MODWT-ANN based Prediction of Stock Data

22 December 2017

Aurik Sarker
asarker1@jhu.edu

Akhil Vasvani
avasvan1@jhu.edu

William Zhang
wzhang53@jhu.edu

***Abstract*—This paper examines the merits of utilizing discrete wavelet transforms in filtering input financial data and various decision making algorithms to buy, sell, or hold stocks. This system shall be split into three components: the pre-processor, the decision maker, and the post-processor. The pre-processor utilizes the discrete wavelet transform, with a wavelet of our choosing, to denoise the data. Next, the decision maker will rely on component autocorrelation to label markets as bull or bear. Finally, the post-processor shall use the states derived from the decision maker to decide whether we buy, sell, or hold on a certain day based on the input data.**

***Index Terms*—finance, stocks, time series, wavelets, discrete wavelet transform, kalman filter, hidden markov models**

MOTIVATION

Our objective is to develop, or perhaps enhance, an asset trading algorithm using the wavelet transform (in some way). Research in this domain is incentivized by the possibility of generating passive income through computer-automated trading.

Information processing of signals, however, has itself become an fundamental part of scientific activity in general, finding its use in many different fields, finance being just one. Signal prediction in particular is necessary in characterizing certain input data and making informed decisions from it, as these signals may not always be smooth and regular. Financial data is no exception. We hope to develop various methodologies (in particular, find the best wavelets) which will accomplish successful signal representation before processing.

The method we pursued is motivated by one particular paper that caught our interest, the Jothimani et al., was one that described a wavelet-machine learning hybrid approach to forecasting and trading [6]. However, their findings were merely at the proof-of-concept level. So, we took it upon ourselves to validate and expand these findings further. Thus we proceeded, guided by both our own ideas and the research suggestions from the authors.

I. RELATED WORK

The literature on creating automated trading algorithms which may perform better than humans is already quite extensive. The academic community has devoted large efforts in the domains of financial modeling, time series forecasting, and statistical determination of hidden economic variable. This includes many different strategies in modeling and prediction from stock data. only a couple of which we shall explain in further detail below.

A. Wavelet Filtering for Time Series Signal Prediction

Currently, many methods exist which attempt robust prediction of non-smooth and irregular signals. These methods typically take in past and present information about a signal (broken up into frames), then pass it through some predictor. These methods have been tested vigorously and their performances compared numerously, and are now quite good when dealing with relatively clean time series data. However, when input data are highly irregular, naïve signal representations tend to break down, and the aforementioned methods are not so accurate as a result. This is a big issue with financial data which can be extremely noisy.

In the absence of reliable methods of combating this problem, Karkanis et al. experimented with performing the one dimensional Discrete Wavelet Transform (DWT) on an input signal before loading it through a processor, in their case a feedforward neural network [7]. They hypothesized that the DWT would better capture the high order noise and discontinuities that exist in financial data, and improve the MLP's function approximation capabilities.

TABLE I
SIGNAL PREDICTION PERFORMANCE OF THE NAÏVE METHOD AND EACH WAVELET METHOD, COMPARED. ADAPTED FROM TABLE 1 [7]

Method	Wavelet	SSE for "Method 1"	SSE for Wavelets	Improvement
2	Haar	1.68	1.53	0.09
3	Coiflet	1.85	1.35	0.27
4	Daubechies	3.94	2.33	0.41
5	Symmlet	3.94	2.33	0.41
6	Vaidyanathan	1.95	1.32	0.33

In an effort to improve the baseline results, the authors chose a select number of wavelets to use in computing the DWT for these signal frames [7]. After the DWT was computed for M samples, $M/2$ detail wavelet coefficients were selected and normalized to create the input vectors to the MLP. Multiple methods of windowing and MLP was used to test the error of this wavelet-based predictor compared to the baseline. Table I displays the results of this analysis, comparing the sum of squared error (SSE) of the original windowing technique to that of its corresponding wavelet transformed signal.

From the table it is evident that the wavelet methods in each case produced much better results than the baseline, with the Daubechies wavelet performing the best out of all six. This wavelet method is quite clearly more robust than the original method, as shown by its improvement on the sum of squared error. This is supported by our understanding of wavelets, as the detail coefficients still carry a lot of information about the original signal. Unfortunately this method also requires an increase in input dimensionality and samples required. Daubechies also may not necessarily be the best wavelet to use in all cases, and much further testing with the financial data will be required in our case.

B. Monte Carlo Simulation with Geometric Brownian Motion

Monte Carlo simulations attempt to predict possible future outcomes by applying a model to a large set of initial random trials. One particular model commonly used in stimulating stock prices is Geometric Brownian Motion, a model whose "random walk" assumption is consistent with the weak Efficient Market Hypothesis, or GBM. The formula for GBM is given below in Equation 1; the change in stock price, ΔS, is given by the product of its volatility and a sum of two terms. These two terms correspond to a drift, the tendency for the stock to drift back to its expected return μ, and a shock, a random shock to its price.

$$\Delta S = S(\mu \Delta t + \sigma \epsilon \sqrt{\Delta t}) \quad (1)$$

After simulating many trials using this model, we can get a distribution of possible future outcomes for the stock. From observing a histogram of this data, however, we observe that while our GBM model assumption was a normal distribution of period returns, the consequent multi-period price levels are actually log-normally distributed. This is explained by the nature of our model. Positive price deltas tend to compound over time; a increase in the price one day means the stock has more to possibly gain the next. The opposite is true for price decreases, as lower prices means less money for the stock to potentially lose the next day.

One major problem with this approach is that the model assumes stock prices are distributed normally with some known mean and standard deviation, when in fact the distribution has many other measurable properties, like skew and kertosis. Additionally, it is not completely accurate to assume the weak form of the EMH, as there have been many proven instances of insider trading which does affect stock prices. These two assumptions make GBM too naïve to accurately predict prices which reflect this information.

C. Discrete Wavelet Transform-Based Prediction

Machine learning is a powerful statistical tool but is ultimately limited by statistics itself. Asset prices tend to be non-linear and non-stationary processes. Hence, one can argue that training a classifier or regressor to predict market state or asset price is a futile effort because the properties of the underlying process will change over time. The immediate counterargument is that one can continuously train a classifier or regressor as new data streams into the system. This is sometimes called online learning. Intuition, however, still points unfavorably on direct learning of asset prices, as the influence of past data on the system will delay the effects of online learning. The way to overcome this is to retrain only on data drawn after the process properties have changed. Of course, it is difficult to determine when the process has changed and where to draw the line when partitioning the time series data.

Jothimani et al. [6] demonstrated that a non-stationary time series can at times be decomposed into stationary components by separating them in wavelet time-scale space. Therefore the prediction performance of statistical (and nonlinear) methods such as multilayer perceptrons may increase when used in tandem with wavelet preprocessing. The paper showed that wavelet-ANN or wavelet-SVR approaches were better at one-step price prediction that an approach using Artificial Neural Network (ANN) or Support Vector Regression (SVR) alone. Furthermore, they demonstrated that returns on a wavelet-SVR trading algorithm could outperform a traditional buy-and-hold. The shortcoming of this study was that it was in essence a case study based on a predicting and trading a single market index using the Haar wavelet.

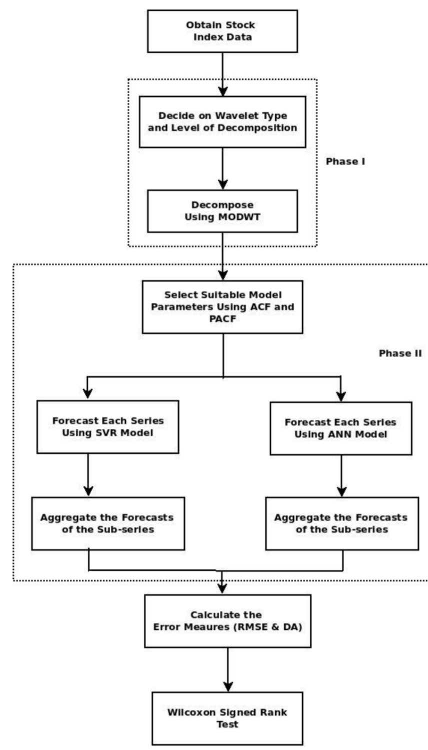


Fig. 1. **Flow chart for the proposed hybrid model [6]**

The authors propose a hybrid model for stock (the National Stock Exchange Fifty) price prediction, combining a decomposition model (Maximal Overlap Discrete Wavelet Transform, or MODWT) and some machine learning model (either ANN or SVR). These models will hereafter be referred to as the MODWT-ANN and the MODWT-SVR. The original time series data is first decomposed into some sub-series, in this case its wavelet coefficients from the MODWT using Haar basis. Each sub-series is then forecasted separately using the chosen machine learning model and recombined to acquire the final forecast. The error is calculated simply by comparing this forecast to the original data. Final comparisons were evaluated using the Wilcoxon-Signed Rank Test. This pipeline can be visualized in the flow chart given in Figure 1.

The above pipeline was performed for four different models: the ANN, the SVR, and the two hybrid models where MODWT was applied beforehand. The results showed both the root mean squared error (RMSE) and directional accuracies (DA %) improved when the data was projected onto a stationary wavelet basis in comparison to when the data was simply fed through the machine learning model.

II. Methods

Asset prices are both non-linear and non-stationary random processes, meaning their probability properties will vary over time. Therefore it is difficult to train a regressor (through some machine learning/statistical method) which predicts asset prices, as the probability of the underlying process keeps changing. However, it may be possible to project these asset prices onto a wavelet basis and decompose the random process into stationary components based on its wavelet level, similar to how the Discrete Wavelet Transform (DWT) outputs different numbers of coefficients based on the level at which you obtain them. These coefficients, as well as their derivatives, are stationary.

Now, Jothimani et al. tested this strategy by projecting National Stock Exchange Fifty prices onto a Haar basis [6]. Upon decomposing these prices onto stationary wavelet components, they applied multilayer perceptrons (MLP/ANN) and support vector regression (SVR) techniques, claiming both that wavelet-ANN & wavelet-SVR were better at predicting prices than non-Wavelet methods alone, and that a wavelet-SVR trading strategy would beat out traditional buy and hold. In light of their findings, our natural objectives were as follows

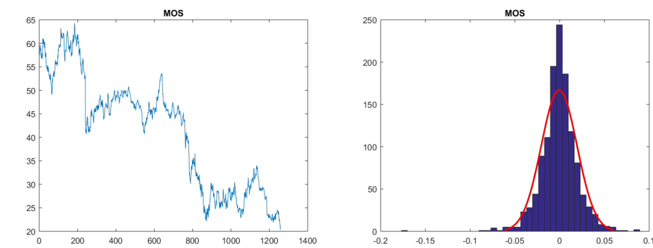
- Generalize their claims on stationarity, probability, and prediction
- Validate whether or not (and to what degree) these findings apply to assets in general
- Examine the effects of different wavelets on prediction and trading

A. S&P 500 Stock Data

The dataset used in these experiments was posted by Cam Nugent on kaggle, and includes a set of historical stock prices from the past five years (August 13, 2012 to August 11, 2017) for 447 companies on the S&P 500 Index. From source code posted on Github, we could import this data into a .mat file which included five things: four matrices corresponding to stock prices (opening, closing, high, low) and a key labeling all included stock names. Each matrix was of shape 1258 by 447, where each column represented a time series vector spanning almost five years for each individual stock. Figure 2a below shows an example of what such a vector may look like.

B. Stock Properties

Further examination of the time series data from MOS, *The Mosaic Company*, tells us a few key details about stocks. Figure 2b shows us the distribution of the returns from this stock, with a fitted probability density function. This is consistent with the literature, which asserts that the return on one point in time to the next can be modeled as independent samples from a Gaussian distribution [1]. "Returns" here is defined as $X_t/X_{t-1} - 1$



(a) Stock Price (b) Distribution

Fig. 2. **Time series stock prices for *The Mosaic Company* (MOS), as well as the distribution of its daily returns**

An Augmented Dickey-Fuller (ADF) test suggests that while the price is a non-stationary process (as we asserted earlier) and mostly uncorrelated with previous price points, the daily returns are a stationary process distributed along a clear bell curve. A plot of the autocorrelation, shown below in Figure 3, illustrates that, apart from the trivial max at the origin, the price at each day does not correlate very much with prices in the recent past.

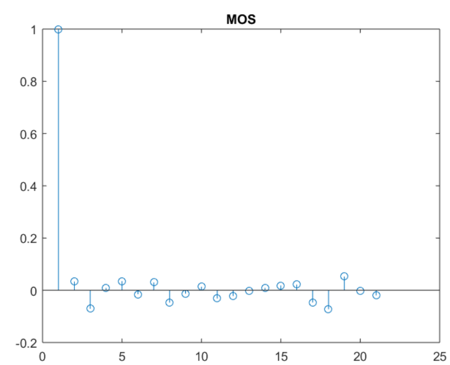


Fig. 3. **Autocorrelation of *The Mosaic Company* (MOS) returns**

C. Artificial Neural Networks

Let us begin with the naive approach and try to predict the price at time $t+1$ using the most recent 10 prices. The method will be a single hidden layer neural network with ten hidden nodes. The prediction is shown below in Figure 4, in red.

The prediction line seems to be an approximate single-delayed version of the price. This result supports our intuition. The daily return (NN-output) is independent of past returns (NN-input) but the daily returns are also distributed along a

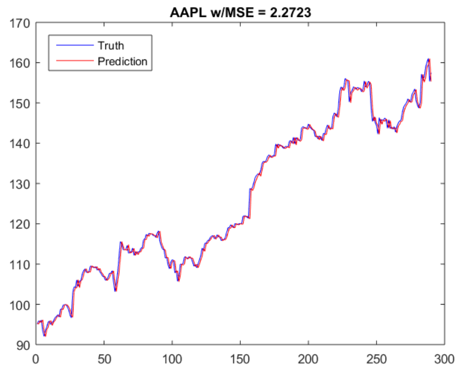


Fig. 4. **Price prediction of *Apple* (AAPL) using ANN**

zero-mean Gaussian distribution. Given no useful inputs, the neural network has learned a guessing rule that predicts that tomorrows return will be the expected value of the Gaussian distribution, zero. Though the MSE of price and prediction is small relative to the energy of the signal, a regressor that always predicts that tomorrows price is the same as todays is not useful for trading, which aims to buy low and sell high in general.

D. Maximally Overlapping Discrete Wavelet Transform

Jothimani et al. [6] suggested improving this result by decomposing the original non-stationary time series signal into some stationary sub-series through projection onto a wavelet basis. The following pipeline takes in times series stock data and applies the aforementioned process:

1) Apply the L-level maximally overlapping discrete wavelet transform (MODWT) on the input time series stock data. In particular, we perform a **3-level** decomposition.
2) This results in L sets of wavelet coefficients and 1 set of scaling coefficients, for a total of $L + 1$ coefficient sets.
3) Perform the inverse MODWT on one set of coefficients.
4) This results in $L + 1$ signals representing a decomposition of the original time series input signal. This can be thought of as a projection of the original signal onto different frequency bands via the wavelet transform.

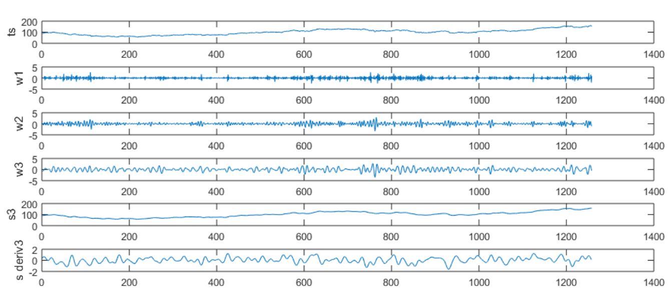


Fig. 5. **Projection of time series AAPL data onto db4 wavelet domain**

Figure 5 above features these time series wavelet/scaling components w1, w2, w3, s3, and s3' (the derivative) for *Apple* (AAPL) stock prices. It also displays the original time series data on top. Figure 6 below now displays autocorrelation for each of the aforementioned wavelet components. Here, we can see a clear correlation between present and past values, implying some stationarity in these components. In particular, the wavelet coefficients are stationary. The scalar coefficient s3 is not, but since its derivative s3' is we may just predict s3' instead of s3.

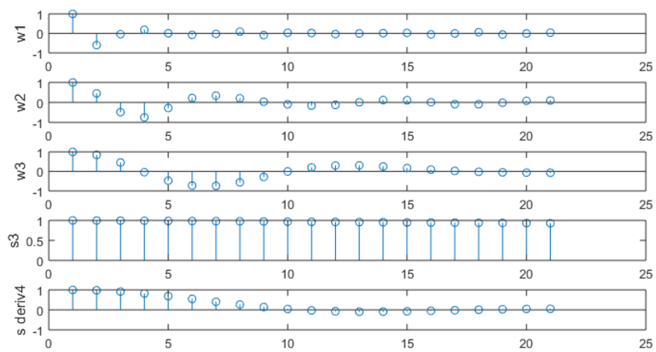


Fig. 6. **Correlations with respect to MODWT level 3 coefficients**

With this new representation, we may address two problems with the artificial neural net. The non-stationarity of the original signal meant that we cannot train the regressor due to the ever changing properties of the random processes. There was also no point in using machine learning or statistical methods if the future values are uncorrelated with present and past values. By looking at the signal in terms of its frequency bands, instead, we find that the process can be defined as a sum of stationary and past-correlated processes. In this case, we may apply statistical methods. This is the underlying logic behind a wavelet-ANN predictor.

Evaluation

From our results on AAPL stock data, we may safely assume the claims from the paper [6] were reasonable, as we were able to successfully extract stationary components from non-stationary input data. Now, we will attempt to perform the same analysis on all 447 stocks, as well as with different levels of MODWT decompositions. From here, we attempt to to determine how many companies follow certain stationarity/correlation requirements, thus making it viable for wavelet-ANN analysis.

TABLE II
CORRELATIONS WITH RESPECT TO LEVEL IN THE WAVELET TRANSFORM

Level (L)	Correlation (p)
1	1.0000
2	1.0000
3	1.0000
4	0.1969
5	0.0000

Each wavelet component needs to show some degree of correlation with recent signal values and little to no correlation with its long term past values. We define "recent past" as signal values not beyond a certain time $x[n-T]$, where T is small. Table II displays the results of checking for stationarity and recent correlation, using the Debauchies 4 Wavelet with $T = 10$. It is a successful demonstration of the ability for the MODWT to output stationary components in general.

E. Decision-Making Algorithm

Once the stock price prediction algorithm is sufficiently accurate, it is possible to make an informed decision on whether to buy, hold, or sell a certain stock. For this, we will use the same trading algorithm as used by Jothimani et al. [6]. For each individual stock, this algorithm follows three rules:

1) If its price tomorrow is greater than its price today, BUY the stock (if not held)
2) If its price tomorrow is smaller than its price today, SELL the stock (if held)
3) If we have held an open long position for three days and have been incorrect about its direction of price movement, SELL the stock (if held)

From this, an alpha value, or excess returns on investment compared to "buy and hold", is reported.

III. Results

After finalizing our full pipeline for making trading decisions on individual stocks, we may apply it to individual stocks to evaluate its performance with relation to the naïve buy and hold method. In particular, we shall compute an $L = 3$ level MODWT, then pass these $L + 1$ components into separate ANNs, each with two hidden layers, ten nodes each, and one output. A correlation significance threshold of 0.15 will be used to determine whether a trading decision can be made or not. The model will train on only the first 958 (out of 1258) data points in the time series, and the trading algorithm will attempt to decide to do for the last 300 data points.

First, let's test the algorithm on just the *Apple* (AAPL) stock using the Daubechies 4 wavelet for the MODWT. Figure 7 shows these individual wavelet components for AAPL. Figure 8 below that shows our results from applying the algorithm.

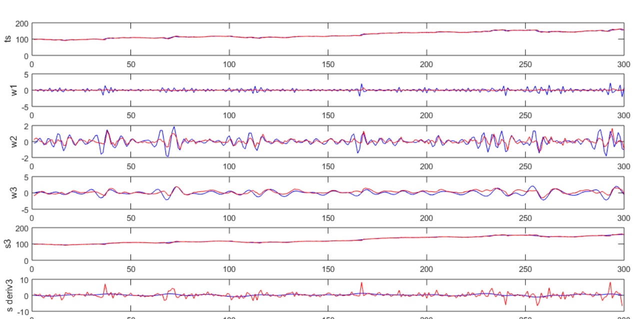


Fig. 7. **Individual db4 wavelet components of *Apple* (AAPL)**

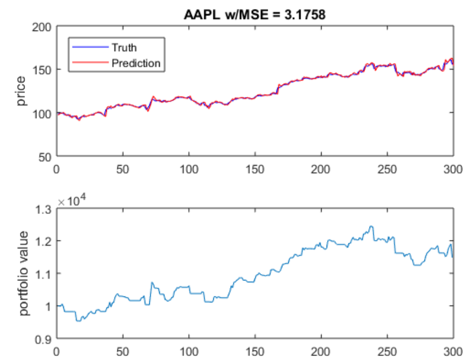


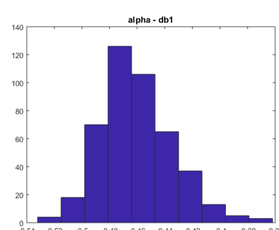
Fig. 8. **AAPL stock: (top) price prediction and (bot) resulting portfolio value using the db4 wavelet-ANN model**

While our alpha (-0.4908) was worse than the naïve method, the wavelet model did improve the mean squared error (3.1758). This confirms that the wavelet-ANN predictor is better at predicting than the ANN model alone, even though the algorithm was still worse than "buy and hold".

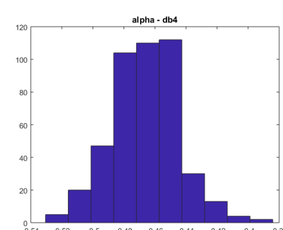
Now, we attempt to apply this method to all 447 stocks using three different wavelets for the MODWT (Haar, Daubechies 4, Daubechies 9). Figure 9 shows, for each wavelet, a histogram of alpha values obtained from running the algorithm on all stocks; Table III just above it summarizes the histograms by their mean, maximum, minimum, and standard deviation.

TABLE III
Alphas obtained from using different wavelets

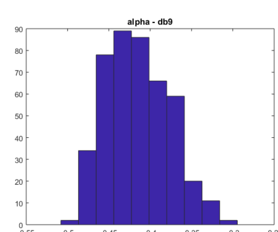
Alpha	Mean	Max	Min	Std
Haar	-0.4602	-0.3619	-0.5322	0.0264
db4	-0.4663	-0.3841	-0.5305	0.0216
db9	0.4147	-0.2946	-0.5081	0.0376



(a) Haar



(b) Daubechies 4



(c) Daubechies 9

Fig. 9. **Histograms of alphas obtained, separated by wavelet type used**

IV. Conclusion

Evaluation

Just as we discovered with the single AAPL input signal, the wavelet-ANN was able to decrease the mean squared error of the predicted stock data across most of the available stocks, signaling a clear improvement upon prediction methods using only an ANN model. However, the alpha value obtained from this method failed to improve upon the naïve method of buy and hold in all cases.

Comparison to Past Results

The paper in question, Jothimani et al., was able to use a similar trading algorithm to get better returns on the National Stock Exchange Fifty than the traditional buy and hold strategy [6], a result which we were unsuccessful in replicating here. It may just be the case that our implementation of the algorithm was poor. We did not use Support Vector Regression in our method, a machine learning model which the authors proved to perform much better than the ANN which we opted for instead.

However, it may also be possible that their results, with respect to trading, simply do not extend to stocks, or that their success here was simply an isolated incident due to the data they were using. They tested their method just once in the report, so it is not feasible for us to validate that their strategy would work for other financial data.

Future

Given more time, it would be beneficial to test this algorithm with much more data and many different models. Of course, implementing a better model like an SVM would have improved our prediction tremendously, possibly leading to better alpha values. In fact, since the paper failed to test with more than one type of wavelet [6] (and we exhibited much better MSE with db9 than the haar), implementing the SVM may yield improved results. As time was an issue for us (running the algorithm on batch stock data would take almost an hour), we could not test on many different correlation thresholds or ANN structures; without a time constraint we could tune ANN components to predict the data better.

Acknowledgment

This research was supported in part by Dr. Trac D. Tran, who supplied us more than enough prerequisite Wavelets knowledge to get us started on this project. It was a pleasure attending his course this semester.

References

[1] Abbassi, Noraddin Mousazadeh, et al. An Integrated System Based on Fuzzy Genetic Algorithm and Neural Networks for Stock Price Forecasting. International Journal of Quality & Reliability Management, vol. 31, no. 3, 2014, pp. 281292.

[2] Aminghafari, Mina, and Jean-Michel Poggi. Nonstationary Time Series Forecasting Using Wavelets and Kernel Smoothing. Communications in Statistics - Theory and Methods, vol. 41, no. 3, 2012, pp. 485499.

[3] Chandar, S. Kumar, et al. Prediction of Stock Market Price Using Hybrid of Wavelet Transform and Artificial Neural Network. Indian Journal of Science and Technology, vol. 9, no. 8, 2016.

[4] Edmundo G. De Souza E Silva, et al. Forecasting Oil Price Trends Using Wavelets and Hidden Markov Models. Energy Economics, vol. 32, no. 6, 2010, pp. 15071519.

[5] Hsieh, Tsung-Jung, et al. Forecasting Stock Markets Using Wavelet Transforms and Recurrent Neural Networks: An Integrated System Based on Artificial Bee Colony Algorithm. Applied Soft Computing, vol. 11, no. 2, 2011, pp. 25102525.

[6] Jothimani, D., Shankar, R., Yadav, S.S., (2015) Discrete Wavelet Transform-Based Prediction of Stock Index: A Study on National Stock Exchange Fifty Index, Journal of Financial Management and Analysis, 28 (2), 35-49.

[7] Karkanis, S. A., D. A. Karras, and B. G. Mertzios. "1-D Signal Prediction Using Wavelets." Available at: rtsimage.di.uoa.gr/ publications/ iwssip98.pdf. (2006).

[8] Manimaran, P., et al. Multiresolution Analysis of Fluctuations in Non-Stationary Time Series through Discrete Wavelets. Physica A: Statistical Mechanics and Its Applications, vol. 388, no. 12, 2009, pp. 23062314.

[9] Tan, Pei P., et al. A Wavelet-Based Evaluation of Time-Varying Long Memory of Equity Markets: A Paradigm in Crisis. Physica A: Statistical Mechanics and Its Applications, vol. 410, 2014, pp. 345358.

[10] Wang, Lipo, and Shekhar Gupta. Neural Networks and Wavelet De-Noising for Stock Trading and Prediction. Time Series Analysis, Modeling and Applications Intelligent Systems Reference Library, 2013, pp. 229247.