Harmonic Recurrent Process for Time Series Forecasting

Shao-Qun Zhang and **Zhi-Hua Zhou**[1]

Abstract. In this paper, we propose the *Harmonic Recurrent Process* (HRP) for forecasting non-stationary time series with period-varying patterns. HRP works by selectively ensembling recurrent period-varying patterns in harmonic analysis. In contrast to classical forecasting approaches that rely on stationary priors and recurrent neural network approaches that are mostly black boxes, our model is able to deal with irregular nonstationary signals, and its working mechanism is reasonably lucid. We also prove that the stochastic process led by HRP under weak dependence condition is predictive PAC learnable. Comprehensive experiments on simulated and practical tasks validate the effectiveness of HRP.

1 Introduction

In time series forecasting (TSF), nonstationary time series is often encountered. Since the sample distribution changes along time, especially when the latent dominant facts change along time, nonstationary time series usually performs aperiodically and the patterns are unequal-interval. An intuitive manifestation is that the peaks or troughs are not equally spaced, such as the coupling cosine signals in Figure 1(d), the automobile registration data in Figure 1(e), and the Yosemite temperature series in Figure 1(f). Predicting future values or trends of nonstationary time series with period-varying patterns has been a long-standing challenge in TSF [11, 13, 38].

There have been great efforts on TSF. The most famous are statistical models, which usually employ common priors for the evolving of time series. The representative approaches include AutoRegressive Moving Average and Multivariate AutoRegressing (MAR) modeling, which assume that the concerned time series is stationary, and formulate the forecasting model as a stochastic differential equation [34, 8]. The cosine function in Figure 1(a) gives an example of conventional stationary sequence with equal period. Even with the help of spectrum analysis, the generalized stationarity assumption [18, 16] can only cope with the regular period signals such as the mixture cosine signals in Figure 1(b) and the electrocardiogram data [19] in Figure 1(c). Furthermore, sufficient evidence [22] points out that the spectrum methods are more suitable for long term analysis, limiting the applicable range of this kind of approach.

To deal with nonstationary time series, some investigators focus on the model flexibility for the unknown evolution structure of observations, leaving the underlying series distributions with wide assumptions. For instance, the local evolution laws of time series can be captured using simple chain update structure and scalable matrix factorization methods [33, 23, 41]. However, this kind of approach often forces time series patterns to be within a preset fixed time window, and thus the models are not guaranteed to be able to characterize nonstationary time series with period-varying patterns. Alternative approaches are based on deep learning, such as RNN, LSTM and their variants [6, 24, 12, 28], which can effectively extract the features, and forcast non-stationary time series with period-varying patterns. However, the deep learning models often lack comprehensibility and make the whole model blackboxes.

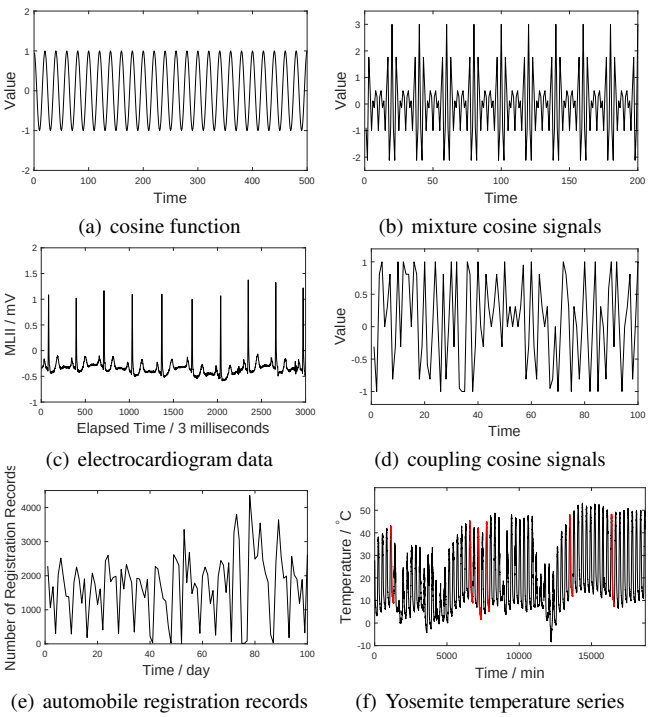

Figure 1. Illustration of stationary (a)-(c) and nonstationary (d)-(f) series.

To tackle the challenge, we try to explore a special type of "recurrent" patterns of non-stationary time series. Different from the patterns of conventional TSF models, the recurrent patterns not only allow unequal periods, but also be compatible with harmonic transformation, e.g., shifting and scaling. The red lines that can overlap each other by shifting or scaling in Figure 1(f) give an illustration of this type of recurrent patterns. By extracting the recurrent period-varying patterns with harmonic transformation, the model can adapt to complicated cases even when the sample distributions change over time.

Inspired by this recognition, we propose the *Harmonic Recurrent Process* (HRP) for forecasting nonstationary time series with period-varying patterns. Based on harmonic decomposition on order-subpermutation deformations, HRP formulates the forecasting model into a specific formula, which can extract the recurrent patterns of

[1] National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China, email: {zhangsq,zhouzh}@lamda.nju.edu.cn

nonstationary time series with self-guided parameters, e.g., the window size of recurrent patterns. By selectively ensembling the detected recurrent patterns, HRP is able to handle complicated signals, including non-stationary time series with period-varying patterns. Note that order-subpermutation is an introduced elastic technology for matching the recurrent but unequal-period patterns, which may be of independent interest. We also prove that the stochastic process led by HRP under weak dependence condition is predictive PAC learnable. Comprehensive experiments are conducted on the simulated coupling cosine signals as well as two real-world data sets, Yancheng automobile registration records and CSI 300. The experimental results show the superiority of HRP over several state-of-the-art statistical models and deep learning models.

The rest of this paper is organized as follows. Section 2 introduces some related notations and concepts. In Section 3, we present the proposed HRP and a concrete implementation. Section 4 gives predictable theoretical analysis about the stochastic process led by HRP. The experiments are conducted on simulated and real-world data sets in Section 5. Finally, we conclude our work in Section 6.

2 Preliminaries

In this section, we review the working mechanism of the stationarity assumption and introduce some related concepts and notations.

Let $\mathcal{Z}$, $\mathcal{N}^+$ and $\mathcal{R}$ denote the set of time stamps, non-negative integers and real numbers, respectively. Consider a doubly infinite sequence of L-dimensional random variables $\boldsymbol{X}_t = (X_t^1, X_t^2, \cdots, X_t^L)^T$ jointly distributed according to a distribution $\mathcal{D}$. We write $\boldsymbol{X}|_t^{t+\tau}$ to denote a multivariable sequence $(\boldsymbol{X}_t, \boldsymbol{X}_{t+1}, \cdots, \boldsymbol{X}_{t+\tau})$ and $\boldsymbol{\mathcal{D}}_t^{t+\tau}$ to denote the joint distribution of $\boldsymbol{X}|_t^{t+\tau}$, where $\tau \in \mathcal{N}^+$. For the univariate time series, $\boldsymbol{X}_t$ only comprises a real-valued X_t or X_t^i. Correspondingly, $X^i|_t^{t+\tau}$ indicates the sequence $(X_t^i, X_{t+1}^i, \cdots, X_{t+\tau}^i)$. A stochastic process $\{\boldsymbol{X}_t, t \in \mathcal{Z}\}$ is said to be (strictly) stationary provided that, for any $t \in \mathcal{Z}$ and any τ, $k \in \mathcal{N}^+$, the distributions of $\boldsymbol{X}|_t^{t+\tau}$ and $\boldsymbol{X}|_{t+k}^{t+\tau+k}$ are the same, that is, $\boldsymbol{\mathcal{D}}_t^{t+\tau} = \boldsymbol{\mathcal{D}}_{t+k}^{t+\tau+k}$.

The finite dimensional distribution function of a stationary time series is only the function of the time interval τ, but independent of the specific time point t. Thereby, stationary time series models work by fitting the apposite finite dimensional distribution function $\boldsymbol{\mathcal{D}}$, which is made up of the tractable characteristic function defined on a fixed window, e.g., the cosine function shown in Figure 1(a). By exploiting spectral decomposition, the time series with regular periods can also be processed with generalized stationarity assumptions, although its patterns still depend on a fixed window and are independent of the current time t [34]. The mixture cosine signal in Figure 1(b) and the electrocardiogram data in Figure 1(c) give a typical illustration.

However, most real-world situations are nonstationary and period-varying, where the patterns are not only related to time windows but also to time, e.g., the coupling cosine signals in Figure 1(d) and two real cases in Figure 1(e) and 1(f). Different to the regular distribution functions of stationary stochastic processes, non-stationary distribution families usually change within a limited range over time and are coupled together in a more loose manner. Formally speaking, for any t, there exist certain τ_i and $k \in \mathcal{N}^+$, satisfying $\boldsymbol{\mathcal{D}}_t^{t+\tau_i} = \boldsymbol{\mathcal{D}}_{t+k}^{t+\tau_i+k}$. The collection $\{\tau_i\}$ of periods is generally finite. In this paper, we are going to cope with the nonstationary TSF challenge with the aforementioned finite-period distribution structure.

3 Harmonic Recurrent Process

In this section, we will introduce the HRP, which can handle non-stationary time series signals by exploring the recurrent period-varying patterns. Firstly, we introduce a couple of elastic metrics for measuring the period-varying patterns. Secondly, we present the core mechanism of HRP. Finally, we give the concrete implementation of two key elements of HRP.

3.1 Order-subpermutation

To explore the period-varying patterns, we need a suitable metric that is able to measure the stochastic processes of different window sizes. Here we develop an alternative oracle by efficiently searching the best period between two stochastic processes. We begin it by introducing the *order-subpermutation matrix*.

Definition 1 *A matrix $\mathbf{P} \in \{0,1\}^{n\times m}$ is an order-subpermutation matrix, if $\mathbf{P}$ satisfies the following conditions:*

- $\sum_{j=1}^{n} \mathbf{P}(j,i) = 1$*;*
- $\mathbf{P}(n,m) = 1$*;*
- *let p_i denote the number of row satisfying $\mathbf{P}(p_i,i) = 1$, then $i \le p_i \le p_{i+1}$,*

where $\mathbf{P}(j,i)$ denotes the entry of the j-th row and i-th column of matrix $\mathbf{P}$.

The order-subpermutation matrix makes a row vector deformed by fixing the end point (due to $\mathbf{P}(n,m) = 1$) and allowing to abandon the first few points in the sequence in order (since p_1 may be greater than 1). From another perspective, the order-subpermutation matrix provides the possibility of order-preserving deformation of the concerned sequence, which includes the best prediction windows. An illustration about the order-subpermutation matrix is shown in Figure 2. As we can see, a 7-dimensional row vector $(1,2,3,4,5,6,7)$ is converted into a 8-dimensional row vector $(2,2,3,4,5,6,7,7)$ by a 7×8 order-subpermutation matrix. The sequence $(p_2,p_2,p_3,p_4,p_5,p_6,p_7,p_7)$ indicates the position encoding deformed by this order-subpermutation matrix. For example, since $\mathbf{P}(p_2,1) = 1$ and $\mathbf{P}(p_2,2) = 1$, both of the first and second elements of the deformation vector are equal to 2, i.e., the second element of the original vector.

Based on order-subpermutation deformation and optimal transport theory [21], we can derive two novel metrics for measuring the discrepancy of two sequences. (1) By deforming an n-dimensional sequence into an m-dimensional version, the single-side distance from sequence $\boldsymbol{x} = (x_1, \cdots, x_n)$ to sequence $\boldsymbol{y} = (y_1, \cdots, y_m)$ can be formalized as $\min_{\mathbf{P}} \|\boldsymbol{x}\,\mathbf{P} - \boldsymbol{y}\|_2$, where $\mathbf{P} \in \{0,1\}^{n\times m}$ is an order-subpermutation matrix. (2) Meanwhile, the distance between sequences $\boldsymbol{x}$ and $\boldsymbol{y}$ can be calculated through an l-dimensional embedding space, that is, $\min_{\{\mathbf{P},\mathbf{Q}\}} \|\boldsymbol{x}\,\mathbf{P} - \boldsymbol{y}\,\mathbf{Q}\|_2$, where $\mathbf{P} \in \{0,1\}^{n\times l}$ and $\mathbf{Q} \in \{0,1\}^{m\times l}$ are both order-subpermutation matrices. Further, we can obtain two meaningful metrics for measuring the distance between multivariate stochastic processes as follows.

Definition 2 *(1) The Single-side Order-subPermutation Distance (SOPD) from a stochastic process $\boldsymbol{\mathcal{D}}_{t+1}^{t+n}$ to another one $\boldsymbol{\mathcal{D}}_{s+1}^{s+m}$ is*

$$SOPD(\boldsymbol{\mathcal{D}}_{t+1}^{t+n}, \boldsymbol{\mathcal{D}}_{s+1}^{s+m}) = \min_{\mathbf{P}_i} \sum_{i=1}^{L} \|(X^i|_{t+1}^{t+n} \cdot \mathbf{P}_i - X^i|_{s+1}^{s+m}) \cdot \mathbf{W}_i\|_2; \tag{1}$$

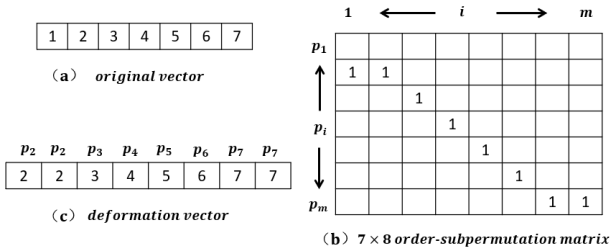

Figure 2. An example of the order-subpermutation deformation.

(2) The Order-subPermutation Distance (OPD) between two stochastic processes $\boldsymbol{\mathcal{D}}_{t+1}^{t+n}$ and $\boldsymbol{\mathcal{D}}_{s+1}^{s+m}$ is

$$OPD(\boldsymbol{\mathcal{D}}_{t+1}^{t+n}, \boldsymbol{\mathcal{D}}_{s+1}^{s+m}) = \min_{\mathbf{P}_i, \mathbf{Q}_i} \sum_{i=1}^{L} \|(X^i|_{t+1}^{t+n} \cdot \mathbf{P}_i - X^i|_{s+1}^{s+m} \cdot \mathbf{Q}_i) \cdot \mathbf{W}_i\|_2, \tag{2}$$

where $X^i|_{t+1}^{t+n}$ denotes the row vector $(X_{t+1}^i, \cdots, X_{t+n}^i)$, L is the number of random variables, $\mathbf{P}_i \in \{0,1\}^{n\times l}$ and $\mathbf{Q}_i \in \{0,1\}^{m\times l}$ are two order-subpermutation matrices, and $\mathbf{W}_i$ is the prior cost matrix, which is set as an m-dimensional square matrix in Equation 1 and an l-dimensional square matrix in Equation 2, respectively.

It is worth noting that SOPD is a directed measure that only deforms the first vector $X^i|_{t+1}^{t+n}$, while OPD is undirected. Thus, OPD is a legal pseudo-metric while SOPD is not.

In fact, there have been great efforts on measuring the similarity or discrepancy of two time series [14]. AutoCorrelation Function (ACF) and Partial AutoCorrelation Function stress on the linear similar correlation between two equal-period time series from a statistical perspective [9]. In conjunction with this, dozens of distance measures for discrepancy of time series, such as the Dynamic Time Warping (DTW) [5], Edit Distance on Real sequence (EDR) [10], and Sequence Weighted Alignment model [31], attempt to develop a local (non-linear) alignment, which is more flexible than the simple Euclidean Distance (ED) [17]. However, most of the existing local alignment methods only consider the editability within a fixed period, making the obtained patterns always retain the information of the two endpoints. In other words, the "elasticity" of these measures is reflected through exploring the "elastic" internal information in a fixed window rather than considering the elasticity of window sizes. The order-subpermutation measurement we proposed is more focused on the editability of the effective window, which allows abandoning the first few points of a sequence in order, as shown in Figure 2. In this way, we can achieve the best match between two time series even if sampling is performed in a fixed window. Besides, compared to other elasticity measures, OPD is a legal pseudo-metric. Thus, the order-subpermutation measurement can adapt to harmonic decomposition, which will be shown in the next section.

3.2 Harmonic Estimation

For TSF tasks, we often receive the observed time series $\{x_t, t \in \mathcal{Z}\}$ and aim at predicting the b-step ahead value x_{t+b} or trend $x_{t+b} - x_t$, where $b \in \mathcal{N}^+$. For convenience, we mark the set of instances as $\{(\boldsymbol{x}_t, y_t) : \boldsymbol{x}_t = (x_{t-a+1} \cdots, x_t), y_t = x_{t+b}\}$, and note that $a \in \mathcal{N}^+$ is the feed-up parameter while b is the horizon parameter. In general, suppose that there is a underlying continuous forecasting function f, satisfying $f(\boldsymbol{x}_t) = y_t$. By exploiting harmonic decomposition and order-subpermutation deformation, we have the following estimation conclusion.

Theorem 1 *Concerning the objective time series $\{x_t, t \in \mathcal{Z}\}$, for a finite set $\{\tau_1, \cdots, \tau_m\}$ and t, if there exist certain τ_i and $k \in \mathcal{N}^+$, satisfying $\boldsymbol{\mathcal{D}}_t^{t+\tau_i} = \boldsymbol{\mathcal{D}}_{t+k}^{t+\tau_i+k}$, then there is a harmonic convergence formula with respect to the forecasting function*

$$\hat{y_t} = f_K(\boldsymbol{x}_t) = \sum_{i=1}^{K} \alpha_i \cdot \Phi(\boldsymbol{x}_t, HR_i), \tag{3}$$

where HR_i is the centroid of basic ball B_i, $\alpha_i = f(HR_i)$ is the corresponding harmonic coefficient, Φ measures the similarity between two unequal-period sequences and K is the number of basic balls.

Especially, if the concerned time series belongs to a mixture Gaussian process, the estimation error of the forecasting function f can be bounded by

$$Error(f) \le M \cdot \exp\left(-\frac{\sigma^2}{2(\delta_2 - \delta_1)^2}\right),$$

where σ is the maximum variance of the mixture Gaussian process, $\delta_2 \ge \delta_1 \in \mathcal{R}$, and M is a constant with respect to $K, \sigma, \delta_1, \delta_2$.

Proof Suppose that f is a continuous function that maps from $\mathcal{R}^n$ to $\mathcal{R}$, i.e., $f \in C[\ (\mathcal{R}^n, \mathcal{R}), \|\cdot\|_D\ ]$, where $\|\cdot\|_D$ is a preset legal metric, then there exists a natural harmonic decomposition in the Hilbert space $\mathcal{H}(\mathcal{R}^n, \|\cdot\|_D)$

$$f(\boldsymbol{x}) = \int_{\mathcal{H}} K_0(\|\boldsymbol{x} - \boldsymbol{z}\|_D) f(\boldsymbol{z})\, d\boldsymbol{z}, \tag{4}$$

where the real kernel function K_0 belongs to a Schwartz space $\mathcal{S}(\mathcal{R}, \mathcal{R})$ and is singular near $\boldsymbol{x} = \boldsymbol{z}$ with the norm $\|\cdot\|_D$.

As mentioned in Section 2, the following formula holds:

$$\|(x_1, \cdots, x_i, \cdots, x_n) - (x_1, \cdots, u_i, \cdots, x_n)\|_D = 0,$$

where $u_i = \max\{x_i\}$. Thus, we have

$$\sup_{x_i} |\ f(\boldsymbol{x})\ | \le c_i,$$

implying that f has a compact support in $\mathcal{H}(\mathcal{R}^n, \|\cdot\|_D)$, that is, the stochastic process X_t can be covered by finite landmark points with corresponding basic balls [37]. In other words, the finite instance space of f can be partitioned by two types of basic balls

$$\begin{cases} B^1(\boldsymbol{x}) = \{\boldsymbol{z} : \|\boldsymbol{x} - \boldsymbol{z}\|_D \le \delta_1\}; \\ B^2(\boldsymbol{x}) = \{\boldsymbol{z} : \|\boldsymbol{x} - \boldsymbol{z}\|_D \le \delta_2\}, \end{cases}$$

and a truncation function

$$\tau(\boldsymbol{z}) = \begin{cases} 1 & , \quad \boldsymbol{z} \in B^1(\boldsymbol{x}) \quad ; \\ \exp\left(-\dfrac{\|\boldsymbol{x} - \boldsymbol{z}\|_D}{\delta_2 - \delta_1}\right), & \boldsymbol{z} \in B^2(\boldsymbol{x}) \setminus B^1(\boldsymbol{x}) \, ; \\ 0 & , \quad \boldsymbol{z} \notin B^2(\boldsymbol{x}) \quad . \end{cases}$$

Figure 3 illustrates this procedure in detail. The whole instance space is completely covered by B^1 balls and B^2 balls. The regions covered by B^1 balls have a confidence of 1, while the rest region, i.e., the truncation partition (denoted as $\mathcal{TP}$), is with a diminishing confidence. Let K denote the number of B^1 balls. By gathering all B^1 balls, we

can estimate the forecasting function with confidence 1 as follows:

$$f_K(\boldsymbol{x}) = \int_{\bigcup B^1(\boldsymbol{x})} K_0(\|\boldsymbol{x}-\boldsymbol{z}\|_D) f(\boldsymbol{z})\, d\boldsymbol{z}. \tag{5}$$

Discretizing Equation 4 and 5 as

$$\begin{cases} f(\boldsymbol{x}) = \sum_{i=0}^{\infty} f(\boldsymbol{z}_i) \cdot K_0(\|\boldsymbol{x}-\boldsymbol{z}_i\|_D), \\ f_K(\boldsymbol{x}) = \sum_{i=0}^{K} \alpha_i \cdot \Phi(\boldsymbol{x}, HR_i)\ , \end{cases}$$

where $HR_i = \mathbb{E}[B^1(\boldsymbol{x}_i)]$, $\alpha_i = f(HR_i)$, and $\Phi(\boldsymbol{x}, HR_i) = K_0(\|\boldsymbol{x} - HR_i\|_D)$. Then we can obtain the core formula shown in Equation 3.

If the concerned time series $\{x_t, t \in \mathcal{Z}\}$ belongs to a mixture Gaussian process, the instance $\boldsymbol{z}$ will be distributed over the truncation partition with probability

$$\rho(\boldsymbol{z}) \le \min_{\boldsymbol{x}\in\bigcup B^1(\boldsymbol{x})} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{z}\|_D^2}{2\sigma^2}\right).$$

Abbreviate $\|\boldsymbol{x}-\boldsymbol{z}\|_D$ as D. Then the estimation error can be bounded by the cost of abandoning the truncation partition, i.e.,

$$\begin{aligned} Error(f) &\le |\ \mathbb{E}_{\boldsymbol{x}\in\mathcal{TP}}[f(\boldsymbol{x})]\ | \\ &\le \int_{\bigcup B^2(\boldsymbol{x})\setminus\bigcup B^1(\boldsymbol{x})} \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{z}\|_D^2}{\delta_2-\delta_1}\right)\rho(\boldsymbol{z})\, d\boldsymbol{z} \\ &\le \frac{K}{\sqrt{2\pi}\sigma} \cdot \int_{\delta_1}^{\delta_2} \exp\left(-\frac{D}{\delta_2-\delta_1}\right) \exp\left(-\frac{D^2}{2\sigma^2}\right) dD \\ &\le M \cdot \exp\left(-\frac{\sigma^2}{2(\delta_2-\delta_1)^2}\right), \end{aligned}$$

where $M = K(\delta_2-\delta_1)/(\sqrt{2\pi}\sigma)$.

Besides, the truncation function has various formations. If we adopt the following formula:

$$\tau(\boldsymbol{z}) = \begin{cases} 1 & , \quad \boldsymbol{z} \in B^1(\boldsymbol{x}) \quad ; \\ \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{z}\|_D^2}{\delta_2-\delta_1}\right), & \boldsymbol{z} \in B^2(\boldsymbol{x}) \setminus B^1(\boldsymbol{x})\ ; \\ 0 & , \quad \boldsymbol{z} \notin B^2(\boldsymbol{x}) \quad , \end{cases}$$

then the estimation error can be bounded by

$$Error(f) \le \frac{K(\delta_2-\delta_1)}{\sqrt{2\pi}\sigma} \cdot \exp\left(\frac{(2\sigma^2+\delta_2-\delta_1)\delta_1^2}{2\sigma^2(\delta_2-\delta_1)}\right).$$

□

In Theorem 1, we cover the finite instance space by two types of basic balls $B^1(\boldsymbol{x})$ and $B^2(\boldsymbol{x})$. The samples in the ball $B^1(\boldsymbol{x})$ are absolutely reliable, i.e., recurrent and clearly distinguishable, while the samples between $B^1(\boldsymbol{x})$ and $B^2(\boldsymbol{x})$ are confusing. So we can extract the recurrent patterns of the observed time series by the centroids of B^1-basic balls. When predicting a future value, a natural idea is to match the current input sequence $\boldsymbol{x}_t$ with all the recurrent patterns HR_i, and then make the combination of the harmonic elements α_i associated with the analogous patterns. By exploiting the recurrent elements HR_i and harmonic elements α_i, the mysterious forecasting function f can be formalized into a specific formula, as shown in Equation 3. In the next section, we will calculate these two types of elements, that is, α_i and HR_i by two concrete phases: *recurrent phase* and *harmonic phase*.

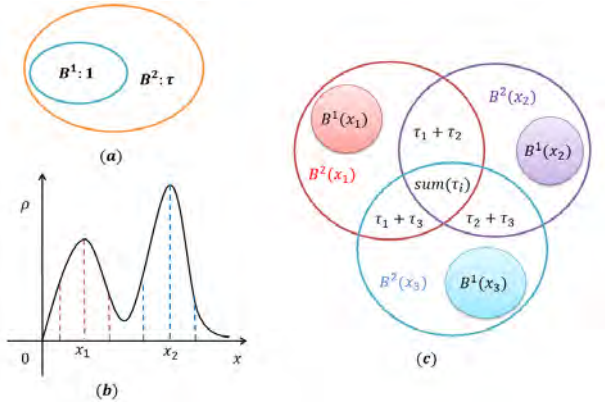

Figure 3. Illustration of truncation coverage and unit decomposition.

3.3 Recurrent Phase

According to $HR_i = \mathbb{E}[B^1(\boldsymbol{x}_i)]$, the recurrent element HR_i can be generated by gathering the similar samples in the basic ball $B^1(\boldsymbol{x}_i)$ and raising a representative one. It is in line with the essence of clustering. In fact, there have been many methods to implement this procedure, such as k-means++ [3], functional subspace clustering [4], and sparse & low-rank decomposition [1].

Here, we develop an improved k-means++ algorithm by incorporating the order-subpermutation metrics SOPD and OPD into k-means++, as described in Algorithm 1. There are two key points in the Order-subpermutation Clustering algorithm. Firstly, the centroids in Step 5 are updated by averaging the order-subpermuted instances, which can achieve the optimal averaging [30]. Secondly, the order-subpermutation metrics here should be used with caution, since improper order-subpermutation transformations of the original sample will cause over-exploitation and lose a lot of necessary information. To avoid this matter, in Step 2, we use OPD to calculate the sampling probability

$$\frac{\min_k OPD(\boldsymbol{x}_i, c_k)}{\sum_i [\ \min_k OPD(\boldsymbol{x}_i, c_k)\ ]}, \tag{6}$$

where c_k denotes the centroid of the k-th cluster and $k < i$. And in Step 4, we employ SOPD to update the centroids of clusters

$$c_k = \frac{\sum_{\boldsymbol{x}_i \in \mathcal{C}_k} \boldsymbol{x}_i \cdot \mathbf{P}_i}{|\mathcal{C}_k|}, \tag{7}$$

where $|\mathcal{C}_k|$ indicates the number of instances in the k-th cluster.

However, the computational complexity of SOPD and OPD is too large if traversing all possible order-subpermutation matrices. To simplify the computation procedure, we can utilize the idea of setting warping window ω in [4], which restricts the search space into a strip area by forcibly limiting that the maximum allowed subpermutation time stamps from point i_1 to the deformation point i_2 is ω. So the computational complexity of traversing $\mathbf{P}$ can be lowered to $\Omega(m\omega)$.

Algorithm 1 Order-subpermutation Clustering

Input: Set of instances $\{\boldsymbol{x}_i\}_{i=1}^N$ and number of clusters K.
Output: Centroids $c_1, \cdots, c_K$ of the clusters $\mathcal{C}_1, \cdots, \mathcal{C}_K$.
Procedure:
1: Arbitrarily choose an initial centroid c_1.
2: Take a new centroid c_k by choosing $\boldsymbol{x}_i$ with the sampling probability, described in Equation 6.
3: Repeat Step 2, until we have taken K centriods altogether.
4: For each instance $\boldsymbol{x}_i$,
find the closest centroid c_k to $\boldsymbol{x}_i$ by $\min_i SOPD(\boldsymbol{x}_i, c_k)$,
record the corresponding optimal order-subpermutation matrix $\mathbf{P}_i$,
and put $\boldsymbol{x}_i$ into the k-th cluster $\mathcal{C}_k$.
5: For each cluster, update its centroid according to Equation 7.
6: Repeat Steps 4 and 5 until $\mathcal{C}_1, \cdots, \mathcal{C}_K$ no longer change.

3.4 Harmonic Phase

In the *harmonic phase*, we are going to estimate the *harmonic element* α_i, which corresponds one-to-one with the recurrent pattern HR_i and is formulated by $\alpha_i = f(HR_i) = f(\mathbb{E}[B^1(\boldsymbol{x}_i)])$.

An intuitive idea is simply to estimate α_i with an average or weighted average, i.e., replace $f(\mathbb{E}[B^1(\boldsymbol{x}_i)])$ with $\mathbb{E}[f(B^1(\boldsymbol{x}_i))]$, which, however, requires that the forecasting function f must be linear. We argue this view and present a global data-based optimization process to calculate α_i. The key idea is that the proper harmonic elements should be able to portray the various harmonic characteristics of the whole data set rather than focusing on a certain cluster. When receiving an instance $\boldsymbol{x}_j$, we dynamically select several closest recurrent elements to $\boldsymbol{x}_j$ by utilizing SOPD and a threshold θ, and then integrate these filtered elements to fit the target value according to the forecasting formula $y_j = \sum_{i=1}^K \alpha_i \, \Phi(\boldsymbol{x}_j, HR_i)$. Thereby, we can obtain the global optimal coefficients $\{\alpha_i\}_{i=1}^K$ by the following two formal optimization issues.

For a certain threshold θ, we find the optimal $\{\alpha_i\}_{i=1}^K$ as follows:

$$\begin{aligned} &\boldsymbol{\alpha}^*(\theta) = \arg\min_{\boldsymbol{\alpha}} \sum_{j=1}^{n} \left(\sum_{i=1}^{K} \alpha_i \boldsymbol{\Psi}_{ij} - y_j \right)^2 \\ &s.t. \;\; \boldsymbol{\Psi}_{ij} = \begin{cases} \Phi(\boldsymbol{x}_j, HR_i), \; if \;\; e^{-SOPD(\boldsymbol{x}_j, HR_i)} \leq \theta; \\ 0 \qquad , \; otherwise. \end{cases} \end{aligned} \tag{8}$$

The other optimization issue is related to θ

$$\theta^* = \arg\min_{\theta} Loss(\theta), \;\; s.t. \;\; \theta \in [0, 1), \tag{9}$$

where

$$Loss(\theta) = \sum_{j=1}^{n} \left(\sum_{i=1}^{K} \alpha_i^*(\theta) \boldsymbol{\Psi}_{ij} - y_j \right)^2.$$

Analogous to least squares estimation, Equation 8 has a closed-form solution: $\boldsymbol{\alpha}^T = (\boldsymbol{\Psi} \cdot \boldsymbol{\Psi}^T)^{-1} \cdot \boldsymbol{\Psi} \cdot \boldsymbol{y}^T$, where $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_K)$ and $\boldsymbol{y} = (y_1, \cdots, y_n)$. To avoid the singular solutions, we sometimes need to add a regularizer to Equation 8. Besides, Equation 9 can be efficiently solved by the Bayesian optimization algorithm [36].

4 Theoretical Analysis

So far, we have discussed the structure of the finite-dimensional distribution families of HRP and present a concrete implementation. In this section, we are going to investigate the predictability of HRP.

Similar to most predictable theoretical results [26, 27], we need to constrain the dependence between random variables. In order to characterize this problem more accurately, we introduce the Copula coefficient according to Sklar's theorem [25].

Lemma 1 *Given a fixed t, there is a unique constant $C_t(k)$ satisfying the following equation:*

$$\boldsymbol{\mathcal{D}}_t^{t+\tau_i} \wedge \boldsymbol{\mathcal{D}}_{t+\tau_i+k}^{t+\tau_i+k+\tau_j} = C_t(k) \cdot \boldsymbol{\mathcal{D}}_t^{t+\tau_i} \cdot \boldsymbol{\mathcal{D}}_{t+\tau_i+k}^{t+\tau_i+k+\tau_j},$$

where for any non-negative integers τ_i, τ_j, $\boldsymbol{\mathcal{D}}_t^{t+\tau_i} \wedge \boldsymbol{\mathcal{D}}_{t+\tau_i+k}^{t+\tau_i+k+\tau_j}$ denotes the joint distribution of $\boldsymbol{X}|_t^{t+\tau_i}$ and $\boldsymbol{X}|_{t+\tau_i+k}^{t+\tau_i+k+\tau_j}$. Here, $C_t(k)$ is said to a Copula coefficient.

Obviously, the Copula coefficient, as a tool for characterizing the interdependence mechanism between variables, contains almost all the dependent information of random variables (that may be non-adjacent). For example, the case that for any t and k, $C_t(k) = 1$, means that random variables are independent of each other. Thus, the Copula coefficient is useful for indicating the correlation between variables without calculating autocorrelation coefficients [32]. For most realistic cases, the futures of a concerned time series would have a sufficiently weak dependence on the distant past [2], that is, $\sup_t C_t(k) \to 0$ as $k \to +\infty$. With this prior, we can claim that the stochastic process led by our HRP is predictive PAC learnable [35].

Theorem 2 *The HRP with weak dependence condition is predictive PAC learnable. In detail, let $a = \max\{\tau_1, \cdots, \tau_m\}$, and $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ denote the sample set. For any $\delta' = \delta - \lfloor \frac{N}{k} \rfloor \beta(k) > 0$, with probability $1 - \delta$, the following holds for all hypotheses $\hbar \in H$:*

$$\mathbb{E}(\hbar) \leq \hat{\mathbb{E}}(\hbar) + \frac{2}{k} \sum_{i=1}^{2k} \mathfrak{R}_p(\mathcal{X}_i) + \frac{2}{N} \sum_{i=1}^{N} d(s_i, t+b) + C\sqrt{\frac{\log \frac{2}{\delta'}}{8p}}, \tag{10}$$

where $\mathbb{E}(\hbar)$ is the generalization error, $\hat{\mathbb{E}}(\hbar)$ denotes the empirical error, C indicates the upper bound of the loss function and $\{s_i\}$ represents a series of time stamps where $s_N = t$. Specially, $\{\mathcal{X}_i\}$ is a partition of the training set that $\{\boldsymbol{x}_{i(p-1)+1}, \boldsymbol{x}_{i(p-1)+2}, \cdots, \boldsymbol{x}_{ip}\}$, where $p = \lceil \frac{N}{k} \rceil$, so the second term is summed by the Rademacher complexity of each subset

$$\mathfrak{R}_p(\mathcal{X}_i) = \frac{1}{p} \, \mathbb{E}\left[\sup_{\hbar \in H} \sum_{j : \boldsymbol{x}_j \in \mathcal{X}_i} \sigma_j \; loss(\hbar(\boldsymbol{x}_j), y_j) \right],$$

where σ_j is the Rademacher random variable and $loss(\hbar(\boldsymbol{x}_j), y_j)$ is the prediction error. And the third term measures the discrepancy between two probability distributions

$$d(s, t) = \sup_{\hbar \in H} |L_s(\hbar) - L_t(\hbar)|,$$

where $L_s(\hbar) = \mathbb{E}_s \left[loss(\hbar(\boldsymbol{x}_s), y_s) | \boldsymbol{X}_{s-a+1}^s \right]$ denotes the path-dependent generalization error.

Theorem 2 reveals an important conclusion that the temporal length of effective recurrent patterns are between 1 and the maximum period $\max\{\tau_1, \cdots, \tau_m\}$. So when generating instances, we can simply set the feed-up parameter a equal to this maximum period.

Proof Here, we highlight the proof logic flows of Lemma 1 and Theorem 2 below. Sklar's theorem [25] states that any multivariate joint distribution can be formalized as a combination function of univariate marginal distributions and a Copula. So the Copula coefficient of the stochastic process can be formulated lucidly by a time-dependent function $C_t(k)$. If $\sup_t C_t(k) \to 0$ as $k \to +\infty$, the concerned time

series meet the weak dependence condition that the futures have a sufficiently weak dependence on the distant past.

Then we can deduce that the stochastic process with weak dependence condition is β-mixing [15] with

$$\beta(k) = \sup_{t} \mathbb{E}_{\boldsymbol{X}_{-\infty}^{t}} \left[\| \boldsymbol{\mathcal{D}}_{t+k}^{+\infty}(\cdot|\boldsymbol{X}_{-\infty}^{t}) - \boldsymbol{\mathcal{D}}_{t+k}^{+\infty}(\cdot) \|_{\infty} \right],$$

where $\boldsymbol{\mathcal{D}}(\cdot|\cdot)$ stands for conditional probability measure. The β-mixing coefficients satisfy

$$\beta(k) \to 0, \ \text{ as } \ k \to +\infty.$$

According to the generalization bound results of the mixing process in [35, 26], we can conclude that our proposed HRP is predictive PAC learnable and the generalization error can be bounded by

$$\mathbb{E}(\hbar) \leq \hat{\mathbb{E}}(\hbar) + \frac{2}{k} \sum_{i=1}^{2k} \mathfrak{R}_p(\mathcal{X}_i) + \frac{2}{N} \sum_{i=1}^{N} d(s_i, t+b) + C\sqrt{\frac{\log \frac{2}{\delta'}}{8p}}.$$

□

5 Experiments

In this section, we are going to examine the practical performance of the proposed HRP in a handful of experiments. In Section 5.1, we evaluate the effectiveness of the introduced order-subpermutation metrics and analyze the sensitivity of hyperparameters K and w, through a simulated task of forecasting the coupling consine signals. In Sections 5.2 and 5.3, we compare HRP with some state-of-the-art TSF models on two real-world data sets, Yancheng Automobile Registration and CSI 300, respectively.

5.1 Coupling Cosine Signals

We first consider the *Coupling Cosine Signals Forecasting* task, which performs a kind of typical nonstationary time series with period-varying patterns. A simpler version of the task was proposed in [29]. Here, we generate a coupling cosine signal train within 1000 input points by exploiting different cos functions with period 5-9. As mentioned in Theorem 2, we fix the maximum period parameter $a = 9$. In the following experimental procedure, we feed up HRP with the first 900 signals and forecast the future 100 values of coupling cosine.

The purposes that we conduct experiments on coupling cosine signals are twofold: (1) to exhibit the advantages of our proposed order-subpermutation measures against other elasticity measures; (2) to evaluate the parameters sensitivity and configurations of our HRP model. Therefore, we first investigate the performance of HRP with $w = 1$ by employing various measures in the recurrent phase. The experimental results are shown in Table 1. In the cases that take 30, 40, or 50 clusters, our proposed order-subpermutation measures are able to perform better, achieving the best mean squared error (MSE) in contrast to ACF, ED, EDR, 1-DTW, and FWS [4].

Table 1. MSE of HRP with different measures for the task of forecasting coupling cosine signals.

Clusters	ACF	ED	EDR	1-DTW	FWS	SOPD / OPD
K=30	0.3960	0.4934	0.3481	0.4946	0.3432	**0.3430**
K=40	0.4109	0.6264	0.3689	0.3996	0.3211	**0.3146**
K=50	0.3399	0.5494	0.3319	0.4016	0.3081	**0.3074**
K=60	**0.3634**	0.5037	0.3723	0.4530	0.4413	0.4326

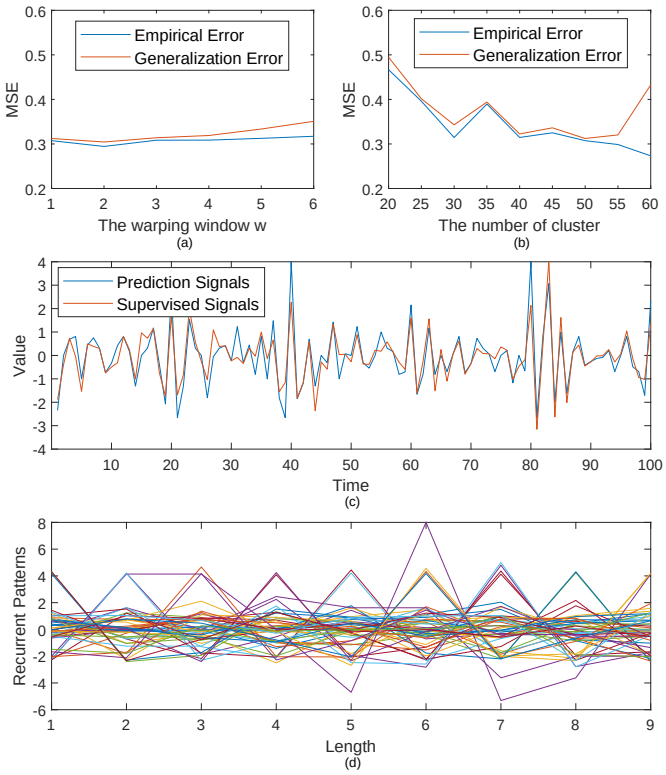

Figure 4. HRP on Coupling Cosine Signals. (a)-(b) Prediction raster plots with respect to the number K of clusters and the warping window w, respectively; (c) Snapshot of HRP forecasting signals with $K = 50$ and $w = 1$; (d) Recurrent patterns of HRP with $K = 50$ and $w = 1$.

The second testing is to demonstrate parameters sensitivity and to determine the best configuration of our HRP model. Figure 4(a) and 4(b) plot the empirical and generalization errors of HRP with various warping window w and the number K of clusters from 20 to 60, respectively. From the prediction raster plots, we can observe that the MSE curves are almost positively proportional to w. In practice, we suggest that w should be between 10% and 20% of the window size. Besides, the better performance can be obtained when K is sampled from the interval [40, 50]. Thereby, HRP is robust to both the number K of clusters and the warping window w.

Furthermore, we displayed the forecasting snapshot of HRP with $K = 50$ and $w = 1$ to indicate how the errors are constructed. The comparison diagrams of the supervised signals (red) and the predicted signals (blue) are shown in Figure 4(c). And the corresponding 50 recurrent patterns are exhibited in Figure 4(d). From these charts, we can find out that our proposed HRP has a good predictive performance.

5.2 Yancheng Automobile Registration

Next, we conduct experiments on the *Yancheng Automobile Registration Forecasting* task [2], which comprises registration records of 5 car brands in nearly 1000 dates. To feed up HRP and other competing approaches with a reasonable setting, we sample 5 consecutive points as an input instance to predict the one-step-ahead increment. In particular, missing points and outliers in this data set are not preprocessed. Additionally, we utilize the first 800 instances to constitute the training set, and the testing set consists of follow-up 100 instances.

[2] https://tianchi.aliyun.com/competition/entrance/231641/information

Table 2. MSE and setting of comparative models for the task of forecasting Yancheng Automobile Registration data.

Types	Models	Settings	MSE (10^5)
Stastical Models	ARIMA	$(p, d, q) = (6, 1, 3)$	84.5129
	MAR [39]	–	92.6458
	AGP [7]	–	41.0147
	KNNs [40]	$(K, w) = (1, 1)$	31.2573
Evolution Models	NARXnet [20]	$\diamond$	28.0549
	LSTM	$\diamond$	15.2490
	GRU [12]	$\diamond$	13.0421
	LSTNet [28]	$\diamond$	10.2070
Our Work	HRP_DTW	$(K, w) = (1, 1)$	9.2737
	HRP	$(K, w) = (1, 1)$	**6.6460**

$^\diamond$ indicates the setting of neural networks, that is, the input size = 7, the hidden size = 50, employing 32 convolution, and running 100 Epochs.

Table 2 lists the comparative results of nine competing approaches and HRP with $K = 50$ and $w = 1$. Note that HRP_DTW denotes the HRP model using DTW as a deformation measure instead of SOPD and OPD. Based on these results, we find that (1) the proposed HRP achieves the best performance, surpassing the competing approaches including HRP_DTW; (2) deep learning approaches (i.e., NARXnet, LSTM, GRU, and LSTNet) are superior to statistical methods (i.e, ARIMA, MAR, AGP, and KNNs), which is consistent with that observed in the previous works [12, 28].

5.3 CSI 300

The last experiment is run on the minute-level stock market prices of the CSI 300 Index, which are accessed from Tushare Pro[3]. Empirically, we raise the price series per morning as the input signals, and correspondingly regard the log return of per afternoon as the target value. Additionally, we set the warping window $w = 10$, since the dimension of input sequence is much larger (more than 100 dimensions). In a bid to balance the ratio of rise and fall, we set the testing data set drawn from February 21st, 2013 to May 20th, 2014, just 100 days, and the training data set consists of the price data from April 25th, 2005 to February 20th, 2013. Finally, we add the evaluation indicator, *Confusion Accuracy*, which consists of *True Positive Rate* (TPR) and *True Negative Rate* (TNR).

Table 3. MSE and confusion accuracy of NARXnet, LSTM, GRU, LSTNet, and HRP ($K = 50, w = 10$) for the task of forecasting CSI 300 Index data.

Models	MSE (10^{-5})	Accuracy (%)	TPR (%)	TNR (%)
NARXnet	19.16	60.00	73.68	41.86
LSTM	20.26	57.00	63.16	48.84
GRU	17.22	62.00	75.44	44.19
LSTNet	12.70	65.00	71.93	53.49
HRP (our work)	**9.80**	**68.00**	**77.19**	**55.81**

Table 3 displays the MSE and confusion accuracy of NARXnet, LSTM, GRU, LSTNet, and HRP on the CSI 300 Index data set. We do not list the performance of some conventional or statistical methods due to their poor achievements in such a complicated case. The results show that HRP can surpass other competing models in both effectiveness and accuracy.

We also conduct an extensive experiment for stocks selection in the CSI 300 Constituent Stocks pooling, containing 400 stocks with minute-level stock market prices. Analogous to the aforementioned setting, we enter the price series per minute in the morning to predict the log return in the afternoon, and divide the whole data set into the training part from January 5th, 2015 to June 22nd, 2017 and the testing part from June 23rd, 2017 to November 16th, 2017.

[3] http://tushare.org/index.html

Table 4. Performance of HRP on a portion of CSI 300 constituent stocks.

Stock Codes	MSE (10^{-3})	TPR (%)	TNR (%)	PR / NR
000926 SHE	6.72	92.17	40.41	2.00
001696 SHE	4.80	66.56	53.73	0.87
002155 SHE	9.30	56.41	68.23	1.14
600300 SHG	4.81	52.33	61.21	1.39
600635 SHG	10.20	57.72	61.80	1.17
600863 SHG	8.40	73.41	45.56	1.41
000006 SHE	8.34	61.32	40.22	0.88
000519 SHE	27.57	70.61	32.93	0.70
600060 SHG	23.30	58.81	36.40	1.11
600141 SHG	120.21	36.27	57.11	1.27

Table 4 lists the MSE and confusion accuracy of the proposed HRP on a portion of representative CSI 300 constituent stocks, where PR / NR indicates the ratio of rise and fall of the log returns on the testing data set. At first glance, HRP does not perform well on all stocks. For example, the stock 600141 SHG has larger MSE and lower confusion accuracy. The imperfect reasons for the under-performing stocks may be twofold: the number of training instances is not large enough to identify the recurrent patterns, or the potential distributions are inherently fickle, which are difficult to predict. However, HRP still holds an effective performance on some stocks in both MSE and confusion accuracy. Particularly, the TPR of stock 000926 SHE even reached 92.17%. We clearly distinguish between the well-performing and under-performing stocks using a spacer line in Table 4.

6 Conclusion

In this paper, we present the HRP for nonstationary TSF. HRP is adept in exploring the recurrent order-subpermutation patterns of time series with harmonic decomposition, and thus, can be applied to more realistic and complex situations. We give an effective implementation of HRP, and also prove that the stochastic process led by HRP under weak dependence condition is predictive PAC learnable. The empirical studies conducted on several tasks confirm the effectiveness of HRP.

Acknowledgment

This research was supported by the National Key R&D Program of China (2018YFB1004300) and NSFC (61921006). The authors would like to thank the anonymous reviewers for constructive suggestions, as well as De-Chuan Zhan and Chao Qian for helpful discussions.

References

[1] Mahdi Abavisani and Vishal M Patel, 'Multimodal sparse and low-rank subspace clustering', *Information Fusion*, **39**, 168–177, (2018).
[2] Pierre Alquier and Olivier Wintenberger, 'Model selection for weakly dependent time series forecasting', *Bernoulli*, **18**(3), 883–913, (2012).
[3] David Arthur and Sergei Vassilvitskii, 'K-means++: The advantages of careful seeding', in *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1027–1035, (2007).
[4] Mohammad Taha Bahadori, David Kale, Yingying Fan, and Yan Liu, 'Functional subspace clustering with application to time series', in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pp. 228–237, (2015).

[5] Donald J Berndt and James Clifford, 'Using dynamic time warping to find patterns in time series', in *Proceedings of the 1994 AAAI Workshop on Knowledge Discovery in Databases (KDD Workshop)*, pp. 359–370, (1994).

[6] Filippo Maria Bianchi, Enrico Maiorino, Michael C Kampffmeyer, Antonello Rizzi, and Robert Jenssen, 'An overview and comparative analysis of recurrent neural networks for short term load forecasting', *arXiv:1705.04378*, (2017).

[7] Maarten Blaauw and J Andrés Christen, 'Flexible paleoclimate age-depth models using an autoregressive gamma process', *Bayesian Analysis*, **6**(3), 457–474, (2011).

[8] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung, *Time Series Analysis: Forecasting and Control*, John Wiley & Sons, 2015.

[9] Peter J Brockwell, Richard A Davis, and Stephen E Fienberg, *Time Series: Theory and Methods*, Springer, 1991.

[10] Lei Chen, M Tamer Özsu, and Vincent Oria, 'Robust and fast similarity search for moving object trajectories', in *Proceedings of the 2005 International Conference on Management of Data (SIGMOD)*, pp. 491–502, (2005).

[11] Changqing Cheng, Akkarapol Sa-Ngasoongsong, Omer Beyca, Trung Le, Hui Yang, Zhenyu Kong, and Satish TS Bukkapatnam, 'Time series forecasting for nonlinear and non-stationary processes: A review and comparative study', *Iie Transactions*, **47**(10), 1053–1071, (2015).

[12] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, 'Learning phrase representations using RNN encoder–decoder for statistical machine translation', in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, (2014).

[13] Jan G De Gooijer and Rob J Hyndman, '25 years of time series forecasting', *International Journal of Forecasting*, **22**(3), 443–473, (2006).

[14] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn Keogh, 'Querying and mining of time series data: Experimental comparison of representations and distance measures', in *Proceedings of the 34th International Conference on Very Large Data Bases (VLDB)*, pp. 1542–1552, (2008).

[15] Paul Doukhan, *Mixing: Properties and Examples*, Springer, 2012.

[16] James B Elsner and Anastasios A Tsonis, *Singular Spectrum Analysis: A New Tool in Time Series Analysis*, Springer, 2013.

[17] Christos Faloutsos, Mudumbai Ranganathan, and Yannis Manolopoulos, 'Fast subsequence matching in time-series databases', *ACM SIGMOD Record*, **23**(2), 419–429, (1994).

[18] Piotr Fryzlewicz, Sébastien Van Bellegem, and Rainer Von Sachs, 'Forecasting non-stationary time series by wavelet process modelling', *Annals of the Institute of Statistical Mathematics*, **55**(4), 737–764, (2003).

[19] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley, 'Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals', *Circulation*, **101**(23), 215–220, (2000).

[20] Sandra M Guzman, Joel O Paz, and Mary Love M Tagert, 'The use of NARX neural networks to forecast daily groundwater levels', *Water Resources Management*, **31**(5), 1591–1603, (2017).

[21] Steven Haker, Zhu Lei, Allen Tannenbaum, and Sigurd Angenent, 'Optimal mass transport for registration and warping', *International Journal of Computer Vision*, **60**(3), 225–240, (2004).

[22] Jan S Hesthaven, Sigal Gottlieb, and David Gottlieb, *Spectral Methods for Time-dependent Problems*, Cambridge University Press, 2007.

[23] Ali Jalali and Sujay Sanghavi, 'Learning the dependence graph of time series with latent factors', *arXiv:1106.1887*, (2011).

[24] Yonghong Jiang, He Nie, and Weihua Ruan, 'Time-varying long-term memory in Bitcoin market', *Finance Research Letters*, **25**, 280–284, (2018).

[25] Harry Joe, *Multivariate Models and Multivariate Dependence Concepts*, Chapman and Hall/CRC, 1997.

[26] Vitaly Kuznetsov and Mehryar Mohri, 'Generalization bounds for time series prediction with non-stationary processes', in *Proceedings of the 25th International Conference on Algorithmic Learning Theory (ALT)*, pp. 260–274, (2014).

[27] Vitaly Kuznetsov and Mehryar Mohri, 'Learning theory and algorithms for forecasting non-stationary time series', in *Advances in Neural Information Processing Systems 28 (NIPS)*, 541–549, (2015).

[28] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu, 'Modeling long-and short-term temporal patterns with deep neural networks', in *Proceedings of the 41st International Conference on Research & Development in Information Retrieval (SIGIR)*, pp. 95–104, (2018).

[29] Lei Li, Nian Fang, Lutang Wang, and Zhaoming Huang, 'Modeling of the multiple superimposed oscillator problem using linear echo state network with leaky-integrator neurons', in *Proceedings of the 6th International Conference on Information Engineering (ICIE)*, pp. 1–5, (2017).

[30] Warissara Meesrikamolkul, Vit Niennattrakul, and Chotirat Ann Ratanamahatana, 'Shape-based clustering for time series data', in *Proceedings of the 16th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pp. 530–541, (2012).

[31] Michael D Morse and Jignesh M Patel, 'An efficient and accurate method for evaluating time series similarity', in *Proceedings of the 2007 International Conference on Management of Data (SIGMOD)*, pp. 569–580, (2007).

[32] Roger B Nelsen, *An Introduction to Copulas*, Springer, 2007.

[33] Lawrence R Rabiner, 'A tutorial on hidden markov models and selected applications in speech recognition', *Proceedings of the IEEE*, **77**(2), 257–286, (1989).

[34] Louis L Scharf and Cédric Demeure, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*, Addison Wesley, 1991.

[35] Cosma Shalizi and Aryeh Kontorovich, 'Predictive PAC learning and process decompositions', in *Advances in Neural Information Processing Systems 26 (NIPS)*, pp. 1619–1627, (2013).

[36] Jasper Snoek, Hugo Larochelle, and Ryan P Adams, 'Practical Bayesian optimization of machine learning algorithms', in *Advances in Neural Information Processing Systems 25 (NIPS)*, pp. 2951–2959, (2012).

[37] Elias M Stein, *Harmonic Analysis: Real-Variable Methods, Orthogonality, and Oscillatory Integrals*, Princeton University Press, 2016.

[38] Andreas S Weigend, *Time Series Prediction: Forecasting the Future and Understanding the Past*, Routledge, 2018.

[39] Chee Sun Won and Robert M Gray, *Stochastic Image Processing*, Springer, 2013.

[40] Hui Yang, Satish TS Bukkapatnam, and Leandro G Barajas, 'Local recurrence based performance prediction and prognostics in the nonlinear and nonstationary systems', *Pattern Recognition*, **44**(8), 1834–1840, (2011).

[41] Hsiang-Fu Yu, Nikhil Rao, and Inderjit S Dhillon, 'Temporal regularized matrix factorization for high-dimensional time series prediction', in *Advances in Neural Information Processing Systems 29 (NIPS)*, pp. 847–855, (2016).