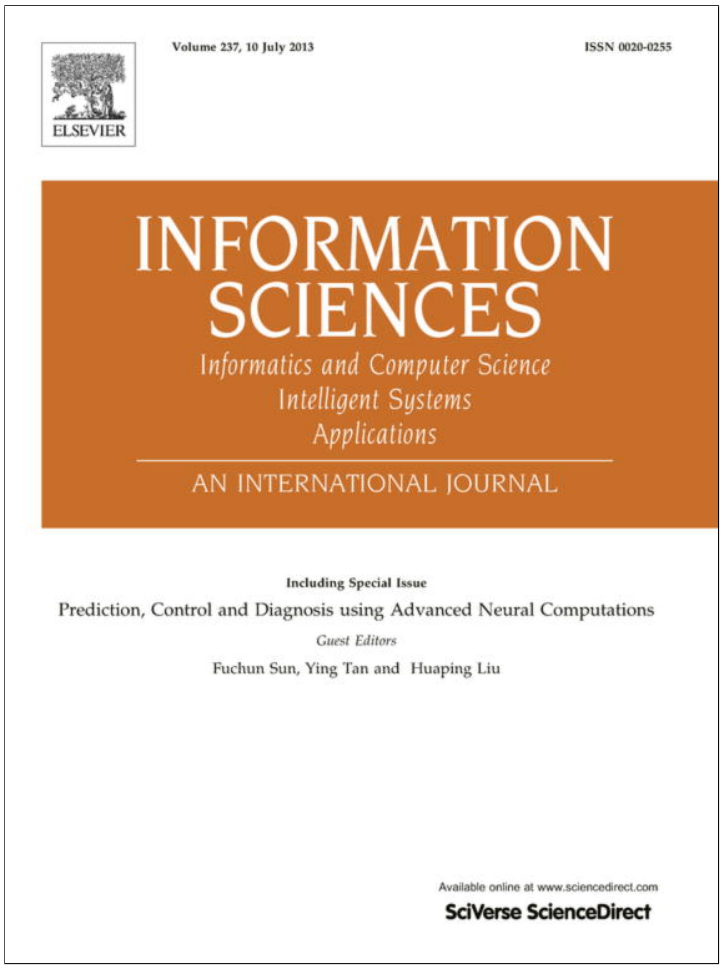

Information Sciences 237 (2013) 29–38

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Recentness biased learning for time series forecasting

Suicheng Gu, Ying Tan *, Xingui He

Key Laboratory of Machine Perception (Ministry of Education), Department of Machine Intelligence, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, PR China

ARTICLE INFO

Article history:
Available online 7 September 2010

Keywords:
Time series forecasting
Recentness biased learning
Concept drifting
Autoregressive process
Feed-forward neural networks
Drift factor
Forgetting factor

ABSTRACT

In recent years, dynamic time series analysis with the concept drift has become an important and challenging task for a wide range of applications including stock price forecasting, target sales, etc. In this paper, a recentness biased learning method is proposed for dynamic time series analysis by introducing a drift factor. First of all, the recentness biased learning method is derived by minimizing the forecasting risk based on *a priori* probabilistic model where the latest sample is weighted most. Secondly, the recentness biased learning method is implemented with an autoregressive process and the multi-layer feed-forward neural networks. The experimental results have been discussed and analyzed in detail for two typical databases. It is concluded that the proposed model has a high accuracy in time series forecasting.

1. Introduction

It is well-known that time series can be found everywhere in our daily life, such as stock price, exchange rate, sensor data, and electrocardiogram [6,1], to name a few. Time series forecasting is of great importance in many applications, for example, prediction of stock prices [3]. Generally speaking, for time series forecasting, we always pay more attention to recent data rather than the data captured long ago. As a result, the recent data should have a big weight in the analysis of the time series for prediction and decision-making. For example, for a stockbroker, the prices of a stock in this week are usually more important than its prices in the last week.

The concept drift for forecasting was introduced in the community of time series analysis for a period of time. Several approaches had been developed for dynamic time series analysis based on this concept. For example, one typical approach is to incrementally maintain a classifier that tracks the patterns in the recent training data, which are usually within the most recent sliding window [4]. However, the number of samples, which reflects a compromise between adequate coverage and effectiveness, is difficult to determine in advance. If too many samples are used, some old samples might be included such that these out-of-date samples are useless for forecasting, in addition to introducing noise. On the contrary, if only a few samples are included, the training data might be insufficient. For these two cases, the learned model will probably carry a large variance due to the over-fitting phenomenon.

Since the 1980s, many researchers have used the concept of forgetting factor in their models to solve these problems [11,14]. The forgetting factor was used in the control theory for the first time. Thereafter, it is naturally introduced into the time series forecasting. Many experiments have shown that, by means of the forgetting factor, the forecasting accuracy of time series could be greatly improved [8,20]. In recent years, some researchers tried to exploit the concept drifting patterns to solve the over-fitting problem in the model [16,13]. Some of them took efforts on the recentness biased feature extraction. For example, Zhao and Zhang [24,25] designed a generalized dimension-reduction framework for recentness

* Corresponding author. Tel./fax: +86 10 62767611.
E-mail addresses: gusuch@gmail.com (S. Gu), ytan@pku.edu.cn (Y. Tan), hexingui@pku.edu.cn (X. He).

0020-0255/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2010.09.004

biased approximations, aiming at making use of traditional dimension reduction techniques for the recentness biased time series analysis. Others tried to find out some efficient learning models [5,18,7]. Wang et al. [17] proposed a general framework to mine the concept drifting in data streams using the weighted ensemble classifiers based on their expected classification accuracy on the test data in a real-time environment. Wu et al. [19] proposed an online-optimization incremental learning framework as an example learning system for tracking the concept drifting. Zhang et al. [23] provided a data-mining based solution to forecast ozone days for the Houston area as well as experience and guidelines to solve problems with similar properties.

Neural network is a universal function approximator [10]. Unlike traditional statistical models, the neural network is a data-driven and non-parametric weak model which lets the data speak for themselves. It is therefore less susceptible to the mis-specification problem than most parametric models. Furthermore, the neural network is more powerful in describing the dynamics of financial time series than the traditional statistical models. Among most of neural networks, the multilayer feed-forward neural network (FNN) is widely used for financial time series prediction due to its strong approximation of nonlinear mapping [21,22,9]. However, FNN currently has a problem in catching the concept drifting of the model. For the purpose of online prediction of the financial time series, Case et al. [2] proposed an online-learning algorithm for the FNN based on an adaptive forgetting factor and an optimized learning rate.

Although many methods have been proposed to deal with concept drifting, they are often difficult to implement or deduce in an optimal way. As a result, a probabilistic model is at first proposed in this paper as a basis of our analysis. Then a sample weighting strategy used in many traditional models is derived in an optimal way based on the established probabilistic model [15]. After that, a recentness biased model is developed as a practical approach under this general strategy. With this strategy, the recent samples in the training dataset will be more heavily weighted while the old samples will be weighted less. Finally, the recentness biased method is implemented by using an autoregressive process and the FNN.

The remainder of this paper is organized as follows: In Section 2, a probabilistic model is constructed, from which a recentness biased method is constructed. In Section 3, the recentness biased model is implemented by an autoregressive process. In Section 4, the recentness biased model is implemented with the FNN. In Section 5, several simulation experiments are conducted to evaluate and test our proposed method. Finally, a conclusion is given in Section 6.

2. Recentness biased learning model

In this section, a probabilistic model is constructed for time series forecasting, from which a sample weighted learning strategy is derived in an optimal way. Finally, a recentness biased learning model is developed and analyzed.

2.1. A general forecasting model

The models of time series in the real-world are complex, evolutionary, and dynamic. It is impossible to construct a general forecasting model for all kinds of time series. In real-world applications, a simple model with a few parameters is always more preferable to that with more parameters, especially when people are expecting to deal with the time series efficiently. However, the model should still be reasonable for the application but just over-simplification processing. According to these principles and rules, single-step forecasting is needed, such as a forecast for the next time interval only. Let $\{S_i\}_{i=0}^{T-1}$ be training samples, the sample S_T at time T is to be forecasted.

In order to predict the expected value of y_T at time T, we know there exists an underlying function

$$y_T = F_T(X_T, V_T), \tag{1}$$

where X_T is a vector with the observable variables and V_T is a vector with unobservable latent variables. F_T is an unknown underlying function.

Functions $\{F_t\}_{t=0}^{T-1}$ for previous samples, with known input and output, are used to estimate the function F_T, where

$$y_t = F_t(X_t, V_t). \tag{2}$$

The function F_T is often not equal to any one of the functions $\{F_t\}_{t=0}^{T-1}$. Usually, the function changes over time. For each time point t, we assume

$$y_T = F_T(X_T, V_T) = F_t(X_T, V_T) + \xi_t, \tag{3}$$

where $\xi_t \sim N(0, \sigma_t^2)$ is a random variable which represents the difference of function F_t between time t and T.

On the other hand, the unobservable variables V_t should be discarded in the model; the exact formation of the functions $\{F_t\}_{t=0}^{T-1}$ is also unknown and then approximated by an assumed function $F*$. The function $F*$ can be a linear or nonlinear function, such as a polynomial, neural networks, etc. Both the variable reducing and the function approximating bring uncertainty (risks) into the model. In order to address the uncertainty, we assume

$$F_t(X, V) = F^*(X) + \epsilon_t, \tag{4}$$

where $\epsilon_t \sim N(0, \sigma^2)$ is an additive white Gaussian noise (AWGN).

Combining Eq. (4) with Eq. (3), we have

$$y_T = F_t(X_T, V_T) + \xi_t = F^*(X_T) + \xi_t + \epsilon_t. \tag{5}$$

From Eq. (5), the forecasting risks mainly come from two sources. One is from the change of the model over time. The other is from the uncertainty of the relation between input and output. Therefore, in order to minimize the forecasting risk, a proper model should be found to first simulate the actual world, then an associated learning algorithm is developed to catch the concept drifting efficiently.

2.2. Minimize the forecasting risk

The probabilistic model of time series forecasting is derived in this subsection based on the assumptions in Section 2.1. Traditionally, each sample has the same weight in a training set. This simplification makes the learning model easier to compute, but it is not necessarily optimal. A more reasonable assumption is that recent samples are more relevant to the sample to be predicted. So our goal is to determine how to assign a proper weight to each sample in the training set. Usually, we want to minimize the forecasting risk.

Let $\{p_t\}_{t=0}^{T-1}$ be the prior probabilities of T training samples. Then the sum of the prior probabilities of the training samples is 1, i.e., p_t satisfies

$$\sum_{t=0}^{T-1} p_t = 1. \tag{6}$$

According to Eq. (5), the forecasting uncertainty from the sample at time t is $\xi_t + \epsilon_t$. Thus, the total forecasting uncertainty of the training model can be

$$u = \sum_{t=0}^{T-1} p_t(\xi_t + \epsilon_t), \tag{7}$$

where ξ_t and ϵ_t are defined in Eqs. (3) and (4), respectively. We have

$$Var(\xi_t) = \sigma_t^2, \tag{8}$$

$$Var(\epsilon_t) = \sigma^2. \tag{9}$$

Hence, the variance of u (also called the risk of forecasting) is

$$Var(u) = Var\left(\sum_{t=0}^{T-1} p_t(\xi_t + \epsilon_t)\right) = \sum_{t=0}^{T-1} p_t^2(\sigma_t^2 + \sigma^2). \tag{10}$$

In order to minimize the risk of forecasting, the optimal p_t is determined by solving the following constrained optimization problem:

$$\begin{aligned} &\text{minimize}: \sum_{t=0}^{T-1} p_t^2(\sigma_t^2 + \sigma^2), \\ &\text{subject to} \sum_{t=0}^{T-1} p_t = 1. \end{aligned} \tag{11}$$

By using Lagrange multipliers, the solution of Eq. (11) is

$$p_t = \frac{1}{\sigma_t^2 + \sigma^2} \Big/ \left(\sum_{\tau=0}^{T-1} \frac{1}{\sigma_\tau^2 + \sigma^2}\right). \tag{12}$$

In order to simplify Eq. (12), let $\sigma_t^2 = \lambda_t \sigma^2$, then Eq. (12) can be rewritten as:

$$p_t = \frac{1}{\lambda_t + 1} \Big/ \left(\sum_{\tau=0}^{T-1} \frac{1}{\lambda_\tau + 1}\right). \tag{13}$$

From Eq. (12), we see that the sample in the training set that is more similar to the sample to be forecasted is assigned a greater weight. That is to say, if a sample is more similar to the sample to be forecasted, then it is more valuable and therefore should be assigned a greater weight.

2.3. Recentness biased learning model

In this subsection, the recentness biased learning model is constructed based on the probabilistic model developed in Section 2.2. This model is specified to be a recentness biased learning model. The number of parameters is reduced to make the model easy to compute and implement.

A number of models can be used to accomplish this task. Here, we just consider a simple model in which the function changes equally with each time unit, i.e.,

$$F_{t+1}(X_T, V_T) - F_t(X_T, V_T) = \xi, \quad \forall t = 0, 1, \ldots, T-1, \tag{14}$$

where $\xi \sim N(0, \lambda\sigma^2)$, and σ is defined in Eq. (4).

The parameter $\lambda \geqslant 0$, called **drift factor**, indicates the change of the function in each time interval. A greater value of λ means the function changes faster.

Given Eq. (14), one can obtain

$$F_{t+\tau}(X_T, V_T) - F_t(X_T, V_T) = \xi_\tau, \quad t = 0, 1, \ldots, T-1, \tag{15}$$

where $\xi_\tau \sim N(0, \tau\lambda\sigma^2)$.

Hence, ξ_t in Eq. (3) satisfies $\xi_t \sim N(0, (T-t)\lambda\sigma^2)$, i.e., $\sigma_t^2 = (T-t)\lambda\sigma^2$. So Eq. (13) can be specified as:

$$p_t = \frac{1}{(T-t)\lambda + 1} \Big/ \left(\sum_{\tau=0}^{T-1} \frac{1}{(T-\tau)\lambda + 1} \right), \tag{16}$$

where $\left(\sum_{\tau=0}^{T-1} \frac{1}{(T-\tau)\lambda+1} \right)$ is independent of the time variable t.

From Eq. (16), one can easily verify that: if $t1 > t2$, then $p_{t1} \geqslant p_{t2}$. Under this circumstance, $p_{t1} = p_{t2}$ if and only if $\lambda = 0$. This means that, in the recentness biased learning model, the latest sample can get a greatest prior probability in the training dataset.

For the convenience of computation and without loss of generality, the scalar $\left(\sum_{\tau=0}^{T-1} \frac{1}{(T-\tau)\lambda+1} \right)$ can be ignored[1,2] from Eq. (16). Thus, p_t can be directly simplified to

$$p_t = \frac{1}{(T-t)\lambda + 1}. \tag{17}$$

Fig. 1 shows the curves of the prior probability (p_t) versus time (t) for different drift factors (λ). It can be seen from the figure that the later sample has a greater prior probability in the model. It is natural that the more recent samples contain more information for forecasting. We can also observe that a great λ is associated with a steep curve. In limited cases, if $\lambda = 0$, then $p_t = 1/T$ and the curve becomes flat. In this case, all of the samples take same prior probability, and the associated model degenerates to the traditional one in which each sample in the training set is assigned the same weight.

3. Autoregressive process and its extension

3.1. AR model

An autoregressive (AR) model is a commonly and widely-used linear model for time series forecasting [1]. Most of time series consist of elements that are serially dependent in the sense that one can estimate a set of coefficients that describe consecutive elements of the time series from specific, time-lagged (previous) elements. This can be summarized as follows:

$$y_t = \sum_{\tau=1}^{\tau=k} w_\tau * y_{t-\tau} + w_0 + \epsilon = X_t^T * W + \epsilon, \tag{18}$$

where $X_t = (1, y_{t-1}, y_{t-2}, \ldots, y_{t-k})$ are inputs of the training set and $W = (w_0, w_1, \ldots, w_k)$ are the autoregressive coefficients. Let

$$Y = (y_0, y_1, \ldots, y_{T-1})^T, \tag{19}$$

$$X = (X_0, X_1, \ldots, X_{T-1})^T. \tag{20}$$

Then the sum of squared errors (SSE) is the cost function given by:

$$E = \sum_{t=0}^{T-1} \left(y_t - X_t^T * W \right)^2 = \|Y - X^T W\|^2. \tag{21}$$

Usually, the famous Least Square (LS) method can be used to solve the following optimization problem:

$$\text{minimize} : \|Y - X^T W\|^2. \tag{22}$$

The coefficient vector W in Eq. (22) can be found by:

$$W = (X^T X)^{-1} X^T Y = (X^T X)^{-1} X^T Y. \tag{23}$$

[1] We have $W = (X^T sPX)^{-1} X^T sPY = (X^T PX)^{-1} X^T PY$ for Eq. (26) in RB-AR model, where $s = \sum_{\tau=0}^{T-1} \frac{1}{(T-\tau)\lambda+1}$.

[2] In FNN model, we can combine s with the learning rate η, i.e., let $\eta_1 = \eta s$.

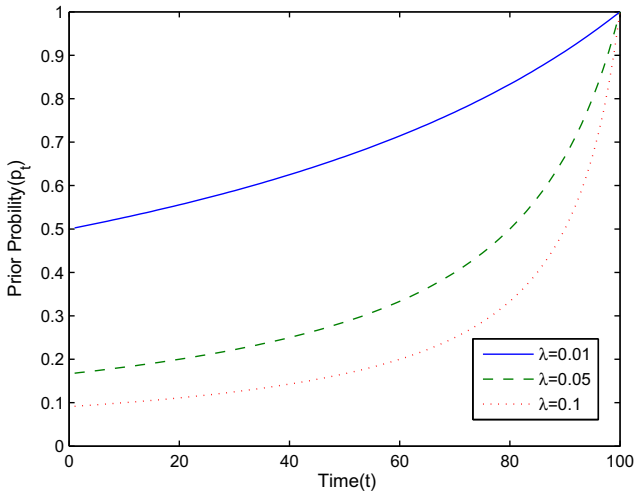

Fig. 1. Prior probability (p_t) versus time t for different drift factors (λ).

Therefore, the output y_T^* can be written as:

$$y_T^* = X_T * W. \tag{24}$$

3.2. Recentness biased autoregressive process (RB-AR)

In this subsection, a recentness biased model is implemented by using the AR process described above. The recentness biased learning model reflects the importance of the recent samples in the training set. As each training sample in the training set will take a different prior probability, the more recent samples in the training set will be assigned the greater weights than the older samples. So, we aim to find an optimal W to minimize the following revised cost function:

$$E = \sum_{t=0}^{T-1} p_t \left(y_t - X_t^T * W\right)^2 = \sum_{t=0}^{T-1} \left(\sqrt{p_t} y_t - \sqrt{p_t} X_t^T * W\right)^2 = \|DY - DX^T W\|^2, \tag{25}$$

where D is a diagonal matrix, i.e., $D = diag\{\sqrt{p_0}, \sqrt{p_1}, \ldots, \sqrt{p_{T-1}}\}$. The coefficient vector W is obtained by:

$$W = \left(X^T D^2 X\right)^{-1} X^T D^2 Y = (X^T P X)^{-1} X^T P Y, \tag{26}$$

where P is also a diagonal matrix, i.e.,

$$P = D^2 = diag\{p_0, p_1, \ldots, p_{T-1}\} = diag\left\{\frac{1}{T\lambda + 1}, \frac{1}{(T-1)\lambda + 1}, \ldots, \frac{1}{1\lambda + 1}\right\}. \tag{27}$$

If $\lambda = 0$, then $P = diag\{1, 1, \ldots, 1\}$. In this case, P degenerates to an identity matrix, and the RB-AR model is reduced to a traditional AR model. Therefore, the traditional AR model is just a special case of our proposed RB-AR model.

In addition, a sliding window strategy can be also regarded as a special recentness biased learning model as the most recent samples in the training set have equal weights while other samples have zero weight.

4. Feed-forward neural network and its extension

The most commonly-used neural network for forecasting is a multi-layer feed-forward neural network (FNN) which can be easily trained by a famous back-propagation algorithm. Consider a three-layer FNN that has k nodes in the input layer, l nodes in the hidden layer and 1 node in the output layer. Mathematically, the basic structure of the FNN can be described by:

$$y_t = \sum_{j=1}^{l} v_j f\left(\sum_{i=1}^{k} w_{ij} y_{t-i} + \theta_j\right) + \theta_0, \tag{28}$$

where $\{y_{t-i}\}_{i=1}^k$ are the inputs, y_t is the output, and w_{ij}, v_j, θ_j are the weights of the FNN. The function f is an activation function of the FNN.

The back-propagation (BP) algorithm is adopted to train the FNN because it is simple and efficient to implement. Essentially, the BP algorithm is to minimize the following cost function:

$$E = \frac{1}{T}\sum_{t=0}^{T-1} e_t^2 = \frac{1}{T}\sum_{t=0}^{T-1}\left(y_t - y_t^*\right)^2, \tag{29}$$

where y_t is the actual output of the FNN and y_t^* is the target of the FNN.

The recentness biased learning model is easily implemented with the FNN trained by the BP algorithm. Similar to the RB-AR model, the recentness biased back-propagation (RB-BP) algorithm is also able to assign the more recent samples in the training set with the greater weights. The cost function (29) can be rewritten as

$$E = \frac{1}{2}\sum_{t=0}^{T-1} p_t\left(y_t - y_t^*\right)^2 = \frac{1}{2}\sum_{t=0}^{T-1}\frac{1}{(T-t)\lambda + 1}\left(y_t - y_t^*\right)^2. \tag{30}$$

According to the BP algorithm, let $e_t = y_t - y_t^*$, so the weight w_{ij} can be modified by:

$$\Delta w_{ij} = -\frac{1}{2}\eta\frac{\partial E}{\partial w_{ij}} = -\eta\sum_{t=0}^{T-1} p_t e_t \frac{\partial y_t}{\partial w_{ij}}, \tag{31}$$

where η is the learning rate and

$$\frac{\partial y_t}{\partial w_{ij}} = \sum_{j=1}^{l} v_j \frac{\partial f}{\partial w_{ij}}. \tag{32}$$

In this paper, a logistic function is selected as the activation function, thus one has

$$f(z) = \frac{1}{1+\exp(-az)}, \tag{33}$$

where a is a parameter of the logistic function.

Since $f'(z) = af(z)[1 - f(z)]$, one has

$$\Delta w_{ij} = -\eta\sum_{t=0}^{T-1} p_t e_t\left[\sum_{j=1}^{l} v_j a f(1-f) y_{t-i}\right] = -\eta\sum_{t=0}^{T-1}\frac{e_t}{(T-t)\lambda+1}\left[\sum_{j=1}^{l} v_j a f(1-f) y_{t-i}\right]. \tag{34}$$

According to Eq. (17), if $\lambda = 0$, $p_t = 1$, then the RB-BP algorithm degenerates to the traditional BP algorithm, thus, Eq. (34) becomes

$$\Delta w_{ij} = -\eta\sum_{t=0}^{T-1} e_t\left[\sum_{j=1}^{l} v_j a f(1-f) y_{t-i}\right]. \tag{35}$$

Similarly, the update formula for other weights can also be deduced like w_{ij} in Eq. (34). Due to a limited space, we do not give them here.

5. Experiments and discussion

Several experiments are conducted to evaluate the proposed recentness biased learning strategy based on two databases: (1) Monash database contains 4 time series data [12], which can be found at http://www-personal.buseco.monash.edu.au/hyndman/forecasting/gotodata.htm. Two examples of the databases are shown in Fig. 2. (2) Stock database contains 93 time series data, each has 3000 stock prices of sequential time points, which can be found at http://www.pmel.noaa.gov/tao/data_deliv/.

5.1. Experimental setup

For the AR model, we simply set $k = 15$ as in Eq. (18), which means each value is assumed to be a weighted sum of the latest 15 values. To test the FNN, the algorithm is kept as simple as possible by avoiding using momentum, weight decay, structure-dependent learning rate, extra padding around the inputs, and averaging instead of sub-sampling. We set $k = 15$ for the input layer, $l = 7$ for the hidden layer, $\eta = 0.1$ for the learning rate and $a = 0.2$ in Eq. (33). We use these values of parameters for all experiments so that we can mainly focus on the drift factor λ in the recentness biased strategy.

For the traditional model and the proposed recentness biased learning model, $T = 1000$ most recent samples are used for training. If there are less than 1000 previous samples, we use all the previous samples for training. If a sample has less than 50 previous samples, then the sample will not be predicted.

Mean square error (MSE) is used as a performance criterion defined by:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - y_i'\right)^2, \tag{36}$$

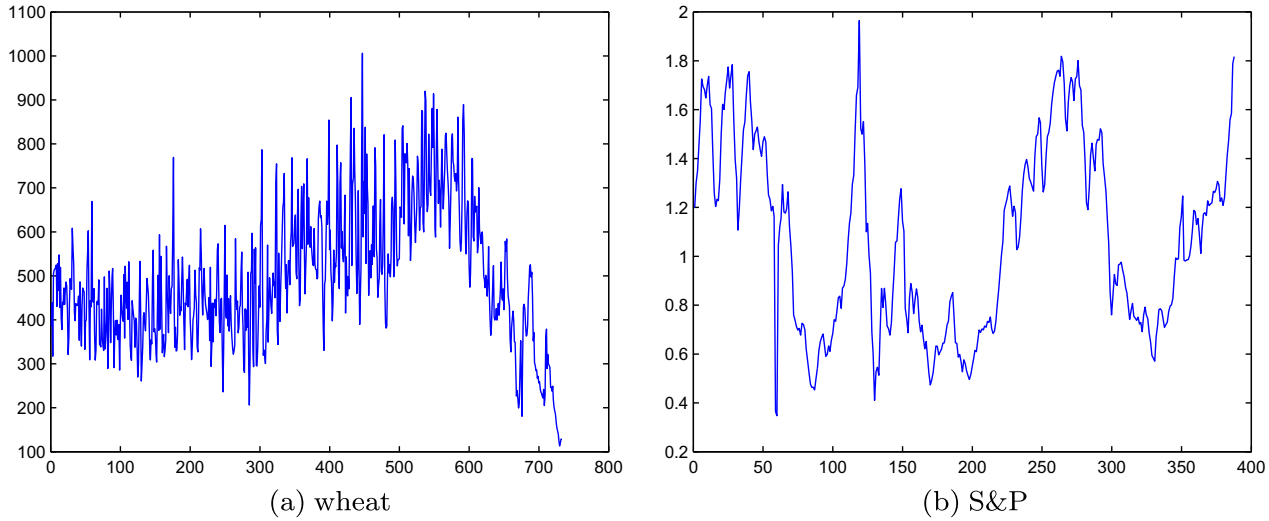

Fig. 2. Samples of two data sets in Monash database.

where N is the number of test samples, y_i is the value of the target and y_i' is the actual output of the AR model or the FNN.

5.2. Selection of drift factor (λ)

The proposed recentness biased learning model has a parameter λ, but the value of λ is not given in our model yet. Thus, we designed an experiment to evaluate how λ affects the accuracy of forecasting. Fig. 3 shows the relationship between mean square error (MSE) and drift factor (λ) on the four data sets on the Monash data set. It turns out that the accuracy of forecasting on each of the four databases is improved by using a proper drift factor. On the other hand, if the drift factor λ is too large, the old samples are not sufficiently weighted, and the model suffers from an over-fitting problem.

Therefore, our task is to select a proper parameter λ? It is impossible to give a fixed optimal value of λ for all situations or cases. However, we can give some suggestions here. In our experiments, all the optimal $\lambda^* \in [0.001, 0.1]$. Furthermore, the optimal λ^* can be found by using cross-validation strategy. While forecasting y_T, we can select an optimal λ^* to forecast the previous data $y_{T-\tau}$ ($\tau = 1, 2, \ldots$) which are already known. Furthermore, the optimal λ^* for the previous data can also be found by searching in the interval $[0.001, 0.1]$ by a gradient descent method.

5.3. Comparisons on Monash database

The wheat data include wheat prices, by pound, from year 1264 to 1996. The S&P data include Quarterly S&P 500 index from year 1900 to 1996. The Wage data include real daily wages in pounds in England from year 1260 to 1994. The milk data include monthly milk production per cow over 14 years.

We select an optimal λ for each datum in our experiments. The comparison between the proposed recentness biased learning models and the traditional models are shown in Table 1. The RB-AR model is equivalent to traditional AR model and the RB-BP algorithm is also equivalent to traditional BP algorithm. When $\lambda > 0$, the recentness biased learning models perform better than the traditional models on all the four data sets. On S& P data set, the RB-AR models obtain smaller MSE than the RB-BP models while the RB-BP models are better on the other three data sets.

It turns out from Eqs. (23) and (26) that the computational complexity of the recentness biased learning model is a little higher than the traditional model, but the difference is insignificant.

5.4. Comparisons on stock database

The stock database is much larger than the Monash database. In the stock database, the first 20 time series data are used in our experiments. Each time series data has 3000 stock prices of sequential time points. The proposed recentness biased learning models are compared with the traditional models and a sliding window model. Here we combine the sliding window strategy with the AR model together as a SL-AR model in which 200 previous samples are used to predict the current sample. For the recentness biased learning models, an optimal λ for each data set is selected. Other parameters in the AR and FNN models are also given in Section 5.1.

The experimental results are given in Table 2 from which the following conclusions can be drawn:

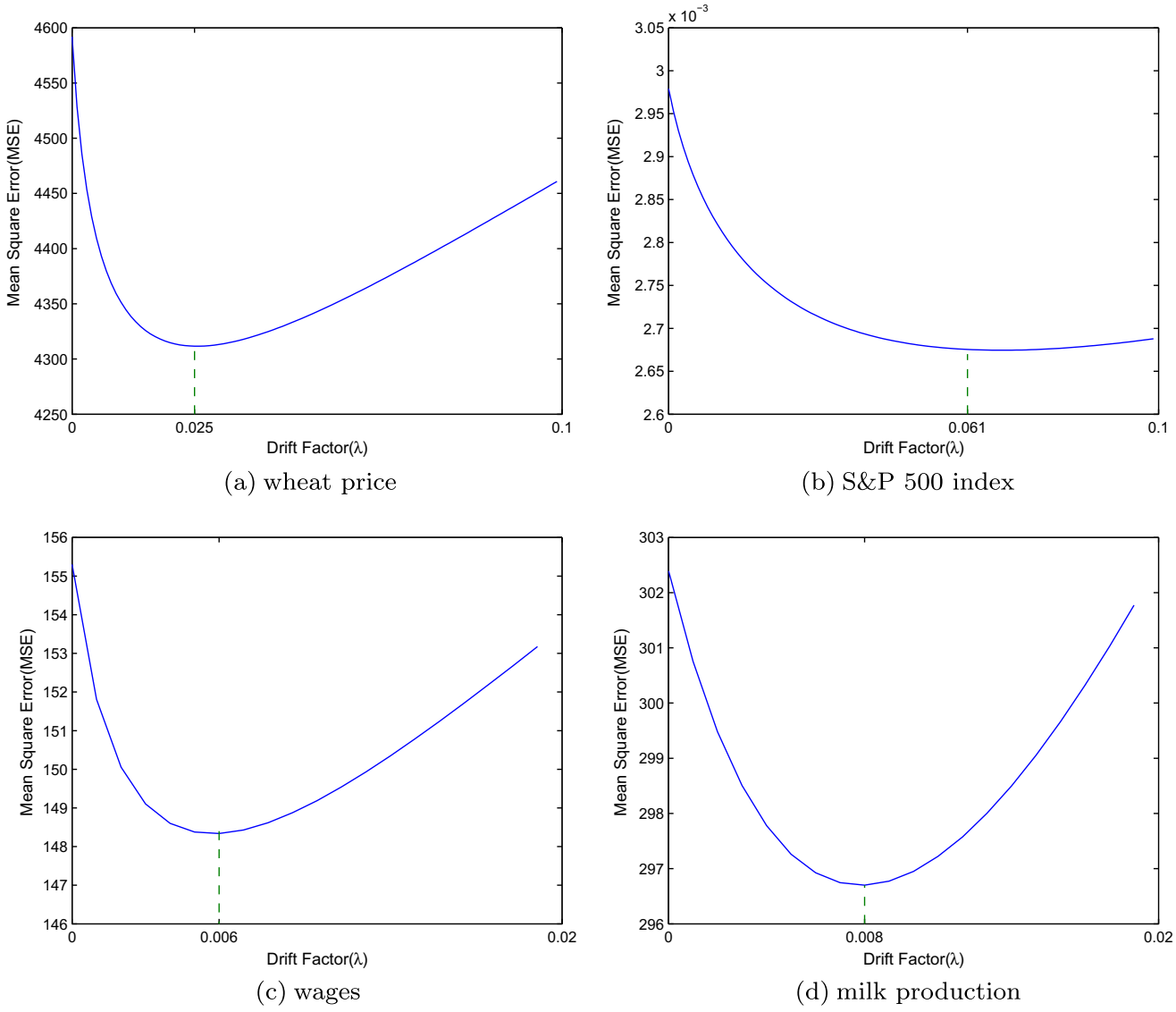

Fig. 3. Mean square error (MSE) versus different drift factors (λ).

Table 1
Comparisons of the mean square errors (MSE) on Monash database. The left column of "Ratio" is the ratio of the MSE of the RB-AR method to that of the AR method.

Databases	AR ($\lambda = 0$)	RB-AR ($\lambda > 0$)	Ratio	BP ($\lambda = 0$)	RB-BP ($\lambda > 0$)	Ratio
wheat	4592	4311	0.94	4318	4121	0.95
S&P	$2.98*10^{-3}$	$2.67*10^{-3}$	0.90	0.034	0.030	0.88
wages	155.3	148.3	0.95	31.9	28.2	0.88
milk	303.4	296.7	0.98	915	736	0.80

- The proposed recentness biased learning methods improve the accuracy of forecasting significantly.
- In some cases, the MSE of the RB-AR method is only 15% of the MSE of the corresponding AR method.
- In some cases, the SL-AR model has smaller MSE than the AR model but in other cases the AR model may have smaller MSE.
- The SL-AR model obtains better performance than the RB-AR model on three data sets, however, the RB-AR model is superior on the remaining 17 data sets.
- In our experiments, the AR models obtain better performances than the FNN models in most cases.

5.5. Discussion

Why does the recentness biased leaning method significantly improve the accuracy of forecasting? By assigning the older samples with smaller weights, the noise or interference introduced by these old samples can be reduced. Furthermore,

Table 2
Comparisons of the mean square errors (MSE) on the Stock databases. The left "Ratio" is the ratio of the MSE of the RB-AR method to the AR method.

Data	MSE of AR ($\lambda = 0$)	MSE of SL-AR	MSE of RB-AR	Ratio	MSE of BP	MSE of RB-BP	Ratio
1	3.3597	4.0298	0.5258	0.16	5.6	5.1	0.91
2	0.2826	0.4738	0.0743	0.26	0.22	0.19	0.86
3	0.3282	1.1577	0.1999	0.61	0.4	0.36	0.9
4	1.4454	3.9292	0.6065	0.42	1.37	1.26	0.92
5	0.0521	0.0640	0.0252	0.48	0.04	0.034	0.85
6	0.115	0.1659	0.0647	0.56	0.45	0.4	0.89
7	0.0105	0.0131	0.006	0.57	0.069	0.062	0.90
8	0.448	0.172	0.282	0.63	0.46	0.36	0.78
9	0.8103	1.046	0.448	0.55	0.60	0.35	0.58
10	8.0247	10.27	1.2127	0.15	6.72	6.16	0.92
11	7.8592	5.0555	3.2254	0.41	12.2	10.4	0.85
12	3.6413	2.3308	2.5313	0.70	7.15	5.76	0.81
13	1.8625	1.1412	1.6121	0.87	9.44	8.76	0.93
14	0.1858	0.1268	0.12	0.64	0.318	0.281	0.88
15	0.1672	0.1034	0.0728	0.44	0.318	0.219	0.69
16	1.1566	0.7835	0.4495	0.39	2.23	1.89	0.85
17	0.4766	0.304	0.0921	0.19	0.316	0.281	0.89
18	0.5169	0.5073	0.4662	0.90	0.865	0.626	0.72
19	1.8775	1.1334	0.8993	0.48	1.43	1.31	0.92
20	0.0281	0.0276	0.0162	0.58	0.0282	0.028	0.99

including more samples in the training set will reduce the over-fitting risk. From Eq. (11), one can see that the sample weighting method is deduced from the "minimizing the risk" problem. In fact, it can be seen from Tables 1 and 2 that the recentness biased learning model indeed improved the accuracy of forecasting. In some situations, the improvement is very significant.

The sliding window strategy can perform better in some cases by throwing out the old data. However, it performs worse in other cases probably due to the over-fitting problem.

In some cases, the BP models perform better than the AR models. But in most cases, the AR models do better than BP models. This suggests that a special model is needed for a specific task. Also, it is better to choose a simple and reasonable model for a given task in a real-world application.

Fan [5] pointed out that using the additional old data does not always help produce a more accurate hypothesis than using the most recent data only. It will increase the accuracy only in some random situations. However, the experiments given in this paper show that the old data would help produce more accurate hypotheses, but the improvement is sometimes insignificant. Also, the recentness biased learning model improved the accuracy of forecasting statistically by using the proposed probabilistic model in Section 2. Therefore, even though the recentness biased learning model is not applicable in a few cases, it is useful for most real-world problems of time series forecasting.

6. Conclusion

In this paper, the probabilistic model of time series forecasting is constructed, from which a recentness biased learning model is deduced in an optimal way. The recentness biased learning model can be implemented by an autoregressive process and the FNN. By utilizing more samples in the training set, the over-fitting risk is reduced greatly. By assigning the old samples with smaller weights, noise probably introduced by the old samples is reduced. Therefore, the accuracy for time series forecasting is greatly increased.

One problem in the recentness biased learning model is how to select an appropriate drift factor in practice. Some suggestions are provided in this paper. The cross-validation method might be a good approach to determine the λ. Although a reasonable and efficient recentness biased learning strategy is given in Section 2.3, it does not guarantee that the strategy is optimal. Therefore, under different assumptions and conditions, different recentness biased models could be developed accordingly in practice.

In Section 2.2, we give the probabilistic model of time series forecasting that not only helps to overcome the problem of concept drift but also helps to solve other problems in time series forecasting. For example, this model can help to solve seasonal time series forecasting. Our model provides a simple and important concept that a sample in the training set will take a greater weight so long as it is more similar to the sample to be predicted.

Acknowledgements

This work was supported by the National High Technology Research and Development program of China (863 Program), with Grant No. 2007AA01Z453, and the National Natural Science Foundation of China under Grant Nos. 60673020 and 60875080.

References

[1] P.J. Brockwell, R.A. Davis, Introduction to Time Series and Forecasting, second ed., Springer, 2002.
[2] J. Case, S. Jain, S. Kaufmann, A. Sharma, F. Stephan, Predictive learning models for concept drift, Theoretical Computer Science 268 (2) (2001) 323–349.
[3] C.G. da Silva, Time series forecasting with a non-linear model and the scatter search meta-heuristic, Information Sciences 178 (16) (2008) 3288–3299.
[4] M. Datar, A. Gionis, P. Indyk, R. Motwani, Maintaining stream statistics over sliding windows, SIAM Journal of Computing 31 (6) (2002) 1794–1813.
[5] W. Fan, Systematic data selection to mine concept-drifting data streams, in: The 10th ACM SIGKDD, 2004, pp. 128–137.
[6] M.M. Gaber, A. Zaslavsky, S. Krishnaswamy, Mining data streams: A review, in: SIGMOD, 2005, pp. 18–26.
[7] J. Gao, W. Fan, J. Han, P.S. Yu, A general framework for mining concept-drifting data streams with skewed distributions, in: SDM, 2007, pp. 3–14.
[8] L. Guo, L. Ljung, P. Priouret, Performance analysis of the forgetting factor RLS algorithm, International Journal of Adaptive Control and Signal Processing 7 (8) (1993) 525–537.
[9] C. Hamzasebi, Improving artificial neural networks performance in seasonal time series forecasting, Information Sciences 178 (23) (2008) 4550–4559.
[10] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice-Hall Inc., Englewood Cliffs, New-Jersey, 1999.
[11] R.M. Johnstone, C.R. Johnson, R.R. Bitmead, B.D.O. Anderson, Exponential convergence of recursive least squares with exponential forgetting factor, in: 21st IEEE Conference on Decision and Control, 1982, pp. 994–997.
[12] Makridakis, Wheelwright, Hyndman, Forecasting: Methods and Applications, third ed., Wiley, New York, 1998.
[13] K. Nishida, Learning and detecting concept drift, Graduate School of Information Science and Technology, Hokkaido University, 2008.
[14] J.E. Parkum, N.K. Poulsen, J. Holst, Recursive forgetting algorithms, International Journal of Control 55 (1) (1992) 109–128.
[15] D. Ruppert, M.P. Wand, Multivariate locally weighted least squares regression, The Annals of Statistics 23 (3) (1994) 1346–1370.
[16] A. Tsymbal, The problem of concept drift: definitions and related work, Technical Report, Department of Computer Science, Trinity College Dublin, Ireland, 2004.
[17] H. Wang, W. Fan, P.S.Yu, J. Han, Mining concept-drifting data streams using ensemble classifiers, in: the ninth ACM SIGKDD, 2002, pp. 226–235.
[18] H. Wang, J. Yin, J. Pei, P.S. Yu, J.X. Yu, Suppressing model overfitting in mining concept-drifting data streams, in: the 11th ACM SIGKDD, 2006, pp. 736–741.
[19] J. Wu, D. Ding, X.-S. Hua, B. Zhang, Tracking concept drifting with an online-optimized incremental learning framework, in: Seventh ACM SIGMM, 2005, pp. 33–40.
[20] L. Yu, S. Wang, K.K. Lai, An online learning algorithm with adaptive forgetting factors for feedforward neural networks in financial time series forecasting, Nonlinear Dynamics and Systems Theory 7 (1) (2007) 51–66.
[21] G. Zhang, B.E. Patuwo, M.Y. Hu, Forecasting with artificial neural networks: The state of the art, International Journal of Forecasting (1998) 35–42.
[22] G.P. Zhang, A neural network ensemble method with jittered training data for time series forecasting, Information Sciences 177 (23) (2007) 5329–5346.
[23] K. Zhang, W. Fan, X. Yuan, I. Davidson, X. Li, Forecasting skewed biased stochastic ozone days, in: ICDM 06, 2006, pp. 753–764.
[24] Y. Zhao, C. Zhang, S. Zhang, Enhancing dwt for recent-biased dimension reduction of time series data, in: AI 2006: Advances in Artificial Intelligence, 2006, pp. 1048–1053.
[25] Y. Zhao, S. Zhang, Generalized dimension-reduction framework for recent-biased time series analysis, IEEE Transactions on KDE 18 (2) (2006) 231–244.