pTSE: A Multi-model Ensemble Method for Probabilistic Time Series Forecasting

Yunyi Zhou[1] , **Zhixuan Chu**[1] , **Yijia Ruan**[1] , **Ge Jin**[1] , **Yuchen Huang**[1] , **Sheng Li**[2]
[1]Ant Group
[2]University of Virginia
{zhouyunyi.zyy, yijia.ryj, elvis.jg}@antgroup.com, chuzhixuan.czx@alibaba-inc.com,
hyc264276@antfin.com, shengli@virginia.edu

Abstract

Various probabilistic time series forecasting models have sprung up and shown remarkably good performance. However, the choice of model highly relies on the characteristics of the input time series and the fixed distribution that the model is based on. Due to the fact that the probability distributions cannot be averaged over different models straightforwardly, the current time series model ensemble methods cannot be directly applied to improve the robustness and accuracy of forecasting. To address this issue, we propose pTSE, a multi-model distribution ensemble method for probabilistic forecasting based on Hidden Markov Model (HMM). pTSE only takes off-the-shelf outputs from member models without requiring further information about each model. Besides, we provide a complete theoretical analysis of pTSE to prove that the empirical distribution of time series subject to an HMM will converge to the stationary distribution almost surely. Experiments on benchmarks show the superiority of pTSE over all member models and competitive ensemble methods.

1 Introduction

The common requirements of time series forecasting are not only predicting the expected value of a future target, namely point estimation but also further measuring the uncertainty of the output by predicting its probability distributions, namely probabilistic forecasting. Probabilistic forecasting methods have been extensively studied in the literature, such as deterministic methods that predict the quantiles of the predictive distribution [Lim *et al.*, 2021], probabilistic methods that sample future values from a learned approximate distribution [Salinas *et al.*, 2020; Rangapuram *et al.*, 2018; Salinas *et al.*, 2019a]), and latent generative models [Yuan and Kitani, 2019; Koochali *et al.*, 2021; Rasul *et al.*, 2020]. These methods are usually motivated by a particular modeling focus, characterizing certain aspects of the input time series [Januschowski *et al.*, 2020]. For instance, Prophet [Taylor and Letham, 2017] shows advantages in explicitly characterizing the fundamental time-domain components, i.e., trend and seasonality of time series, while the method in [Shih *et al.*, 2019; Chu and Li, 2023; Chu *et al.*, 2023] digests the frequency domain information for time series to overcome the nonstationary problem. On the other hand, the training objective of such probabilistic forecasting methods is usually maximizing a likelihood function that is conventionally assumed to be a fixed distribution, e.g., Gaussian. However, this is not always true for time series. According to [Ravagli, 2021], real world time series data is more likely to be asymmetric and multi-modal with a mixture of distributions. Both aforementioned facts bring challenges for single models. Therefore, this calls for a new way to integrate the advantages and specificity of diverse models.

To combine the advantages of different models, a popular and competitive solution is model ensemble [Oliveira and Torgo, 2015]. Generally, there are two main categories of ensemble techniques for time series. The first one learns an optimal linear combination of the predicted values returned by each member model [Akyuz *et al.*, 2017; Adhikari, 2015] by searching for the optimal weight of each model output. The second one trains the member models as weak predictors and then combines them via a boosting-like ensemble step, where member model information, including input features, is typically required for loss reduction [Qiu *et al.*, 2017; Liu *et al.*, 2019; Godahewa *et al.*, 2021; Qiu *et al.*, 2014]. However, to the best of our knowledge, all of the aforementioned studies need to treat all member models as point estimation models, where the ensemble step handles target values not distributions, and thus cannot be directly applied to fulfill probabilistic forecasting model ensemble. In addition to such practical shortcomings, an adapted theoretical foundation aimed at a probabilistic forecasting ensemble is also highly desired.

This paper intends to fill these gaps by designing *pTSE* ("probabilistic time series ensemble"), a semi-parametric method, to perform distribution ensemble for probabilistic forecasting of time series. We adopt the idea from the Hidden Markov Model (HMM) to treat the collection of member models as a hidden state space, where the distribution of each observation is determined by its hidden state. Then, we incorporate "mixture quantile estimation" (MQE) into the classic Baum-Welch algorithm to estimate the distribution of model residuals, which is subsequently used to compute a distribution quantile at the prediction stage. In order to guarantee the generality, we use weighted kernel density estimation

(WKDE), a non-parametric method, to approximate the residual distributions, where the sample weight of each residual for a member model is the probability of the corresponding model in the hidden state. A bootstrap method is specifically designed to calculate the optimal bandwidth parameter for each model, which is crucial to the WKDE performance. The whole ensemble step is achieved by inferring the stationary distribution of the HMM, of which the quantile is used for forecasting. We provide a complete theoretical analysis of pTSE to prove that the empirical distribution of time series subject to an HMM will converge to the stationary distribution almost surely, and such a result is non-trivial to guess. *That is to say, the whole time series approximately subjects to the stationary distribution (ensemble distribution) inferred by pTSE within any period of time.* It is worth noting that pTSE takes off-the-shelf model outputs without further requiring implementation details of the member model. This makes pTSE plug-and-play and easily integrated into existing models. We evaluate pTSE on synthetic datasets to confirm the theoretical results and then on public data sets to show its superiority over single-model methods as well as ensemble methods designed for point estimation models.

The main contributions of this paper are three folds: (1) We propose pTSE, a multi-model ensemble method for probabilistic forecasting, which only takes off-the-shelf outputs from member models without requiring further information about each model; (2) We theoretically verify the ensemble distribution discovered by our method, which the time series approximately subject to within any period of time; (3) We demonstrate on real-world data sets that our ensemble method produces better performance than all member models as well as competitive point estimation model ensemble methods.

2 The pTSE Framework

In this section, we begin with giving a brief overview of HMM as a preliminary in Section 2.1. Then the core idea of pTSE is introduced in Section 2.2. The distribution evaluation procedure with a mixture quantile estimation method and its parameter selection procedure are described in Section 2.3 and Section 2.4, respectively. Section 2.5 presents the complete procedure of pTSE, including the parameter estimation stage and the prediction stage, where forecasting is made based on the ensemble distribution.

2.1 Preliminaries

We first give a brief review of HMM and then introduce its fitting process.

HMM: An HMM is a probabilistic model describing the joint probability of a collection of random variables $\{O_1, \ldots, O_T, S_1, \ldots, S_T\}$[Bilmes and others, 1998]. The O_t variables, either continuous or discrete, represent the observations that we can acquire in the real world, while the S_t variables are hidden states corresponding to each O_t. Under an HMM, the random process $\{S_t : t \in \mathbb{N}\}$ is a Markov process satisfying

$$p(S_{t+1}|S_1, \ldots, S_t) = p(S_{t+1}|S_t).$$

Given the state S_t, O_t satisfies

$$p(O_t|S_1, \ldots, S_T, O_1, \ldots, O_T) = p(O_t|S_t), \qquad (1)$$

which demonstrates that the distribution of O_t is only determined by the hidden state.

In other words, for continuous variables, Equation (1) defines the Probability Density Function (PDF) of O_t given S_t, and the PDF denoted as $f_k(o) := p(O_t = o|S_t = k)$, is conventionally termed the "emission function".

Fitting an HMM: Fitting an HMM with a total number of K states to a dataset $\{O_t : t \in \mathbb{N}, 1 \le t \le T\}$ requires determining the following parameters: (1) transition matrix $\boldsymbol{A} = (a_{i,j})_{1 \le i,j \le K}$ of the underlying Markov process $\{S_t\}$, where $a_{ij} = p(S_{t+1} = j|S_t = i)$, (2) a parameter set $\Theta = \{\theta_k\}_{k=1}^{K}$ of the emission function $f_k(o; \theta_k)$, and (3) initial distribution $\pi = (\pi_1, \ldots, \pi_K)$, where $\pi_k = p(S_0 = k)$.

The fitting is done by maximum likelihood estimation (MLE) or equivalently

$$\operatorname*{argmax}_{\boldsymbol{A}, \pi, \Theta} p(\{O_t\}_{t=1}^{T}|\boldsymbol{A}, \pi, f_k(O_t; \theta_k \in \Theta)). \qquad (2)$$

However, for a dataset governed by an HMM, the undergoing state process $\{S_t : S_t \in \mathbb{N}, 1 \le S_t \le K\}$ is unknown in most cases. Therefore, the parameter estimation problem of an HMM is generally solved via an Expectation-Maximization (EM) fashion, which particularly deals with MLE problems with missing data, such as the hidden states. This EM method for fitting HMM is known as the Baum-Welch algorithm [Bilmes and others, 1998].

2.2 Framework Basics

We now introduce the pTSE framework for probabilistic forecasting. A probabilistic forecasting problem usually requires to estimate the conditional distribution of y_t, given a trained model M and a feature vector X_t, where X_t may contain the history of the time series along with known future information without randomness. In other words, probabilistic forecasting aims to estimate $p(y_t|M(X_t))$.

Suppose a total number of K probabilistic forecasting models, $\{M_k\}_{k=1}^{K}$, are independently fitted to the same data set $\{y_t\}_{t=1}^{T}(T \in \mathbb{N})$. We first assume that at each time t, there exists an optimal model $M_{k_t} \in \{M_k\}_{k=1}^{K}$ such that the distribution of y_t is determined by the optimal model given the feature X_t. Or equivalently, $y_t \sim p(y_t|M_{k_t}(X_t))$.

Next, for y_{t+1}, we assume that M_{k_t} will randomly transfer to a new optimal model $M_{k_{t+1}}$ with a probability of $p_{k_t,k_{t+1}}$ (bold arrow in Figure 1), and this transition process is a **Markov process**. We illustrate this idea in Figure 1. In order to derive an executable methodology, we denote the PDF $p(y_t|M_{k_t}(X_t))$ as $f_{M_{k_t}}^{X_t}(y_t)$, for convenience.

Recall that we intend to use HMM for the model ensemble, we clarify the concepts as follows: the optimal model M_{k_t} corresponds to the hidden state S_t, the time series y_t corresponds to the observation O_t, and the PDF $f_{M_k}^{X_t}(y_t)$ corresponds to the emission function $f_k(o)$, as introduced in Section 2.1. Hence by now, we have established a framework using **HMM** to capture the relation between probabilistic forecasting models and the target time series.

We now introduce the method for evaluating the ensemble weights for each member model M_k. As illustrated by the

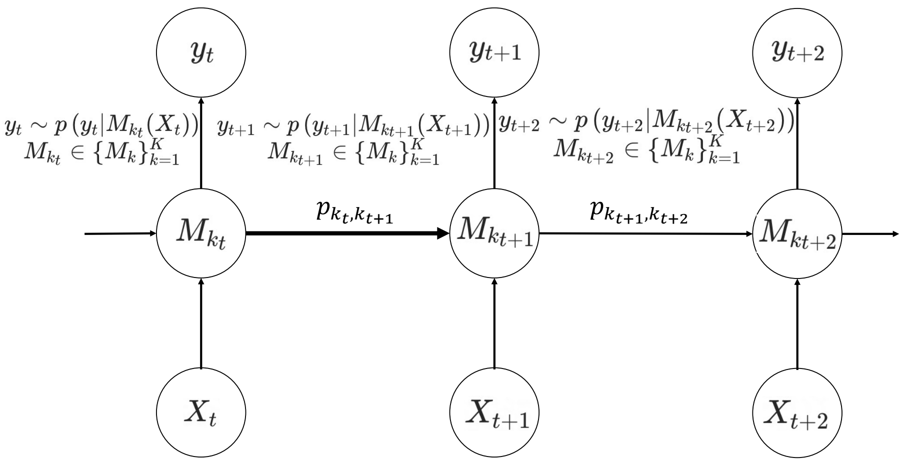
y_t
y_{t+1}
y_{t+2}
$y_t \sim p\left(y_t|M_{k_t}(X_t)\right)$
$M_{k_t} \in \{M_k\}_{k=1}^K$
$y_{t+1} \sim p\left(y_{t+1}|M_{k_{t+1}}(X_{t+1})\right)$
$M_{k_{t+1}} \in \{M_k\}_{k=1}^K$
$y_{t+2} \sim p\left(y_{t+2}|M_{k_{t+2}}(X_{t+2})\right)$
$M_{k_{t+2}} \in \{M_k\}_{k=1}^K$
$p_{k_t,k_{t+1}}$
$p_{k_{t+1},k_{t+2}}$
M_{k_t}
M_{kt+1}
M_{kt+2}
X_t
X_{t+1}
X_{t+2}

Figure 1: Markov Property of the Optimal Model Transition Process. At each time t, we assume there exists an optimal model $M_{kt} \in \{M_k\}_{k=1}^K$ such that the distribution of y_t is determined by $M_{kt}(X_t)$, where $\{M_k\}_{k=1}^K$ is the set of all K models fitted to the time series $\{y_t\}_{t=1}^T$ and X_t is the feature obtained at t for predicting y_t. The optimal model at time t will transfer to a new optimal model M_{kt+1} at time $t+1$ with a probability of $p_{k_t,k_{t+1}}$ (black bold arrow), and we assume this random transition process to be Markov.

law of total probability that

$$p(y_t) = \sum_{k=1}^{K} p(y_t|M_k(X_t))p(M_k(X_t)),$$

the time series subjects to an ensemble distribution of the component PDF defined by each $M_k(X_t)$ with the weights defined as the probability of each M_k being the optimal model, i.e., $p(M_k(X_t))$, where X_t is a known vector without randomness. For simplicity, we estimate $p(M_k(X_t))$ as the average chance of M_k being the optimal model within any period of time, denoted as π_k^*, and the ensemble distribution is estimated as

$$\widehat{f}(y_t) = \sum_{k=1}^{K} \pi_k^* f_{M_k}^{X_t}(y_t). \tag{3}$$

We illustrate this idea in Figure 2.

Surprisingly, $(\pi_1^*, \ldots, \pi_K^*)$ is nothing but the stationary distribution of the hidden Markov process, which determines the expected frequency of reaching each state. For a Markov process with transition matrix $\boldsymbol{A}$, the stationary distribution is a row vector, $\pi^* = (\pi_1^*, \ldots, \pi_K^*)$, satisfying $\pi^* \boldsymbol{A} = \pi^*$, whose elements are non-negative and sum up to 1. In other words, a key step of our ensemble method is to acquire the stationary distribution of the hidden Markov process, which requires fitting an HMM to the time series $\{y_t\}_{t=1}^T$ given $\{M_k\}_{k=1}^K$ and $\{X_t\}_{t=1}^T$. By substituting the variables of our interest into Equation (2), the **ultimate problem** we need to solve is defined as

$$\underset{\boldsymbol{A},\pi,f_{M_k}^{X_t}}{\operatorname{argmax}}\, p\left(\{y_t\}_{t=1}^T \middle| \boldsymbol{A}, \pi, f_{M_k}^{X_t}\right). \tag{4}$$

Although it may seem to be slightly arbitrary to use the stationary distribution to evaluate the ensemble distribution, we have theoretically proved that the time series $\{y_t : t \in \mathbb{N}\}$ approximately subject to this ensemble distribution within any period of time in Section 3.

2.3 Mixture Quantile Estimation

Generally, for a probabilistic forecasting method, the PDF $f_{M_k}^{X_t}(y_t)$ is not directly evaluated in the prediction stage; instead, it tends to estimate a quantile of y_t, where the $q-$th quantile of y_t is a constant τ satisfying $p(y_t \leq \tau) = q$. Therefore, we exploit an MQE framework [Wu and Yao, 2016] to estimate Equation (3).

The MQE framework first formalizes a q-th quantile forecasting model M as $y = M(X) + \epsilon_q$, where X is the vector of model inputs, and the error term ϵ_q is a random variable whose $q-$th quantile is equal to zero. Let $f_{\epsilon_q}(\cdot)$ be the PDF of ϵ_q and the PDF of y given $M(X)$ is $f_M(y) = f_{\epsilon_q}(y - M(X))$. For multiple models, the MQE framework is used to estimate the error term PDF for each model and ensure the $q-$th quantile of ϵ_q is zero for each model.

Each y_t is then a random variable whose PDF is defined as $f_{M_k}^{X_t}(y_t) = f_{\epsilon_q}^k\left(y_t - M_k(X_t)\right)$. The function $f_{\epsilon_q}^k(\cdot)$ is the error term PDF of the member model M_k and can be estimated by the MQE framework.

2.4 Kernel Density Estimation

To perform MQE, we incorporate WKDE, a non-parametric method for PDF estimation. The WKDE is used specifically to estimate the PDF of error term $f_{\epsilon_q}^k(\cdot)$. A WKDE implies the importance of each sample is unequal, which in our case, is due to the diverse probabilities of different member models being the optimal model for a certain y_t. The relation between sample weights and member models will be further discussed in Section 2.5.

A WKDE obtained from a dataset $\{y_t\}_{t=1}^T$ is defined as $\hat{f}(y) = \frac{1}{\sigma \sum_{t=1}^T w_t} \sum_{t=1}^T w_t K\left(\frac{y-y_t}{\sigma}\right), w_t > 0$, where the bandwidth parameter σ essentially influences the performance. We select the optimal σ by a bootstrap method [Faraway and Jhun, 1990]. For a set of candidate bandwidth parameters $\Sigma = \{\sigma_i\}$, the bootstrap method first chooses an

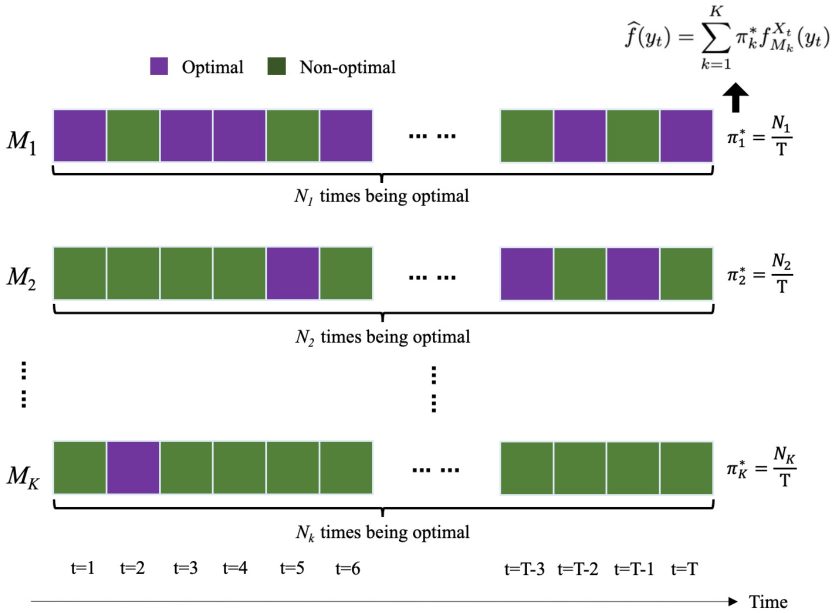
$\hat{f}(y_t) = \sum_{k=1}^{K} \pi_k^* f_{M_k}^{X_t}(y_t)$
Optimal
Non-optimal
M_1
$\pi_1^* = \frac{N_1}{T}$
N_1 times being optimal
M_2
$\pi_2^* = \frac{N_2}{T}$
N_2 times being optimal
M_K
$\pi_K^* = \frac{N_K}{T}$
N_k times being optimal
t=1
t=2
t=3
t=4
t=5
t=6
t=T-3
t=T-2
t=T-1
t=T
Time

Figure 2: Ensemble Distribution. The ensemble PDF is defined as the weighted average of the PDF defined by each member model, where the weights correspond to the average chance of each member model being the optimal model.

initial value σ_0. Next, it constructs a total number of B sample sets by resampling from the distribution determined by a WKDE with σ_0. For each $\sigma_i \in \Sigma$, a WKDE is performed on each resampled sample set, resulting in a PDF $\hat{f}^b_{\sigma_i}(\cdot)$ (corresponding to the bth sample set). The selected σ^* is the one that minimizes the bootstrap integrated mean squared error (BIMSE), or technically

$$\sigma^* = \text{argmin}_{\sigma_i \in \Sigma} \frac{1}{B} \sum_{b=1}^{B} \int \left(\hat{f}^b_{\sigma_i}(o) - \hat{f}_{\sigma_0}(o) \right)^2 \mathrm{d}o. \tag{5}$$

In this work, we adopt the Gaussian kernel for performing WKDE. An estimated PDF $f^k_{\epsilon_q}(\cdot)$ with a selected σ_k is denoted as $f^k_{\epsilon_q}(\cdot;\sigma_k)$ for explicitness.

2.5 The MQE Baum-Welch Algorithm

We now derive the MQE Baum-Welch Algorithm for inferring both π^* and $f^{X_t}_{M_k}(y_t)$ in equation 3.

Recall the problem defined in Equation 4 in Section 2.1, by MQE which refines $f^{X_t}_{M_k}(y_t)$ as $f^k_{\epsilon_q}(y_t - M_k(X_t);\sigma_k)$, the objective function for the MQE Baum-Welch Algorithm is

$$\underset{\boldsymbol{A},\pi,\{\sigma_k\}_{k=1}^{K}}{\text{argmax}} \; p\left(\{y_t\}_{t=1}^{T} \middle| \boldsymbol{A}, \pi, f^k_{\epsilon_q}(\cdot;\sigma_k) \right). \tag{6}$$

The MQE Baum-Welch Algorithm is still an EM method for resolving Equation 6.

During the **E step**, four quantities $\alpha_k(t)$, $\beta_k(t)$, $\gamma_k(t)$ and $\xi_{i,j}(t)$, need to be prepared, whose definition will be given in the following content. The **E step** uses the same update equations in the general Baum-Welch Algorithm introduced in [Bilmes and others, 1998]. Based on the **E step**, the **M step** is designed to update the transition matrix $\boldsymbol{A}$, the initial distribution π, and the PDF $f^k_{\epsilon_q}(\epsilon)$ for each member model, where the MQE method and the bootstrap procedure are performed. The **EM steps** are repeated until all parameters converge.

The **stationary distribution** π^* in Equation (3) is obtained by setting $\pi^* = \pi$ and repeating $\pi^* = \pi^* \boldsymbol{A}$ until π^* converges, where π is the estimated initial distribution. Once π^* and $f^k_{\epsilon_q}(\epsilon)$ are determined, the ensemble PDF of y_{T+h} provided with the output of each member model $M_k(X_{T+h})$, i.e. $\hat{f}(y_{T+h};\pi^*, f^k_{\epsilon_q}, M_k(X_{T+h}))$, is defined as

$$\hat{f}(y_{T+h};\pi^*, f^k_{\epsilon_q}, M_k(X_{T+h})) = \sum_{k=1}^{K} \pi^*_k f^k_{\epsilon_q}(y_{T+h} - M_k(X_{T+h})). \tag{7}$$

MQE Baum-Welch Algorithm

E step

Updating $\alpha_k(t)$, $\beta_k(t)$, $\gamma_k(t)$ and $\xi_{i,j}(t)$: For $1 \leq i, j \leq K$, define $\alpha_k(t)$ and $\beta_k(t)$ as

$$\alpha_k(1) = \pi_k f^k_{\epsilon_q}(y_1 - M_k(X_1)), \; k = 1, \ldots, K, \tag{8}$$

$$\alpha_j(t+1) = \left(\sum_{i=1}^{K} \alpha_i(t) a_{ij} \right) f^j_{\epsilon_q}(y_{t+1} - M_j(X_{t+1});\sigma_k), \tag{9}$$

and

$$\beta_k(T) = 1, \; k = 1, \ldots, K, \tag{10}$$

$$\beta_j(t) = \left(\sum_{i=1}^{K} a_{ij} f^j_{\epsilon_q}(y_{t+1} - M_j(X_{t+1});\sigma_k) \right) \beta_j(t+1), \tag{11}$$

where $\alpha_k(t)$, $\beta_k(t)$ are termed forward probability and backward probability respectively.

Based on $\alpha_k(t)$ and $\beta_k(t)$, $\gamma_k(t)$ and $\xi_{i,j}(t)$ are defined as:

$$\gamma_k(t) = \frac{\alpha_k(t)\beta_k(t)}{\sum_j \alpha_j(t)\beta_j(t)}), \tag{12}$$

which is the probability of being in state k at time t, and

$$\xi_{i,j}(t) = \frac{\gamma_i(t) a_{i,j} f^j_{\epsilon_q}(y_{t+1} - M_j(X_{t+1}); \sigma_k)\beta_j(t+1)}{\beta_i(t)}, \tag{13}$$

which is the probability of being in state i at time t and in state j at time $t+1$.

M step

Updating $\boldsymbol{A}$ and $\boldsymbol{\pi}$: Each element a_{ij} of the transition matrix $\boldsymbol{A}$ is estimated as the expected number of transitions from state i to state j relative to the expected total number of transitions away from i [Bilmes and others, 1998], which is

$$a_{i,j} = \frac{\sum_{t=1}^{T_1} \xi_{i,j}(t)}{\sum_{t=1}^{T_1} \gamma_i(t)}. \tag{14}$$

For the initial distribution $\pi = (\pi_1, \ldots, \pi_K)$, the k-th element is estimated as the expected relative frequency in the k-th state at time 1, or

$$\pi_k = \gamma_k(1). \tag{15}$$

In summary, Equations (8)-(15) aim at updating the transition matrix $\boldsymbol{A}$ and the initial distribution π, with $f^k_{\epsilon_q}(\epsilon)$ fixed.

Selecting $\sigma_\mathbf{k}$ for $\mathbf{f}^\mathbf{k}_{\epsilon_\mathbf{q}}(\epsilon)$: Our MQE method subsequently updates the emission function or the PDF $f^k_{\epsilon_q}(\epsilon)$. It first selects a bandwidth parameter σ_k for each $f^k_{\epsilon_q}(\epsilon)$ via the bootstrap method. In the bootstrap procedure, $\gamma_k(t)$ is used as the sample weight of each ϵ^t_k, where $\epsilon^t_k = y_t - M_k(X_t)$. We use $\gamma_k(t)$ as sample weight because it is the probability of M_k being the hidden state of y_t [Bilmes and others, 1998]. Therefore, for each member model, samples with larger probability should be paid more attention to.

Updating $\mathbf{f}^\mathbf{k}_{\epsilon_\mathbf{q}}(\epsilon)$ by MQE: Based on σ_k, a simple weighted-KDE-like update equation for $f^k_{\epsilon_q}(\epsilon)$ would be $\sum_{t=1}^{T} \frac{\gamma_k(t)}{\sigma_k \sum_t \gamma_k(t)} K\left(\frac{\epsilon - \epsilon^t_k}{\sigma_k}\right)$. However, this update equation ignores which quantile is being estimated by the member models. To make the method focus on the q-th quantile, $f^k_{\epsilon_q}(\epsilon)$ is updated as

$$f^k_{\epsilon_q}(\epsilon) = \sum_{t=1}^{T} \sum_{l=1}^{2} \mathbf{I}^k_{t,l} W^k_l \gamma_k(t) \frac{1}{\sigma_k} K\left(\frac{\epsilon - \epsilon^t_k}{\sigma_k}\right),$$

where $\mathbf{I}^k_{t,1} = \mathbb{I}_{\{\epsilon^t_k \le 0\}}$[1] and $\mathbf{I}^k_{t,2} = \mathbb{I}_{\{\epsilon^t_k > 0\}}$ [Wu and Yao, 2016]. The constants W^k_1 and W^k_2 constrain $f^k_{\epsilon_q}(\epsilon)$ to have 0 as the $q-$th quantile, while normalizing the integral of $f^k_{\epsilon_q}(\epsilon)$ equal to 1. By defining $v_{k,t} = \int_{-\infty}^{0} \frac{1}{\sigma_k} K\left(\frac{\epsilon - \epsilon^t_k}{\sigma_k}\right) \mathrm{d}\epsilon$, W^k_1 and W^k_2 are acquired by solving the following linear equation systems,

$$\begin{cases} \sum_{t=1}^{T} \sum_{l=1}^{2} \mathbf{I}^k_{t,l} W^k_l \gamma_k(t) = 1, \\ \sum_{t=1}^{T} \sum_{l=1}^{2} \mathbf{I}^k_{t,l} v_{k,t} W^k_l \gamma_k(t) = q. \end{cases}$$

Prediction Step

Once the learning procedure stops, the $q-$th quantile τ at time $T+h$ is obtained by solving Equation (16)

$$\int_{-\infty}^{\tau} \sum_k \pi^*_k f^k_{\epsilon_q}(y - M_k(X_{T+h}))\, \mathrm{d}y = q \tag{16}$$

Therefore, the future q-th quantile for a time series is estimated from ensemble PDF with Equation (16) and can be directly used at the prediction stage.

3 Theoretical Analysis

In this section, we present the theoretical results of our work. The core idea is to evaluate the limit of the empirical distribution of a sample set originated by an HMM. The results are non-trivial because a random process $\{O_t\}_{t=0}^{+\infty}$ generated from an HMM is not necessarily a Markov process. Strictly speaking, Equation (17) is not always equal to Equation (18) for a given initial distribution π, where $\mathbf{F}(o) = (f_1(o), \ldots, f_K(o))^T$ and $f_k(o)$ is the emission function as introduced in 2.1.

$$p(O_{t+1} \le \tau_{t+1} | O_0 \le \tau_0, \ldots, O_t \le \tau_t) \tag{17}$$
$$= \frac{\pi \boldsymbol{A} \int_{-\infty}^{\tau_0} \mathrm{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o \cdots \boldsymbol{A} \int_{-\infty}^{\tau_{t+1}} \mathrm{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o\mathbf{1}}{\pi \boldsymbol{A} \int_{-\infty}^{\tau_0} \mathrm{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o \cdots \boldsymbol{A} \int_{-\infty}^{\tau_t} \mathrm{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o\boldsymbol{A}\mathbf{1}}.$$

$$p(O_{t+1} \le \tau_{t+1} | O_t \le \tau_t) \tag{18}$$
$$= \frac{\pi \boldsymbol{A}^{t-1} \int_{-\infty}^{\tau_t} \mathrm{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o\boldsymbol{A} \int_{-\infty}^{\tau_{t+1}} \mathrm{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o\mathbf{1}}{\pi \boldsymbol{A}^{t-1} \int_{-\infty}^{\tau_t} \mathrm{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o\boldsymbol{A}\mathbf{1}}.$$

We summarize our theoretical work in four lemmas and one theorem. Lemma 3.1 is a straightforward statement of the property of a transition matrix. Both Lemma 3.2 and Lemma 3.3 illustrate convergence. Lemma 3.2 computes the final limit with the help of Lemma 3.1. Lemma 3.3 assures the existence of the limit of the sum of a random variable sequence, providing fast dependence decay. Lemma 3.4, indicating the exponential convergence rate of a Markov Process, turns out to be a guarantee for the condition in Lemma 3.3. Remarkably, based on Lemma 3.1-3.4, Theorem 3.1 indicates that the empirical distribution of a dataset sampled from an HMM converges to $\sum_{k=1}^{K} \pi^*_k f_k(o)$ almost surely. In other words, these HMM samples are approximately subject to an ensemble distribution of the emission functions, $f_k(o)$, with the weights determined by the stationary distribution regardless of the time window. In the case of pTSE, each emission function $f_k(y_t)$ is defined as $f^k_{\epsilon_q}(y_t - M_k(X_t))$. Hence as a corollary, within any period of time, the time series $\{y_t : t \in \mathbb{N}\}$ approximately subjects to the ensemble distribution in Equation (7), estimated by the MQE Baum-Welch Algorithm.

[1] The indicator function $\mathbb{I}_{\{\cdot\}}$ is a 0/1 valued function, which is defined as $\mathbb{I}_{\{c\}} = 1$ if c is true, and 0 otherwise.

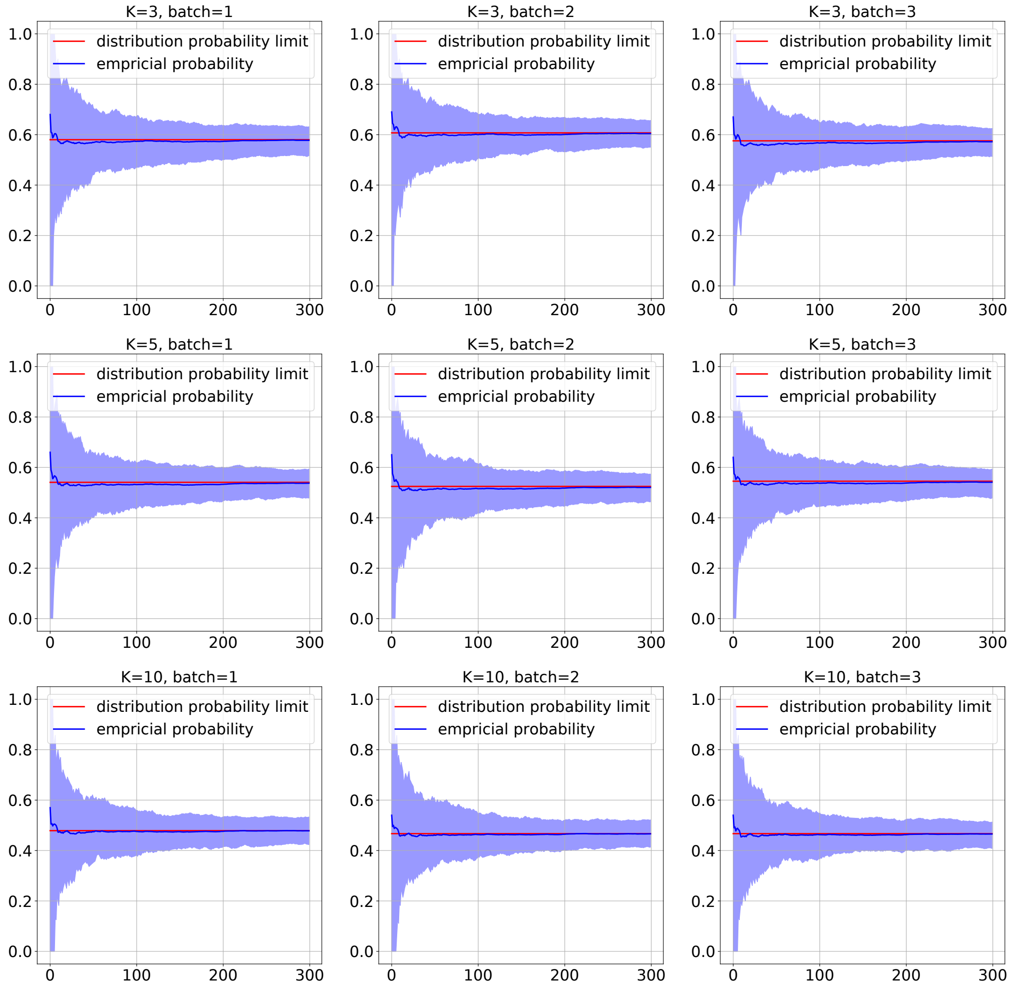

Figure 3: Average empirical probability (blue line) across 100 simulations vs. theoretical limit (red line) when $\tau = 0.5$. The blue shadow represents the 95% confidence interval. Three batches of a 100-time-simulation dataset are presented for each K.

Lemma 3.1. *Let A be the transition probability matrix of an HMM, and let*

$$\mathbf{1} = (1, 1, \ldots, 1)^{\mathrm{T}},$$

then $\mathbf{1}$ *is a right eigenvector of eigenvalue 1 of A, i.e.*

$$\boldsymbol{A}\mathbf{1} = \mathbf{1}$$

The proof is straightforward.

Lemma 3.2. *Let A be the transition probability matrix of an HMM, π^* be the stationary distribution, and π be any initial distribution, let $\mathbf{F}(o)$ be the vector-valued emission function of all states, then*

$$\lim_{T\to+\infty} \frac{1}{T}\sum_{t=0}^{T-1} \pi\boldsymbol{A}^t \int_{-\infty}^{\tau} \operatorname{diag}\left(\mathbf{F}(o)\right) \mathrm{d}o \boldsymbol{A}^{T-1-t}\mathbf{1} = \pi^* \int_{-\infty}^{\tau} \mathbf{F}(o)\mathrm{d}o \tag{19}$$

Lemma 3.3. *Let $\{E_n : n \geq 1\}$ be a sequence of events and $S_n = \sum_{k=1}^{n} \mathbb{I}_{E_k}$, if there exists $\{\rho_n\}_{n=1}^{\infty}$ satisfying $\sum_n |\rho_n| \leq +\infty$, such that, for any $i \neq j$,*

$$p(E_i \cap E_j) - p(E_i)p(E_j) \leq \rho_{|i-j|}\sqrt{p(E_i)p(E_j)},$$

and if $\lim_{n\to+\infty} \mathbb{E}(S_n) = +\infty$,*then,*

$$\lim_{n\to+\infty} \frac{S_n}{\mathbb{E}(S_n)} = 1, a.s.$$

Lemma 3.4. *Let A be the transition probability matrix of a Markov Process with non-zero elements and π^* be the stationary distribution. Then, there exists a constant C, such that the following inequality holds,*

$$||\boldsymbol{A}^t - \mathbf{1}\pi^*||_F \leq Ct^{J-1}\lambda_*^{t-J+1},$$

(**A**)	Traffic			Electric			Solar Energy		
Model	0.5-risk	0.9-risk	AVG	0.5-risk	0.9-risk	AVG	0.5-risk	0.9-risk	AVG
DeepAR	0.151	0.302	0.227	0.083	0.070	0.076	**0.435**	0.253	0.344
SFF	0.235	0.471	0.353	0.084	0.047	0.066	0.509	0.278	0.393
TFT	0.184	0.367	0.275	0.118	0.062	0.090	0.487	**0.184**	0.336
Transformer	0.163	0.326	0.244	0.087	0.058	0.072	0.499	0.270	0.385
pTSE (Ours)	**0.150**	**0.106**	**0.128**	**0.079**	**0.042**	**0.061**	0.451	0.210	**0.331**

(**B**)	Traffic			Electric			Solar Energy		
Model	0.5-risk	0.9-risk	AVG	0.5-risk	0.9-risk	AVG	0.5-risk	0.9-risk	AVG
ModelRank	**0.150**	0.108	0.129	**0.073**	0.049	**0.061**	0.457	**0.210**	0.334
FFORMA	0.165	0.112	0.139	0.081	0.070	0.076	0.459	0.284	0.372
pTSE (Ours)	**0.150**	**0.106**	**0.128**	0.079	**0.042**	**0.061**	**0.451**	**0.210**	**0.331**

Table 1: The experimental results of our model (pTSE), member models, and another two ensemble models on three real datasets. The best results are marked in bold (lower is better).

where J is the size of the largest Jordan block of $\boldsymbol{A}$, and λ_*^{t-J+1} is the largest absolute value of the eigenvalues smaller than 1 of $\boldsymbol{A}$. $||\cdot||_F$ is the Frobenius Norm.

Theorem 3.1. *Let $\{O_t\}_{t=1}^T$ be the observations generated from an HMM whose transition matrix has non-zero elements, π^* be the stationary distribution, and $\mathbf{F}(o)$ be the vector-valued emission function of all states. Define $\hat{F}_T(\tau)$, the empirical distribution of $\{O_t\}_{t=1}^T$, as*

$$\hat{F}_T(\tau) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}_{\{O_t<=\tau\}}$$

then,

$$\lim_{T\to+\infty} \hat{F}_T(\tau) = \pi^* \int_{-\infty}^{\tau} \mathbf{F}(o)\mathrm{d}o,\ a.s.$$

4 Numerical Experiments

In this section, we first present a few synthetic data experiments to verify our theoretical results. Next, we apply the pTSE to publicly available datasets to test the performance.

4.1 Synthetic Data Analysis

We simulated random sequences governed by HMM structures. We set $K = 3,\ 5,\ 10$ and $T = 1000$. The transition matrix $\boldsymbol{A}$ is chosen by first generating a matrix of uniformly distributed random numbers and then normalizing the matrix to ensure the sum of elements of each row equals 1. The emission function $f_k(o)$ for each state k is simply set to a Gaussian distribution as $\mathcal{N}(0.2k, \sqrt{k}+1)$, $(k = 1, \ldots, K)$. For each set of $\{K, T, \boldsymbol{A}, \mathbf{F}(o)\}$, we run the simulation procedure for 100 times, where during each time, an initial distribution π^0 is randomly chosen. The results are presented in Figure 3. The empirical probability, $\hat{F}(\tau)$, shows fast convergence to $\pi^* \int_{-\infty}^{\tau} \mathbf{F}(o)\mathrm{d}o$, after $T = 50$ for all simulated datasets, regardless of the state number K or the transition matrix $\boldsymbol{A}$.

4.2 Real World Data Analysis

We evaluate the performance of pTSE on three challenging real-world benchmark datasets, i.e., solar energy [2], electricity, traffic [Yu *et al.*, 2016].The solar energy, electricity, and traffic datasets contain hourly measurements from 137, 370, and 963 time series, respectively. For these three datasets, each model would perform an iterative prediction task of forecasting the future values for a 24-hour horizon after being trained on the past 168-hour (past week) data. The model performance would be evaluated on a 7-day-horizon test set.

Four of the most popular probabilistic forecasting models are selected as the member models: SimpleFeedForwardEstimator (SFF), Transformer, DeepAR, and TemporalFusionTransformer (TFT). As in [Salinas *et al.*, 2019b], we use q-risk metrics (quantile loss) to quantify the accuracy of a q-th quantile of the predictive distribution. Table 1 presents 0.5-risk, 0.9-risk, and the average risk of the output corresponding to each method. We also present a comparison of pTSE with existing time series model ensemble methods, FFORMA [Montero-Manso *et al.*, 2020], a meta-learning approach, and a model ranking-based ensemble method [Adhikari *et al.*, 2015]. Hyperparameters of the two ensemble methods are set up as recommended in the original papers. Results in Table 1 show that pTSE outperforms all member models on average across all datasets. The performance of pTSE is relatively more stable in the electricity and traffic datasets, as it ranked highest in all three competitions in each case. On the solar energy dataset, pTSE maintained the best performance in the average loss. Compared with the other two ensemble methods, pTSE shows a significant advantage on three datasets.

5 Conclusion

We present pTSE, a semi-parametric multi-model ensemble methodology for probabilistic forecasting. Our method takes off-the-shelf model outputs without requiring further information. We theoretically prove the empirical distribution of time series subject to an HMM will converge to the stationary distribution almost surely. We use the synthetic data to verify the validity of our theory and conduct extensive experiments on three benchmark datasets to demonstrate the superiority of pTSE over each member model and other model ensemble methods. Nevertheless, it should be pointed out that the improvement by pTSE or any other ensemble methods is essentially limited by the performance of member models.

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

References

[Adhikari *et al.*, 2015] Ratnadip Adhikari, Ghanshyam Verma, and Ina Khandelwal. A model ranking based selective ensemble approach for time series forecasting. *Procedia Computer Science*, 48:14–21, 2015. International Conference on Computer, Communication and Convergence (ICCC 2015).

[Adhikari, 2015] Ratnadip Adhikari. A neural network based linear ensemble framework for time series forecasting. *Neurocomputing*, 157:231–242, 2015.

[Akyuz *et al.*, 2017] A. Okay Akyuz, Mitat Uysal, Berna Atak Bulbul, and M. Ozan Uysal. Ensemble approach for time series analysis in demand forecasting: Ensemble learning. In *2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, pages 7–12, 2017.

[Bilmes and others, 1998] Jeff A Bilmes et al. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510):126, 1998.

[Chu and Li, 2023] Zhixuan Chu and Sheng Li. Continual treatment effect estimation: Challenges and opportunities. *arXiv preprint arXiv:2301.01026*, 2023.

[Chu *et al.*, 2023] Zhixuan Chu, Ruopeng Li, Stephen Rathbun, and Sheng Li. Continual causal inference with incremental observational data. *arXiv preprint arXiv:2303.01775*, 2023.

[Faraway and Jhun, 1990] Julian J Faraway and Myoungshic Jhun. Bootstrap choice of bandwidth for density estimation. *Journal of the American Statistical Association*, 85(412):1119–1122, 1990.

[Godahewa *et al.*, 2021] Rakshitha Godahewa, Kasun Bandara, Geoffrey I Webb, Slawek Smyl, and Christoph Bergmeir. Ensembles of localised models for time series forecasting. *Knowledge-Based Systems*, 233:107518, 2021.

[Januschowski *et al.*, 2020] Tim Januschowski, Jan Gasthaus, Yuyang Wang, David Salinas, Valentin Flunkert, Michael Bohlke-Schneider, and Laurent Callot. Criteria for classifying forecasting methods. *International Journal of Forecasting*, 36(1):167–177, 2020. M4 Competition.

[Koochali *et al.*, 2021] Alireza Koochali, Andreas Dengel, and Sheraz Ahmed. If you like it, gan it—probabilistic multivariate times series forecast with gan. In *Engineering Proceedings*, volume 5, page 40. Multidisciplinary Digital Publishing Institute, 2021.

[Lim *et al.*, 2021] Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 2021.

[Liu *et al.*, 2019] Fagui Liu, Muqing Cai, Liangming Wang, and Yunsheng Lu. An ensemble model based on adaptive noise reducer and over-fitting prevention lstm for multivariate time series forecasting. *IEEE Access*, 7:26102–26115, 2019.

[Montero-Manso *et al.*, 2020] Pablo Montero-Manso, George Athanasopoulos, Rob J. Hyndman, and Thiyanga S. Talagala. Fforma: Feature-based forecast model averaging. *International Journal of Forecasting*, 36(1):86–92, 2020. M4 Competition.

[Oliveira and Torgo, 2015] Mariana Oliveira and Luis Torgo. Ensembles for time series forecasting. In Dinh Phung and Hang Li, editors, *Proceedings of the Sixth Asian Conference on Machine Learning*, volume 39 of *Proceedings of Machine Learning Research*, pages 360–370, Nha Trang City, Vietnam, 26–28 Nov 2015. PMLR.

[Qiu *et al.*, 2014] Xueheng Qiu, Le Zhang, Ye Ren, Ponnuthurai N Suganthan, and Gehan Amaratunga. Ensemble deep learning for regression and time series forecasting. In *2014 IEEE symposium on computational intelligence in ensemble learning (CIEL)*, pages 1–6. IEEE, 2014.

[Qiu *et al.*, 2017] Xueheng Qiu, Ye Ren, Ponnuthurai Nagaratnam Suganthan, and Gehan A.J. Amaratunga. Empirical mode decomposition based ensemble deep learning for load demand time series forecasting. *Applied Soft Computing*, 54:246–255, 2017.

[Rangapuram *et al.*, 2018] 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*, 31:7785–7794, 2018.

[Rasul *et al.*, 2020] 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.

[Ravagli, 2021] Davide Ravagli. *Mixture autoregressive models with applications to heteroskedastic time series.* The University of Manchester (United Kingdom), 2021.

[Salinas *et al.*, 2019a] David Salinas, Michael Bohlke-Schneider, Laurent Callot, Roberto Medico, and Jan Gasthaus. High-dimensional multivariate forecasting with low-rank gaussian copula processes. *arXiv preprint arXiv:1910.03002*, 2019.

[Salinas *et al.*, 2019b] David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 2019.

[Salinas *et al.*, 2020] 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.

[Shih *et al.*, 2019] Shun-Yao Shih, Fan-Keng Sun, and Hung-yi Lee. Temporal pattern attention for multivariate time series forecasting. *Machine Learning*, 108(8):1421–1441, 2019.

[Taylor and Letham, 2017] Sean J. Taylor and Benjamin Letham. Forecasting at scale. *PeerJ Prepr.*, 5:e3190, 2017.

[Wu and Yao, 2016] Qiang Wu and Weixin Yao. Mixtures of quantile regressions. *Computational Statistics & Data Analysis*, 93:162–176, 2016.

[Yu *et al.*, 2016] Hsiang-Fu Yu, Nikhil Rao, and Inderjit S Dhillon. Temporal regularized matrix factorization for high-dimensional time series prediction. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[Yuan and Kitani, 2019] Ye Yuan and Kris Kitani. Diverse trajectory forecasting with determinantal point processes. *arXiv preprint arXiv:1907.04967*, 2019.