

# A variable order hidden Markov model with dependence jumps

Anastasios Petropoulos, Sotirios P. Chatzis, and Stelios Xanthopoulos



**Abstract**—Hidden Markov models (HMMs) are a popular approach for modeling sequential data, typically based on the assumption of a first- or moderate-order Markov chain. However, in many real-world scenarios the modeled data entail temporal dynamics the patterns of which change over time. In this paper, we address this problem by proposing a novel HMM formulation, treating temporal dependencies as latent variables over which inference is performed. Specifically, we introduce a hierarchical graphical model comprising two hidden layers: on the first layer, we postulate a chain of latent observation-emitting states, the temporal dependencies between which may change over time; on the second layer, we postulate a latent first-order Markov chain modeling the evolution of temporal dynamics (dependence jumps) pertaining to the first-layer latent process. As a result of this construction, our method allows for effectively modeling non-homogeneous observed data, where the patterns of the entailed temporal dynamics may change over time. We devise efficient training and inference algorithms for our model, following the expectation-maximization paradigm. We demonstrate the efficacy and usefulness of our approach considering several real-world datasets. As we show, our model allows for increased modeling and predictive performance compared to the state-of-the-art in the considered scenarios, for competitive computational complexity.

**Index Terms**—Temporal dynamics, hidden Markov models, expectation-maximization, variable order, dependence jumps.

## 1 INTRODUCTION

Modeling sequential data continues to be a fundamental task and a key challenge in the field of machine learning, encountered in a plethora of real-world applications, including bioinformatics, document analysis, financial engineering, speech processing, and computer vision, to name just a few. In this paper, we focus on the problem of *sequence prediction*, dealing with *continuous*, possibly *high-dimensional* observations (time-series). Machine learning literature comprises a rather extensive corpus of proposed prediction algorithms for sequences of continuous observations. Among them, the hidden Markov model (HMM) is one of the most popular methods, used in a great variety of application contexts. This popularity is mainly due to the fact that HMMs are flexible enough to allow for modeling complex temporal patterns and structures in sequential data. Specifically, HMMs are popular for their provision of a convenient way of modeling observations appearing in a sequential manner and tending to cluster or to alternate between different possible components (subpopulations) [1].

Most popular HMM formulations are based on the postulation of first-order Markovian dependencies; in other

words, only one-step-back temporal dynamics are considered. Such an assumption allows for increased simplicity and low computational complexity of the resulting model training and inference algorithms. However, postulating first-order temporal dynamics does also entail ignoring the possibility of the modeled data comprising longer temporal dynamics. Even though this assumption might be valid in some cases, it is well-known to be unrealistic in several application scenarios, including handwriting recognition, molecular biology, speech recognition, and volatility prediction in financial return series, thus undermining the modeling effectiveness.

To resolve this problem, several researchers have attempted to introduce HMM-type models with higher-order dependencies. Characteristic examples are the methods presented in [2] and [3], with successful applications to the problem of speech recognition, the method presented in [4], applied to handwriting recognition, the method of [5], designed to address challenges related to pattern recognition tasks in molecular biology, and the method presented in [6], which was successfully applied to the field of robotics. However, a major drawback of such higher-order HMM approaches is their considerably increased computational costs, which become rather prohibitive as model order increases. An effort to ameliorate these issues of higher-order HMMs is presented in [7]. In that work, instead of directly training  $R$ -th order HMMs on the data, a method of fast incremental training is used that progressively trains HMMs from first to  $R$ -th order.

Note, though, that using higher-order HMMs gives rise to a source of significant burden for researchers and practitioners, namely the need to determine the most appropriate order for the postulated models. This procedure entails fitting multiple models to the available data to choose from, and application of some cross-validation procedure, which, apart from computationally cumbersome, is also likely to become prone to overfitting [8]. Finally, another limitation of the existing higher-order HMM formulations concerns their static and homogeneous assumptions, i.e. their consideration that the temporal dynamics order in the modeled data does not change over time. Indeed, sequential data with variable order in the entailed temporal dynamics are quite often encountered in real-world application scenarios [9], [10], [11], [12]. Therefore, allowing for capturing more complex structure of temporal dynamics in the modeled data, where effective model order may change over time

as a result of dynamic switching between different temporal patterns, is expected to result in much better modeling and predictive performances.

To address these problems of conventional higher-order HMMs, some researchers have proposed appropriate models with variable order Markovian dynamics assumptions. For instance, a variable order Markov model is presented in [10] to address the problem of prediction of discrete sequences over a finite alphabet; the method is successfully applied to three different domains, namely English text, music pieces, and proteins (amino-acid sequences). More recently, [11] presented a simple and effective generalization of variable order Markov models to full online Bayesian estimation. Generalization of variable order Markov models in this context enables perpetual model improvement and enrichment of the learned temporal patterns by accumulation of observed data, without any need for human intervention. Despite these merits, a drawback of both these approaches concerns their inability to model sequential data comprising continuous observations, i.e. sequences each frame of which is a (probably high-dimensional)  $D$ -dimensional vector of real values, defined in  $\mathbb{R}^D$ . Finally, [13] propose a two-stage modeling approach towards variable order HMMs: the first stage consists in discovering repetitive temporal patterns of variable length, while the second stage consists in performing prediction by means of a separate simple HMM fit to the temporal pattern determined to be relevant at each specific time point. Similar to the previous approaches, a major limitation of [13] consists in its incapability to model sequential observations taking *continuous* values in  $\mathbb{R}^D$ .

In a different vein, a maximum-margin classifier for sequential data with (theoretically) infinitely-long temporal dependencies is presented in [9]. That paper devises a novel margin-maximizing model with convex objective function that allows for capturing arbitrarily-long temporal dependencies in sequential datasets. This is effected by utilizing a recently proposed nonparametric Bayesian model of label sequences with infinitely-long temporal dependencies, namely the *sequence memoizer* (SM) [14]. Training and inference for this model can be efficiently performed by employing a versatile mean-field-like approximation [15], [16]; this approximation allows for increased computational efficiency, almost comparable to analogous (large-margin) first-order HMM formulations, e.g. [17]. Further, since design of the model of [9] is limited to classification tasks, a generalization of this model allowing to also perform sequence prediction was recently presented in [12].

As discussed in [9] and [12], the proposed models, postulating infinitely-long temporal dependencies, perform *inference* over temporal dependence patterns. In other words, they do *not* try to determine the most appropriate model setup for a considered dataset. Instead, they essentially learn a posterior distribution over all possible temporal dependence patterns. During prediction, these models effectively perform marginalization over all possible temporal dependence modeling assumptions, with each assumption being given a different probability at each time point. As a result, the methods of [9] and [12] do not suffer from issues regarding appropriate model selection, namely the need to fit multiple models and perform cross-validation, and the associated overfitting proneness. In addition, they

inherently allow for handling the case where the form of temporal dependencies changes over time. Note also that both the methods presented in [9] and [12] can model discrete as well as continuous observations, contrary to previous approaches which can handle only discrete observations.

Despite these merits, two major limitations of these approaches are:

- (i) The need to come up with a (rather brute-force) approximation to allow for deriving efficient model training and inference algorithms, namely the mean-field approximation. Indeed, although mean-field approximation does not affect the nature of the model, which takes into account infinitely-long histories of latent temporal states, it results in omitting the fluctuations of higher-order temporal states when performing training and inference [9], [12]. This procedure leads to suboptimal training and inference results, that do not exploit the full modeling capacity of the methods, and with *no* theoretical convergence guarantees.
- (ii) The need to perform inference for the employed model of (arbitrarily-long) temporal dynamics (state-transitions), i.e. a postulated sequence memoizer [18]. SM is a nonparametric Bayesian method recently proposed for modeling sequential data with discrete values and dependencies over infinitely-long time-windows. While effective in modeling sequential data with long temporal dynamics, inference for this model suffers from high computational costs, which increase with the length of the modeled sequences. As such, obviating the need of using an SM in the modeling pipeline is expected to significantly reduce computational costs.

In this paper, we address all the aforementioned shortcomings of the current state-of-the-art, by introducing an HMM variant capable of capturing *jumps* in the temporal *dependence patterns* of modeled sequential data. Specifically, we introduce a hierarchical graphical model comprising two hidden layers: on the *first layer*, we postulate a *chain of latent observation-emitting states*, the *dependencies* between which may *change over time*; on the *second layer*, we postulate a *latent first-order Markov chain* modeling the *evolution* of temporal dynamics (*dependence jumps*) pertaining to the first-layer latent process. As a result of this construction, our model allows for effectively modeling non-homogeneous observed data, where the patterns of temporal dependencies may change over time. To allow for tractable training and inference procedures, our model considers *temporal dependencies* taking the form of *variable order dependence jumps*, the order of which is *inferred* from the data as part of the model inference procedure.

Our method is designed to allow for modeling *both* discrete and continuous observations; it allows for capturing seasonal effects in the modeled sequences, and enhances modeling in the implied autocorrelation structure of the observed sequences. In addition, contrary to the related methods of [9] and [12], our method does *not* require utilization of any kind of approximation to perform model training and inference. Indeed, both model training and inference can be performed *exactly* and in a computationally efficient way, using elegant algorithms derived under the expectation-maximization paradigm [19]. We demonstrate the efficacy of our approach in the task of sequential data prediction, considering real-world application scenarios.

The remainder of this paper is organized as follows: In Section 2, we introduce our proposed model and derive its training and inference algorithms. In Section 3, we experimentally evaluate our approach, and exhibit its advantages over existing approaches. Finally, in Section 4 we conclude this paper, summarizing and discussing our results.

## 2 PROPOSED APPROACH

### 2.1 Motivation

In real-world applications, it is often the case that stochastic processes are characterized by non-homogeneous evolution, exhibiting higher-order dependencies. For example, time series of financial asset returns are known to exhibit variable autocorrelation and non-stationarity [20]; such forms of dynamics in the modeled data cannot be sufficiently captured by using a simple Markov process. In the same vein, historical volatility of financial asset returns usually exhibits long temporal interdependencies, slow autocorrelation decay, fat distribution tails, as well as temporal pattern switching over time, e.g. shifting between low volatility and high volatility regimes [21], [22], [23], [24], which are manifested as jumps driven by shocks or unexpected news [25], [26].

Several studies have examined whether conventional HMM formulations are capable of capturing such stylized facts in modeled time-series. For example, [27] examined the efficacy of simple first-order HMMs; further, [28] used hidden semi-Markov models (HSMMs) as an alternative solution allowing for better capturing the autocorrelation structure. However, the outcome of all these studies has been quite unsatisfactory compared to the state-of-the-art in the literature pertaining to the related applications, e.g. the literature on financial return series modeling. Motivated from these results, in this work we aim to come up with an elegant and computationally efficient HMM variant capable of accommodating the above-mentioned stylized facts in observed time-series, namely: (i) distributions with fat tails; (ii) seasonality and temporal clustering dynamics; and (iii) non-homogeneous temporal dynamics patterns, exhibiting dependence jumps over time.

### 2.2 Model Definition

As we have already discussed, in this work we are seeking to devise an HMM variant allowing for modeling sequential data with *variable temporal dependence patterns*, i.e. a model capable of determining *dependence jumps* in the chain of observation-emitting latent states. For this purpose, we postulate an HMM variant, the hierarchical construction of which comprises *two hidden layers*: The *first layer* essentially consists of the chain of *observation-emitting* latent states, the dependencies between which may *change form* over time. The *second layer* comprises a latent *first-order Markov chain* that determines (and generates) the *dependence jumps* taking place in the observation-emitting latent chain of the first layer.

Let us postulate  $N$  observation-emitting states on the chain of the first layer of our model, where the hidden emission density of each state is modeled by a  $M$ -component finite mixture model. Let us also postulate a latent first-order Markov chain comprising  $K$  states on the second

layer;  $K$  is essentially the number of alternative temporal dependence patterns considered on the first layer of the model. Even though multiple alternative configurations could be considered for the form of the modeled temporal dependence patterns of the first-layer observation-emitting chain, in this work we limit ourselves to pairwise latent emitting state transitions between the *current* emitting state and *some previous state that occurred at a time point a number of steps back*; this number of steps back is determined from the latent values generated from the *second-layer dependence jumps-generating Markov chain* of our model.

Let us introduce here some useful notation. We denote as  $O = \{\mathbf{o}_t\}_{t=1}^T$  an observed data sequence, with  $\mathbf{o}_t \in \mathbb{R}^D$ . The latent (unobserved) data associated with this sequence comprise: (i) the corresponding *emitting state* sequence  $Q = \{q_t\}_{t=1}^T$ , where  $q_t = 1, \dots, N$  is the indicator of the state the  $t$ th observation is emitted from; (ii) the sequence of *temporal dependence form* indicators  $Z = \{z_t\}_{t=1}^T$  that indicate the pairwise emitting states transition that is relevant (“active”) at time  $t$ , where  $z_t = 1, \dots, K$ ; and (iii) the sequence of the corresponding *mixture component* indicators  $L = \{l_t\}_{t=1}^T$ , where  $l_t = 1, \dots, M$  indicates the mixture component density that generated the  $t$ th observation. A graphical illustration of the generative model and the latent interdependencies assumptions of our model is provided in Fig. 1.

The above-described VDJ-HMM model comprises the set of parameters  $\Theta = \{\Phi, \Psi\}$ , where  $\Phi$  denotes the parameters set of the emission distributions of the model, and  $\Psi$  denotes the set of parameters of the postulated latent processes pertaining to the observed data dynamics (first-layer process) and the dependence jump dynamics (second-layer process). Specifically, since the second-layer process is a simple first-order Markov chain, it comprises the parameters

$$\hat{\omega}_k \triangleq p(z_1 = k) \quad (1)$$

that denote the (prior) probabilities of the initial state of this Markov chain, and the parameters

$$\hat{\pi}_{kk'} = p(z_t = k' | z_{t-1} = k) \quad \forall t > 1 \quad (2)$$

denoting the transition (prior) probabilities of this Markov chain. From the above model definition, we observe that, if the transition probability  $\hat{\pi}_{11}$  in the above-defined transition matrix  $\hat{\Pi} \triangleq [\pi_{kk'}]_{k,k'}$  is close to one, then the observation-emitting process of our model (first model layer) almost reduces to a first-order Markov chain. In this paper, for simplicity we set  $\hat{\omega}_k = \frac{1}{K} \quad \forall k$  and  $\hat{\pi}_{kk'} = \frac{1}{K} \quad \forall k, k'$ ; in other words, we consider all dependence forms a priori of equal probability. These assumptions, although relatively limiting, allow for deriving tractable and computationally efficient model training and inference algorithms, as we show further on.

In a similar fashion, the postulated first-layer process of our model comprises the parameters

$$\varpi_i \triangleq p(q_1 = i) \quad (3)$$

denoting the (prior) probabilities of the initial observation-emitting state, with  $\varpi \triangleq [\varpi_i]_{i=1}^N$ . In addition, turning to the variable-form temporal dynamics of this process, we also

introduce the set of *dependence form-conditional* transition (prior) probability matrices  $\{\mathbf{\Pi}^k\}_{k=1}^K$ , with

$$\mathbf{\Pi}^k \triangleq [\pi_{ij}^k]_{i,j=1}^N \quad (4)$$

where

$$\begin{aligned} \pi_{ij}^k &\triangleq p(q_t = j | q_{t-1}, \dots, q_{t-k} = i; z_t = k) \\ &= p(q_t = j | q_{t-k} = i; z_t = k) \end{aligned} \quad (5)$$

In other words, we consider different (pairwise) state transition probabilities, depending on the inferred dependence form  $k$  (number of steps back) generated from the postulated second-layer process.

Having defined the latent processes of our model, with effective parameters set  $\Psi = \{\varpi, \{\mathbf{\Pi}^k\}_{k=1}^K\}$ , we can now proceed to the definition of the (conditional on the first-layer states) emission distributions of our model. For this purpose, and in order to allow for effective modeling of continuous-valued observations, we postulate  $M$ -component finite mixture models, as we have already discussed. Specifically, in our work, *to allow for modeling distributions with fat tails*, we consider two alternative selections: (i) multivariate Gaussian mixture models, yielding

$$p(\mathbf{o}_t | q_t = i) = \sum_{m=1}^M w_{im} \mathcal{N}(\mathbf{o}_t | \boldsymbol{\mu}_{im}, \boldsymbol{\Sigma}_{im}) \quad (6)$$

where  $\mathcal{N}(\cdot | \boldsymbol{\mu}, \boldsymbol{\Sigma})$  is a multivariate Gaussian with mean  $\boldsymbol{\mu}$  and covariance matrix  $\boldsymbol{\Sigma}$ ; and (ii) multivariate Student's- $t$  mixture models, yielding

$$p(\mathbf{o}_t | q_t = i) = \sum_{m=1}^M w_{im} \mathcal{S}(\mathbf{o}_t | \boldsymbol{\mu}_{im}, \boldsymbol{\Sigma}_{im}, \nu_{im}) \quad (7)$$

where  $\mathcal{S}(\cdot | \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu)$  is a multivariate Student's- $t$  distribution with parameters  $\boldsymbol{\mu}$ ,  $\boldsymbol{\Sigma}$ , and  $\nu$  degrees of freedom. On this basis, the parameters set  $\Phi$  yields  $\Phi = \{w_{im}, \boldsymbol{\mu}_{im}, \boldsymbol{\Sigma}_{im}\}_{i,m}$  or  $\Phi = \{w_{im}, \boldsymbol{\mu}_{im}, \boldsymbol{\Sigma}_{im}, \nu_{im}\}_{i,m}$ , respectively. As discussed in [29], HMM-type models with Student's- $t$  mixture emission distributions allow for better modeling sequential data stemming from populations with long tails, which are quite common in real-world application scenarios.

This concludes the definition of our model. We dub our approach the variable dependence jump HMM (VDJ-HMM). From Eqs. (1)-(7), the joint distribution of VDJ-HMM yields:

$$\begin{aligned} p(O, Q, Z | \Theta) &= \hat{\omega}_{z_1} \varpi_{s_1} \prod_{t=1}^{T-1} \hat{\pi}_{z_t, z_{t+1}} \prod_{t>1} \pi_{s_t - z_t, s_t}^{z_t} \\ &\times \prod_{t=1}^T p(\mathbf{o}_t | q_t = i) \end{aligned} \quad (8)$$

Note that, as observed from (8), a major advantage from the computational point of view of the proposed VDJ-HMM model compared to higher-order HMM formulations (e.g., [2], [6], [7]) is the much fewer number of parameters postulated from VDJ-HMM. As a result, VDJ-HMM is capable of capturing seasonal effects in the modeled data while allowing for significantly more efficient training and inference algorithms compared to existing alternatives. In addition, the lower number of trainable parameters reduces

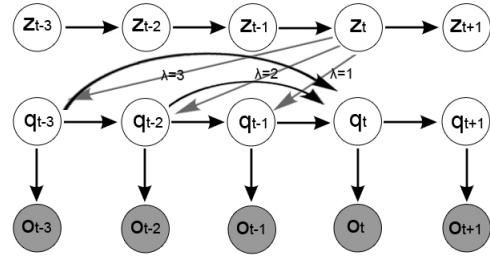


Figure 1: Graphical illustration of the generative model and the latent interdependencies assumptions of VDJ-HMM. Here,  $\lambda$  denotes the *active* dependence form inferred at time  $t$  by the model. Effectively, the value of  $\lambda$  determines which past observation-emitting latent state currently affects the temporal dynamics of observation generation.

the tendency of the model to overfitting, as well as the associated requirements in training data availability to ensure effective model training.

### 2.3 Model Training

To perform training for our model given a sequence  $O = \{\mathbf{o}_t\}_{t=1}^T$ , we resort to the familiar expectation-maximization (EM) paradigm [generalization of the here-derived algorithm for the case of training with multiple sequences is straightforward]. Based on the definition of VDJ-HMM [Eqs. (1)-(7)], the complete data of our model comprise the observable sequence  $O$ , the corresponding emitting state sequence  $Q = \{q_t\}_{t=1}^T$ , the dependence form sequence  $Z = \{z_t\}_{t=1}^T$ , and the sequence of corresponding mixture component indicators  $L = \{l_t\}_{t=1}^T$ . In addition, based on the derivations of [29], in the special case of considering multivariate Student's- $t$  mixture models as the emission distributions of VDJ-HMM, to allow for effective model training and inference procedures, we resort to expressing the multivariate Student's- $t$  distributions as scale-mixtures of Gaussians, yielding [29]:

$$p(\mathbf{o}_t | q_t = i; \{u_{imt}\}_{m=1}^M) = \sum_{m=1}^M w_{im} \mathcal{N}(\mathbf{o}_t; \boldsymbol{\mu}_{im}, \boldsymbol{\Sigma}_{im}/u_{imt}) \quad (9)$$

where  $u_{imt}$  is a precision scalar corresponding to the observation  $\mathbf{o}_t$  given it is generated from the  $j$ th component density of the  $i$ th emitting state, and is Gamma-distributed as [29]

$$u_{imt} \sim \mathcal{G}\left(\frac{\nu_{im}}{2}, \frac{\nu_{im}}{2}\right) \quad (10)$$

Under this setup, the above introduced set of precision scalars  $\{u_{imt}\}$  is also regarded as part of the complete data configuration of our model.

The EM algorithm comprises optimization of the posterior expectation of the complete data log-likelihood of the treated model [19]

$$Q(\Theta; \hat{\Theta}) \triangleq E_{\hat{\Theta}}(\log L_c(\Theta) | O) \quad (11)$$

where  $\hat{\Theta}$  denotes the currently obtained estimator of the model parameters set  $\Theta$ , and  $\log L_c(\Theta)$  is the expression of

the complete data log-likelihood of the model, which reads (ignoring constant terms)

$$\begin{aligned} \log L_c(\Theta) = & \sum_{h=1}^N \left\{ \mathbb{I}[q_1 = h] \log \varpi_h \right. \\ & + \sum_{k=1}^K \sum_t \mathbb{I}[z_t = k] \sum_{i=1}^N \mathbb{I}[q_{t-k} = h, q_t = i] \log \pi_{hi}^k \left. \right\} \\ & + \sum_{t=1}^T \sum_{i=1}^N \mathbb{I}[q_t = i] \log L_c(\mathbf{o}_t | q_t = i) \end{aligned} \quad (12)$$

where  $\mathbb{I}[\cdot]$  is the boolean operator. In Eq. (12),  $\log L_c(\mathbf{o}_t | q_t = i)$  is the complete data log-likelihood of the emission distribution of the  $i$ th hidden state with respect to  $\mathbf{o}_t$ , and the associated latent variables  $l_t$  and (in case of Student's- $t$  models)  $\{u_{imt}\}_m$ . In the case of Gaussian mixture emission distributions,  $\log L_c(\mathbf{o}_t | q_t = i)$  yields

$$\begin{aligned} \log L_c(\mathbf{o}_t | q_t = i) = & \sum_{m=1}^M \mathbb{I}[l_t = m] \left\{ \log w_{im} - \frac{1}{2} \log |\Sigma_{im}| \right. \\ & \left. - \frac{1}{2} d(\mathbf{o}_t, \boldsymbol{\mu}_{im}; \Sigma_{im}) \right\} \end{aligned} \quad (13)$$

where  $d(\mathbf{o}_t, \boldsymbol{\mu}_{im}; \Sigma_{im})$  is the Mahalanobis distance between  $\mathbf{o}_t$  and  $\boldsymbol{\mu}_{im}$ , with covariance matrix  $\Sigma_{im}$ . On the other hand, in the case of Student's- $t$  mixture emission distributions,  $\log L_c(\mathbf{o}_t | q_t = i)$  yields

$$\begin{aligned} \log L_c(\mathbf{o}_t | q_t = i) = & \sum_{m=1}^M \mathbb{I}[l_t = m] \left\{ -\log \Gamma\left(\frac{\nu_{im}}{2}\right) + \frac{\nu_{im}}{2} \times \right. \\ & \left[ \log\left(\frac{\nu_{im}}{2}\right) + \log u_{imt} - u_{imt} \right] + \log w_{im} \\ & \left. - \frac{u_{imt}}{2} d(\mathbf{o}_t, \boldsymbol{\mu}_{im}; \Sigma_{im}) - \frac{1}{2} \log |\Sigma_{im}| \right\} \end{aligned} \quad (14)$$

where  $\Gamma(\cdot)$  is the Gamma function.

As usual, the EM algorithm for our model is an iterative procedure, each iteration of which comprises an E-step and an M-step. On the E-step of the algorithm, we compute a set of posterior expectations pertaining to the latent variables of our model (sufficient statistics), using the current estimator of the model parameters set  $\hat{\Theta}$ . Subsequently, on the M-step of the algorithm, we optimize the model parameters set  $\Theta$  using the sufficient statistics computed previously, in order to obtain an updated estimator of the model parameters set,  $\hat{\Theta}$ .

### 2.3.1 E-step

From (11) and (12), it directly follows that the E-step of our algorithm consists in computing the posterior probabilities of the latent states on the first and second hidden layers of our model, as well as the corresponding state transition posteriors. It also comprises computation of the emitting state-conditional mixture component posteriors, as well as the posteriors of the precision scalars  $u_{imt}$ , when considering Student's- $t$  mixture emission distributions.

Let us begin with the mixture component posteriors, hereafter denoted as  $\xi_{imt}$ ; we have

$$\xi_{imt} \triangleq E_{\Theta}(l_t = m | \mathbf{o}_t, q_t = i) = \frac{p(\mathbf{o}_t | q_t = i, l_t = m)}{\sum_{h=1}^M p(\mathbf{o}_t | q_t = i, l_t = h)} \quad (15)$$

This expression yields

$$\xi_{imt} = \frac{w_{im} \mathcal{N}(\mathbf{o}_t | \boldsymbol{\mu}_{im}, \Sigma_{im})}{\sum_{h=1}^M w_{ih} \mathcal{N}(\mathbf{o}_t | \boldsymbol{\mu}_{ih}, \Sigma_{ih})} \quad (16)$$

when considering Gaussian mixture emissions, and

$$\xi_{imt} = \frac{w_{im} \mathcal{S}(\mathbf{o}_t | \boldsymbol{\mu}_{im}, \Sigma_{im}, \nu_{im})}{\sum_{h=1}^M w_{ih} \mathcal{S}(\mathbf{o}_t | \boldsymbol{\mu}_{ih}, \Sigma_{ih}, \nu_{ih})} \quad (17)$$

in the case of Student's- $t$  mixture emissions.

Regarding the posterior expectations of the precision scalars  $u_{imt}$  (if applicable), we have

$$\hat{u}_{imt} \triangleq E_{\Theta}(u_{imt} | \mathbf{o}_t) = \frac{\nu_{im} + D}{\nu_{im} + d(\mathbf{o}_t, \boldsymbol{\mu}_{im}; \Sigma_{im})} \quad (18)$$

Further, to obtain the rest of the sought posteriors, we need to define a set of auxiliary distributions, which can be computed by means of a variant of the well-known forward-backward algorithm [30], [1]. Specifically, let us define the forward probabilities

$$\alpha_t(i, k) \triangleq p(\{\mathbf{o}_\tau\}_{\tau=1}^t; q_t = i | z_t = k) \quad (19)$$

These probabilities can be computed iteratively, with initialization

$$\alpha_1(i, k) = \begin{cases} \varpi_i p(\mathbf{o}_1 | q_1 = i), & k = 1 \\ 0, & k > 1 \end{cases} \quad (20)$$

and recursion

$$\alpha_t(j, k) = p(\mathbf{o}_t | q_t = j) \sum_i \sum_{\lambda} \pi_{ij}^k \alpha_{t-k}(i, \lambda) \quad (21)$$

In a similar way we define the backward probabilities of our model, which yield

$$\beta_t(i, k) \triangleq p(\{\mathbf{o}_\tau\}_{\tau=t+1}^T | q_t = i; z_{t+k} = k) \quad (22)$$

These probabilities can also be computed iteratively, with initialization

$$\beta_T(i, k) = 1, \quad \forall k \quad (23)$$

and recursion

$$\beta_t(i, k) = \sum_{j=1}^M \sum_{\lambda} \pi_{ij}^k p(\mathbf{o}_{t+k} | q_{t+k} = j) \beta_{t+k}(j, \lambda) \quad (24)$$

Having obtained the forward and backward probabilities of our model, we can now proceed to obtain the remaining sought posteriors. For the emitting state posteriors, hereafter denoted as  $\gamma_{jt}$ , we have

$$\gamma_{jt} \triangleq p(q_t = j | O) \propto \left[ \sum_{k=1}^K \zeta_{kt} \alpha_t(j, k) \right] \left[ \sum_{k'=1}^K \zeta_{k', t+k'} \beta_t(j, k') \right] \quad (25)$$

Similarly, the emitting state transition posteriors yield

$$\begin{aligned} \gamma_{ijt}^{\lambda} & \triangleq p(q_t = i, q_{t+\lambda} = j | z_{t+\lambda} = \lambda; O) \\ & \propto \sum_{k, k'=1}^K \alpha_t(i, k) \beta_{t+\lambda}(j, k') \pi_{ij}^{\lambda} p(\mathbf{o}_{t+\lambda} | q_{t+\lambda} = j) \end{aligned} \quad (26)$$

Finally, regarding the (“active”) dependence form posteriors, hereafter denoted as  $\zeta_{kt}$ , we have

$$\zeta_{kt} \triangleq E(z_t = k|O) \propto \sum_{i=1}^N \alpha_t(i, k) \sum_{\lambda} \zeta_{\lambda, t+\lambda} \beta_t(i, \lambda) \quad (27)$$

This concludes the E-step of our algorithm.

### 2.3.2 M-step

Having obtained the required posterior expectation expressions on the E-step of the training algorithm of our model, we now proceed to optimization of the objective function (11) over the model parameters to obtain the expressions of the model parameter updates. Let us introduce the notation

$$r_{imt} \triangleq \gamma_{it} \xi_{imt} \quad (28)$$

We then have

$$\pi_i = \gamma_{i1} \quad (29)$$

$$\pi_{hi}^\lambda = \frac{\sum_t \gamma_{hit}^\lambda}{\sum_t \gamma_{ht}} \quad (30)$$

$$w_{im} = \frac{\sum_{t=1}^T r_{imt}}{\sum_{t=1}^T \gamma_{it}} \quad (31)$$

Further, the parameters of the emission distributions yield the following expressions:

(i) In case of Gaussian mixture emissions, we have

$$\boldsymbol{\mu}_{im} = \frac{\sum_{t=1}^T r_{imt} \mathbf{o}_t}{\sum_{t=1}^T r_{imt}} \quad (32)$$

$$\boldsymbol{\Sigma}_{im} = \frac{\sum_{t=1}^T r_{imt} (\mathbf{o}_t - \boldsymbol{\mu}_{im})(\mathbf{o}_t - \boldsymbol{\mu}_{im})^T}{\sum_{t=1}^T r_{imt}} \quad (33)$$

(ii) In case of Student's- $t$  mixture emissions, we have

$$\boldsymbol{\mu}_{im} = \frac{\sum_{t=1}^T r_{imt} \hat{u}_{imt} \mathbf{o}_t}{\sum_{t=1}^T r_{imt} \hat{u}_{imt}} \quad (34)$$

$$\boldsymbol{\Sigma}_{im} = \frac{\sum_{t=1}^T r_{imt} \hat{u}_{imt} (\mathbf{o}_t - \boldsymbol{\mu}_{im})(\mathbf{o}_t - \boldsymbol{\mu}_{im})^T}{\sum_{t=1}^T r_{imt}} \quad (35)$$

while the degrees of freedom are obtained by solving w.r.t.  $\nu_{im}$  the equation

$$\begin{aligned} & 1 - \psi\left(\frac{\nu_{im}}{2}\right) + \log\left(\frac{\nu_{im}}{2}\right) \\ & + \psi\left(\frac{\hat{\nu}_{im} + D}{2}\right) - \log\left(\frac{\hat{\nu}_{im} + D}{2}\right) \\ & + \frac{1}{\sum_{t=1}^T r_{imt}} \sum_{t=1}^T r_{imt} (\log \hat{u}_{imt} - \hat{u}_{imt}) = 0 \end{aligned} \quad (36)$$

where  $\hat{\nu}_{im}$  is the current estimate of the degrees of freedom  $\nu_{im}$ , and  $\psi(\cdot)$  is the Digamma function.

This concludes the training algorithm of our model. An outline of the EM algorithm for VDJ-HMM is provided in Alg. 1.

---

### Algorithm 1 EM Algorithm for the VDJ-HMM model.

---

Initialize the model parameters estimate  $\hat{\Theta}$ . Set the maximum number of iterations,  $MAXITER$ , and the convergence threshold of the EM algorithm.

For  $MAXITER$  iterations or until convergence of the objective function  $Q(\Theta; \hat{\Theta})$  do:

- 1) Conduct the forward-backward algorithm to obtain the forward probabilities  $\alpha_t(j, k)$  and the backward probabilities  $\beta_t(i, k)$ , using Eqs. (20)-(21) and (23)-(24), respectively.
  - 2) Effect the E-step of the algorithm by computing the posteriors pertaining to the mixture components,  $\xi_{imt}$ , the precision scalars,  $\hat{u}_{imt}$ , the chain of observation-emitting states,  $\gamma_{jt}$  and  $\gamma_{ijt}^\lambda$ , and the Markov chain of dependence jumps,  $\zeta_{kt}$ . For this purpose, use Eqs. (15), (18), (25)-(26), and (27), respectively.
  - 3) Effect the M-step by computing the new estimates of the model parameters  $\pi_i, \pi_{hi}^\lambda, w_{im}, \boldsymbol{\mu}_{im}, \boldsymbol{\Sigma}_{im}$ , and  $\nu_{im}$ , using Eqs. (29)-(36), respectively.
- 

## 2.4 Inference Algorithm

A first inference problem we consider in this work is the problem of predicting the next emitting state, say at time  $t + 1$ , denoted as  $q_{t+1}$ , given the values of the currently observed data, i.e. the observations set  $\{\mathbf{o}_\tau\}_{\tau=1}^t$ . From the definition of our model, it is easy to deduce that the probability of the emitting state at time  $t + 1$ , given the sequence of past observations  $\{\mathbf{o}_\tau\}_{\tau=1}^t$ , can be written in the form

$$\begin{aligned} p(q_{t+1} = j | \{\mathbf{o}_\tau\}_{\tau=1}^t) &= \sum_k \sum_{i=1}^N p(q_{t-k+1} = i | \{\mathbf{o}_\tau\}_{\tau=1}^t) \\ &\quad \times p(q_{t+1} = j | q_{t-k+1} = i; z_{t-k+1} = k) \\ &= \sum_k \sum_{i=1}^N \pi_{ij}^k \gamma_{i, t-k+1} \end{aligned} \quad (37)$$

where the emitting state posteriors  $\gamma_{jt}$  are computed by (25), using the sequence of observations  $\{\mathbf{o}_\tau\}_{\tau=1}^t$ . On this basis, determination of the first-layer state of our model, say  $\hat{q}$ , that is most likely to emit the (next) observation at time  $t + 1$  can be performed by maximization of the conditionals  $p(q_{t+1} = j | \{\mathbf{o}_\tau\}_{\tau=1}^t)$ , yielding:

$$\hat{q} \triangleq \underset{j}{\operatorname{argmax}} p(q_{t+1} = j | \{\mathbf{o}_\tau\}_{\tau=1}^t) \quad (38)$$

Another inference problem quite common in the related literature is the task of determining the probability of a given sequence w.r.t. a trained VDJ-HMM model. For this purpose, we can resort to the forward algorithm of our model, similar to conventional HMMs. Specifically, let us consider a sequence  $O = \{\mathbf{o}_t\}_{t=1}^T$  and a trained VDJ-HMM model with parameter estimate  $\hat{\Theta}$ . Then, following the definition of our model, the probability of sequence  $O$  w.r.t. the available VDJ-HMM model yields

$$p(O | \hat{\Theta}) = \sum_{i=1}^N \sum_{k=1}^K \alpha_T(i, k) \quad (39)$$

Finally, as discussed in the Introduction, the key *inference* problem we focus on in this work is the problem of *sequence prediction*. Let us consider a sequence  $\{\mathbf{o}_\tau\}_{\tau=1}^t$ . Then, the sequence prediction problem we consider here is the problem of performing an one-step ahead forecast, i.e. predicting the observation value  $\mathbf{o}_{t+1}$  at time  $t+1$ , given the values  $\{\mathbf{o}_\tau\}_{\tau=1}^t$ . To address this problem, we exploit the above obtained results regarding computation of the next-state probabilities,  $p(q_{t+1} = j | \{\mathbf{o}_\tau\}_{\tau=1}^t)$ . Specifically, we effect the sequence prediction task at time  $t+1$  as follows:

(i) We use Eq. (38) to obtain the emitting state probabilities at the following time point ( $t+1$ ), given the current set of observations (up to time  $t$ ), i.e.  $p(q_{t+1} = j | \{\mathbf{o}_\tau\}_{\tau=1}^t)$ .

(ii) We set the generated predicted value  $\hat{\mathbf{o}}_{t+1}$  of the observation at time  $t+1$  equal to the mean value of the modeled variable  $\mathbf{o}$  at time  $t+1$ , based on the fitted VDJ-HMM model with parameters set  $\hat{\Theta}$ . Specifically, considering mixtures of Gaussians or Student's- $t$  densities as the emission distributions of our model, as discussed previously, this procedure yields:

$$\hat{\mathbf{o}}_{t+1} = \sum_{n=1}^N \sum_{m=1}^M p(q_{t+1} = n | \{\mathbf{o}_\tau\}_{\tau=1}^t) w_{nm} \boldsymbol{\mu}_{nm} \quad (40)$$

## 2.5 Computational Complexity

We conclude this section with a short discussion on the computational complexity of our model. We first focus on the training algorithm of our model: From Eqs. (19)-(27), we can easily observe that the main difference between VDJ-HMM model training and training of a simple first-order HMMs concerns computation of the set of forward and backward probabilities,  $\{\alpha_t(j, k)\}_{t,j,k}$  and  $\{\beta_t(j, k)\}_{t,j,k}$ , respectively, which are distinct for each possible temporal dependence pattern,  $k = 1, \dots, K$ . Indeed, turning to each one of the probabilities in these sets, from Eqs. (21) and (24) we observe that their computation imposes computational costs similar to the corresponding quantities (forward and backward probabilities) pertaining to simple first-order HMMs. Thus, the only difference consists in repeating this computation procedure for each postulated temporal dependence pattern,  $k = 1, \dots, K$ . Given the fact that the dominant computational costs related to these quantities concerns computation of the emitting state density functions, which are shared across the considered temporal dependence patterns,  $k = 1, \dots, K$ , it is easy to deduce that, with proper algorithm implementation, VDJ-HMM model training imposes only negligible computational overheads compared to conventional HMM formulations.

Similar results can be obtained regarding the computational costs of inference using our model. Specifically, from Eqs. (37)-(40) we observe that the extra computational costs of the inference procedures of the VDJ-HMM model compared to conventional first-order HMMs are related to the need of computing different state transition posteriors and different forward and backward probabilities for each postulated temporal dependence pattern  $k = 1, \dots, K$ . Thus, inference using our model induces only negligible computational overheads compared to conventional HMM formulations, especially when the number of postulated temporal dependence patterns  $K$  is considered to be low

(which is usually the case in real-world applications, since a relatively low  $K$  value selection typically suffices for optimal model performance).

## 3 EXPERIMENTS

In this section, we perform an extensive evaluation of the proposed VDJ-HMM model. For this purpose, we consider a set of time-series forecasting experiments dealing with real-world applications from the computational finance domain. Specifically, we first consider *volatility forecasting* in financial return series; further, we consider the problem of predicting the future *return values* for a set of considered assets. Broad empirical evidence (see, e.g. [21], [23], [25]) has shown that financial return series exhibit variable order non-linear temporal dependencies, as well as dependence jumps, both when it comes to *volatility forecasting* and concerning *future value prediction*. As such, leveraging the merits of our model in the context of these applications is expected to yield a significant performance improvement over the competition.

To provide some comparative results, apart from our method we also evaluate the related HMM $^\infty$  model [12], which postulates infinitely-long temporal dependencies at each time point, baseline first-order HMMs, and explicit-duration HSMMs [31]. In addition, we cite the performance of methods yielding the state-of-the-art results in the considered experimental scenarios, as they have been reported in the recent literature. In all cases, to ensure the validity of our comparisons, we perform model training following exactly the same experimental setup as in the case of the papers reporting the cited state-of-the-art results.

Our experimental setup is the following: For each one of the considered applications, we split the available data into a training sample, a validation sample, and a testing sample; we adopt the same splits as the authors of the state-of-the-art methods reported in the literature, to render our performance measurements comparable with these results. We use the available training samples to train multiple VDJ-HMM models with different configurations; specifically, we evaluate models with different maximum allowed numbers of alternative temporal dependence patterns (maximum steps back)  $K$ , numbers of emitting states  $N$ , and numbers of mixture components per emitting state  $M$ . We select the optimal model configuration on the basis of the obtained predictive performances on the available validation samples. Finally, we use the available test samples to obtain the reported performance figures. Similar is the experimental setup we adopt for the considered competitors. In all cases, to alleviate the effect of random model initialization on the reported performance results, we repeat our experiments 10 times, with different model initializations each time, and report average performance figures over these repetitions.

### 3.1 Volatility Forecasting

In this set of experiments, we apply our model to prediction of the volatility in daily returns of financial assets. Consider a modeled asset with price  $P_t$  at time  $t$ ; then, its daily return at time  $t$  is defined as the logarithm  $r_t \triangleq \log \frac{P_t}{P_{t-1}}$ . On this basis, (historic) volatility is defined as the square of the return series  $r_t^2$ ; as discussed in [32], this groundtruth

Table 1: EUR-USD exchange rate volatility: Optimal VDJ-HMM model configuration.

Parameter	Value
$K$	4
$N$	2
$M$	3

measurement constitutes one of the few consistent ways of volatility measuring. As our performance metric used to evaluate the considered algorithms, we consider the root mean squared error (RMSE) between the model-estimated volatilities and the squared returns of the modeled return series (except for the case of the experiments in Sections 3.1.3 and 3.1.4, where we use the mean square error (MSE) and mean absolute error (MAE), following the approach adopted in the related literature).

### 3.1.1 Euro-United States Dollar exchange rate volatility

Our first experimental scenario regarding volatility forecast is dealing with the EUR-USD exchange rate time series<sup>1</sup>. Specifically, for the purposes of this experiment, we use data from the period 5/17/2007 – 8/10/2008 as our training set, and data pertaining to the period 9/10/2008 - 2/3/2009 as our validation set. To perform model evaluation, we consider three distinct test samples, pertaining to the periods: 3/3/2009 - 10/12/2009, 10/13/2009 - 5/25/2010, and 5/26/2010 - 12/30/2010, respectively. This way, we allow for evaluating model performance in periods with different levels of inherent volatility in the European economy. In all cases, the evaluated methods are trained using a rolling window of the previous 60 days of returns to make daily volatility forecasts for the following 10 days; we retrain the models every 10 days. In our experiments, all the considered HMM-based methods are evaluated using both Gaussian mixtures and Student’s- $t$  mixtures as their state-conditional emission distributions. In the case of the HSMM method, we consider Poisson, Negative Binomial, Geometric, and Logarithmic densities for modeling state duration.

In Table 1, we depict the optimal configuration parameters of our model, obtained by utilizing the available validation set, as described previously. In Tables 2 and 3, we illustrate the obtained performances of the evaluated methods. Note that these results are obtained for optimal model configuration (as determined in the validation set) both in the case of our model and the considered competitors. As we observe, in all cases our VDJ-HMM model yields the best performance among the evaluated methods. In addition, it appears that utilization of Student’s- $t$  mixture emission distributions yields in most cases only negligible performance improvements over models postulating Gaussian mixture emission distributions. We also observe that the HSMM model yielded best performance when postulating Geometric state duration distributions (we omit the results pertaining to different HSMM model configurations for brevity).

Finally, in Figs. 2a-2c, we illustrate how model performance changes by varying model configuration, i.e. the

1. The used data have been obtained from the official website of the European Central Bank.

hyperparameter values  $K$  (maximum order of dependence jumps),  $N$  (number of emitting states), and  $M$  (number of mixture components). It is apparent that model configuration plays a critical role in the obtained performance. This is especially true for the maximum order of dependence jumps  $K$ : selecting too big a value results in performance deterioration, while values close to  $K = 1$  (i.e., reducing to a simple first-order HMM) yield inferior performance compared to a fully-fledged VDJ-HMM.

### 3.1.2 Time-series of multiple correlated exchange rates and market indices

In this set of experiments, we consider three application scenarios:

- In the first scenario, we model the return series pertaining to the following *currency exchange rates*, over the period December 31, 1979 to December 31, 1998 (daily closing prices):
  1. (AUD) Australian Dollar / US \$
  2. (GBP) UK Pound / US \$
  3. (CAD) Canadian Dollar / US \$
  4. (DKK) Danish Krone / US \$
  5. (FRF) French Franc / US \$
  6. (DEM) German Mark / US \$
  7. (JPY) Japanese Yen / US \$
  8. (CHF) Swiss Franc / US \$.
- In the second scenario, we model the return series pertaining to the following *global large-cap equity indices*, for the business days over the period April 27, 1993 to July 14, 2003 (daily closing prices):
  1. (TSX) Canadian TSX Composite
  2. (CAC) French CAC 40
  3. (DAX) German DAX
  4. (NIK) Japanese Nikkei 225
  5. (FTSE) UK FTSE 100
  6. (SP) US S&P 500.
- Finally, in the third scenario, we model the return series pertaining to the following seven *global large-cap equity indices* and *Euribor rates*, for the business days over the period February 7, 2001 to April 24, 2006 (daily closing prices for the first 6 indices, and annual percentage rate converted to daily effective yield for the last index):
  1. (TSX) Canadian TSX Composite
  2. (CAC) French CAC 40
  3. (DAX) German DAX
  4. (NIK) Japanese Nikkei 225
  5. (FTSE) UK FTSE 100
  6. (SP) US S&P 500
  7. (EB3M) Three-month Euribor rate.

These series have become standard benchmarks for assessing the performance of volatility prediction algorithms [33], [34], [35]. In our experiments, we follow an evaluation protocol similar to [34], [36]. We adopt the same data split as in [36]; all the evaluated methods are trained using a rolling window of the previous 120 days of returns to make daily volatility forecasts for the following 10 days; we retrain the models every 7 days.

To begin with, we consider modeling each asset with a different VDJ-HMM model; i.e. we postulate as many VDJ-

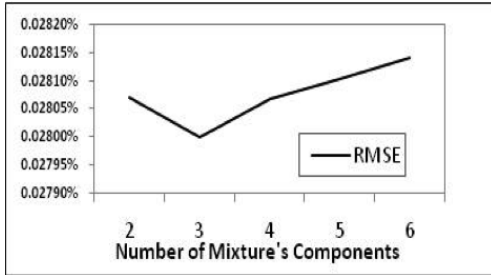


Table 2: EUR-USD exchange rate volatility: Performance (RMSE %) of the evaluated methods.

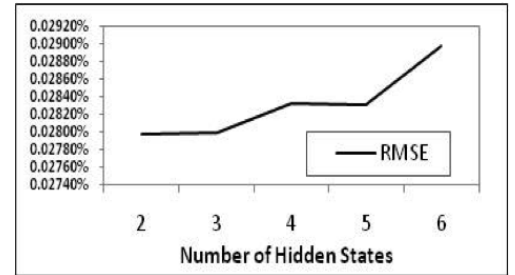
	HMM (Gaussian)	HMM (Student's- $t$ )	HMM $^\infty$ (Gaussian)	HMM $^\infty$ (Student's- $t$ )
3/3/2009 - 10/12/2009	1.607	1.559	1.591	1.534
10/13/2009 - 5/25/2010	0.738	0.721	0.730	0.713
5/26/2010 - 12/30/2010	0.683	0.696	0.677	0.691
Total	1.094	1.07	1.086	1.059

Table 3: EUR-USD exchange rate volatility: Performance (RMSE %) of the evaluated methods (cont.).

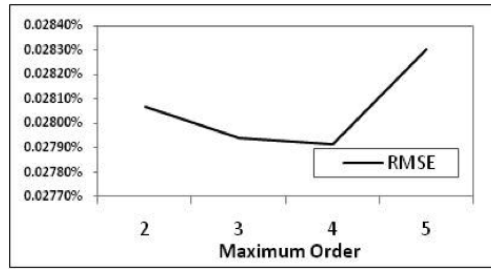
	HSMM (Geometric - Gaussian)	HSMM (Geometric - Student's- $t$ )	VDJ-HMM (Gaussian)	VDJ-HMM (Student's- $t$ )
3/3/2009 - 10/12/2009	1.689	1.74	1.504	1.435
10/13/2009 - 5/25/2010	0.717	0.703	0.7	0.702
5/26/2010 - 12/30/2010	0.681	0.687	0.672	0.669
Total	1.113	1.146	1.028	1.011



(a)



(b)



(c)

Figure 2: EUR-USD exchange rate volatility: Performance (RMSE %) fluctuation obtained by varying model configuration (validation set).

HMM models as the assets modeled in each scenario. The same *univariate* setup is also adopted for the considered HMM-based competitors<sup>2</sup>. Under this setup, the determined optimal configuration for our model is provided in Table 4. In Table 5, we provide the obtained results for the three considered scenarios (for optimal model configuration, as determined in the validation set). These results are computed over all the assets modeled in each scenario (averages). The performances of the state-of-the-art methods GARCH [37], [38], mixGARCH [39], VHGP [40], and GPMCH [36] have been cited from [36]. We observe that VDJ-HMM performs better than the competition in all scenarios, with the obtained performance differences becoming more significant in the case of scenario #1, which involves *only* currency exchange rates in the set of modeled assets. We tend to attribute this finding to the fact that currency exchange rates have a unique *mean-reverting property* [41], which seems

2. All HMM-based models are evaluated using Gaussian mixture emission distributions.

that our proposed VDJ-HMM model is capable of capturing much better than the competition.

Further, we consider the case of jointly modeling all the assets available in each scenario. For this purpose, we essentially postulate VDJ-HMM models with  $D$ -variate emission distributions, where  $D$  is the number of jointly modeled assets. The same holds for all the considered HMM-type competitors of our method. In Table 6, we report the determined optimal configuration of our model for this experimental setup. The corresponding predictive performances are reported in Table 7. In this table, we also cite the performance of the multioutput GPMCH model (using Clayton copulas), as reported in [36]. As we observe, our approach yields results comparable to or slightly better than the state-of-the-art in all cases. Note also that this performance improvement does also come for a significantly lower computational complexity compared to the second best performing method in these experiments, i.e. the GPMCH method.

Table 5: Time-series of multiple correlated exchange rates and market indices: Performance (RMSE %) obtained under the univariate modeling setup.

	HMM	HSMM	HMM $^\infty$	GARCH	mixGARCH	VHGP	GPMCH	VDJ-HMM
Scenario #1	0.0442	0.0292	0.0235	0.0705	0.0625	0.0146	0.0121	0.0108
Scenario #2	0.0841	0.0589	0.0353	0.2785	0.2623	0.0552	0.0360	0.0351
Scenario #3	0.0744	0.0578	0.0331	0.0552	0.0550	0.0542	0.0345	0.0329

Table 7: Time-series of multiple correlated exchange rates and market indices: Performance (RMSE %) obtained under the multivariate modeling setup.

	HMM	HMM $^\infty$	GPMCH	VDJ-HMM
Scenario #1	0.0345	0.0333	0.0341	0.0330
Scenario #2	0.0712	0.0609	0.0557	0.0605
Scenario #3	0.1512	0.1109	0.9905	0.0744

Table 4: Time-series of multiple correlated exchange rates and market indices: Optimal VDJ-HMM model configuration under the univariate modeling setup.

Parameter	Value
$K$	3
$N$	2
$M$	2

Table 6: Time-series of multiple correlated exchange rates and market indices: Optimal VDJ-HMM model configuration under the multivariate modeling setup.

Parameter	Value
$K$	3
$N$	2
$M$	2

### 3.1.3 Oil price time-series volatility

Further, we consider the problem of volatility forecasting in oil prices. For this purpose, and similar to the experimental setup of [42], we use the daily price data of the Brent index and the West Texas Intermediate (WTI) index from January 6, 1992, to December 31, 2009 (prices expressed in US dollars per barrel). From these time-series, the data pertaining to the last three years, i.e., 2007 to 2009, are used to evaluate the predictive performance of the evaluated models, while the data pertaining to the period 1/3/2006 - 12/29/2006 are used as our validation sample (and the rest for model training). All the evaluated methods are trained using a rolling window of the previous 60 days of returns to make daily volatility forecasts; we retrain the models every 5 days.

In Table 8, we report the optimal configuration of our model for our experiments with both time-series (Brent and WTI). In Table 9, we provide the obtained performances of the evaluated models. Note that all HMM-based models are evaluated using Gaussian mixture emission distributions. The performances of ARCH and its variants have been reported from [42]. As we observe, the proposed VDJ-HMM model consistently yields the best observed performance expressed in terms of the resulting MSE metric, with significant performance differences from all the considered competitors. On the other hand, when evaluation is performed using the MAE metric, we observe that our method manages to yield performance comparable to the state-of-the-art, but it cannot obtain further improvements; note though that the reported state-of-the-art MAEs are already exception-

Table 8: Oil price time-series volatility: Optimal VDJ-HMM model configuration.

Parameter	Value (Brent time-series)	Value (WTI time-series)
$K$	3	3
$N$	2	2
$M$	3	4

Table 10: Gold market time-series volatility: Optimal VDJ-HMM model configuration.

Parameter	Value
$K$	3
$N$	3
$M$	2

ally low, and therefore the room for further performance improvement is rather limited.

### 3.1.4 Gold market time-series volatility

Finally, we explore the performance of VDJ-HMM in volatility prediction for daily return series of Gold. The dataset used for this experiment consists of the daily Gold fixing prices of the London Bullion Market<sup>3</sup>. Specifically, following [43], we use the daily PM fixings price released at 15:00, and forecast the daily volatility during the second semester of 2008. This is an interesting and quite challenging experimental scenario, since the considered forecast period coincides with the period when the recent financial crisis took place. Similar to [43], our training and validation samples pertain to the period 1/4/1999 - 6/30/2008, while evaluation is performed using the MSE and MAE metrics.

In Table 10, we report the optimal configuration of our model. In Tables 11-12, we provide the obtained performances of the evaluated models. Note that all HMM-based models are evaluated using Gaussian mixture emission distributions. The performances of the reported state-of-the-art competitors, namely historical mean (HM), autoregressive models (AR( $k$ )), moving average models (MA( $k$ ) and EWMA), ARMA, as well as several GARCH variants [38], [37], have been cited from [43]. As we observe, the proposed VDJ-HMM model yields a quite satisfactory performance in this experiment, yielding error figures comparable to the state-of-the-art results reported in the recent literature.

3. Data obtained from the official website of the London Bullion Market Association ([www.lbma.org.uk](http://www.lbma.org.uk)).

Table 9: Oil price time-series volatility: Performance (MSE and MAE) of the evaluated approaches.

Method	Brent: MSE	Brent: MAE	WTI: MSE	WTI: MAE
GARCH	0.698	0.065	0.933	0.693
IGARCH	0.856	0.000	0.690	0.000
GJR	0.987	0.811	0.847	0.000
EGARCH	0.609	0.000	0.058	0.000
APARCH	0.557	0.002	0.846	0.031
FIGARCH	0.083	0.111	0.514	0.074
FIAPARCH	0.157	0.586	0.501	0.668
HYGARCH	0.080	0.030	0.546	0.000
HMM	0.087	0.095	0.200	0.067
HSMM	0.100	0.090	0.181	0.090
HMM <sup>∞</sup>	0.079	0.088	0.191	0.071
VDJ-HMM	0.050	0.001	0.044	0.000

Table 11: Gold market time-series volatility: Performance (MSE and MAE) of the evaluated approaches.

	HM	MA(20)	MA(40)	MA(120)	HMM	HSMM	HMM <sup>∞</sup>
MSE	105.24	84.64	83.29	87.97	85.77	85.5	84.52
MAE	5.43	5.96	5.72	5.40	5.69	5.82	5.63

Table 12: Gold market time-series volatility: Performance (MSE and MAE) of the evaluated approaches (cont.).

	AR(5)	MAD(5)	ARMA	EMWA	GARCH	GARCH-M	VDJ-HMM
MSE	86.08	90.08	84.24	83.81	86.94	86.35	84.16
MAE	5.67	5.54	5.68	5.84	5.56	5.68	5.60

### 3.2 Return Value Prediction

In this set of experiments, we apply our model to prediction of the future values of the daily return series of modeled financial assets,  $r_t$ . Specifically, under our experimental setup, we are interested in correctly predicting the sign of the return value at future time points. This sign can be used as the foundation of a simple portfolio management policy as follows: If the predicted future return sign is positive, then the policy suggests that the asset be retained by the investment portfolio manager; on the other hand, if the predicted future return sign is negative, then the policy creates a “sell” signal. All HMM-based models evaluated in these experiments postulate Gaussian mixture models as their emission distributions.

#### 3.2.1 Euro-United States Dollar exchange rate

We begin with evaluating our method considering future value prediction for the EUR-USD exchange rate. We use a training sample pertaining to the period 1/17//2002 – 5/16/2008, a validation sample pertaining to the period 5/17/2008 - 3/2/2009, and a test sample pertaining to the period 3/3/2009 - 12/30/2010. All the evaluated methods are trained using a rolling window of the previous 60 days of returns to make daily price prediction for the following 10 days; we retrain the models every 5 days.

On this basis, model evaluation is performed according to: (i) the comparison of the signs of the generated predictions with the actual ones (hereafter referred to as *directional prediction*); and (ii) the resulting annualized return of the aforementioned portfolio management policy, defined as the mean obtained profit adjusted for the return standard deviation over the whole forecasting period.

In Table 13, we depict the optimal configuration of our VDJ-HMM model as determined by utilizing the available validation set. In Figs. 3a-3c, we show how VDJ-HMM

Table 13: EUR-USD exchange rate price prediction: Optimal VDJ-HMM model configuration.

Parameter	Value
$K$	3
$N$	2
$M$	2

model performance changes by varying the adopted configuration (results obtained on the available validation set). As we observe, model configuration plays a crucial role to the obtained performance. Further, another interesting finding is that, similar to the volatility forecasting experiment, model performance reaches its optimal value for a moderate value of  $K$ , while experiencing a significant decrease for too high values of  $K$  or when  $K = 1$ .

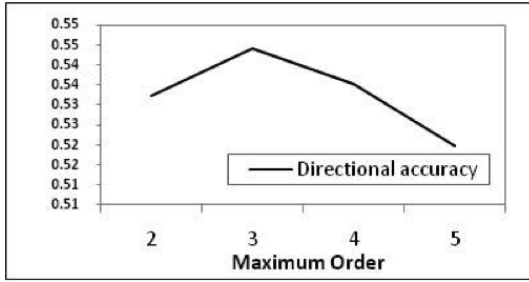
In Table 14, we provide the obtained performance results for the evaluated methods (for optimal model configuration). Note that the performances of the methods  $k$ -nearest neighbor (KNN), Naïve Bayes, back-propagation neural network (BP), support vector machine (SVM) [44], and random forest (RF) [45] have been cited from [46]. As we observe, our method completely outperforms the competition, yielding the state-of-the-art result in this dataset.

#### 3.2.2 Taiwan stock market

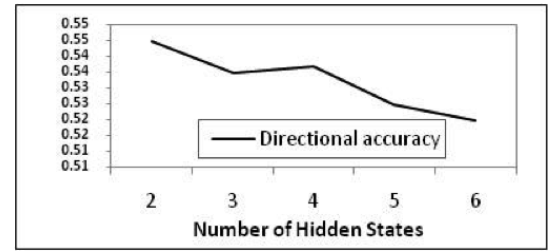
Finally, in this experiment, we apply our VDJ-HMM model to predicting the level of the Taiwan stock market (TAIEX). For transparency, in our experimental evaluations we use data pertaining to a seven-year period of the TAIEX, from 1999/1/4 to 2005/12/31, and split them as described in the state-of-the-art work presented in [47]. Specifically, we use the data from the first ten months of each year in the considered time period for model training and validation, and the data from the last two months as our test set. All the evaluated methods are trained using a rolling window of the

Table 14: EUR-USD exchange rate price prediction: Performance of the evaluated models.

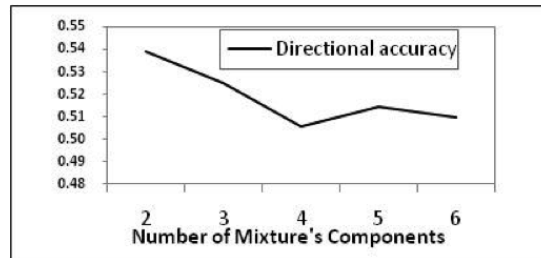
Statistic	KNN	Naïve Bayes	BP	SVM	RF	HMM	HSMM	HMM <sup>∞</sup>	VDJ-HMM
Directional Prediction Accuracy	50.11%	48.83%	50.12%	52.65%	53.50%	52.5%	51.2%	53.18%	54.05%
Annualized Return	-2.26%	-3.08%	1.59%	3.98%	7.28%	4.05%	1.5%	6.44%	9.50%



(a)



(b)



(c)

Figure 3: EUR-USD exchange rate price prediction: Performance fluctuation (directional prediction accuracy) obtained by varying model configuration (validation set).

Table 15: Taiwan stock market price prediction: Optimal VDJ-HMM model configuration.

Parameter	Value
$K$	3
$N$	2
$M$	4

previous 120 days of returns to make daily price prediction for the following 5 days; we retrain the models every 5 days.

In Table 15, we depict the optimal configuration of our VDJ-HMM model as determined by utilizing the available validation set. In Table 16, we provide the obtained performance figures pertaining to our method and the considered competitors. Performance evaluation is conducted on the grounds of the resulting RMSE between the predicted prices and the actual ones. Note that, apart from HMM and HMM<sup>∞</sup>, performance of the rest of the considered competitors is cited from [47]. Based on the reported RMSE results, we deduce that VDJ-HMM achieves a clearly competitive performance compared to existing state-of-the-art approaches. Specifically, in some years our VDJ-HMM model achieves the lowest RMSE among the considered competitors, while retaining a very competitive performance in the rest of the examined years.

## 4 CONCLUSIONS

In this paper, we focused on the problem of modeling sequential data the temporal dynamics of which may switch

between different patterns over time. To address this problem, we introduced a hierarchical model comprising two hidden chains of temporal dependencies: on the *first layer*, our model comprises a *chain of latent observation-emitting states*, the dependencies between which may change over time; on the *second layer*, our model utilizes a *latent first-order Markov chain* modeling the *evolution* of temporal dynamics pertaining to the first-layer latent process. To allow for tractable training and inference procedures, our model considers *temporal dependencies* taking the form of *variable order dependence jumps*, the order of which is *inferred* from the data as part of the model inference procedure. We devised efficient model training and inference algorithms under the maximum-likelihood paradigm.

To evaluate the capacity of our method in effectively modeling non-homogeneous observed sequential data, where the patterns of temporal dependencies may change over time, we considered a number of computational finance applications. Specifically, we considered both *volatility forecasting* applications as well as *value prediction* applications dealing with financial return series for sets of considered assets. As we discussed, this setting allows for an objective evaluation of whether our approach does actually achieve its goals, since empirical evidence has shown that financial return series exhibit variable order non-linear temporal dependencies, as well as dependence jumps.

Our experimental results provided strong evidence that our method is actually capable of delivering on its goals, allowing for obtaining much better performance compared to: (i) baseline (first-order) HMMs; (ii) the HMM<sup>∞</sup> model,

Table 16: Taiwan stock market price prediction: Performance (RMSE) of the evaluated models.

Models	1999	2000	2001	2002	2003	2004	2005
Linear regression	164	420	1070	116	329	146	-
Method of [48]	-	139	144	82	73	-	-
Method of [49]	109	152	130	84	56	79	69
Method of [50]	-	-	122	94	55	69	65
Fuzzy Symmetric Method [47]	103	130	120	68	55	56	54
Fuzzy Asymmetric Method [47]	109	122	125	68	58	58	53
HMM	112	154	116	71	60	59	56
HSMM	111	155	120	75	61	58	59
HMM <sup>∞</sup>	109	148	113	70	56	58	54
VDJ-HMM	109	145	111	68	55	55	54

designed for capturing arbitrarily long temporal dependencies; and (iii) state-of-the-art methods in the considered application domains, e.g. methods belonging to the GARCH family. As we showed, these encouraging performance results come at a very low additional computational cost compared to existing approaches; thus, our method offers a favorable performance/complexity trade-off.

An issue we have not fully addressed in this work is how we could allow for automatic determination of the optimal model configuration, without the need of resorting to cross-validation (as we did in our experimental evaluations). For this purpose, one could resort to devising a nonparametric Bayesian construction for the VDJ-HMM model, by imposing appropriate priors over the model parameters (e.g., Dirichlet process priors [51] over the transition probability matrices of our model), and performing Bayesian inference instead of maximum-likelihood training. This issue remains to be addressed in our future work.

## REFERENCES

- [1] O. Cappé, E. Moulines, and T. Rydén, *Inference in Hidden Markov Models*. New York: Springer Series in Statistics, 2005.
- [2] J. Mari, D. Fohr, and J. Junqua, "A second-order HMM for high-performance word and phoneme-based continuous speech recognition," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 1996, pp. 435–438.
- [3] J.-F. Mari, J.-P. Haton, and A. Kriouile, "Automatic word recognition based on second-order hidden Markov models," *IEEE Trans. Speech Audio Process.*, vol. 5, no. 3, pp. 22–25, 1997.
- [4] E.-M. Nel, J. D. Preez, and B. Herbst, "Estimating the pen trajectories of static signatures using hidden Markov models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1733–1746, 2005.
- [5] C. Eng, A. Thibessard, S. Hergalant, J.-F. Mari, and P. Leblond, "Data mining using hidden Markov models (HMM2) to detect heterogeneities into bacteria genomes," in *Journées Ouvertes Biol. Inf. Math. (JOBIM)*, 2005.
- [6] O. Aycard, J.-F. Mari, and R. Washington, "Learning to automatically detect features for mobile robots using second-order hidden Markov models," *Int. J. Adv. Robotic Syst.*, vol. 1, no. 4, pp. 231–245, 2004.
- [7] H. Engelbrecht and J. du Preez, "Efficient backward decoding of high-order hidden Markov models," *Pattern Recognition*, vol. 43, no. 1, pp. 99–112, 2010.
- [8] G. McLachlan and D. Peel, *Finite Mixture Models*. Wiley Series in Probability and Statistics, 2000.
- [9] S. P. Chatzis, "Margin-maximizing classification of sequential data with infinitely-long temporal dependencies," *Expert Systems with Applications*, vol. 40, no. 11, pp. 4519–4527, 2013.
- [10] R. Begleiter, R. El-Yaniv, and G. Yona, "On prediction using variable order Markov models," *Journal of Machine Learning Research*, vol. 22, pp. 385–421, 2004.
- [11] C. Dimitrakakis, "Bayesian variable order Markov models," in *Proc. AISTATS*, 2010, pp. 161–168.
- [12] S. P. Chatzis, D. I. Kosmopoulos, and G. M. Papadourakis, "A non-stationary hidden Markov model with approximately infinitely-long time-dependencies," in *Proc. ISVC*, vol. 2, 2014, pp. 51–62.
- [13] M. J. Zaki, C. D. Carothers, and B. K. Szymanski, "VOGUE: A Variable Order Hidden Markov Model with Duration based on Frequent Sequence Mining," *ACM Transactions on Knowledge Discovery from Data*, vol. 4, no. 1, pp. 1–31, 2010.
- [14] J. Gasthaus and Y. W. Teh, "Improvements to the Sequence Memoizer," in *Proc. NIPS*, 2011.
- [15] G. Celeux, F. Forbes, and N. Peyrard, "EM procedures using mean field-like approximations for Markov model-based image segmentation," *Pattern Recognition*, vol. 36, no. 1, pp. 131–144, 2003.
- [16] J. Zhang, "The mean field theory in EM procedures for Markov random fields," *IEEE Transactions on Image Processing*, vol. 2, no. 1, pp. 27–40, 1993.
- [17] F. Sha and L. K. Saul, "Large margin Gaussian mixture modeling for phonetic classification and recognition," in *Proceedings of ICASSP 2006*, Toulouse, France, 2006, pp. 265–268.
- [18] F. Wood, J. Gasthaus, C. Archambeau, L. James, and Y. W. Teh, "The sequence memoizer," *Communications of the ACM*, vol. 54, no. 2, pp. 91–98, 2011.
- [19] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, B*, vol. 39, no. 1, pp. 1–38, 1977.
- [20] C. Stărică and C. Granger, "Nonstationarities in stock returns," *Review of economics and statistics*, vol. 87, no. 3, pp. 503–522, 2005.
- [21] S. J. Taylor, *Modelling financial time series*. World Scientific Publishing Company, 2007.
- [22] R. Cont, "Empirical properties of asset returns: stylized facts and statistical issues," *Quantitative Finance*, vol. 1, pp. 223–236, 2001.
- [23] Z. Ding, C. W. Granger, and R. F. Engle, "A long memory property of stock market returns and a new model," *Journal of empirical finance*, vol. 1, no. 1, pp. 83–106, 1993.
- [24] V. Todorov and G. Tauchen, "Volatility jumps," *Journal of Business & Economic Statistics*, vol. 29, no. 3, pp. 356–371, 2011.
- [25] B. Eraker, "Do stock prices and volatility jump? Reconciling evidence from spot and option prices," *The Journal of Finance*, vol. 59, no. 3, pp. 1367–1404, 2004.
- [26] B. Eraker, M. Johannes, and N. Polson, "The impact of jumps in volatility and returns," *The Journal of Finance*, vol. 58, no. 3, pp. 1269–1300, 2003.
- [27] T. Rydén, T. Teräsvirta, and S. Åsbrink, "Stylized facts of daily return series and the hidden Markov model," *Journal of applied econometrics*, vol. 13, no. 3, pp. 217–244, 1998.
- [28] J. Bulla and I. Bulla, "Stylized facts of financial time series and hidden semi-Markov models," *Computational Statistics & Data Analysis*, vol. 51, no. 4, pp. 2192–2209, 2006.
- [29] S. P. Chatzis, D. I. Kosmopoulos, and T. A. Varvarigou, "Robust sequential data modeling using an outlier tolerant hidden Markov model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 9, pp. 1657–1669, 2009.
- [30] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 245–255, 1989.
- [31] S. Yu, "Hidden semi-Markov models," *Artificial Intelligence*, vol. 174, no. 2, pp. 215–243, 2010.
- [32] C. T. Brownlees, R. F. Engle, and B. T. Kelly, "A practical guide to volatility forecasting through calm and storm," 2009, available at SSRN: <http://ssrn.com/abstract=1502915>.
- [33] B. McCullough and C. Renfro, "Benchmarks and software standards: A case study of GARCH procedures," *Journal of Economic and Social Measurement*, vol. 25, pp. 59–71, 1998.

- [34] A. G. Wilson and Z. Ghahramani, "Copula processes," in *Advances in Neural Information Processing Systems*, 2010.
- [35] C. Brooks, S. Burke, and G. Persaud, "Benchmarks and the accuracy of GARCH model estimation," *International Journal of Forecasting*, vol. 17, pp. 45–56, 2001.
- [36] E. A. Platanios and S. P. Chatzis, "Gaussian process-mixture conditional heteroscedasticity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 888–900, May 2014.
- [37] T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *Journal of Econometrics*, vol. 94, pp. 238–276, 1986.
- [38] R. Engle, "Autoregressive conditional heteroskedasticity models with estimation of variance of United Kingdom inflation," *Econometrica*, vol. 50, pp. 987–1007, 1982.
- [39] M. Haas, S. Mittnik, and M. Paolella, "Mixed normal conditional heteroskedasticity," *Journal of Financial Econometrics*, vol. 2, no. 2, pp. 211–250, 2004.
- [40] M. Lázaro-Gredilla and M. Tsitsias, "Variational heteroscedastic Gaussian process regression," in *Proc. 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.
- [41] J. R. Lothian, "Some new stylized facts of floating exchange rates," *Journal of International Money and Finance*, vol. 17, no. 1, pp. 29–39, 1998.
- [42] Y. Wei, Y. Wang, and D. Huang, "Forecasting crude oil market volatility: Further evidence using GARCH-class models," *Energy Economics*, vol. 32, no. 6, pp. 1477–1484, 2010.
- [43] S. Trück and K. Liang, "Modelling and forecasting volatility in the gold market," *International Journal of Banking and Finance*, vol. 9, no. 1, pp. 48–80, 2012.
- [44] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
- [45] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [46] K. Theofilatos, S. Likothanassis, and A. Karathanasopoulos, "Modeling and trading the EUR/USD exchange rate using machine learning techniques," *Engineering, Technology & Applied Science Research*, vol. 2, no. 5, pp. 269–272, 2012.
- [47] T.-L. Chen, "Forecasting the Taiwan stock market with a novel momentum-based fuzzy time-series," *Review of Economics & Finance*, vol. 2, pp. 38–50, 2012.
- [48] K. H. Huarng and Y. H. K., "A type-2 fuzzy time-series model for stock index forecasting," *Physica A*, vol. 353, pp. 445–462, 2005.
- [49] K. H. Huarng and Y. T. H. K., "The application of neural networks to forecast fuzzy time series," *Physica A*, vol. 363, no. 2, pp. 481–491, 2006.
- [50] C. Su, T. Chen, C. Cheng, and Y. Chen, "Forecasting the stock market with linguistic rules generated from the minimize entropy principle and the cumulative probability distribution approaches," *Entropy*, vol. 12, no. 2, pp. 2397–2417, 2010.
- [51] D. M. Blei and M. I. Jordan, "Variational inference for Dirichlet process mixtures," *Bayesian Analysis*, vol. 1, no. 1, pp. 121–144, 2006.