wang, Xiaolin Li, Charles E Hughes, and Yanjie Fu. Semi-supervised drifted stream learning with short lookback. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1504–1513, 2022.

[Tang *et al.*, 2020] Xianfeng Tang, Huaxiu Yao, Yiwei Sun, Charu Aggarwal, Prasenjit Mitra, and Suhang Wang. Joint modeling of local and global temporal dynamics for multivariate time series forecasting with missing values. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5956–5963, 2020.

[Tashiro *et al.*, 2021] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in Neural Information Processing Systems*, 34:24804–24816, 2021.

[Van Buuren and Groothuis-Oudshoorn, 2011] Stef Van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45:1–67, 2011.

[Vanschoren, 2018] Joaquin Vanschoren. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*, 2018.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wu *et al.*, 2020] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 753–763, 2020.

[Yi *et al.*, 2024a] Kun Yi, Qi Zhang, Wei Fan, Hui He, Liang Hu, Pengyang Wang, Ning An, Longbing Cao, and Zhendong Niu. Fouriergnn: Rethinking multivariate time series forecasting from a pure graph perspective. *Advances in Neural Information Processing Systems*, 36, 2024.

[Yi *et al.*, 2024b] Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Ning An, Defu Lian, Longbing Cao, and Zhendong Niu. Frequency-domain mlps are more effective learners in time series forecasting. *Advances in Neural Information Processing Systems*, 36, 2024.

[Yick *et al.*, 2008] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[Yu *et al.*, 2016] Hsiang-Fu Yu, Nikhil Rao, and Inderjit S Dhillon. Temporal regularized matrix factorization for high-dimensional time series prediction. *Advances in neural information processing systems*, 29, 2016.

[Zeng *et al.*, 2023] 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, pages 11121–11128, 2023.

[Zhao *et al.*, 2019] Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE transactions on intelligent transportation systems*, 21(9):3848–3858, 2019.