Research Article
A Hybrid Neural Network and H-P Filter Model for Short-Term Vegetable Price Forecasting

Youzhu Li, 1 Chongguang Li, 2 and Mingyang Zheng 1

1 College of Public Administration, Huazhong Agricultural University, Wuhan 430070, China
2 College of Economics and Management, Huazhong Agricultural University, Wuhan 430070, China

Correspondence should be addressed to Chongguang Li; chonggli@163.com

Received 2 February 2014; Accepted 20 May 2014; Published 23 June 2014

Academic Editor: Wei-Chiang Hong

Copyright © 2014 Youzhu Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper is concerned with time series data for vegetable prices, which have a great impact on humans’ lives. An accurate forecasting method for prices and an early warning system in the vegetable market are urgent needs in people’s daily lives. The time series price data contain both linear and nonlinear patterns. Therefore, neither a current linear forecasting nor a neural network can be adequate for modeling and predicting the time series data. The linear forecasting model cannot deal with nonlinear relationships, while the neural network model alone is not able to handle both linear and nonlinear patterns at the same time. The linear Hodrick-Prescott (H-P) filter can extract the trend and cyclical components from time series data. We predict the linear and nonlinear patterns and then combine the two parts linearly to produce a forecast from the original data. This study proposes a structure of a hybrid neural network based on an H-P filter that learns the trend and seasonal patterns separately. The experiment uses vegetable prices data to evaluate the model. Comparisons with the autoregressive integrated moving average method and back propagation artificial neural network methods show that our method has higher accuracy than the others.

1. Introduction

Among all the price fluctuations in the market, the prices of agricultural products have the most obvious and basic impact on the cost of living. Many countries have established price early warning systems to monitor and evaluate grain prices so that the price can be timely adjusted and controlled when it is in an abnormal state, in order to guarantee that the grain economy develops in a sustainable, healthy, and stable way [1, 2].

To predict a time series price is a challenging problem according to current studies [3, 4]. In recent decades, several linear and nonlinear prediction models have been developed for time series data forecasting. The autoregressive integrated moving average (ARIMA) model [5] is one of the most popular methods based on time series and has been widely used for data prediction [6]. Bianco et al. proposed ARIMA and transfer function models for the prediction of electricity consumption [7]. Linear regression models have also been proposed for the prediction of energy consumption [8].

However, time series data forecasting is usually a nonlinear problem, so linear approaches may fail to capture the nonlinear dynamics of the process. There are some nonlinear methods for forecasting future prices using a machine learning model [9–12]. As the price of vegetables has seasonal cyclical factors, some authors use signal processing methods to analyze the historic data and predict future prices [6, 13]. The limitation of this method is that it assumes that the period of price changes is fixed and will follow the same cycle in the future. As the price is impacted by many factors, it cannot have a fixed change cycle. Those methods cannot handle both linear and nonlinear patterns at the same time.

In this paper, a hybrid approach, combining H-P filtering [14] and a neural network model [15, 16], is developed to predict short-term time series price data. Due to the high correlation of time series data, they often contain a trend component. Hence, using an H-P filter to decompose the demand time series into its trend and cyclical components is proposed as an effective technique for time series data forecasting. The use of combined models in water quality time
series data could assist in capturing patterns in the data and
could improve prediction accuracy. The motivation behind
this hybrid approach is large that water quality problems
are often complex in nature and so any individual model
may not be able to capture all the different patterns equally
well.

This paper is organized as follows. Section 2 briefly
describes the situation of the current vegetable market in
China and the factors affecting prices. In Section 3, we
introduce some technical background knowledge. Section 4
describes the proposed H-P filter based hybrid neural net-
work model. We discuss the experimental results in Section 5.
Our conclusion is presented in Section 6.

2. A Brief Description of the Vegetable
Market in China

The Chinese vegetable market plays an important role in
people’s daily life and in the agricultural industry. In 2011,
the area planted with vegetables was about 19.7 square
kilometers, and vegetable production reached 6.79 million
tons. According to a report from the Food and Agriculture
Organization (FAO), China has 43 percent of the planted
area of the world and 50 percent of the production of the
world, and so it is ranked as the first in the world vegetable
production.

However, vegetable prices have been volatile in recent
years. There are many factors that have an influence on
vegetable prices, such as population, national policies, area
of arable land, international financial markets, price of alter-
atives, economic growth, and international trade. Here, we
study the hidden patterns behind the history of vegetable
prices to see if we can build a more accurate model for price
forecasting.

3. Background

There are several methods for extracting the trend and
cyclical components from an original single set of data.
As the prediction results from the two components need
to be merged together, we have to find a linear filter to
separate the original single set of data. Here, we choose the
linear Hodrick-Prescott filter to get the trend and cyclical
components [17]. Then, we use a traditional neural network
to learn from the patterns.

3.1. Hodrick-Prescott Filter Model. The H-P filter [17] decom-
poses time series data into trend and cyclical components:
\[ Y_t^C = Y_t^T + Y_t^C, \quad t = 1, 2, \ldots, T, \]
where \( Y_t^T \) and \( Y_t^C \) are the trend and cyclical components, respectively. This decomposition assumes that the trend component does not contain any seasonality and, because the cycle is derived residually, it does not separate out the cycle from any irregular movements. Hodrick and Prescott [17] minimize the variance of \( Y_t^C \), subject to a penalty for variations in the second difference of the growth term. Their filter is given by
\[
\min_{Y_t^T} \left\{ \sum_{t=1}^{T} (Y_t^C - Y_T^C)^2 + \lambda \sum_{t=2}^{T-1} \left[ (Y_{t+1}^T - Y_t^T) - (Y_t^T - Y_{t-1}^T) \right] \right\}.
\]

The parameter \( \lambda \) controls the smoothness of \( Y_t^C \). The minimization of (1) provides a mapping from \( Y_t \) to \( Y_t^T \), with \( Y_t^C \) determined residually. The estimate of potential output using the H-P filter depends on the choice of \( \lambda \), the smoothing parameter. A \( \lambda \) of zero corresponds to an extreme real business cycle model where all of the fluctuations in real output are caused by technology shocks, because the H-P trend is the same as the series being detrended. Conversely, as \( \lambda \) tends to infinity, the H-P trend moves towards a deterministic time trend. Following Hodrick and Prescott, researchers typically set \( \lambda \) at 1600 for use with quarterly data [17] but test the robustness of their results with different values.

3.2. Principles of Neural Networks. Artificial neural networks
(ANN) are popular models for studying nonlinear relation-
ship functions. One of the most significant advantages of
an ANN model is that it can approximate a large class of
functions with a high degree of accuracy [18]. This capability
comes from the parallel processing of the information. No
prior assumption of the model form is required in the model
building process. Instead, the network structure is mainly
determined by the features of the data. The key element of
an artificial neural network is an artificial neuron. For a given
neuron, there are multiple \((m+1)\) inputs and one output. The
output of the kth neuron is \( y_k = \varphi(\sum_{i=0}^m w_k x_i) \), where \( \varphi(\cdot) \) is
called the action function.

A single hidden layer feed-forward network is the most
widely used model form for time series modeling and fore-
casting [19]. The model is characterized by a network of three
layers of simple processing units connected by links, as shown
in Figure 1. The output \( y_t \) and the inputs \((y_{t-1}, \ldots, y_{t-T})\) have the following relationship:
\[
y_t = w_0 + \sum_{j=1}^q w_j \cdot g \left( w_{0j} + \sum_{i=1}^p w_{ij} \cdot y_{t-i} \right) + \epsilon_t,
\]
where \( w_j \) and \( w_{ij} \) \((i = 0, 1, \ldots, p, j = 1, 2, \ldots, q)\) are
parameters often called connection weights, \( p \) is the
number of input nodes, and \( q \) is the number of nodes in the hidden
layer. There are several types of activation function. The
most widely used activation function for the output layer is
the linear function, as a nonlinear activation function may
introduce distortion to the predicted output. The logarithmic
and hyperbolic functions are often used as the hidden
Mathematical Problems in Engineering

Input layer Hidden layer Output layer

\[ X_1, X_2, X_3, X_4, X_5, X_n \]

\[ \text{Out}_1, \text{Out}_2, \text{Out}_m \]

**Figure 1:** Structure of the three layers of a neural network.

\[ Y_t, Y_{t-1}, \ldots, Y_{t-k} \]

\[ \text{H-P filter} \]

\[ \text{Trend component} \]

\[ \text{Cyclic component} \]

\[ \text{ANN}_T, \text{ANN}_C \]

\[ Y_{t+1}, Y_{t+1}^T, Y_{t+1}^C \]

**Figure 2:** Framework of the proposed forecasting approach.

layer transfer function and they are shown in (3) and (4), respectively. Consider

\[
\text{sig}(x) = \frac{1}{1 + \exp(-x)}, \quad (3)
\]

\[
\text{tanh}(x) = \frac{1 - \exp(-2x)}{1 + \exp(-2x)}. \quad (4)
\]

Hence, the neural network model of (2) acts as a non-linear function mapping from past observations to the future value \( y_t \). The function can be represented as

\[
y_t = f(y_{t-1}, \ldots, y_{t-T}, w) + \epsilon_t, \quad (5)
\]

where \( w \) is a vector and \( f(\cdot) \) is the mapping function that we want to find.

The function learning process uses a back propagation training algorithm [20] to minimize the errors \( E \) between the output results and the desired results. This minimization is done by adjusting the parameters \( w_{ij} \) of the neural network by an amount \( \Delta w_{ij} \) according to the following formula:

\[
\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}}. \quad (6)
\]

Finally, the estimated model is evaluated using a separate hold-out sample that has not been exposed to the training process.

4. Model of H-P Filter Based Hybrid Neural Network

In this section, we will introduce the proposed forecasting model. The whole system framework will be presented first. Then, the model formulation will be given. Finally, we will discuss the workflow of our forecasting scheme.

4.1. Framework of the Proposed Approach. The module description of our proposed time series data forecast framework is presented in Figure 2. As shown in Figure 2, the proposed approach includes three main stages: data preprocessing, data forecasting, and data merge.

In the first stage, the time series price data are passed through the H-P filter. Trend and cyclical components are generated. This decomposition allows us to model the trend and the cyclical fluctuations of the time series separately and more accurately. As the H-P filter is a linear filter, we can merge these two components after forecasting. It must be noted that the trend and the cyclical components are separately learned by \( \text{ANN}_T \) and \( \text{ANN}_C \). Then, in the next stage, we select suitable features to be applied to the ANN models and forecast each component individually. In the third stage, we use a linear function to merge the two components together as the original data series forecasting result.

4.2. Hybrid Model Formulation. The behavior of vegetable prices may not easily be captured by stand-alone models because the price time series data could include a variety
of characteristics such as seasonality, heteroskedasticity, or a non-Gaussian error. Our hybrid model aims to reduce the dimension of the raw data. As the raw data contains multidimensional variables, we need to find a more complex function to fit the data. The artificial neural network needs more nodes and more layers to learn the complex function. This also requires more time to train the ANN, which will increase the risk of failure and make the results less accurate. After we have reduced the dimension of the raw data, the artificial neural networks are easier to train and the predictive performance will be improved in the combined models.

Based on Box’s work in linear modeling, time series data are considered as a nonlinear function of several past observations and random errors as follows:

\[
y_t = f \left[ \left( y_{t-1}, y_{t-2}, \ldots, y_{t-m} \right), \left( e_{t-1}, e_{t-2}, \ldots, e_{t-n} \right) \right],
\]

where \( f(\cdot) \) is a nonlinear function learned by the neural network, the noise \( e_t \) is the residual at time \( t \), and \( m \) and \( n \) are integers. Suppose the time series \( f(t) \) is composed of a trend and a cyclical component:

\[
Y_t = f(t) = Y^T_t + Y^C_t = f^T(t) + f^C(t),
\]

where \( f^T \) and \( f^C \) are the trend and the cyclical components. The H-P filter separates the function \( f(\cdot) \) into \( f^T \) and \( f^C \). This is the first stage shown in Figure 2. In the second stage, two neural networks are used in order to model the nonlinear relationships \( f^T \) and \( f^C \). Consider

\[
Y^T_t = f^T \left[ \left( y_{t-1}, y_{t-2}, \ldots, y_{t-m} \right), \left( e_{t-1}, e_{t-2}, \ldots, e_{t-n} \right) \right],
\]

\[
Y^C_t = f^C \left[ \left( y_{t-1}, y_{t-2}, \ldots, y_{t-m} \right), \left( e_{t-1}, e_{t-2}, \ldots, e_{t-n} \right) \right].
\]

As we can see from (9), we still use the original data as the inputs for the two neural networks, and the output result is the trend value and the cyclical value. Then, in the forecasting process, we use the following function to compute the future data:

\[
Y_{t+1} = f^T(t+1) + f^C(t+1)
\]

\[
= f^T \left[ \left( y_t, y_{t-1}, \ldots, y_{t-m+1} \right), \left( e_t, e_{t-1}, \ldots, e_{t-n+1} \right) \right] + f^C \left[ \left( y_t, y_{t-1}, \ldots, y_{t-m+1} \right), \left( e_t, e_{t-1}, \ldots, e_{t-n+1} \right) \right].
\]

4.3. Forecasting Scheme. The overall flowchart of the proposed H-P filter based neural network forecasting is shown in Figure 3. The proposed training scheme is briefly described in the following steps.

**Step 1.** Prepare the raw data of the vegetable price time series \( Y_t \) and compute the cycle \( T \) of the time series data using Fourier transform.

**Step 2.** Preprocess the raw price data using the H-P filter and generate two series of data: \( Y^T_t \) and \( Y^C_t \).
Step 3. According to the time series data cycle $T$, build the training set for the neural network.

Step 4. Use the heuristic algorithm to select the optimal number of neurons in the hidden layers of ANN$_T$ and ANN$_C$ and initialize the parameters of each neural network.

Step 5. Use the back propagation algorithm to train the neural network.

Step 6. Repeat Steps 4 and 5 until the best fitness value satisfies the minimum requirement, or the given count of total generations is reached.

The proposed forecasting scheme is briefly described in the following steps.

Step 1. Select the latest data within a cycle as the input data.

Step 2. Preprocess these data using the H-P filter.

Step 3. Use the trained ANN$_T$ and ANN$_C$ to forecast the price at the next date.

Step 4. Combine the results of the two networks as the final forecast price value.

5. Experimental Results

In this section, we will use two popular forecasting models to compare with our proposed method. These time series come from the monthly price data for five types of vegetable from 2012 to 2013. We randomly choose the price data for one cycle as the testing dataset and the remaining data as the training dataset. All the experiments are done using Matlab R2013b in the Windows 7 platform. We choose the neural network function in the Matlab toolbox [22]. Now, we study the characteristics of the prices of those five kinds of vegetable.

5.1. Study of Vegetable Price Data. We choose the prices of five types of vegetable: cabbages, peppers, cucumbers, green beans, and tomatoes. The price trends of the five types of vegetable are shown in Figure 4. As we can see from Figure 4, the price series data have a significant cyclical character. This cycle is mainly influenced by the seasonality of agricultural production. We can also note that there is a growth trend in this series data. This is mainly due to the effects of inflation.

In Table 1, we illustrate the price characteristics for the five types of vegetable. They have very similar cycles as they have annual changes. The prices of peppers and green beans have a higher variance than the other prices. It can be seen from Figure 4 that the prices of those two vegetables have relatively larger fluctuations than the prices of the others.

5.2. Experiment Modeling. We implement two traditional forecasting models ARIMA and ANN to compare with the proposed model.

5.2.1. ARIMA Modeling. In the present study, several trials were carried out to choose the optimal ARIMA model parameters. The model parameters that satisfy the statistical residual diagnostic checking were chosen for the ARIMA forecasting model. In an ARIMA $(p, d, q)$ model, the future value of a variable is assumed to be a linear function of the past observations and random errors; that is, the underlying process that generates the time series with the mean has the form

$$\phi(B) \nabla^d (y_t - \mu) = \theta(B) a_t,$$

where $y_t$ and $a_t$ are the actual value and random error at time period $t$.

The Box [21] methodology includes three iterative steps: model identification, parameter estimation, and diagnostic checking. The basic idea of model identification is that if a time series is generated from an ARIMA process, it should have theoretical autocorrelation properties. Box [21] proposed the use of the autocorrelation function (ACF) and the partial autocorrelation function (PACF) of the sample data as the basic tools for identifying the order of the ARIMA model. Based on the ARIMA model, we build five ARIMA models, to forecast the price of each of the five types of vegetable. The five ARIMA models are Cabbage ARIMA $(2, 1, 5)(1, 1, 1)^{12}$, Pepper ARIMA $(2, 1, 2)(1, 1, 1)^{12}$, Cucumber ARIMA $(1, 1, 1)(1, 1, 1)^{12}$, Green bean ARIMA $(2, 1, 1)(1, 1, 1)^{12}$, and Tomatoes ARIMA $(1, 1, 1)(1, 1, 1)^{12}$.

From Tables 2, 3, 4, 5, and 6, we can see that the ARIMA models can basically predict the results. However, we need to calculate the different parameters each time for each model, and it is difficult to find an easy way to get the parameters.
Figure 5: Regression of the predictive data using ANN.
Table 2: Cabbage ARIMA (2, 1, 5)(1, 1, 1)\(_{12}\) model predictions contrast (unit: RMB/kg).

<table>
<thead>
<tr>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-01</td>
<td>1.52</td>
<td>1.44</td>
<td>2012-07</td>
<td>2.18</td>
<td>2.48</td>
</tr>
<tr>
<td>2012-02</td>
<td>1.65</td>
<td>1.5</td>
<td>2012-08</td>
<td>2.40</td>
<td>3.05</td>
</tr>
<tr>
<td>2012-03</td>
<td>1.85</td>
<td>2</td>
<td>2012-09</td>
<td>2.41</td>
<td>2.57</td>
</tr>
<tr>
<td>2012-04</td>
<td>2.31</td>
<td>2.61</td>
<td>2012-10</td>
<td>2.08</td>
<td>1.81</td>
</tr>
<tr>
<td>2012-05</td>
<td>2.25</td>
<td>2.82</td>
<td>2012-11</td>
<td>1.84</td>
<td>1.61</td>
</tr>
<tr>
<td>2012-06</td>
<td>2.06</td>
<td>2.52</td>
<td>2012-12</td>
<td>1.85</td>
<td>1.85</td>
</tr>
</tbody>
</table>

Table 3: Pepper ARIMA (2, 1, 2)(1, 1, 1)\(_{12}\) model predictions contrast (unit: RMB/kg).

<table>
<thead>
<tr>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-01</td>
<td>9.84</td>
<td>7.10</td>
<td>2012-07</td>
<td>4.25</td>
<td>4.02</td>
</tr>
<tr>
<td>2012-02</td>
<td>7.95</td>
<td>7.57</td>
<td>2012-08</td>
<td>4.16</td>
<td>4.09</td>
</tr>
<tr>
<td>2012-03</td>
<td>9.57</td>
<td>7.78</td>
<td>2012-09</td>
<td>4.26</td>
<td>4.40</td>
</tr>
<tr>
<td>2012-04</td>
<td>7.96</td>
<td>7.20</td>
<td>2012-10</td>
<td>4.13</td>
<td>4.84</td>
</tr>
<tr>
<td>2012-05</td>
<td>6.56</td>
<td>5.74</td>
<td>2012-11</td>
<td>4.39</td>
<td>5.32</td>
</tr>
<tr>
<td>2012-06</td>
<td>4.75</td>
<td>4.43</td>
<td>2012-12</td>
<td>5.67</td>
<td>6.12</td>
</tr>
</tbody>
</table>

Table 4: Cucumber ARIMA (1, 1, 1)(1, 1, 1)\(_{12}\) model predictions contrast (unit: RMB/kg).

<table>
<thead>
<tr>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-01</td>
<td>8.17</td>
<td>6.08</td>
<td>2012-07</td>
<td>3.11</td>
<td>2.89</td>
</tr>
<tr>
<td>2012-02</td>
<td>6.35</td>
<td>6.32</td>
<td>2012-08</td>
<td>3.8</td>
<td>3.39</td>
</tr>
<tr>
<td>2012-03</td>
<td>6.21</td>
<td>5.74</td>
<td>2012-09</td>
<td>3.43</td>
<td>3.62</td>
</tr>
<tr>
<td>2012-04</td>
<td>5.04</td>
<td>4.71</td>
<td>2012-10</td>
<td>3.44</td>
<td>3.96</td>
</tr>
<tr>
<td>2012-05</td>
<td>3.56</td>
<td>3.68</td>
<td>2012-11</td>
<td>4.38</td>
<td>4.67</td>
</tr>
<tr>
<td>2012-06</td>
<td>2.71</td>
<td>2.84</td>
<td>2012-12</td>
<td>5.69</td>
<td>5.56</td>
</tr>
</tbody>
</table>

Table 5: Green bean ARIMA (2, 1, 1)(1, 1, 1)\(_{12}\) model predictions contrast (unit: RMB/kg).

<table>
<thead>
<tr>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-01</td>
<td>9.95</td>
<td>8.65</td>
<td>2012-07</td>
<td>4.78</td>
<td>5.10</td>
</tr>
<tr>
<td>2012-02</td>
<td>9.3</td>
<td>9.18</td>
<td>2012-08</td>
<td>5.6</td>
<td>5.63</td>
</tr>
<tr>
<td>2012-03</td>
<td>10.12</td>
<td>9.64</td>
<td>2012-09</td>
<td>5.58</td>
<td>6.06</td>
</tr>
<tr>
<td>2012-04</td>
<td>9.2</td>
<td>8.56</td>
<td>2012-10</td>
<td>5.19</td>
<td>6.26</td>
</tr>
<tr>
<td>2012-05</td>
<td>6.29</td>
<td>7.12</td>
<td>2012-11</td>
<td>6.96</td>
<td>7.06</td>
</tr>
<tr>
<td>2012-06</td>
<td>4.64</td>
<td>5.097</td>
<td>2012-12</td>
<td>8.13</td>
<td>8.05</td>
</tr>
</tbody>
</table>

Table 6: Tomato ARIMA (1, 1, 1)(1, 1, 1)\(_{12}\) model predictions contrast (unit: RMB/kg).

<table>
<thead>
<tr>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
<th>Date</th>
<th>Observation</th>
<th>Forecast</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-01</td>
<td>5.82</td>
<td>5.26</td>
<td>2012-07</td>
<td>3.75</td>
<td>3.36</td>
</tr>
<tr>
<td>2012-02</td>
<td>5.92</td>
<td>5.48</td>
<td>2012-08</td>
<td>4.61</td>
<td>3.65</td>
</tr>
<tr>
<td>2012-03</td>
<td>6.42</td>
<td>5.28</td>
<td>2012-09</td>
<td>5</td>
<td>3.95</td>
</tr>
<tr>
<td>2012-04</td>
<td>6.69</td>
<td>5.15</td>
<td>2012-10</td>
<td>4.41</td>
<td>4.31</td>
</tr>
<tr>
<td>2012-05</td>
<td>5.17</td>
<td>4.42</td>
<td>2012-11</td>
<td>4.75</td>
<td>4.58</td>
</tr>
<tr>
<td>2012-06</td>
<td>3.78</td>
<td>3.49</td>
<td>2012-12</td>
<td>5.46</td>
<td>4.90</td>
</tr>
</tbody>
</table>

The artificial neural network is easier to use because we just need to determine the number of layers.

5.2.2. Neural Network Modeling. A three-layer feed-forward neural network model was developed for the prediction of the price series data using an optimized back propagation training algorithm. In the present study, the scaled conjugated gradient algorithm was selected as the optimized training method. The network structure is shown in Figure 1. We choose 12 neural nodes for the input data and 8 nodes in the hidden layer. The output has one node with the “purelin” function. In what follows, artificial neural network model performances were validated for flow prediction under a monthly time-step condition. The forecast results are shown in Figures 5(a), 5(b), 5(c), 5(d), and 5(e) for the prices of the five different types of vegetable for one season.

From Figure 5, we can see that the dashed lines are very close to the solid lines. This means that our ANN model can successfully learn the patterns of the time series price data and predict the future results. The neural network is more robust than the ARIMA model, because it does not need us to analyze the characteristics of the original data and find suitable parameters. ANN is easier to use and has similar accuracy.

5.2.3. Hybrid Modeling. Our proposed model uses the same data from the previous 12 months as the testing dataset. First, we need to extract the trend and cyclical components. Figure 6 is an example for the pepper price data after using an H-P filter. We use the decomposed data as the training datasets. The forecast results are shown in Figures 7(a), 7(b), 7(c), 7(d), and 7(e) for the prices of the five different types of vegetable for one season.

The neural network uses the same structure as we used in Section 5.2.2. As the cycle of the price data is 12 months, we use the latest 12 months’ data as the input datasets. The input
Figure 7: Regression of the proposed predictive data and real data.
training dataset is illustrated in (12). The left column is the label data, which are the prices for the latest 12 months. The 12 columns on the right are the historical data, which are used as the 12 input values for the neural networks. Consider

\[
\begin{bmatrix}
y_1 & y_2 & \cdots & y_1 & y_0 \\
y_3 & y_4 & \cdots & y_3 & y_2 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
y_n & y_{n-1} & \cdots & y_{n-11} & y_{n-12} \\
\end{bmatrix}
\]

(12)

Figure 7 shows a cycle’s real data and the predicted results. We can see that the proposed method has good performance in learning the trend and cyclical patterns of the original data. The forecasting results are more accurate than the traditional results. The performance comparison is presented in the next section.

5.3. Model Verification and Comparison. To illustrate the accuracy of the method, two different forecast consistency measures are used for the different types of vegetables. The root mean squared error (RMSE, (13)) is used as the error criterion; it is the ratio of the root mean squared error to the variance of the time series. The mean absolute error (MAE, (14)) is also employed as a performance indicator. The RMSE and the MAE are defined as follows:

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{x}_i)^2},
\]

(13)

\[
MAE = \frac{1}{N} \sum_{i=1}^{N} |x_i - \hat{x}_i|.
\]

(14)

Table 7 contains a statistical analysis of the performance of the three forecasting models. We can see that our proposed model has the best performance in predicting the future prices for the vegetables. The ARIMA model is unstable for different kinds of time series data. It did well for price data for cabbage but worse for the data for the price of peppers. This is because we did not find the best parameters for the ARIMA model. Our proposed model is more stable when handling different kinds of time series data. The ANN model has a middling performance.

### 6. Conclusion

Time series forecasting is one of the most important quantitative models and has received a considerable amount of attention in the literature. This study presents a novel adaptive approach to extending the artificial neural network model; adaptive metrics of the inputs and a new mechanism for mixing the outputs are proposed for time series prediction. Due to the individual modeling of the trend and cyclical components, the forecasting accuracy is improved. The experimental results generated by a set of consistent performance measures with different metrics (RMSE, MAE) show that this new method can improve the accuracy of time series prediction. The performance of the proposed method is validated by time series data for five sets of vegetables.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgments

This work was supported in part by the Major Program of National Social Science Foundation of China (Project no. 12&ZD048) and by the National Social Science Foundation of China (Project no. 13CJY104).

### References


Submit your manuscripts at http://www.hindawi.com