A Hybrid System Using Multiple Cyclic Decomposition Methods and Neural Network Techniques for Point Forecast Decision Making

Taeksoo Shin, Ingoo Han
Graduate School of Management, Korea Advanced Institute of Science and Technology, Seoul, Korea
tsshin@msd.kaist.ac.kr, ighan@kgsm.kaist.ac.kr

Abstract

Data filtering methods are so much crucial to get good performance in time series forecasting. There are a few preprocessing methods (i.e. ARMA outputs as time domain filters, and Fourier transform or wavelet transform as time-frequency domain filters) for handling time series. In particular, the time-frequency domain filters describe the fractal structure of financial markets better than the time domain filters without frequency information.

We study the issues of integrated methods of joint time-frequency analysis and neural network techniques as the application of multi-cyclic decomposition methods to the neural networks for short-term point forecast decision-making. The issues include the appropriate selection of neural network model architecture, for example, what type of neural network learning architecture is selected and what input size should be selected for our time series forecasting. We analyze these problems in particular with recurrent neural network learning and embedding dimension as chaos analysis. This study is also applied to a case study of daily Korean won / U.S. dollar exchange returns. Finally we suggest an integration framework for future research from our experimental results.

1. Introduction

Making predictions and building trading models are central goals for financial institutions as an investor or financial manager. But, the difficulty in forecasting time series such as economic or financial data is usually attributed to the limitation of many conventional forecasting models, but could encourage many researchers to develop the more predictable forecasting models. So the models using artificial intelligence such as neural network techniques have been recognized as more useful forecasting models than the conventional statistical forecasting models ([24], [41], [42]).

Recently, more intelligent forecasting models have been developed with integration methods between neural network techniques and other learning algorithms. This study introduces joint time-frequency analysis as multi-cyclic decomposition method and focuses on the integration of joint time-frequency analysis (as Fourier and wavelet transform) and neural network techniques to gain more meaningful time series features for the efficient and effective learning. For example, a discrete wavelet transform (DWT) allows us to compute wavelet coefficients from coarse to fine scale levels efficiently and so reduces the training time of the neural network.

The purpose of this study is to introduce new filtering methodology based on Fourier and wavelet transforms that use signal decomposition algorithms and so can forecast with greater accuracy than our benchmark models. For this purpose, the original series of daily Korean won / U.S. dollar returns as a case study of financial markets are applied to several models including random walk, mean reverting, ARMA, and artificial neural networks as benchmark models. The benchmark models are then compared with our integrated model of Fourier and wavelet transforms and neural network models.

The comparison of two approaches shows whether or not the use of the decomposed series which is the special smoothed version of the original series in forecasting gets better results than applying the models directly to the original series. Furthermore, the performances of our integrated models using Fourier transform and wavelet transform are compared each other according to different hybrid model types.

This paper is organized as follows. The next section reviews the financial market heterogeneity. The third and the fourth section describe time series decomposition method and filtering methods. The fifth section presents the integration methods of joint time-frequency analysis and neural networks learning and then its experimental results and the conclusion contains final comments.

2. Financial Market Heterogeneity

Market heterogeneity suggests that the different intentions among market participants result in sensitivity by the market to several different time-scales (see Fig. 1). Müller *et al.* ([30], [31]) present that the heterogeneous market hypothesis has been associated with fractal

phenomena in the empirical behavior of foreign exchange rate (FX) markets.

For example, short-term traders are constantly watching the market; they re-evaluate the situation and execute transactions at a high frequency. Long-term traders may look at the market less frequently. A quick price increase of 0.5% followed by a quick decrease of the same size, for example, is a major event for an FX intraday trader but a non-event for central banks and long-term investors. Sometimes, small, short-term price moves may have a certain influence on the timing of long-term traders' transactions but never on their investment decisions as such. Long-term traders are interested only in large price movements and these normally happen only over long time intervals. Therefore, long-term traders with open positions have no need to watch the market every minute. In other words, they judge the market, its prices and also its volatility with a coarse time grid. It means that a coarse time grid reflects the view of long-term trader and a fine time grid reflects that of a short-term trader.
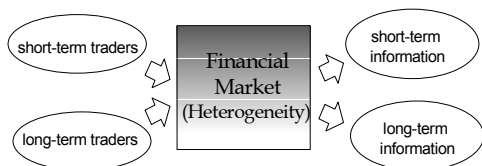

Figure 1. Heterogeneous Financial Market Structure

In summary, these different types of traders create the multiscale dynamics of the time series. Particularly the multi-scale nature of wavelet analysis makes it useful for detection and characterization of self-similar or scaling behavior, such as is common for chaotic processes.

3. Time Series Decomposition Methods

Conventionally, time series have been thought to consist of a mixture of trend (T_t), seasonal (S_t), cycle (C_t), and irregular components (e_t). We can write the time series Z_t as

$$Z_t = f(T_t, S_t, C_t, e_t) \qquad (1)$$

But, when the coverage of forecasting is much short-term, time series mainly consist of cyclic and irregular components.

The idea of decomposing a complicated signal into simpler forms is very attractive for both signal and system analysis. In time series analysis, the focus has also been on regular, periodic cycles. On the contrary, in chaos theory, nonperiodic cycles exist. These cycles have an average duration, but the exact duration of a future cycle is unknown. In that situation, we need a more robust tool

for cycle analysis, a tool that can detect both periodic and nonperiodic cycles.
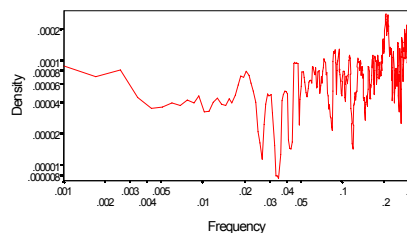

Figure 2. Fourier power spectrum on a log-log scale of Korean Won/U.S. Dollar (1992. 1. - 1996. 6.)

Fig. 2 shows an example of the cyclic features from a foreign exchange market. That is, the power spectra of Korean won / U.S. dollar series are flat and represent a broadband power spectrum. It shows that the initially straight downward trend of the power spectrum on a log-log plot is characteristic for a fractal (i.e. possibly chaotic) signal.

4. Signal Decomposition Filtering Methods

Filter in our study is defined as a means of separating the various periodic components of a time series into individual series. It means that the filtering of the input signal produces some transformation of it, a lowpass filter and a highpass filter according to the information of frequency or time scale. In this section, we present two different filtering methods, time domain filters and joint time-frequency domain filters as follows.

4.1. Time Domain Filters-ARMA Filter

Time domain filters mean linear filters using statistical parameters of such statistical models as ARMA in our study.

4.1.1. Moving average (MA) models

$$Y_t = \sum_{j=0}^{q} b_j e_{t-j} = b_0 e_t + b_1 e_{t-1} + \ldots + b_q e_{t-q} \qquad (2)$$

This equation describes a convolution filter: the new series x is generated by an qth-order filter with coefficients $b_0, \ldots, b_q$ from the series e. It is called an qth-order moving average model, MA(q).

The moving averages technique performs quite well when the market is in a state of trend but performs rather poorly around turning points and/or oscillations, because it receives delayed signals of the abrupt changes [36].

4.1.2. Autoregressive (AR) models

MA filters operate in an open loop without feedback; they can only transform an input that is applied to them. If we do not want to drive the series externally, we need to provide some feedback (or memory) in order to generate internal dynamics:

$$Y_t = \sum_{i=1}^{p} a_i Y_{x_{t-i}} + e_t \qquad (3)$$

This is called a pth-order autoregressive model, AR(p) or an infinite impulse response (IIR) filter (because the output can continue after the input ceases). Depending on the application, e_t can represent either a controlled input to the system or noise.

4.1.3. ARMA models (Weigend [49])

The next step in complexity is to allow both AR and MA parts in the model. This is call an ARMA(p,q) model.

$$Y_t = \sum_{i=1}^{p} a_i Y_{t-i} + \sum_{j=0}^{q} b_j e_{t-j} \qquad (4)$$

Equation (4) has the mixed features of AR and MA model shown from Equation (2) and (3).

4.2. Joint Time-Frequency Domain Filters

4.2.1. Fourier Transform

The Fourier transform [34] is a technique for determining the frequency spectrum of a signal. Knowledge of the frequency components provides a means of estimating where in the cycle the present time series has reached, with important consequences for predicting future behavior. Thus a Fourier spectrum indicates the frequencies, and their strengths, inherent in a time series. However this assumes the signal, or signals, to be stationary. For a general nonstationary time series the Fourier transform provides no information on the time localization of spectral components, rather the frequency spectrum would reveal wide band features characteristic of noise. A single abrupt change in the time series would, for example, affect all the components of the frequency spectrum. A transform designed for stationary signals cannot resolve features of a nonstationary signal.

$$y_t = \sum_{k=0}^{N/2} [a_k \cos(\omega_k t) + b_k \sin(\omega_k t)] + \varepsilon_t, \quad t = 1,2,...,N. \qquad (5)$$

$$s.t. \begin{pmatrix} a_k = R_k \cos(\theta_k) \\ b_k = -R_k \sin(\theta_k) \end{pmatrix}$$

ω : sinusoidal frequency
R : amplitude of the variation
θ : phase
ε : stationary random series

Equation (5) is called the Fourier series of the sequence yt. The ak and bk are called Fourier coefficients. By the Fourier coefficients, all the application-specific filter implementations can be grouped into four general filter types: lowpass, highpass, bandpass, and bandtop filter.

The Fourier transform's utility lies in its ability to analyze a signal in the time domain for its frequency content. The transform works by first translating a function in the time domain into a function in the frequency domain. The signal can then be analyzed for its frequency content because the Fourier coefficients of the transformed function represent the contribution of each sine and cosine function at each frequency. An inverse Fourier transform transforms data from the frequency domain into time domain (Fig. 3).
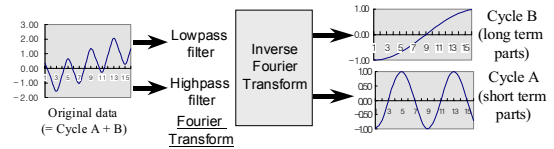

Figure 3. An Example of Using Fourier Transforms

4.2.2. Wavelet Transform

Recently, a new decomposition method known as wavelet decomposition was introduced, which is accomplished through the use of an orthogonal basis consisting of so-called 'wavelets'. The aim of a wavelet transform is to decompose any signal f into a summation of all the possible wavelet bases at the different scales [1].

While the wavelet theory has brought about significant advancements in representation of functions, not much work on its applicability to forecasting has been made. Although an initial attempt to provide the statistical framework for wavelet analysis was made by Basseville *et al.* ([3], [4]), the applicability of wavelet analysis to forecasting needs to be further developed. Tak [40] introduced new methodologies based on wavelet decomposition that can forecast with greater accuracy than existing models and utilized the simplest wavelets and a univariate case. Besides, there has been a rapidly-growing literature on the use of wavelets for denoising and smoothing ([10], [11], [12], [13], [14], [15]). Wavelets are well suited to this type problem because of their properties of time (or space) localization. When approximating a signal, wavelets can preserve local features (discontinuities, turning points, etc.) while still removing noise.

There are many different types of DWTs which have been explored since the original work in the 1980's and each have their own advantages and disadvantages when

performing data analysis. Unlike sines and cosines, which define a unique Fourier transform, there is not one single unique set of wavelets; in fact, there are infinitely many possible sets (i.e. scaling functions). Roughly, the different sets of wavelets make different trade-offs between how compactly they are localized in space and how smooth they are.

Daubechies [8] has discovered that the WT can be implemented with a specially designed pair of FIR filters called a quadrature mirror filter (QMF) pair. A FIR filter performs the dot product (or sum of products) between the filter coefficients and the discrete data samples. The act of passing a set of discrete samples, representing a signal, through a FIR filter is a discrete convolution of the signal with the coefficients of the filter.

The DWT consists of applying a wavelet coefficient matrix hierarchically, first to the full data vector of length N, then to the smooth vector of length N/2, then to the smooth-smooth vector of length N/4, and so on until only a trivial number of smooth-,...-smooth components (usually 2) remain. The procedure is sometimes called a pyramidal algorithm. The output of the DWT consists of these remaining components and all the detail components that were accumulated along the way.

Suppose the finest scale provides the original data, x_N = x, and the approximation at scale m is x_m where usually m = 2^0, 2^1,..., 2^N. The incremental detail added in going from x_m to x_{m+1}, the detail signal, is yielded by the wavelet transform. If e_m is this detail signal, then the following holds:

$$x_{m+1} = H^T(m)x_m + G^T(m)e_m \qquad (6)$$

where H(m) (i.e. defined as scaling function or father wavelet in Fig.4-(c)) and G(m) (i.e. defined as mother wavelet in Fig.4-(d)) are matrices (linear transformations) depending on the wavelet chosen, and T denotes transpose (adjoint). An intermediate approximation of the original signal is immediately possible by setting detail components $e_{m'}$ to zero for m' $\geq$ m (thus, for example, to obtain x_2, we use only x_0, e_0, and e_1). Alternatively we can de-noise the detail signals before reconstituting x and this has been termed wavelet regression ([7], [32]).

Throughout our study, we use the Daubechies wavelet transform with order 4 (DAUB4) as examples of these orthogonal wavelets (Fig. 4).

(a) Wavelet Transform Matrix   (b) Inverse Wavelet Transform Matrix

$$c_0 = \frac{1+\sqrt{3}}{4\sqrt{2}} \quad c_1 = \frac{3+\sqrt{3}}{4\sqrt{2}} \quad c_2 = \frac{3-\sqrt{3}}{4\sqrt{2}} \quad c_3 = \frac{1-\sqrt{3}}{4\sqrt{2}}$$
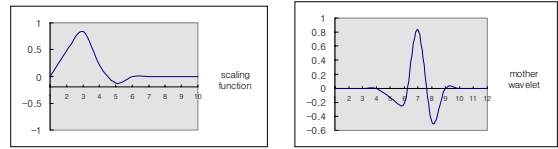
(c) Scaling Function (or Father Wavelet)   (d) Mother Wavelet

Figure 4. Daubechies Wavelet with Order 4

5. Integration of Filtering Methods and Neural Networks

In this section, we present three types of neural networks for univariate time series forecasting as follows.

(1) NN technique using the real values
(2) NN technique using the time domain filters (ARMA Models)
(3) NN technique using the time-frequency domain filters (Fourier transform, wavelet transform)

In this study, we use a fast Fourier transform (FFT) as a Fourier transform method and Daubechies wavelet as a wavelet transform method of the log price change data.

In this section, we suggest two integrated models of filtering methods and neural network techniques. One of them is to forecast one period-ahead return using single neural network. In the other integrated model, a multiscale transform of time-varying data allows forecasts of each scale (i.e. lowpass filtered data and highpass filtered data) using multiple neural networks, followed by combining of the individual forecasts.

5.1. Recurrent Neural Networks (RNN)

The RNN which has feedback connections between layers allows the dynamics of the signal to be captured. The RNNs employed in this research can be also described by a nonlinear ARMA model.

In this paper, we select the input size or order based on chaos analyses from the experimental results. That is, The number of input nodes used in the neural networks is chosen based on the nonlinear dynamic analysis (estimating embedding dimension) and the analysis was proved useful from a set of values we tested because it gave nearly the smallest in-sample prediction error.
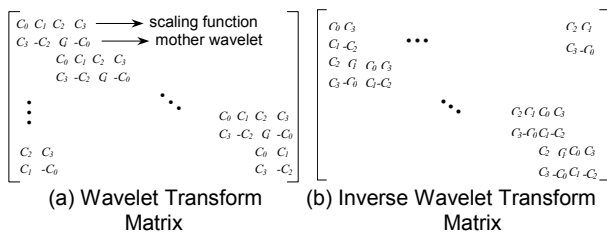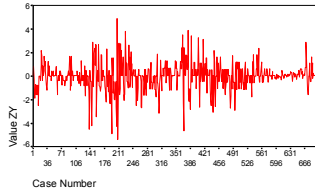
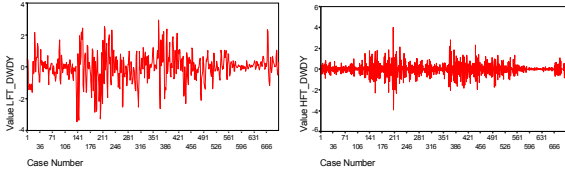**Figure 5A. Original Time Series - Daily Korean Won / U.S. Dollar Returns (lnX_t-lnX_{t-1})**

a) Lowpass Filter   b) Highpass Filter

Figure 5B. Fourier Transformed Time Series

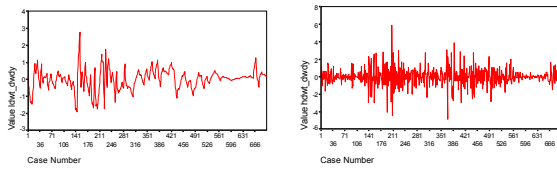a) Lowpass Filter   b) Highpass Filter

Figure 5C. Wavelet Transformed Time Series

Therefore, throughout our study, the neural network structure we used basically has 4 inputs (see Section 6.1.1.). The features of inputs are shown in Fig. 5A, 5B, and 5C.

Lowpass filters pass all frequencies below the specified frequency, and they are usually employed for smoothing. Highpass filters pass all frequencies above the specified frequency. They are usually used to extract information on local variation while suppressing overall signal levels. Bandpass filters pass only those periodic components in the vicinity of the specified frequency.

Several researchers have confirmed the superiority of RNNs over feedforward networks when performing nonlinear time series prediction. The recurrent neural network whose output is fed back to the input layer can provide longer term multi-step predictions, however, a large network size is often needed and its training generally requires an excessively long history of input. In general, context units of RNNs accumulate a weighted moving average or trace of the past output values. By making α in Fig. 6 closer to 1 the memory can be made to extend further back into past, at the expense of loss of sensitivity to detail. So the value of α should be chosen so that the decay rate matches the characteristic time scale of the input sequence.

Stornetta *et al.* Model [38] in Fig. 6 – RNN(1) can perform sequence recognition tasks. The only feedback is now from the context units to themselves, give them decay properties, but their input is now the network input

itself, which only reaches the rest of the network via the context units.

Elman [17] suggested the architecture shown in Fig. 6 – RNN(2). The input layer is divided into two parts: the true input units and the context units. The context units simply hold a copy of the activations of the hidden units from the previous time step. Modified Elman model in Fig. 6 –RNN(2) has additional parts, i.e. the feedback from the context units to themselves.
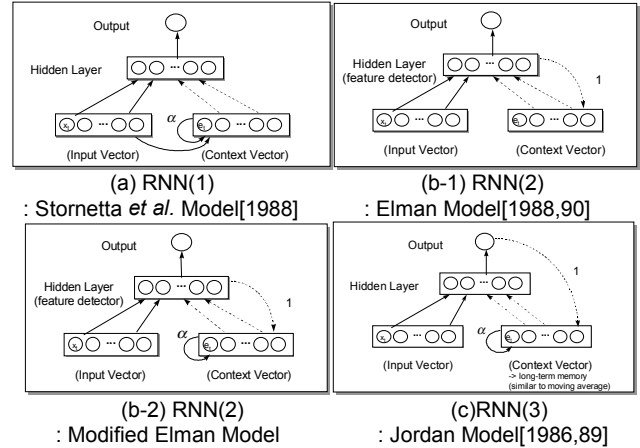

(a) RNN(1)
: Stornetta *et al.* Model[1988]

(b-1) RNN(2)
: Elman Model[1988,90]

(b-2) RNN(2)
: Modified Elman Model

(c)RNN(3)
: Jordan Model[1986,89]

Figure 6. Recurrent Neural Network Architectures

Fig.6–RNN(3) shows the Jordan (1986, 1989) architecture. It has the context units fed from the output layer and also from themselves.

Generally RNN(1) is less well suited than RNN(2) and RNN(3) to generating or reproducing sequences.

Based on the above models, we suggest the following integration models. First of all, we use a general integration model of the time domain filtering methods and neural networks as benchmark integration model (Fig. 7A).

Next, we describe a new integration model of joint time-frequency filtering methods and neural networks. It consists of two types of integration approach, i.e. (1) single recurrent neural network model (RNN) combined with the filtering methods and (2) multiple recurrent neural network models (MRNN) combined with the filtering methods (Fig. 7B).

In our integration models (Fig. 7B), we use each filter type such as highpass type (long term part) and lowpass type (short term part) as joint time-frequency domain filters applied to RNN forecasting models. And also we design the second model of two integration models on a condition that time series consist of highpass filtered parts and lowpass filtred parts (i.e. time series = highpass part + lowpass part) as follows. That is, two sub-integration models separately forecast one day ahead each lowpass output and highpass output and then two forecast filtered outputs are summed up for forecasting final output.
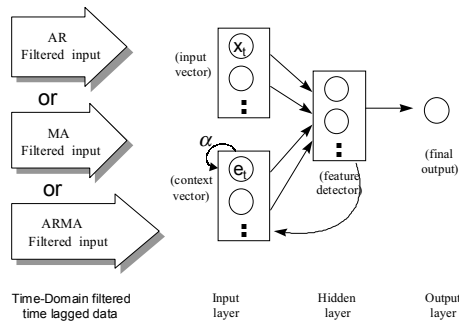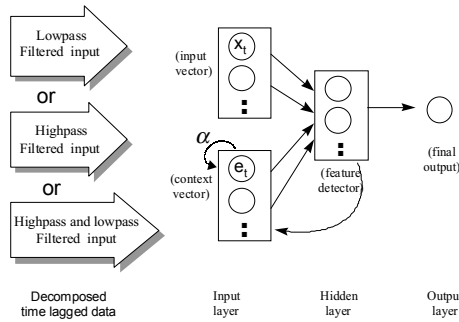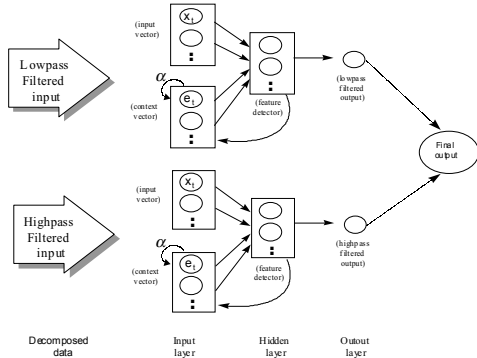
Figure 7A. Integration of Time Domain Filtering Methods and Neural Network Learning Algorithms

(1) Single Recurrent Neural Network (RNN) Model Combined with Frequency Domain Filtering Method

(2) Multiple Recurrent Neural Network (MRNN) Models Combined with Frequency Domain Filtering Method (Output = lowpass filtered data + highpass filtered data)

Figure 7B. Recurrent Neural Network Model Combined with Time-Frequency Filtering Analysis

6. Empirical Analysis and Results: A Case Study of Korean Won / U.S. Dollar Exchange Rate Market

The daily Korean won / U.S. dollar exchange rates are transformed to the returns using the logarithm and through standardization from January 10, 1992 to June 25, 1996. That is, the returns are defined as the logarithm of today's exchange rate divided by the logarithm of yesterday's exchange rate. The learning phase involves observations from January 10, 1992 to August 4, 1995, while the testing phase runs from August 7, 1995 to June 25, 1996.

In our study, the returns are basically decomposed into an approximation part and a detail part of the daily series.

Generally wavelet decomposes the input signal into detail signals, and a residual or the time series into varying scales of temporal resolution. On the contrary, the original signal can be expressed as an additive combination of the wavelet coefficients, at the different resolution levels.

We show the effectiveness (performance) of different forecasting models through two comparative analyses of different filtering methods in the following section.

6.1. Nonlinear Dynamic Analysis

6.1.1. Embedding dimension

Neural networks for univariate time series forecasting are a kind of nonlinear AR model and so the choice of order for the models is based on the embedding dimension of the series in our study because the chaos analysis is a good method to analyze nonlinear dynamics in the series.
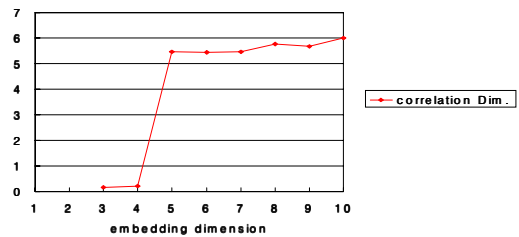

Figure 8. Correlation Dimension vs. Embedding Dimension

That is, neural networks provide a reliable basis for modeling nonlinear, dynamic market signals. And nonlinear dynamics and chaos theory can provide an information about important input sizes (i.e. time lags) for the design of forecasting systems using neural networks [18].

The results of the chaos analysis in Fig. 8 indicate a saturating tendency for the correlation dimension, leading to a fractal dimension of about 6. The embedding dimension (i.e. the dimension of the phase space for which saturation in the correlation dimension occurs) is 5.

The embedding dimension of 5 indicates that 4 data must be shown to a neural network to predict the 5th data point of the time series. But, while the short-term predictions are in principle possible for a chaotic time series, they might prove difficult in practice because of the high value of the correlation dimension.

6.2. Selecting the Order of the ARMA Model

In this section, we show the model building process of conventional ARMA models and ARMA filters. There are

several heuristics to find the right order such as the Akaike Information Criterion (AIC), Schwartz's Bayesian Criterion (SBC). First, based on the standard tools, we select ARMA(0,1) or MA(1) Model as the optimal ARMA model as shown in Table 1.

Table 1. Estimates of ARMA Parameter Identification

ARMA(p,q)	Standard error[a]	Log likelihood	AIC[b]	SBC[c]
ARMA(1,0)	1.0792	-1042.65	2089.30	2098.40
ARMA(2,0)	1.0779	-1041.30	2088.61	2102.25
ARMA(0,1)	1.0767	-1041.03	**2086.07**	**2095.16**
ARMA(0,2)	1.0773	-1040.91	2087.83	2101.47
ARMA(1,1)	1.0773	-1040.92	2087.84	2101.49
ARMA(2,1)	1.0780	-1040.89	2089.78	2107.97
ARMA(1,2)	1.0775	-1040.57	2089.15	2107.34
ARMA(2,2)	1.0749	-1038.39	2086.78	2109.53

a. *Residual Standard Deviation,*
b. *AIC = -2ln(maximum likelihood) +2k,*
c. *SBC = -2ln(maximum likelihood) + kln(n),*
(*k= the number of total parameters, n = sample size*).

Second, we also evaluate ARMA models as optimal time domain filters in terms of their neural network performance. To this purpose, we use alternative time-domain filters such as AR(p) and ARMA(p,q) model (p,q ≠ 0). Based on AIC and SBC statistics, we select AR(1), MA(1), and ARMA(1,1) as alternative time-domain filters. Table 3 shows AR(1) filter has a little better performance than any other time-domain filter in test samples.

6.3. Determining Thresholding Criteria of Time-Frequency Filter

Whether Fourier transformation is lowpass, highpass, or bandpass is characterized by two parameters, i.e. frequency and width. For lowpass filters, the frequency is the cutoff below which periodic components are passed and above which periodic components are obstructed. The width is the transition range over which the response of the filter goes from one extreme (unimpeded passage) to the other extreme (total cutoff).

The width parameter is trickier to specify. There is no simple calculation to provide the correct value. It is arbitrary choice. Unfortunately, a real tradeoff is involved.

The simplest wavelet shrinkage technique is so-called hard thresholding. Wavelet coefficients are replaced by 0 if they are smaller in absolute value than a fixed threshold λ as a filtering criteria. The threshold λ is a tuning parameter of wavelet shrinkage. Donoho and Johnstone ([11], [12]) propose several thresholds (i.e., universal, SURE), as well as several thresholding policies. Nason [33] suggests the well-known cross-validation method to find a proper threshold, which minimizes the mean integrated error for use with wavelets. A few other references in threshold selection and wavelet shrinkage applications are Gao [21] and Vidakovic ([45], [46]).

The ability of the network to capture dynamic behavior over higher resolution levels deteriorates quite fast. The higher the order of the resolution scale, the smoother the curve, and thus the less information the network can retrieve.

In this study, we select 25% and 50% as a threshold λ value (i.e. band levels of threshold filtered coefficients divided by total bands of threshold coefficients) of Daubechies wavelet transform and FFT based on root mean square errors (RMSE) of our models changing the threshold value from 10 % to 90%.

6.4. Comparative Analysis

We make the following comparative analyses in this section. They consist of (1) the comparative effects of non-filters and filters on the model performances, (2) the comparative effects of different filter types (ARMA filters as time domain filter and, Fourier filters and wavelet filters as time-frequency domain filter) on the model performances.

Table 2. Performance Comparison of Non-Filtered Forecasting Models as Benchmark Models Using Root Mean Square Error (RMSE) Measure

Model	(I-C-H-O)[d]	Training Set	Test Set
BPN	(4-0-4-1)	0.835638	0.877505
RNN(1)[a]	(4-4-4-1)	0.854616	0.894718
RNN(2)[b]	(4-4-4-1)	0.835809	**0.871192**
RNN(3)[c]	(4-1-4-1)	0.846807	0.885184
ARMA(0,1)	-	0.854402	0.882932
Random Walks	-	1.070294	1.062600
Mean Reverting	-	0.946250	0.945900

a. Stornetta *et al.* Model (1988), b. Modified Elman Model (1988, 1990), c. Jordan Model (1986, 1989), d. (I: Input Units, C: Context Units, H: Hidden Units, O: Output Units).

Table 2 summarizes the RMSE (root mean square error) on the two distinct subsets of the time series in each model. First of all, we use random walks, mean reverting, ARMA(p,q), back-propagation neural networks (BPN), and RNNs without filtering as a benchmarking model and compare each other in the predictability of the models. That result is that recurrent neural networks outperform the other models and Elman model has a little better performance than any other model in test samples. So we use this model as our main model to use filtered data in the following experiments.

Table 3. The Integration Performance of ARMA Filters and Modified Elman RNN Models (Unit: RMSE)

Filter Type	(I-C-H-O)[a]	Training Set	Test Set
AR(1)	(4-4-4-1)	0.886288	**0.947860**
MA(1)	(4-4-4-1)	0.886420	0.948424
ARMA(1,1)	(4-4-4-1)	0.886114	0.949303

a. (I: Input Units, C: Context Units, H: Hidden Units, O: Output Units).

In Table 3, by trial and error methods we find the optimal pth or qth order of AR(p), MA(q), or ARMA(p,q) filter combined to RNN models. Based on its experimental results we evaluate the results shown in Table 1 in viewpoints of final model performance. That is, AR(1) or ARMA(1,0) filter among ARMA(p,q) filters has the best model performance. But the results didn't show a significant difference compared to other filters.

Table 4. The Integration Performance of Joint Time-Frequency Filters and Modified RNN(2) Models (Unit: RMSE)

Model	(I-C-H-O)[i]	Training Set	Test Set
LFT_RNN(2)[a]	(4-4-4-1)	0.889336	0.939239
HFT_RNN(2)[b]	(4-4-4-1)	0.889659	0.948427
LWT_RNN(2)[c]	(4-4-4-1)	0.842508	0.939097
HWT_RNN(2)[d]	(4-4-4-1)	0.889990	0.948492
FT_RNN(2)[e]	(8-4-4-1)	0.263899	0.26952
FT_MRNN(2)[f]	(4-4-4-1)s	0.251940	0.253377
WT_RNN(2)[g]	(8-4-4-1)	0.735958	0.80823
WT_MRNN(2)[h]	(4-4-4-1)s	0.718174	0.792379

a. Lowpass Fourier Filter + Single Modified RNN(2),
b. Highpass Fourier Filter + Single Modified RNN(2),
c. Lowpass Wavelet Filter + Single Modified RNN(2),
d. Highpass Wavelet Filter + Single Modified RNN(2),
e. Both Fourier Filters + Single Modified RNN(2),
f. Both Fourier Filters + Multiple Modified RNN(2)s,
g. Both Wavelet Filters + Single Modified RNN(2),
h. Both Wavelet Filters + Multiple Modified RNN(2)s,
i. (I: Input Units, C: Context Units, H: Hidden Units, O: Output Units).

Table 4 compares different joint time-frequency filters integrated with RNNs and shows the following results. First, compared with Table 3, the RNNs combined with joint time-frequency filters outperformed both RNNs without filtering and RNNs with time-domain filters. Namely all the integration of joint time-frequency analysis and neural networks were significantly superior to the other models such as the statistical models (ARMA, random walk, or mean reverting) and the integration of time domain filters and neural network models. Second, each time-frequency filter type presents different validation results in integration with RNNs. Before explaining the results, we assume that each filter type, i.e. highpass type

(long term part) or lowpass type (short term part) of joint time-frequency domain filters has its own information on RNN forecasting models. Our experimental results show that RNN considering both highpass and lowpass filer types significantly outperform RNN considering partially highpass or lowpass types. Finally in predictability between FT_MRNN and WT_MRNN, FT_MRNN outperformed WT_MRNN.

Table 5. Comparative Analysis of Different Forecasting Models using Test Set

(a) The Average Performances (RMSE) of Different Forecasting Models

Model	Filter type	Test Set
ARMA(0,1)	-	0.882932
Modified RNN(2)	-	0.871192
Modified RNN(2)	AR(1)	0.947859
Multiple Modified RNN(2)s	Fast Fourier Transform	0.253377
Multiple Modified RNN(2)s	Daubechies Wavelet Transform (DAUB4)	0.792379

(b) Paired Samples t -Test for the Differences in RMSE

Model	RNN (2)	AR(1) RNN(2)	FT MRNN(2)	WT MRNN(2)
ARMA(0,1)	1.21 (.228)	-2.89 (.004)***	6.98 (.000)***	2.03 (.043)**
RNN(2)		2.29 (.004)***	7.27 (.000)***	1.83 (.069)*
AR(1) _RNN(2)[a]			6.78 (.000)***	3.61 (.000)***
FT _MRNN(2)				-5.87 (.000)***

*. significant at 10% level , **. significant at 5% level,
***. significant at 1% level, a. AR(1) Filter + Modified RNN(2).

Table 5 shows the following summary results from Table 2,3, and 4. They consist of five types of models, i.e. ARMA(0,1) and modified RNN(2) without any filters, modified RNN(2) with time domain filter, multiple modified RNN(2) models with FFT filter, and multiple modified RNN(2) models with DAUB4 filter. That is, first, modified RNN(2) models basically outperform the statistical ARMA model and second, modified RNN(2) with time-frequency domain filters outperform modified RNN(2) models without any filters. Third, time-frequency domain filtered modified RNN(2) models outperform the time domain filtered modified RNN(2) models. Finally, modified RNN(2) model with FFT filter outperforms modified RNN(2) model with DAUB4 filter.
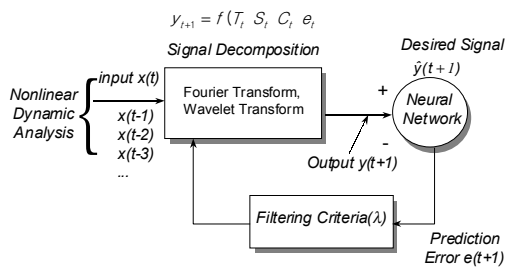
$$y_{t+1} = f(T_t, S_t, C_t, e_t)$$

Figure 9. A Proposed Integration Model Framework of Joint Time-Frequency Analysis and Neural Networks

Based on the above results, we propose a new integration model framework of joint time-frequency analysis and neural networks for future study (Fig. 9). That is, our proposed model, first of all, considers chaos analysis used in nonlinear dynamics for reducing the dimensionality (i.e. the size of time lagged input variables) of our models. After that analysis, our framework shows that according to filtering criteria, two types of control parameters, i.e. 1) thresholding criteria for time-frequency filters and 2) learning parameters for neural network should be simultaneously optimized by one evaluation function (i.e. prediction error) within our proposed model framework.

7. Conclusions

Our study investigated several filtering methodologies in the context of forecasting using daily Korean won / U.S. dollar series. First, we found that the use of the approximation part (or lowpass filtered data) and the detail part (or highpass filtered data) in forecasting gets better results than applying the models directly to the original series. Second, in integration with neural network models we detected that joint time-frequency filtering (Fourier or wavelet transform) outperformed the time domain filtering. Third, the performances of the Fourier transform method got much better results than that of the wavelet transform method. Some of the reasons are related to the following.

First, a wavelet transform has a feature detecting sharply defined structures (such as abrupt change, singularities, discontinuities, edges and bumps) better than a Fourier transform. Its performance is different according to unique time series distribution or its features. Second, we used a Daubechies wavelet as a popular wavelet transform method. However, there are a lot of wavelet transform methods in wavelet analysis. That is, we didn't fully consider that each wavelet method has its own features in our study. Third, the wavelet filter frequencies and widths in this study were not optimally analyzed.

Therefore a future research considering the above issues will be able to have better validity about our empirical results. The research includes analyzing the differences of various wavelet transform types and detecting the optimal features (e.g. filtering criteria) fitting artificial intelligent techniques for financial time series forecasting.

8. References

[1] Alsberg, B.K., A.M. Woodward, and D.B. Kell, "An Introduction to Wavelet Transforms for Chemometricians: A Time-Frequency Approach," *Chemometrics and Intelligent Laboratory Systems* 37, (1997), 215~239.

[2] Azoff, E., *Neural Network Time Series Forecasting of Financial Markets*, John Wiley & Sons, (1994), 33-35.

[3] Basseville, M., A. Benveniste, and A.S. Willsky, "Multiscale Autoregressive Processes, Part I: Schur-Levinson Parameterizations," *IEEE Transactions on Signal Processing* 40, (1992a), 1915~1934.

[4] Basseville, M., A. Benveniste, and A.S. Willsky, "Multiscale Autoregressive Processes, Part II: Lattice Structures for Whitening and Modeling," *IEEE Transactions on Signal Processing* 40, (1992b), 1935~1953.

[5] Baxter, M., and R.G. King, "Measuring Business Cycles: Approximate Band-pass Filters for Economic Time Series," *Working paper*, University of Virginia, (1995).

[6] Brock, W. and W. Dechart, "Theorems on Distinguishing Deterministic from Random Systems," in *Dynamic Econometric Modeling,* Barnett, W., Berndt, R.J., and White, P.J. (eds.), Cambridge: Cambridge University Press, (1991).

[7] Bruce, A. and H.Y. Gao, *S+Wavelets User's Manual, Version 1.0*, Seattle, WA: StatiSci Division, MathSoft Inc. (1994).

[8] Daubechies, I., "Orthogonal Bases of Compactly Supported Wavelets," *Communications on Pure and Applied Mathematics*, Vol. 41, (1988), 909~996.

[9] Deboeck, G.J. (ed.), *Trading on the Edge - Neural, Genetic, and Fuzzy Systems for Chaotic Financial Markets*, John Wiley& Sons, Inc., (1994).

[10] Donoho, D.L., "De-noising by Soft-thresholding," *Technical Report 409*, Dept. of Statistics, Stanford University, CA, (1992).

[11] Donoho, D.L. and I.M. Johnstone, "Ideal Spatial Adaptation by Wavelet Shrinkage," *Biometrika*, 81, (1994), 425~455.

[12] Donoho, D.L. and I.M. Johnstone, "Adapting to Unknown Smoothing via Wavelet Shrinkage," *Journal of American Statist. Association*, 90, (1995a), 1200~1224.

[13] Donoho, D.L. and I.M. Johnstone, "Minimax Estimation via Wavelet Shrinkage," *Submitted to the Annals of Statistics,* (1995b).

[14] Donoho, D.L., I.M. Johnstone, G. Kerkyacharian, and D. Picard, "Density Estimation by Wavelet Thresholding," *Preprint*, Dept. Statistics, Stanford University, (1993).

[15] Donoho, D.L., I.M. Johnstone, G. Kerkyacharian, and D. Picard, "Wavelet Shrinkage: Asymptopia?" *Journal of Royal Statistical Society, B.*, (1994).

[16] Elman, J.L. and D. Zipser, "Learning the Hidden Structure of Speech," *Journal of the Acoustical Society of America* 83, (1988), 1615~1626.

[17] Elman, J.L., "Finding Structure in Time," *Cognitive*

Science 14, (1990), 179~211.

[18] Embrechts, M., M. Cader, and G.J. Deboeck, "Nonlinear Dimensions of Foreign Exchange, Stock, and Bond Markets," in *Trading on the Edge - Neural, Genetic, and Fuzzy Systems for Chaotic Financial Markets*, Deboeck, G.J. (ed.), John Wiley& Sons, Inc., (1994), 297~313.

[19] Episcopos, A. and J. Davies, "Predicting Returns on Canadian Exchange Rates with Artificial Neural Networks and EGARCH-M Models," *Neural Computing and Application*, (1995).

[20] Freeman, J.A. and D.M. Skapura, *Neural Networks Algorithms, Applications, and Programming Techniques*, Addison-Wesley Publishing Company, (1991), 49~53.

[21] Gao, H.Y., "Choice of Thresholds for Wavelet Estimation of the Log Spectrum," *Technical Report 438*, Dept. Statistics, Stanford University, (1993).

[22] Grauwe, D., H. Dewachter, and M. Embrechts, *Exchange Rate Theory - Chaotic Models of Foreign Exchange Markets*, Blackwell Oxford UK & Cambridge USA, (1993), 94~95.

[23] Hartle, T., "Preprocessing Data and Fast Fourier Transform," *Technical Analysis of Stock and Commodities*, (1994).

[24] Hill, T., L. Marquez, M. O'Connor, and W. Remus, "Artificial Neural Networks for Forecasting and Decision Marking," *International Journal of Forecasting*, 10, (1994), 5~15.

[25] Hurst, H.E., R.P. Black, and Y.M. Simaika, *Long-Term Storage: An Experimental Study*, Constable - London U.K. (1965).

[26] Jordan, M.I., "Attractor Dynamics and Parallelism in a Connectionist Sequential Machine," In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society* , Amherst, (1986), 531~546.

[27] Jordan, M.I., "Serial Order: A Parallel, Distributed Processing Approach," *In Advances in Connectionist Theory: Speech*, J.L. Elman and D.E. Rumelhart (eds.), (1989).

[28] Kuan, C.M. and T. Lim, "Forecasting Exchange Rates Using Feedforward and Recurrent Neural Networks," BEBR *Working Paper*, 92-0128, University of Illinois, Urbana-Champaign, (1992).

[29] Masters, T., *Neural, Novel & Hybrid Algorithms for Time Series Prediction*. John Wiley & Sons, Inc. New York, (1995).

[30] Müller, U. A., M.M. Dacorogna, R.D. Dave, O.V. Pictet, R.B. Olsen, and J.R. Ward, "Fractals and Intrinsic Time - A Challenge to Econometricans," *XXXIXth International Conference of the Applied Econometrics Association*, Real Time Ecnometrics – Submonthly Time Series, 14-15th Oct. (1993).

[31] Müller, U. A, M.M. Dacorogna, R.B. Olsen, O.V. Pictet, and J.E. von Weizsacker, "Volatilies of Different Time Resolutions - Analyzing the Dynamics of Market Components," *The First International Conference on High Frequency Data in Finance*, Zurich, March, (1995).

[32] Murtagh, F., "Wedding the Wavelet Transform and Multivariate Data Analysis," *Journal of Classification*, Submitted, (1996).

[33] Nason, G.P., "Wavelet Regression by Cross-Validation,"

Technical Report 447, Dept. Statistics, Stanford University, (1994).

[34] Oppenheimer, A.V. and R.W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall, (1989).

[35] Press, W., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge Univ. Press, (1992).

[36] Refenes, A. (ed.), *Neural Networks in the Capital Markets*, John Wiley & Sons, (1995).

[37] Rioul, O. and M. Vetterli, "Wavelet and Signal Processing," *IEEE Signal Processing Magazine*, October (1991), 14~35.

[38] Stornetta, W.S., T. Hogg, and B.A. Huberman, "A Dynamical Approach to Temporal Pattern Processing," In *Neural Information Processing Systems* (Denver 1987), ed. D.Z. Anderson, New York: American Institute of Physics., (1988), 750~759.

[39] Strang, G., "Wavelets," *American Scientist 82*, April, (1994), 250~255.

[40] Tak, B., "A New Method for Forecasting Stock Prices Using Artificial Neural Network and Wavelet Theory," *Ph.D. Dissertation*, Depart. Economics, University of Pennsylvania, (1995).

[41] Tang, Z., C. Almeida, and P.A. Fishwick, "Time Series Forecasting Using Neural Networks vs Box-Jenkins Methodology," *Simulation* 57 (5), (1991), 303~310.

[42] Tang, Z. and P.A. Fishwick, "Feedforward Neural Nets as Models for Time Series Forecasting," *ORSA Journal on Computing* 5 (4), (1993), 374~385.

[43] Torrence, C. and G. Compo, " A Practical Guide to Wavelet Analysis," *Bull Amer. Met. Soc.,* (1997).

[44] Tsui, F., M. Sun, C. Li, and R.J. Sclabassi, "Recurrent Neural Networks and Discrete Wavelet Transform for Time Series Modeling and Prediction," *Proceedings of ICASSP-95*, Vol.5, (1995).

[45] Vidakovic, B., "Nonlinear Wavelet Shrinkage with Bayes Rules and Bayes Factors," *Discussion Paper* 94-24, ISDS, Duke University, (1994).

[46] Vidakovic, B., "Unbalancing Data with Wavelet Transformations," *Proceedings SPIE 1995, 2569, Wavelet Applications in Signal and Image Processing III*, (1995).

[47] Ville, W., "Theorie et Applications de la Notion de Signal Analytique", *Cables et Transmission*, Vol. 2a, (1948), 61~74. (Translated into English by I. Selin, RAND Corp. Report T-92, Santa Monica, CA, Aug, 1958.).

[48] Wei, W.W.S., *Time Series Analysis - Univariate and Multivariate Methods*, Addison-Wesley Publishing Company, (1990).

[49] Weigend, A., et al. (eds.), Time Series Prediction: Forecasting the Future and Understanding the Past, Addison-Wesley Publishing Company, (1994), 14~15.