Multivariate Time Series Prediction with PID-based Residual Compensation

Jinxin Liu, Qiangxing Tian and Donglin Wang†
School of Engineering, Westlake University, Hangzhou, 310024, China
Institute of Advanced Technology, Westlake Institute for Advanced Study, Hangzhou, 310024, China
{liujinxin, tianqiangxing, wangdonglin}@westlake.edu.cn

***Abstract*—Multivariate time series prediction has fundamental importance to various practical domains. Many useful techniques have been proposed in literature for improving the accuracy and efficiency of prediction. However, due to the complicated dependency between time stamps and the interactive dependency between multiple time series, it is very difficult to pursue a high accuracy and alleviate the prediction error. In this paper, we propose a residual compensation scheme for multivariate time series prediction, where the prediction error is modeled by a PID-based residual network (PID-R) to cover various linear, cumulative and differential effects. To evaluate the effectiveness of the residual compensation, a hybrid structure integrating the Seq2Seq model with the classic vector auto-regression (VAR) is built as an initial predictor. The proposed final predictor with residual compensation (FPRC) incorporates the initial predictor and PID-R. Extensive experiments show that the FPRC achieves superior accuracy in comparison to state-of-the-art methods and ablation analysis demonstrates the effectiveness of residual compensation.**

***Index Terms*—time series, prediction, residual model**

I. Introduction

As is well known, time series modeling and prediction is of indispensable significance to numerous practical fields, such as business, finance, economics, climate, society, science and engineering [1]–[6]. Thus a lot of active research works during the past few years are still going on in this subject. Many important models have been proposed in literature to improve the accuracy and efficiency of time series modeling and prediction [7]. These prominent methods can be largely categorized into two classes: statistical models based prediction and deep network based prediction. Statistical models assume that time series follows certain function form like a linear or quadratic function [8] while deep network based prediction constructs a model for mimicking the intelligence of human brain and attempts to recognize regularities and patterns from the past data [9], [10].

In general, statistical models for time series data may have many forms and represent different stochastic processes [7]. By employing traditional statistical analysis, a variety of methods have been proposed for time series prediction, containing both linear and nonlinear techniques. Linear statistical models primarily consist of auto-regression (AR), moving average (MA) and their variants [8], [11]. Auto-regressive integrated moving averages (ARIMA) is one of the most popular and effective method, which works well for stationary univariate series [12]. Seasonal ARIMA (SARIMA) further considers the seasonal effect in the sales dataset [13]. Vector auto-regression (VAR) considers the dependency among time series while ARIMA does not [14], [15]. VAR addressed the high computational cost problem faced by ARIMA in high dimensional time series prediction ([11], [16]). Nonlinear statistical models primarily consist of the non-linear autoregressive model, the nonlinear moving average model, kernel methods, ensemble methods, and Gaussian processes [7]. The drawback is that most of these approaches employ a predefined nonlinear form and may not be able to capture the true underlying nonlinear relationship appropriately [8], [12].

Deep learning gives rise to a prosperity of time series modeling and prediction. Recurrent neural network (RNN) is known as a connectionist model able to capture the sequential nonlinear dynamics via node cycles while suffering from the problem of vanishing gradient [17]. Long short-term memory network (LSTM) and and the gated recurrent unit (GRU) have overcome this limitation and achieved great success in various applications [18]. Seq2Seq is a popular method for time series modeling and prediction [17], [19]. Convolutional neural network (CNN) is a critical alternative for univariate or multivariate time series prediction [20]. Recent temporal convolutional network (TCN) uses a hierarchy of temporal convolutions to capture features for further prediction [21]. The attention-based encoder-decoder network employs an attention mechanism to adapt a longer input sequence [22]–[24]. Therefore, it is natural to consider state-of-the-art deep network as a starting baseline for time series prediction.

By combining statistical analysis and deep network, hybrid models are invented to improve the prediction accuracy. The prior study in [25] considers a coupling of full-connected neural network and the traditional ARIMA. LSTNet-skip uses the CNN and RNN to extract short-term local dependency patterns and the long-term trend, and uses the AR model to tackle the scale insensitivity [26]. On the basis of LSTNet-skip, LSTNet-Attn leverages the attention mechanism to capture the long-term dependency of generated features at the convolutional layer [26]. Wavelet-Trans feeds the scalgram and temporal signal into CNN and RNN respectively, and fuse them using a multi-layer perceptron (MLP) [27]. Although statistical analysis, deep learning and hybrid models show the benefit for many practical forecasting problems, it is very difficult to

†Corresponding author.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

achieve a precise prediction in complex environments due to the highly dynamics in both global and local aspects.

Most recently, a couple of research works consider to estimate the residual error and use it as a compensation for time series prediction [19], [28], [29]. Residual recurrent neural network (R2N2) considers the traditional AR model as the initial predictor and uses the RNN to estimate the residual error [28]. Dest-ResNet in [19] is designed and dedicated for the traffic speed prediction, where a Seq2Seq-based residual network is introduced to alleviate the difference between the predicted traffic speed and the groundtruth speed. As concluded, the compensation for the residual error greatly improves the accuracy of time series prediction.

In this paper, we propose a novel proportion-integration-differentiation (PID) based residual network (PID-R) to compensate multivariate time series prediction, where the inner PID module and MLP are combined to calculate the residual error. By considering a variety of linear effect, cumulative effect and differential effect of the prediction error at different time stamps, the residual network is designed to reduce the difference between the groundtruth and the initial prediction. Furthermore, we combine the most general Seq2Seq and the traditional VAR as the initial predictor, different from [28] only considering the AR. Thus, the initial predictor introduced in this paper could be easily extended or transferred to other advanced hybrid models without obstacle. On the other hand, the proposed PID-based residual network is capable of fitting well for a general initial predictor. By incorporating the initial predictor and PID-R, the proposed final predictor with residual compensation (FPRC) is demonstrated by real experiments to be superior to state-of-the-art methods. Please be noted that, different from the piror work [19], [28], the final prediction in FPRC is further fed back into both the initial predictor and the residual network for the next prediction.

The main contribution of this paper can be summarized as

- A novel PID-based residual network named PID-R is proposed to compensate and reduce the predict error.
- A novel initial predictor is proposed by combining the Seq2Seq model and traditional VAR, which could be incorporated in the final predictor.
- By combining the initial predictor and the PID-R, we propose the final predictor, where the prediction at t is fed back for the prediction at $t+1$.
- Extensive experiments demonstrate the superiority of our proposed method in comparison to state-of-the-art approaches.

The remaining of this paper is organized as follows. Section II introduces the problem definition and the overview of our scheme. The methodology is detailed in Section III, including the FPRC, the initial predictor and the PID-R. Experiment results on three real datasets are reported in Section IV, followed by Conclusion in Section V.

II. Problem Definition and Overview

A. Problem Definition

Consider a multivariate time series $\mathbf{X}_T$, which consists of m time series and T past time stamps. Define $\mathbf{X}_T = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T) \in \mathbb{R}^{m\times T}$, where $\mathbf{x}_k = \left(x_k^1, x_k^2, ..., x_k^m\right)^{\mathrm{T}}$ denotes the multivariate series at time stamp k. The task of prediction is to provide an accurate estimation for $\{\mathbf{x}_{T+1}, \mathbf{x}_{T+2}, \cdots, \mathbf{x}_{T+H}\}$ based on the known past data $\mathbf{X}_T$.

Without loss of generality, the task of prediction is based on the assumption that the value of future samples in time series is related to that of the past samples and the test data follows the same distribution as the training data [8]. By exploiting the feature representation from the past samples of multivariate time series, we are able to build a proper model to predict the future samples. For convenience, we define $\mathbf{Y}_{T+H} \triangleq (\mathbf{y}_{T+1}, \mathbf{y}_{T+2}, \cdots, \mathbf{y}_{T+H})$, where each $\mathbf{y}_{T+t}$ denotes the estimation of the ground truth $\mathbf{x}_{T+t}$. In other words, we aim to find the predictor as

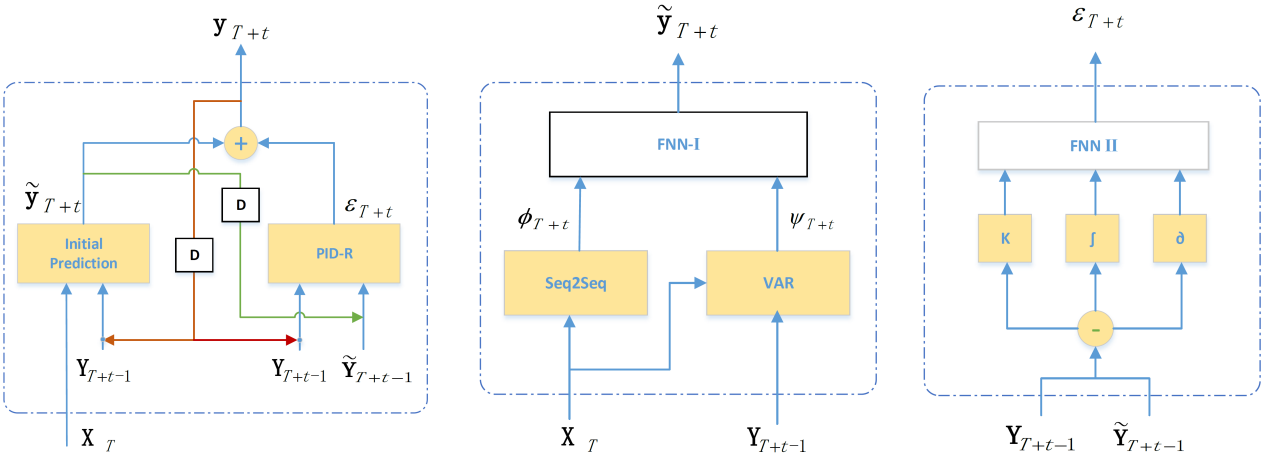
$$\mathbf{y}_{T+t} = f(\mathbf{X}_T, \mathbf{y}_{T+t-1}), \quad 1 \le t \le H \tag{1}$$

where $f(\cdot)$ denotes the mapping function from $\mathbf{X}_T$ to $\mathbf{Y}_{T+H}$, i.e. rolling prediction of H samples at once.

B. Overview of Scheme

1) Final Prediction with Residual Compensation (FPRC): Fig. 1(a) shows the overall prediction scheme with residual compensation named FPRC, which incorporates a proposed initial prediction and a residual network PID-R that will be respectively described as below. The initial predictor takes $\mathbf{X}_T$ and the delayed final prediction $\mathbf{Y}_{T+t-1}$ as inputs and generates the prediction $\tilde{\mathbf{Y}}_{T+t}$. The PID-R takes the delayed initial prediction $\tilde{\mathbf{Y}}_{T+t-1}$ and the delayed final prediction $\mathbf{Y}_{T+t-1}$ as inputs and generates the current residual error ε_{T+t}. The final prediction in the FPRC is the summation of the initial prediction $\tilde{\mathbf{y}}_{T+t}$ and the current residual error ε_{T+t}. Please be noted that the final prediction $\mathbf{Y}_{T+t}$ is delayed and fed back into both the initial prediction and the PID-R for more precise prediction.

2) Initial Predictor: The initial predictor is proposed for evaluating the effectiveness of the FPRC. Fig. 1(b) illustrates the block diagram of our proposed initial predictor, which is one kind of hybrid models. As shown in Fig. 1(b), the initial predictor combines the LSTM-based Seq2Seq model and the VAR model, where the Seq2Seq module takes $\mathbf{X}_T$ as input and generates a basic prediction ϕ_{T+t}, and the VAR module takes $\mathbf{X}_T$ and the delayed final prediction $\mathbf{Y}_{T+t-1}$ as its inputs and generates its own basic prediction ψ_{T+t}. Please be noted that, when the initial predictor is utilized independently for ablation analysis, the feedback is thus assigned by $\tilde{\mathbf{Y}}_{T+t-1} = (\tilde{\mathbf{y}}_{T+1}, \tilde{\mathbf{y}}_{T+2}, \cdots, \tilde{\mathbf{y}}_{T+t-1})$ instead of $\mathbf{Y}_{T+t-1}$ in FPRC. The feedforward neural network (FNN-I) is employed to combine ϕ_{T+t} and ψ_{T+t} to generate the initial hybrid prediction $\tilde{\mathbf{y}}_{T+t}$. Last but not least, the initial predictor introduced in this paper could be easily extended or transferred to other advanced hybrid models without obstacle.



(a) FPRC. (b) Initial predictor. (c) Residual estimator PID-R.

Fig. 1. Block diagram of the proposed models.

3) The PID-R: Fig. 1(c) shows the proposed PID-R, which is designed to evaluate the current residual error ε_{T+t}. Firstly, PID-R takes the delayed initial hybrid prediction $\tilde{\mathbf{Y}}_{T+t-1}$ and the delayed final prediction $\mathbf{Y}_{T+t-1}$ as its inputs and calculates the historical residual errors $\{\varepsilon_{T+1}, \varepsilon_{T+2}, ..., \varepsilon_{T+t-1}\}$. Then the residual errors are processed by using PID modules for covering a variety of linear effect, cumulative effect and differential effect. As shown in Fig. 1(c), all historical residual errors $(\varepsilon_{T+1}, \varepsilon_{T+2}, ..., \varepsilon_{T+t-1})$ are passed through the *Proportion* unit K, the *Integration* unit $\int$ and the *Differentiation* unit ∂, respectively. The resulting outputs are concatenated and processed by a MLP called FNN-II. Please be noted that the proposed PID-based residual network is capable of fitting well for a general initial predictor.

III. Methodology

In this section, we will describe the details of the FPRC, the initial predictor and the PID-R.

A. Final Prediction with Residual Compensation (FPRC)

Fig. 1(a) shows the FPRC, which incorporates the initial predictor and the PID-R for prediction's compensation, where the final prediction is delayed and fed back into two separate modules for more precise prediction.

Specifically, given $\mathbf{X}_T = (\mathbf{x}_1, \mathbf{x}_t, \cdots, \mathbf{x}_T)$, the initial predictor generates its prediction $\tilde{\mathbf{y}}_{T+t}$ by taking the historical final prediction $\mathbf{Y}_{T+t-1}$ as its input. On the other hand, by virtue of $\tilde{\mathbf{Y}}_{T+t-1}$ and $\mathbf{Y}_{T+t-1}$, the PID-R generates the current residual error ε_{T+t}. The final prediction at the time stamp $T+t$ can be obtained as

$$\mathbf{y}_{T+t} = \tilde{\mathbf{y}}_{T+t} + \varepsilon_{T+t}. \tag{2}$$

Considering the next time stamp $T+t+1$, the prediction $\mathbf{y}_{T+t}$ and $\tilde{\mathbf{y}}_{T+t}$ are added into $\mathbf{Y}_{T+t}$ and $\tilde{\mathbf{Y}}_{T+t}$ respectively to predict $\mathbf{y}_{T+t+1}$ until the end we obtain all $\mathbf{Y}_{T+H}$.

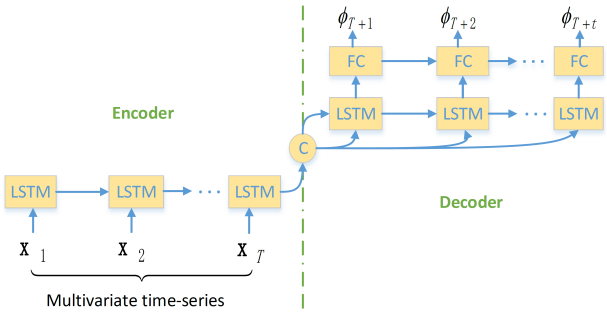


Fig. 2. Seq2Seq model (For easy clarification).

B. Initial Predictor

The initial predictor contains the LSTM based Seq2Seq, the feedback-induced VAR and the FNN-I module.

1) Seq2Seq based Prediction: The LSTM-based seq2seq module contains an encoder, a decoder and a context vector $\mathbf{C}$ [17]. For convenient clarification, we simply sketch its block diagram in Fig. 2. Given $\mathbf{X}_T = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T)$, the encoder takes each $\mathbf{x}_k$ as an input and updates the hidden state $\mathbf{h}_k$ and cell state $\mathbf{c}_k$. The final state $\mathbf{h}_T$ and $\mathbf{c}_T$ are assigned to the vector $\mathbf{C}$. The decoder is to generate the serial prediction based on the hidden state and the vector $\mathbf{C}$ from the encoder.

Specifically, in Seq2Seq module, the hidden state $\mathbf{h}_k$ can be calculated based on the prior hidden state $\mathbf{h}_{k-1}$ and each time series $\mathbf{x}_k$, for $1 \le k \le T$,

$$\begin{aligned}
\mathbf{i}_k &= \sigma(W_i \cdot [\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_i) \\
\mathbf{f}_k &= \sigma(W_f \cdot [\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_f) \\
\tilde{\mathbf{c}}_k &= \tanh(W_c \cdot [\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_c) \\
\mathbf{c}_k &= \mathbf{i}_k \odot \tilde{\mathbf{c}}_k + \mathbf{f}_k \odot \tilde{\mathbf{c}}_{k-1} \\
\mathbf{o}_k &= \sigma(W_o \cdot [\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_o) \\
\mathbf{h}_k &= \mathbf{o}_k \odot \tanh(\mathbf{c}_k),
\end{aligned} \tag{3}$$

where $\mathbf{i}_k$ represents the input gate, $\mathbf{f}_k$ represents the forget gate, $\mathbf{o}_k$ represents the output gate and $\mathbf{c}_k$ represents the cell state;

σ is the Sigmoid function, $[\cdot\ ,\ \cdot]$ represents concatenation, and $\odot$ is the element-wise Hadamard product; W_i, W_f, W_c and W_o represent weights; $\mathbf{b}_i$, $\mathbf{b}_f$, $\mathbf{b}_c$ and $\mathbf{b}_o$ represent the corresponding biases. The context vector $\mathbf{C}$ is thus assigned by

$$\mathbf{C} = [\mathbf{h}_T, \mathbf{c}_T]\,. \tag{4}$$

In the decoder, by passing $\mathbf{C}$ into each LSTM block as shown in Fig. 2, we similarly obtain the hidden states $\mathbf{h}_{T+\tau}$, $1 \le \tau \le t$, as

$$\begin{aligned}
\mathbf{i}_{T+\tau} &= \sigma(W_i^{'} \cdot [\mathbf{h}_{T+\tau-1}, \mathbf{C}] + \mathbf{b}_i^{'}) \\
\mathbf{f}_{T+\tau} &= \sigma(W_f^{'} \cdot [\mathbf{h}_{T+\tau-1}, \mathbf{C}] + \mathbf{b}_f^{'}) \\
\tilde{\mathbf{c}}_{T+\tau} &= \tanh(W_c^{'} \cdot [\mathbf{h}_{T+\tau-1}, \mathbf{C}] + \mathbf{b}_c^{'}) \\
\mathbf{c}_{T+\tau} &= \mathbf{i}_{T+\tau} \odot \tilde{\mathbf{c}}_{T+\tau} + \mathbf{f}_{T+\tau} \odot \tilde{\mathbf{c}}_{T+\tau-1} \\
\mathbf{o}_{T+\tau} &= \sigma(W_o^{'} \cdot [\mathbf{h}_{T+\tau-1}, \mathbf{C}] + \mathbf{b}_o^{'}) \\
\mathbf{h}_{T+\tau} &= \mathbf{o}_{T+\tau} \odot \tanh(\mathbf{c}_{T+\tau}),
\end{aligned} \tag{5}$$

where $W_i^{'}$, $W_f^{'}$, $W_c^{'}$ and $W_o^{'}$ represent weights; $\mathbf{b}_i^{'}$, $\mathbf{b}_f^{'}$, $\mathbf{b}_c^{'}$ and $\mathbf{b}_o^{'}$ represent the corresponding biases.

So the output at time stamp $T+\tau$, i.e. $\phi_{T+\tau}$, can be obtained by combining $\mathbf{h}_{T+\tau}$ and the prior prediction $\phi_{T+\tau-1}$ as

$$\phi_{T+\tau} = W_{FC} \cdot [\phi_{T+\tau-1}, \mathbf{h}_{T+\tau}] + \mathbf{b}_{FC}, \tag{6}$$

where W_{FC} and $\mathbf{b}_{FC}$ represent weights and biases, which could be learned from the training data. The final ϕ_{T+t} can thus be obtained after t iterations.

2) VAR based Prediction: The orginal data $\mathbf{X}_T$ and the feedback at the past time stamps to generate its independent prediction ψ_{T+t} at the current time stamp. By taking the final prediction $\mathbf{Y}_{T+t-1}$ as a feedback, according to the operation of VAR, we have

$$\psi_{T+t} = \sum_{i=1}^{T} A_i \cdot \mathbf{x}_i + \sum_{i=1}^{t-1} A_{T+i} \cdot \mathbf{y}_{T+i} + \zeta, \tag{7}$$

where each A_i is a time-invariant $m \times m$ matrix, for $1 \le i < T+t$, and ζ is a $m \times 1$ constant vector. Please be noted that, for the ablation analysis later, when this module is utilized independently for the initial prediction, the feedback at the time stamp $T+t$ is given by $\tilde{\mathbf{y}}_{T+t}$.

3) Hybrid Initial Prediction: We employ the MLP for the FNN-I module. The hybrid initial prediction is obtained by integrating the basic prediction of the Seq2Seq module ϕ_{T+t} and that of the VAR module ψ_{T+t}. So we have

$$\tilde{\mathbf{y}}_{T+t} = f_I\left(W_{\mathrm{I}} \cdot [\phi_{T+t}, \psi_{T+t}] + \mathbf{b}_{\mathrm{I}}\right), \tag{8}$$

where W_{I} and $\mathbf{b}_{\mathrm{I}}$ denote the weights and biases in the MLP, which could be learned from the training data; $f_{\mathrm{I}}(\cdot)$ represents the activation function.

C. PID-R for Residual Estimation

As shown in Fig. 1(c), the PID-R model takes the historical initial prediction $\tilde{Y}_{T+t-1} = (\tilde{\mathbf{y}}_{T+1}, \tilde{\mathbf{y}}_{T+2}, \ldots, \tilde{\mathbf{y}}_{T+t-1})$ and the historical final prediction $Y_{T+t-1} = (\mathbf{y}_{T+1}, \mathbf{y}_{T+2}, \cdots, \mathbf{y}_{T+t-1})$ as its inputs. The historical residual error ε_{T+i} can thus be obtained as

$$\varepsilon_{T+i} = \mathbf{y}_{T+i} - \tilde{\mathbf{y}}_{T+i}, \tag{9}$$

for all $i = 1, 2, \cdots, t-1$. We pass all historical residual errors $(\varepsilon_{T+1}, \varepsilon_{T+2}, ..., \varepsilon_{T+t-1})$ through the *Proportion* unit K, the *Integration* unit $\int$ and the *Differentiation* unit ∂, respectively.

Among them, the *Proportion* unit considers a linear function in terms of the residual error at the time stamp $T+t-1$. In other words,

$$P_{T+t} = K \cdot \varepsilon_{T+t-1}, \tag{10}$$

where P_{T+t} is the output of the *Proportion* unit and K denotes a proportion factor.

The *Integration* unit focuses on the cumulative effect of errors and the output can be generated by

$$I_{T+t} = \frac{\sum_{i=1}^{t-1} \varepsilon_{T+i}}{t-1}, \tag{11}$$

where $t \ge 2$ and $I_{T+1} = 0$.

The *Differentiation* unit focuses on the differential control strategy, implying that the current residual error is affected by the trend of historical residual errors. So, we have

$$D_{T+t} = \varepsilon_{T+t-1} - \varepsilon_{T+t-2}, \tag{12}$$

where D_{T+t} is the output of the *Differentiation* unit.

The outputs from the PID module, P_{T+t}, I_{T+t} and D_{T+t}, are concatenated and then passed through the FNN-II module, where another MLP is employed for fusion. So the output of PID-R can be expressed as

$$\varepsilon_{T+t} = f_{\mathrm{II}}\left(W_{\mathrm{II}} \cdot [P_{T+t}, I_{T+t}, D_{T+t}] + \mathbf{b}_{\mathrm{II}}\right), \tag{13}$$

where ε_{T+t} is the residual error at time stamp $T+t$; W_{II} and $\mathbf{b}_{\mathrm{II}}$ are weights and biases, which could be learned from the training data; $f_{\mathrm{II}}(\cdot)$ represents the activation function.

Please be noted that, in the proposed PID-R, the current residual error ε_{T+t} is modeled as the combination of various linear, cumulative and differential effects from all historical residual errors $(\varepsilon_{T+1}, \varepsilon_{T+2}, ..., \varepsilon_{T+t-1})$. This makes sense intuitively due to the complexity and correlation of residual errors at consecutive time stamps.

IV. Experiments

A. Data Sets

- **ENSO**[1] [28]: ENSO phenomenon is associated with a band of warm ocean water in the pacific, which consists of 7 indices with monthly surface temperature from 1951 to 2018. We use NINO 1-2, NINO 3, NINO 3-4 and NINO 4 as one of our datasets named **ENSO**, which consists of 816 multivariate (4 features) measurements.
- **Stock High Prices**[2]: The stock prices in Yahoo! from Jan. 3, 2007 to Dec. 25, 2018 are collected as our dataset, containing daily high prices among 6 sectors.

[1]https://www.esrl.noaa.gov/psd/data/climateindices/

[2]https://pydata.github.io/pandas-datareader/stable/remote_data.html

TABLE I
PARAMETER SETTING

Parameters	Description	Range
T	length of past samples	{32, 64}
H	length of prediction	{2, 4, 8}
h_{size}	hidden size of encoder/decoder	{64, 128}
b_{size}	the batch size	{8, 16, 32}
lr	the learning rate	{0.01, 0.001}
lr_{decay}	$lr = lr \cdot lr_{decay}$	{0.95}
w_{decay}	weight decay with L2 reg.	$\{10^{-5}\}$

- **Oil Revenue**: Our dedicated dataset provided by an high-technology company refers to various daily revenue of tens of oil stations from 2015 to 2019.

B. Compared Methods

- **VAR** [8]: VAR is a classic statistical approach to model the dependencies among multivariate time series.
- **Seq2Seq** [17]: Seq2Seq is a widely used method for machine translation and other general prediction.
- **LSTNet-skip** [26]: LSTNet-skip firstly uses the CNN and RNN to extract short-term local dependency patterns and the long-term trend, and then uses the AR to tackle the scale insensitivity.
- **LSTNet-Attn** [26]: LSTNet-Attn adds the attention mechanism on the basis of LSTNet-skip.
- **Wavelet-Trans** [27]: It feeds the scalgram and temporal signal into CNN and RNN respectively, and then fuse them using a MLP, where the scalgram is obtained by using the wavelet transform.
- **TCN** [21]: Temporal convolutional network (TCN) uses a hierarchy of temporal convolutions to capture features for further prediction.
- **R2N2** [28]: Residual recurrent neural network (R2N2) is a residual based model, which uses the AR model as the initial prediction and estimates the residual error using RNN.
- **FPRC**: FPRC is our proposed final predictor with residual compensation.

C. Evaluation Metrics and Parameters

To measure the performance of multivariate time series prediction, we use two evaluation metrics defined as follows:
1) Mean relative squared error (MRSE)

$$MRSE = \frac{\sqrt{\sum_{i=1}^{m}\sum_{t=1}^{T}(Z_t^i - \hat{Z}_t^i)^2}}{\sqrt{\sum_{i=1}^{m}\sum_{t=1}^{T}(Z_t^i - \bar{Z}_t^i)^2}},$$

2) Relative error (RE)

$$RE = \frac{\sqrt{\sum_{i=1}^{m}\sum_{t=1}^{T}(Z_t^i - \hat{Z}_t^i)^2}}{\sqrt{\sum_{i=1}^{m}\sum_{t=1}^{T}(Z_t^i)^2}},$$

where Z and $\hat{Z} \in \mathbf{R}^{m\times T}$ are the ground truth and prediction, respectively; Z_t^i indicates the dimension i at time stamp t; $\bar{Z}^i = \frac{1}{T}\sum_{t=1}^{T} Z_t^i$.

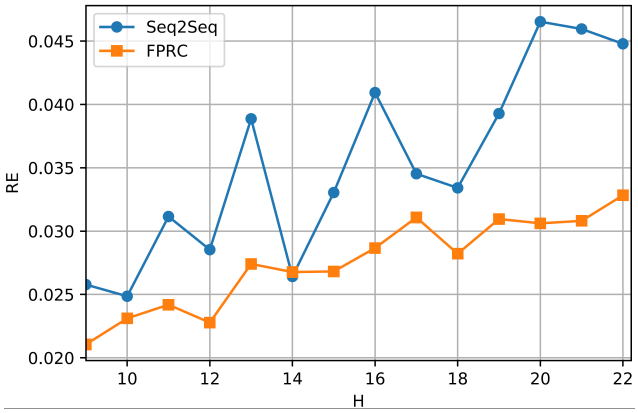


Fig. 3. Longer prediction on ENSO dataset with $H \in \{9, 22\}$.

Table I summarizes the parameter setting in our experiments. The method is optimized by performing mini-batch stochastic gradient descent (SGD) with the Adam optimizer.

D. Results and Discussion

To demonstrate the effectiveness of our proposed residual network PID-R and the resulting predictor FPRC, we conduct the following experiments and ablation analysis.

1) Comparison to All Baselines on Three Datasets: In this experiment, we aim to compare our proposed FPRC to all seven baselines mentioned above on all three datasets, where the number of past samples to be used is selected as $T = \{32, 64\}$ and the number of future samples to be predicted is selected as $H = 8$. Table II illustrates the overall experimental results, where the lowest MRSE and RE in each dataset are highlighted in boldface. As observed, the FPRC shows the best performance for all cases in terms of all seven baselines.

More detailed, it is observed that deep learning based methods and hybrid models outperform statistical analysis based method VAR in most cases, indicating the powerful capability of deep network in time series prediction. Deep learning based methods Wavelet-Trans, TCN and Seq2Seq have a better performance than hybrid models in ENSO and Stock datasets because Wavelet-Trans alleviates frequency features while TCN and Seq2Seq can keep a longer memory for prediction. However, Considering the residual compensation, the R2N2 and our proposed FPRC outperforms various deep learning based baselines Seq2Seq, Wavelet-Trans and TCN, and hybrid models LSTNet-skip and LSTNet-Attn. Furthermore, our proposed FPRC integrating hybrid model with residual compensation is better than the R2N2 that combines deep network and residual estimation. This observation indicates the significant role of the residual compensation.

2) Extension and More Observations: In this experiment, we extend the experiment above to both short-term prediction and a longer prediction on a single dataset **ENSO**, where only the baseline Seq2Seq is regarded as a representative to compare with the proposed FPRC due to its wide significance in various prediction applications.

TABLE II
PERFORMANCE OF LONG-TERM PREDICTION OF THE FPRC AND ALL BASELINES ON THREE DATASETS, WHERE $T = \{32, 64\}$ AND $H = 8$.

models	metrics	ENSO Dataset		Stock High Prices		Oil Revenue Dataset	
		32 - 8	64 - 8	32 - 8	64 - 8	32 - 8	64 - 8
VAR	MRSE	1.240	1.182	1.262	1.215	1.205	1.405
	RE	0.066	0.066	0.039	0.023	0.163	0.187
Seq2Seq	MRSE	0.547	0.477	0.885	1.178	1.263	1.827
	RE	0.030	0.027	0.027	0.022	0.171	0.243
LSTNet-skip	MRSE	1.177	0.803	1.096	1.241	1.402	1.674
	RE	0.064	0.045	0.034	0.023	0.190	0.222
LSTNet-Attn	MRSE	1.343	0.811	0.980	1.083	1.685	1.779
	RE	0.072	0.046	0.030	0.020	0.228	0.236
Wavelet-Trans	MRSE	0.549	0.597	0.888	1.710	1.132	1.745
	RE	0.030	0.034	0.027	0.032	0.153	0.232
TCN	MRSE	0.468	0.508	0.583	0.752	1.391	2.303
	RE	0.025	0.029	0.018	**0.018**	0.188	0.306
R2N2	MRSE	0.806	0.603	1.289	1.183	1.255	1.701
	RE	0.043	0.034	0.040	0.022	0.170	0.226
FPRC	MRSE	**0.403**	**0.467**	**0.531**	**0.608**	**1.062**	**1.151**
	RE	**0.022**	**0.026**	**0.016**	**0.018**	**0.144**	**0.153**

TABLE III
PREDICTION ON ENSO DATASET WITH $H = \{2, 4, 8\}$ AND $T = \{32, 64, 128\}$.

T:H	Seq2Seq		FPRC	
	MRSE	RE	MRSE	RE
32:2	0.316	0.017	**0.293**	**0.016**
32:4	0.348	0.019	**0.341**	**0.018**
32:8	0.547	0.030	**0.403**	**0.022**
64:2	0.290	0.016	**0.254**	**0.014**
64:4	0.358	0.020	**0.327**	**0.018**
64:8	0.477	0.027	**0.467**	**0.026**
128:2	**0.259**	**0.013**	0.284	0.014
128:4	0.336	0.017	**0.316**	**0.015**
128:8	0.463	0.023	**0.385**	**0.019**

With $T = \{32, 64, 128\}$ and $H = \{2, 4, 8\}$, Table III illustrates the experimental results for more observations with different combination. It is observed that the FPRC outperforms the seq2seq in general. In terms of MRSE and RE, the seq2seq has only 2 of 18 results better than our proposed FPRC. In other words, our scheme exceeds the seq2seq model on other 16 results, i.e. 88.9%. The only exceptional experiment at $[T, H] = [128, 2]$ shows that the FPRC performs approximately equal to Seq2Seq although not better.

On the other hand, we consider a longer prediction where T is fixed while H ranges from 9 to 22. Fig. 3 shows the generated RE, indicating the superiority of our proposed FPRC in a longer prediction. It is observed that the advantage becomes more evident when H gets larger.

3) Ablation Analysis: In order to examine whether the residual network PID-R is useful in multivariate time series prediction, we conduct the ablation analysis. To do so, we remove the PID-R from the proposed FPRC and the resulting predictor is named as FPwoRC.

Fig. 4 shows the performance of FPRC and FPwoRC, where we consider $T = \{32, 64\}$ and $H = 8$ on all three datasets. It is observed that there is a big difference on all cases between FPRC and FPwoRC, indicating the effectiveness of the residual compensation.

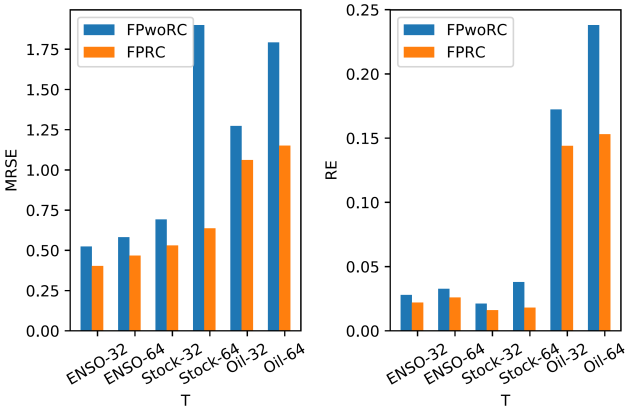


Fig. 4. Ablation analysis on the proposed FPRC.

V. CONCLUSION

In order to improve time series prediction especially the long-term prediction, we propose the residual compensation and the final predictor FPRC. The residual network is proposed based on PID modules, which considers the combination of all effects from historical residual errors owing to the complexity and correlation of residual errors at consecutive time stamps. The FPRC as well as the PID-R is demonstrated by both comparison results and ablation analysis to be greatly valuable in multivariate time series prediction.

REFERENCES

[1] E. Guresen, G. Kayakutlu, and T. U. Daim, "Using artificial neural network models in stock market index prediction," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10 389–10 397, 2011.

[2] C. Yang, X. Shi, L. Jie, and J. Han, "I know you'll be back: Interpretable new user clustering and churn prediction on a mobile social application," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 914–922.

[3] L. Zhang, C. Aggarwal, and G.-J. Qi, "Stock price prediction via discovering multi-frequency trading patterns," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 2141–2149.

[4] S. Scher, "Toward data-driven weather and climate forecasting: Approximating a simple general circulation model with deep learning," *Geophysical Research Letters*, vol. 45, no. 22, pp. 12–616, 2018.

[5] H. A. Haenssle, C. Fink, R. Schneiderbauer, F. Toberer, T. Buhl, A. Blum, A. Kalloo, A. B. H. Hassen, L. Thomas, A. Enk *et al.*, "Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists," *Annals of Oncology*, vol. 29, no. 8, pp. 1836–1842, 2018.

[6] Q. Tian, J. Liu, D. Wang, and A. Tang, "Time series prediction with interpretable data reconstruction," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, ser. CIKM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2133–2136. [Online]. Available: https://doi.org/10.1145/3357384.3358141

[7] B. Liao, J. Zhang, C. Wu, D. McIlwraith, T. Chen, S. Yang, Y. Guo, and F. Wu, "Deep sequence learning with auxiliary information for traffic prediction," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 537–546.

[8] R. Adhikari and R. K. Agrawal, "An introductory study on time series modeling and forecasting," *arXiv preprint arXiv:1302.6613*, 2013.

[9] B. Oancea and Ş. C. Ciucu, "Time series forecasting using neural networks," *arXiv preprint arXiv:1401.1333*, 2014.

[10] B. C. Csáji *et al.*, "Approximation with artificial neural networks," *Faculty of Sciences, Etvs Lornd University, Hungary*, vol. 24, no. 48, p. 7, 2001.

[11] C. Chatfield, *Time-series forecasting*. Chapman and Hall/CRC, 2000.

[12] H. Lütkepohl, *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.

[13] G. Janacek, "Time series analysis forecasting and control," *Journal of Time Series Analysis*, vol. 31, no. 4, pp. 303–303, 2010.

[14] G. Amisano and C. Giannini, *Topics in structural VAR econometrics*. Springer Science & Business Media, 2012.

[15] R. B. Litterman, "Forecasting with bayesian vector autoregressions—five years of experience," *Journal of Business & Economic Statistics*, vol. 4, no. 1, pp. 25–38, 1986.

[16] J. H. Stock and M. W. Watson, "Vector autoregressions," *Journal of Economic perspectives*, vol. 15, no. 4, pp. 101–115, 2001.

[17] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[19] B. Liao, J. Zhang, M. Cai, S. Tang, Y. Gao, C. Wu, S. Yang, W. Zhu, Y. Guo, and F. Wu, "Dest-resnet: A deep spatiotemporal residual network for hotspot traffic speed prediction," in *2018 ACM Multimedia Conference on Multimedia Conference*. ACM, 2018, pp. 1883–1891.

[20] Z. Cui, W. Chen, and Y. Chen, "Multi-scale convolutional neural networks for time series classification," *Computing Research Repository*, vol. abs/1603.06995, 2016.

[21] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[23] K. Cho, B. van Merrienboer, Çaglar Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *Empirical Methods in Natural Language Processing*, 2014.

[24] Y. Qin, D. Song, H. Cheng, W. Cheng, G. Jiang, and G. W. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," in *International Joint Conferences on Artificial Intelligence*, 2017.

[25] G. P. Zhang, "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[26] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 2018, pp. 95–104.

[27] Y. Zhao, Y. Shen, Y. Zhu, and J. Yao, "Forecasting wavelet transformed time series with attentive neural networks," in *IEEE International Conference on Data Mining*, 2018, pp. 1452–1457.

[28] H. Goel, I. Melnyk, and A. Banerjee, "R2n2: residual recurrent neural networks for multivariate time series forecasting," *arXiv preprint arXiv:1709.03159*, 2017.

[29] S. Huang, D. Wang, X. Wu, and A. Tang, "Dsanet: Dual self-attention network for multivariate time series forecasting," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, ser. CIKM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2129–2132. [Online]. Available: https://doi.org/10.1145/3357384.3358132