RESEARCH PAPERS OF WROCŁAW UNIVERSITY OF ECONOMICS No. 104
Business Informatics 16 2010

Jerzy Korczak, Krzysztof Drelczuk
Wrocław University of Economics, Wrocław, Poland
jerzy.korczak@ue.wroc.pl, kdrelczuk@hotmail.com

EFFECT OF WAVELET COMPRESSION OF HIGH FREQUENCY TIME SERIES ON THE QUALITY OF INFORMATION AND PREDICTION

Abstract: In recent times research work on the use of wavelet theory in data mining has increased significantly. In most cases, these works relate to specific applications. In this paper the general compression method of time series will be presented and adapted to financial time series analysis where dimensionality reduction is crucial. This hypothesis proposes that a double compression using Daubechies 4 wavelet does not significantly affect the quality of information carried by a time series. The reduction of dimensionality significantly affects the algorithmic complexity and improves its quality of prediction. In order to verify this hypothesis the highly frequent time series will be evaluated in terms of forecasting quality where future value is predicted only on the basis of the past quotations. In this project as a predictive algorithm we used ARAR due to its good results in forecasting of the real financial time series.

Keywords: time series analysis, discrete wavelet transform, Daubechies 4, prediction.

1. Introduction

Empirical evidence shows that the dimensionality reduction not only significantly affects the computing time of the classifiers, but also the quality of classification results, whereas in Euclidean space increasing the number of dimensions diminishes the distance between vectors. This has particularly important impact on the process of clustering. The space of solutions that is divided into clusters has the same dimensionality as entering vectors. In the case of financial time series where data are taken from, for example, 60 recent observations, the space which the algorithm will have to share will be 60-dimensional. Clustering consists of assigning elements to the respective clusters by means of pre-defined metrics (for example, the Euclidean norm). With a large number of dimensions of difference between the nearest and farthest neighbour it becomes less important and it is a serious obstacle to partition the space into the significant clusters [Beyen et al. 1999]. Too many dimensions can also cause overlapping multidimensional clusters impeding effective classification.

This hypothesis sets out to prove that double compression by Daubechies 4 wavelet does not significantly affect the quality of information carried by a time series in comparison to the original, raw time series. In other words, the double wavelet compression does not influence the deterioration of the time series, as the information source. In the case of clustering where the computational complexity is exponential, proving such an assumption has a significant impact on the usefulness of the clustering algorithms. Taking also into account the reduction of distance between clusters in the Euclidean space with increasing dimensionality, this demonstration would positively influence the quality of the prediction of time series.

In this work the authors made use of discrete wavelet transform for lossless compression of time series; lossless in the sense of preservation of the same quality of information as an untreated time series. In various systems of time series analysis, such as predictive systems, classification systems, archiving systems, the possibility to assure the lossless compression is of great importance. In the next section of this paper the wavelet theory will be briefly presented and detailed later in the fourth section on Daubechies 4 wavelets. Then in section 3, the classification algorithm, ARAR, will be presented. In the section 5 the experiments will be described and selected time series. In the concluding part we have collected the results of empirical research on FOREX time series, which have been used to verify the hypothesis.

2. Discrete wavelet transform

Discrete wavelet transform, first described in [Mallat 1989], is very often applied in preliminary data analysis. With it one can reduce the number of dimensions of input vector to the target system, such as the classifier or predictive system, as well as remove some of the information considered as noise or data redundancies, in terms of Shannon's lossless data compression [Shannon 1948]. In most cases, the signal or function can be better explored, described and processed, when it can be defined as a linear combination of the form:

$$f(t) = \sum_{\ell} a_{\ell} \psi_{\ell}(t), \tag{1}$$

where $l \in Z$ denotes the index of a sum (finite or infinite), but ψ_{ℓ}, $a_{\ell} \in \Re$ is a collection of real functions. If this distribution is well-defined, then the base set of functions ψ_t is called a functional space (instead of the vector space where the vectors have the features, while the scalars are real or complex numbers). If the scalar product of all the functions ψ_{ℓ} is equal to zero, then the base is called orthogonal, and can be written as:

$$\langle \psi_k(t), \psi_{\ell}(t) \rangle = \int \psi_k(t) \cdot \psi_{\ell}(t) dt = 0, \underset{k,\ell \in Z}{\forall} k \neq \ell. \tag{2}$$

This allows us to determine the coefficients a_{ℓ} using scalar product as follows:

$$a_{\ell} = \langle f(t), \psi_{\ell}(t) \rangle = \int f(t) \cdot \psi_{\ell}(t) dt. \tag{3}$$

Thus the defined space, spanned on the functions ψ_k, is called Hilbert space $L^2(\Re)$or the space of integrable squared functions. To recall, the real variable function $f(t)$ belongs to the space $L^2(\Re)$ if and only if $|f(t)|^2$ it is integrable, that is, if:

$$\int_{t\in\Re} |f(t)|^2 \, dt < \infty \tag{4}$$

with the metric described as:

$$\|f\| = \sqrt{\langle f, f\rangle} = \sqrt{\int_{t\in\Re} |f(t)|^2 \, dt}. \tag{5}$$

A wavelet is defined as a real function $\psi(u) \in L^2(\Re)$ [Daubechies 1992] if the following constraints are satisfied:

$$\begin{aligned} &\int_{-\infty}^{\infty} \psi^2(u)\, du = 1, \\ &\int_{-\infty}^{\infty} \psi(u)\, du = 0. \end{aligned} \tag{6}$$

The wavelet transform will be called a set of functions $\psi_{a,b}$, such as:

$$\psi_{a,b}(t) = w(a) \cdot \psi\left(\frac{t-b}{a}\right) dt, \tag{7}$$

where a and b are scaling parameters, $w(a)$ is the weighting function to ensure that the wavelet energy does not change with the change of scale, i.e. $\|\psi_{a,b}\| = \|\psi\| = a \in \Re^+$. In most cases, in formula (7) $a^{-1/2}$ replaces $w(a)$. Then the wavelet transform becomes:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) dt. \tag{8}$$

However, the formula is suitable only for continuous signals. In this paper, the input signal is discrete. To apply this transform it is necessary to sample the input signal. This will be done using a logarithmic discretization of scale a depending on the size of the step between the distances from the next parameter b. Such a discrete wavelet transform has the form:

$$\psi_{m,n}(t) = \frac{1}{\sqrt{a_0^m}} \cdot \left(\frac{t - nb_0 a_0^m}{a_0^m}\right), \tag{9}$$

where a_0 and b_0 are parameters defining the orthogonal base of function $\psi_{a,b}$ in space L^2. Most popular values of the parameters are 2 and 1 [Daubechies 1992]. We then say about a dyadic sampling. The wavelet then takes the following form:

$$\psi_{m,n}(t) = 2^{-m/2} \cdot \psi\left(2^{-m} t - n\right). \tag{10}$$

Distribution of entry signal is named multi-resolution signal [Mallat 1989]. Signal $f(t) \in L^2$ is decomposed into the constituents localized in the sub-spaces spanned on the scaling functions. The scaling function has the same form as the wavelet:

$$\varphi_{m,n}(t) = 2^{-m/2} \cdot \varphi\left(2^{-m} t - n\right). \tag{11}$$

Signal belonging to the space L^2 can be defined by:

$$x_m(t) = \sum_{n=-\infty}^{\infty} S_{m,n} \cdot \varphi_{m.n}(t), \tag{12}$$

where approximation coefficients $S_{m,n}$ are defined as follows:

$$S_{m,n} = \int^{\infty} x(t) \cdot \varphi_{m,n}(t) dt. \tag{13}$$

The procedures of ψ searching having ϕ are well defined in the literature [Daubechies 1992, chap. 5.1]. One possibility, described in [Daubechies 1992] and applied in our approach, consists of definition of wavelet function in the following way:

$$\psi(t) = \sum_{n} (-1)^n h_{n-1} \sqrt{2} \cdot \varphi(2t - n). \tag{14}$$

Signal included in the compliment of the sub-space spanned on wavelet functions can be computed from:

$$d_m(t) = \sum_{n=-\infty}^{\infty} T_{m,n} \cdot \varphi_{M.n}(t), \tag{15}$$

where the transform is defined as follows:

$$T_{m,n} = \int_{-\infty}^{\infty} x(t) \cdot \psi_{m,n}(t) dt. \tag{16}$$

Complete signal in L^2 can be obtained from the following expression:

$$x(t) = \sum_{n=-\infty}^{\infty} S_{m,n} \cdot \varphi_{m.n}(t) + \sum_{m=-\infty}^{m_0} \sum_{n=-\infty}^{\infty} T_{m,n} \cdot \varphi_{m.n}(t). \tag{17}$$

Summing up, the signal is represented as the synthesis of approximation on a given level (12) and sum of details until a given level included (15). Approximations are a compressed time series that carry out the same information in terms of Shannon.

3. Algorithm ARAR

The chosen algorithm ARAR is a modification of previous algorithm ARARMA [Newton, Parzen 1984]. Its characteristic is the application of *memory-shortening* transformation for each time series, and then fitting with a model ARMA [Brockwell, Davis 2002]. Suppose that we have to find in a given time series $\{Y_t, t = 1, 2, \ldots, n\}$ its lengths of time dependency. There exist 3 possibilities:

1. D series $\{Y_t\}$ has a long time dependency.
2. W series $\{Y_t\}$ has a relatively long time dependency.
3. K series $\{Y_t\}$ is a short memory.

If it is proven that the time series is of type D or W, then the transformation is executed until the time series becomes of type K.

The specification below describes the algorithm of classification.

ARAR description

1. For each $\tau = 1,2,3,..,15$ find a value $\hat{\varphi}(\tau)$ such that

$$E(\varphi,\tau) = \frac{\sum_{t=\tau+1}^{n}\left(Y_t - \varphi Y_{t-\tau}\right)^2}{\sum_{t=\tau-1}^{n} Y_t^2}$$

is minimal

2. Let us define

$$E(\tau) = E\left(\hat{\phi}(\tau),\tau\right),$$

and delay $\hat{\tau}$ in a such way that value τ, gives minimum of $E(\hat{\tau})$. Then, if:

1) $E(\hat{\tau}) \le 8/n$ time series is a type D.
2) $\hat{\phi}(\hat{\tau}) \ge 0.93$ and $\hat{\tau} > 2$ time series is a type *D*
3) $\hat{\phi}(\hat{\tau}) \ge 0.93$ and $\hat{\tau} = 1 \vee 2$ describe the values $\hat{\phi}_1, \hat{\phi}_2$, where $\sum_{t=3}^{n}\left(Y_t - \varphi_1 Y_{t-1} - \varphi_2 Y_{t-2}\right)^2$ is minimal; time series is a type W.
4) $\hat{\phi}(\hat{\tau}) < 0.93$ time series is a type K.

3. Define the time series after transformation $\{S_t, t=k+1, \ldots, n\}$ and $\overline{S}$ as average of $S_{k+1}, \ldots, S_n$. Autoregressive model is

$$X_t = \varphi_1 X_{t-1} + \varphi_{l_1} X_{t-l_1} + \varphi_{l_2} X_{t-l_2} + \varphi_{l_3} X_{t-l_3} + Z_t,$$

where $Z_t \sim N\left(0,\sigma^2\right)$, and for delays l_1, l_2 i l_3, coefficients φ_j and variance σ^2 computed from the Yule-Warker equation:

$$\begin{bmatrix} 1 & \hat{\rho}(l_1-1) & \hat{\rho}(l_2-1) & \hat{\rho}(l_3-1) \\ \hat{\rho}(l_1-1) & 1 & \hat{\rho}(l_2-l_1) & \hat{\rho}(l_3-l_1) \\ \hat{\rho}(l_2-1) & \hat{\rho}(l_2-l_1) & 1 & \hat{\rho}(l_3-l_2) \\ \hat{\rho}(l_3-1) & \hat{\rho}(l_3-l_1) & \hat{\rho}(l_3-l_2) & 1 \end{bmatrix} \begin{bmatrix} \varphi_1 \\ \varphi_{l_1} \\ \varphi_{l_2} \\ \varphi_{l_3} \end{bmatrix} = \begin{bmatrix} \hat{\rho}(1) \\ \hat{\rho}(l_1) \\ \hat{\rho}(l_2) \\ \hat{\rho}(l_3) \end{bmatrix}$$

and

$$\sigma^2 = \hat{\gamma}(0)\left[1-\varphi_1\hat{\rho}(1)-\varphi_{l_1}\hat{\rho}(l_1)-\varphi_{l_2}\hat{\rho}(l_2)-\varphi_{l_3}\hat{\rho}(l_3)\right]$$

where $\hat{\gamma}(0)$ and $\hat{\rho}(j)$, j=0, 1, 2, …, auto-covariance and auto-correlation of time series $\{Y_t\}$.

The above described algorithm will be applied for prediction and as a reference for comparison of source and compressed time-series.

4. Compression using Daubechies 4 wavelet

Daubechies wavelets are a family of orthogonal wavelets described in detail in [Daubechies 1992]. Wavelets, labelled D2-D20 (only the even index denotes nonzero coefficients of the scaling functions), are very often used, inter alia, because of very low computational complexity, that is $O(n)$. Comparatively, a widely used Fast Fourier Transformation (FFT) has the computational complexity $O(n \cdot \log n)$.

In our project, the applied wavelet is D4 wavelet with four coefficients of scaling function.

The scaling function for $n = 4$ can be defined as follows:

$$\varphi(t) = \sqrt{2}\left(h_0\varphi(2t) + h_1(2t-1) + h_2(2t-2) + h_3(2t-3)\right)$$

and its corresponding wavelet function:

$$\psi(t) = \sqrt{2}\left(h_3\varphi(2t) - h_2(2t-1) + h_1(2t-2) - h_0(2t-3)\right).$$

Solving the above equation, assuming the orthogonality of wavelets, a normalizing parameter $\sqrt{2}^{-1}$, the following coefficients are obtained:

$$h_0 = \frac{1+\sqrt{3}}{4} \quad h_1 = \frac{3+\sqrt{3}}{4} \quad h_2 = \frac{3-\sqrt{3}}{4} \quad h_3 = \frac{1-\sqrt{3}}{4}$$

and

$$h_0 = \frac{1-\sqrt{3}}{4} \quad h_1 = \frac{3-\sqrt{3}}{4} \quad h_2 = \frac{3+\sqrt{3}}{4} \quad h_3 = \frac{1+\sqrt{3}}{4}.$$

The first set of coefficients corresponds to the scaling function $\phi(t)$, the second one to $\phi(-t)$.

Decomposing the input signal into components, each one step gets rid of a certain amount of signal interferences. Given the space orthogonality, we can be sure that there is no redundancy and every detail will be eliminated once. The input signal can be represented as the sum of the following details with the last approximation. Signal decomposition is performed recursively using the formula for synthesizing approximation at a given level and the sum of details of the level inclusively. This is illustrated in Figure 1.

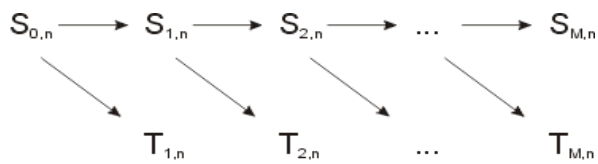


Figure 1. Time series decomposition

Source: [Burrus 2001].

One may notice that n is successively doubled at each iteration; for $m = 1$ we have $2^M/2^1 = N/2$ approximation coefficients, for $m = 2$ there is $2^M/2^2 = N/4$. Decomposing a vector of 8 elements ($S_{0,0}, S_{0,1}, S_{0,2}, S_{0,3}, S_{0,4}, S_{0,5}, S_{0,6}, S_{0,7}$) after the first iteration a vector ($S_{1,0}, S_{1,1}, S_{1,2}, S_{1,3}, T_{1,0}, T_{1,1}, T_{1,2}, T_{1,3}$) is obtained, after the next ($S_{2,0}, S_{2,1}, T_{2,0}, T_{2,1}, T_{1,0}, T_{1,1}, T_{1,2}, T_{1,3}$), and so on. This imposes a constraint on the length of entry vector that must be a power of 2. Automatically, it returns the length of the compressed vector (after removing details) that is also a power of 2 as illustrated on Figure 2.

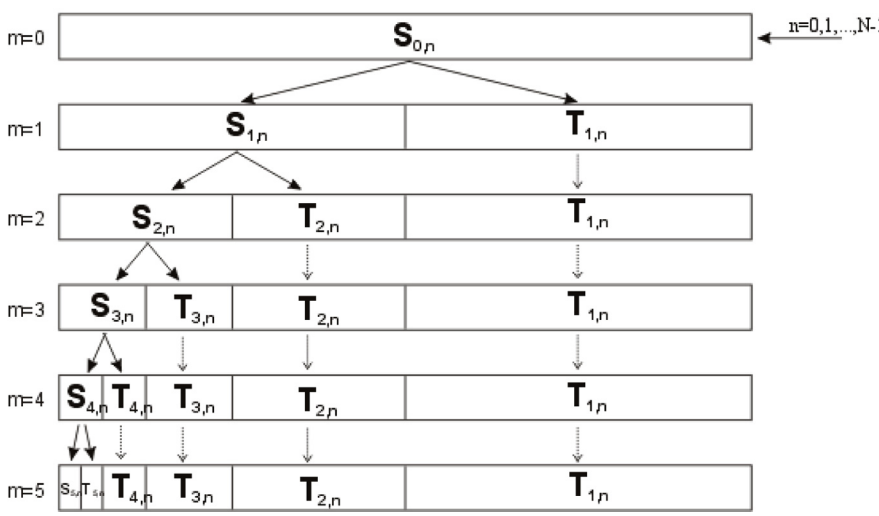


Figure 2. Schema of vector decomposition

Source: [Burrus 2001].

Below the source code of the compression algorithm is presented to show its simplicity and low algorithmic complexity.

Extract from the source code of discrete wavelet transformation

```
public static double[] D4Transform(double[] input, int approximationSize)
{
    double h0 = (1 + Math.Sqrt(3)) / 4 * Math.Sqrt(2);
    double h1 = (3 + Math.Sqrt(3)) / 4 * Math.Sqrt(2);
    double h2 = (3 – Math.Sqrt(3)) / 4 * Math.Sqrt(2);
    double h3 = (1 – Math.Sqrt(3)) / 4 * Math.Sqrt(2);
    double g0 = h3;     double g1 = -h2;
    double g2 = h1;     double g3 = -h0;
    int I = 0, j = 0;
    int half = approximationSize >> 1;
    double[] tmp = new double[approximationSize << 1];
    for (j = 0; j < approximationSize – 3; j = j + 2)
    {
      tmp[i] = a[j] * h0 + a[j + 1] * h1 + a[j + 2] * h2 + a[j + 3] * h3;
      tmp[I + half] = a[j] * g0 + a[j + 1] * g1 + a[j + 2] * g2 + a[j + 3] * g3;
      i++;
    }
    tmp[i] = a[n – 2] * h0 + a[n – 1] * h1 + a[0] * h2 + a[1] * h3;
    tmp[I + half] = a[n – 2] * g0 + a[n – 1] * g1 + a[0] * g2 + a[1] * g3;
    return tmp;
}
```

Input parameters are the input vector to be subjected to transformation (with a length which is the power of two) and the number of approximation coefficients which are to be created from it (which is also the power of two). The result is an array of approximation and details. For example, if we introduce a time series $S_{0,0}, S_{0,1}, S_{0,2}, S_{0,3}, S_{0,4}, S_{0,5}, S_{0,6}, S_{0,7}$, and the second parameter 4, then the result will be a table of four approximations and four details: $S_{1,0}, S_{1,1}, S_{1,2}, S_{1,3}, T_{1,0}, T_{1,1}, T_{1,2}, T_{1,3}$.

The wavelet D4 has been chosen because of its high-speed (low computational complexity) and the simplicity of implementation. In addition, it has been well studied in the literature from the viewpoint of its usefulness in the pre-processing efficiency of highly frequent time series.

5. Experiments

The tests have been carried out in two stages. In the first stage a single, randomly selected, financial time series was examined, extracted from the period of two months. The aim was to examine the different behaviour of financial time series. In

the second stage, one financial time series was selected corresponding to 12 hours of quotations divided into equal size parts using sliding windows. The goal was to discover changes in time series. This study of the impact of wavelet compression on the time series information using wavelets Daubechies 4 was carried out on the currency market FOREX. This market operates 24 hours a day, from Sunday 11.00 pm to 10.00 pm on Friday (according to Central European Time). According to statistics published in 2008, most transactions concerned the pairs [Bank for International Settlements 2008]:

– EUR/USD 27%,
– USD/JPY 13%,
– GBP/USD 12%.

Given these observations the authors have chosen the time series describing historical transactions for these three pairs of currencies. The selection of samples for the testing were purely random. The intervals (the first day and the last day of the month) were generated randomly using the pseudo-generator from the platform . NET. The generated values indicate days of the time series to evaluate. To ensure the objectivity of research, we have drawn five time series of two consecutive months for the same days (if the number indicated the day when the FOREX is close, the next closest date was taken on which the transactions take place), and all pairs (EUR/USD, USD/JPY, GBP/USD) have been selected from those days. In this way, 30 time series have been extracted for the experiment. Time series of the length of 256 have been created from the aggregated data to one minute; the first value represents the aggregated transactions within the first minute of the day (0:01),the last transactions 256 minutes later (at 4:17). The time series were grouped within the pairs of currencies. So the result was three sets of data, three sets of the average relative errors (computed for a single currency).

The test consisted of two phases. In the first, we have examined the amount of information in time series, evaluating the effectiveness of the prediction algorithm ARAR. Each of the selected series was divided into two series of length of 128. The first 128 values served as a learning set, and 20 consecutive values (the first 20 values from the second series) were treated as a validation set. Then, the mean relative error (*MRE*) for each of the quotes (from 1 to 20) was calculated according to the formula:

where x_0 is an expected value, x is a real value. *MRE* was obtained by dividing *RE*

$$RE = \frac{\Delta x}{x} = \frac{x_0 - x}{x} = \frac{x_0}{x} - 1,$$

into a number of samples.

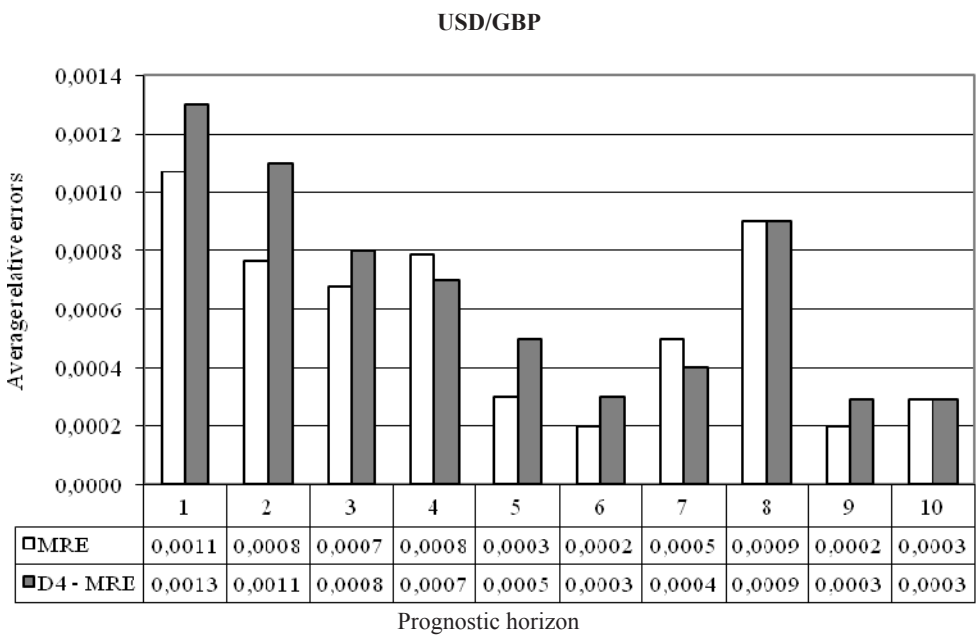
The second stage was to demonstrate if the quantity and quality of information afforded by the compressed time series significantly differed from the uncompressed one. To achieve this a series of length 256 was compressed by the discrete transform wavelet. The resulting series of length 128 (twice the compression) was divided into

two series of length of 64. Values of the first series have been used as a learning set, and a set of next 20 values (the first 20 values from the second row) as a validation set. The last step was the calculation of the average relative errors (*MRE*) for each of the quotes using the same formula as for the series of the first stage.

As a result, three groups of the test series were performed, which included the values of the average relative errors ranked according to the period. Recall that the objective of this study was to demonstrate that the double wavelet compression did not significantly worsen the average prediction error. Such a result confirmed the hypothesis put at the beginning of the work that the double compression using wavelet Daubechies 4 does not affect significantly the quality of information carried by a time series in comparison with the original ones. To validate the hypothesis the compatibility the Kołmogorov–Smirnov test was applied. The test confirmed that the two populations had the same distribution, which is equivalent to saying that the two samples come from the same population.

In this project the algorithm ARAR for time series prediction and the authoring program to compress the time series were applied. In the experiments, the randomly selected quotes from January 8, 12, 16, 17, 19, 20, and February 9, 12, 14, 17, 19, 2009 were tested. For the second study, the quotes come from February 25, 2009.

The average relative errors for the original and transformed time series are presented in Figures 3-5. The differences are practically negligible. Comparison of



	1	2	3	4	5	6	7	8	9	10
MRE	0,0011	0,0008	0,0007	0,0008	0,0003	0,0002	0,0005	0,0009	0,0002	0,0003
D4 - MRE	0,0013	0,0011	0,0008	0,0007	0,0005	0,0003	0,0004	0,0009	0,0003	0,0003

Figure 3. Comparison of average relative errors for original and compressed time series. Pair USD/GBP

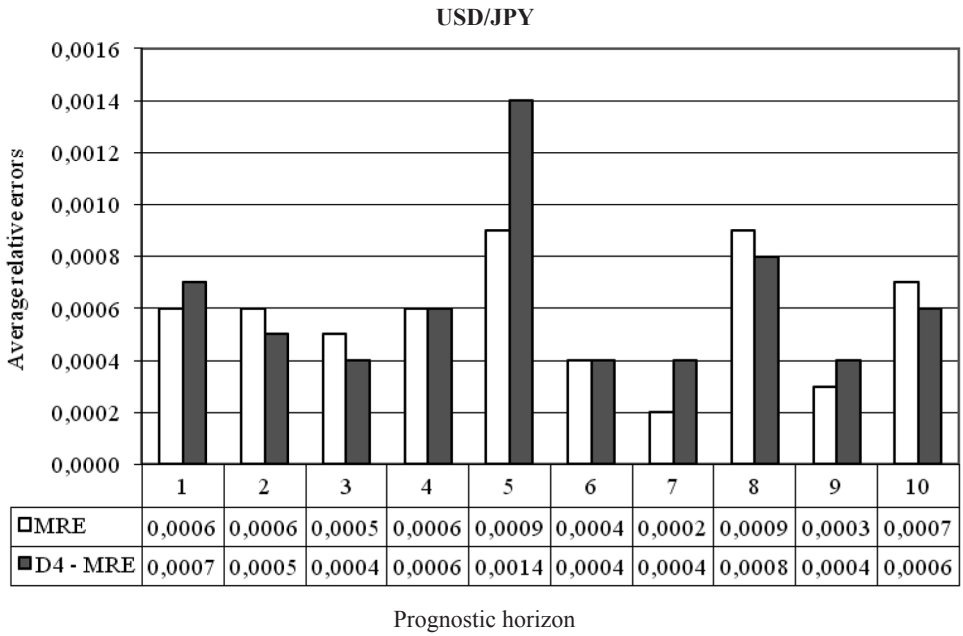


Figure 4. Comparison of average relative errors for original and compressed time series. Pair USD/JPY

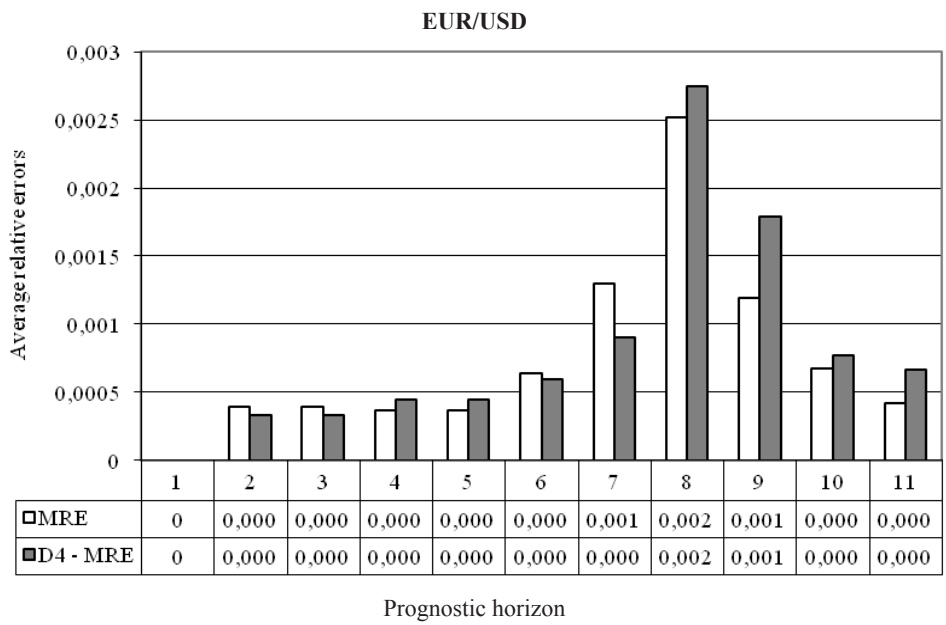


Figure 5. Comparison of average relative errors for original and compressed time series. Pair EUR/USD

the cumulative average of errors is shown in Figures 6-8. The solid lines represent the values for the transformed time series, while the dashed lines illustrate original ones. Values are almost similar; only in case of USD/GBP the difference is greater. When comparing the maximum and minimum values we have noticed that in some cases better prediction results were achieved using the transformed time series, and

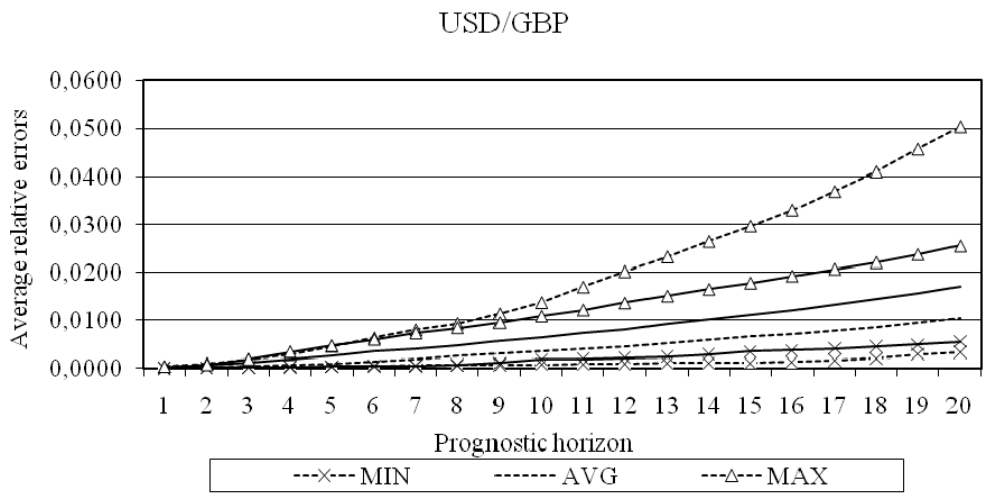


Figure 6. Comparison of average cumulative errors (max, min and average) for original and compressed time series. Pair USD/GBP

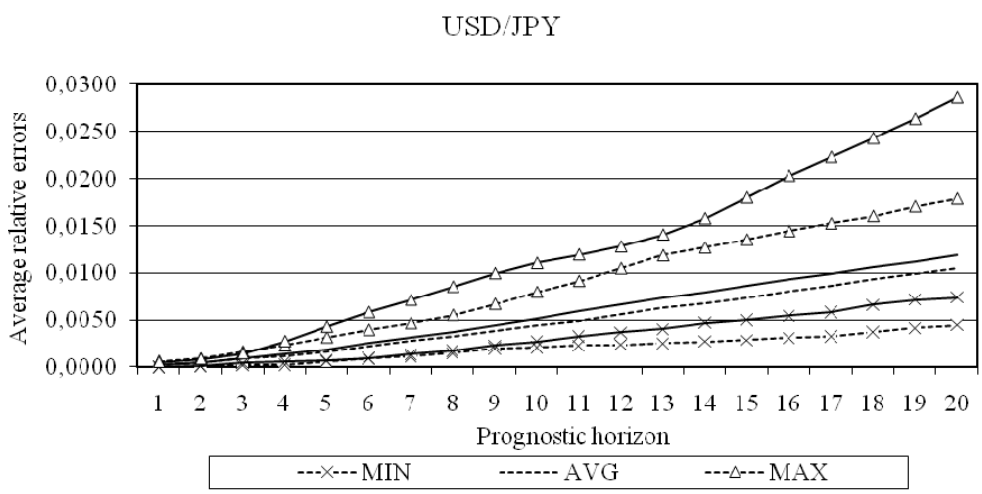


Figure 7. Comparison of average cumulative errors (max, min and average) for original and compressed time series. Pair USD/JPY

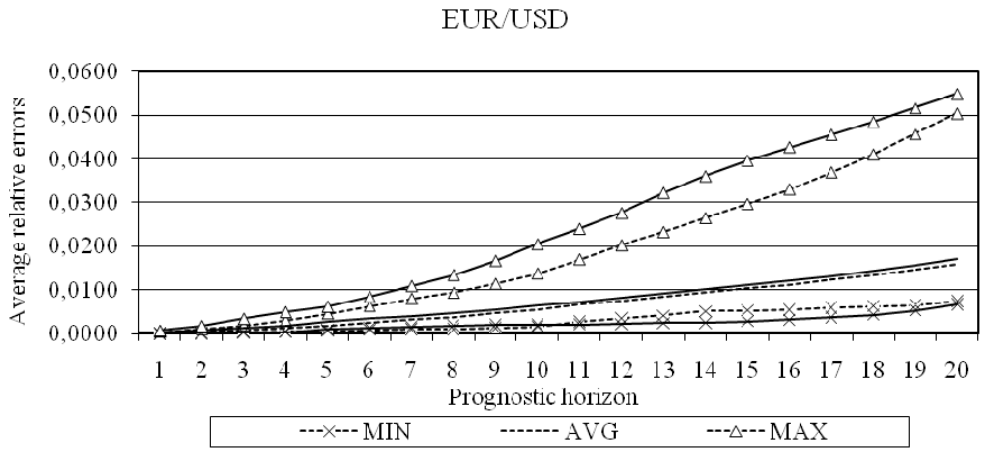


Figure 8. Comparison of average cumulative errors (max, min and average) for original and compressed time series. Pair EUR/USD

in some, when predicting based on the original time series. Although in these cases the differences were small (except the USD/GBP where the wavelet compression significantly improved the least effective prediction).

The goodness-of-fit Kołmogorov–Smirnov test was also carried out. Null hypothesis was assumed that the distributions of the average relative prediction error in the original and compressed time series were the same. To validate the hypothesis the following statistic has been used:

$$\lambda_n = \sqrt{n} \cdot \sup\left|F_{n_1}(x) - F_{n_2}(x)\right|,$$

where

$$n = \frac{n_1 \cdot n_2}{n_1 + n_2}$$

and $F_{n_1}(x), F_{n_2}(x)$ are the empirical distribution functions computed on the basis of samples. Graphical presentation of the distribution is shown in Figure 9.

Values n_1, n_2 mean the sum of averages of relative errors of predictions. The value of empirical statistic was computed and it was equal to 0.004468. The limit λ-Kołmogorov distribution at the confidence level α = 0.01 is equal to 1.63. So, based on the relation:

$$\lambda_e < \lambda_\alpha$$

there is no reason to reject the null hypothesis. So, we say with 99% probability that the average error distributions are the same in the case of time series transformed by Daubechies 4 wavelet and as the raw time series.

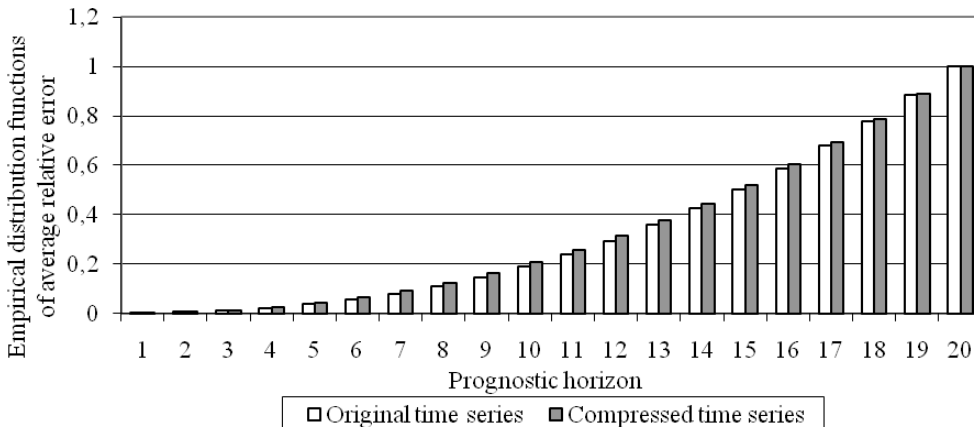


Figure 9. Comparison of average error distributions for original and compressed time series

In the second experiment, the differences were less noticeable for the average and minimum values. However, in all three cases the results were better for compressed time series than the uncompressed one.

Taking into consideration that the predictions were made on double-compressed time series, it can be stated that the result was very encouraging. Not only was the computing time reduced, but also the prediction accuracy was improved.

6. Conclusions and future works

The results of this research have confirmed the hypothesis established at the beginning of the work that the information carried by uncompressed time series is qualitatively identical to the information carried by double-compressed time series using Daubechies 4 wavelet. The consequences are important. Colloquially speaking, it makes no sense to use the original time series since the use of a time series of two times shorter (after compressing by D4) assures the same results. Given the computing complexity of the classification algorithms, it is of utmost importance. It should be also noted that in the case of a long time series (covering 12-hours period) wavelet compression improved the quality of prediction. This would mean that, at least for these three examined time series, noise and redundant information have been eliminated by the compression process.

To determine the usefulness of wavelet compression in financial time series in general, it would be recommended to test them on significantly greater empirical material coming from various stock markets. One can also consider trying other wavelets of the Daubechies family. In the paper, we were focused on the computational complexity and its reduction in the context of prediction systems. It should be mentioned here that the wavelet D1 has lesser complexity than the D4, which could make it more useful for larger data sets or in real time systems. But one must have in

mind that D1 is less sensitive to subtle, local changes in the original time series, and it is less efficient in analysis of highly frequent time series.

Summing up, although we have indicated that further research is required, the use of compression is fully justified if we are interested to reduce the multidimensional space and we do not want to lose any significant information contained in the original time series.

References

Bank for International Settlements (2008), Triennial Central Bank Survey, http://www.bis.org/publ/rpfxf07t.pdf.

Beyen K., Goldstein J., Ramakrishnan R., Shaft U. (1999), When is nearest neighbor meaningful?, [in:] *Proceedings of the 7th International Conference on Database Theory*, Eds. G. Goos, J. Hartmanis, J. Van Leeuwen, Jerusalem, pp. 217-235.

Brockwell P.J., Davis R.A. (2002), *Introduction to Time Series and Forecasting*, Springer, New York.

Burrus C.S. (2001), *Introduction to Wavelets and Wavelet Transform*, Prentice Hall, New York.

Daubechies I. (1992), *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia.

Mallat S.G. (1989), A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, pp. 647-693.

Newton H.J., Parzen E. (1984), Forecasting and time series model types of 111 economic time series, [in:] *The Forecasting Accuracy of Major Time Series Methods*, Eds. S. Makridakis et al., John Wiley and Sons, New York–Chichester.

Shannon A. (1948), A mathematical theory of communication, *Bell System Technical Journal*, Vol. 27, No. 3, pp. 379-423 & 623-656.

WPŁYW KOMPRESJI SZEREGÓW CZASOWYCH O DUŻEJ CZĘSTOTLIWOŚCI FALKĄ DAUBEUCHIES 4 NA JAKOŚĆ ZAWARTYCH W NICH INFORMACJI

Streszczenie: ostatnio ilość prac na temat zastosowań teorii falek w eksploracji danych znacząco wzrasta. W większości przypadków prace te dotyczą zastosowania jej z konkretnym algorytmem. W niniejszym opracowaniu będzie zaprezentowana ogólna metoda kompresji szeregów czasowych, która będzie mogła być łatwo zaadaptowana do wielu problemów, w których redukcja wielowymiarowości ma kluczowe znaczenie.. W pracy tej autorzy spróbują udowodnić hipotezę, iż dwukrotna kompresja falką Daubechies 4 nie wpływa znacząco na jakość informacji niesioną przez szereg czasowy w stosunku do szeregu nieskompresowanego lub inaczej, dwukrotna kompresja falką Daubechies 4 nie wpływa znacząco na pogorszenie jakości szeregu czasowego jako nośnika danych. W celu weryfikacji hipotezy badane szeregi czasowe zostaną ocenione pod kątem jakości prognozy algorytmu, który przewiduje przyszłe wartości jedynie na podstawie analizy informacji niesionej przez wartości przeszłe. Jako algorytm predykcyjny został użyty ARAR ze względu na bardzo dobre wyniki w prognozowaniu rzeczywistych finansowych szeregów czasowych, a także dlatego, iż został on dokładnie zbadany i opisany w literaturze tematu.