

A Comparison of Two Hölder Regularity Functions to Forecast Stock Indices by ANN Algorithms

Yeliz Karaca[1(✉)] and Carlo Cattani[2,3]

[1] University of Massachusetts Medical School, Worcester, MA 01655, USA
yeliz.karaca@ieee.org, yeliz.karaca@umassmemorial.org
[2] Engineering School (DEIM), University of Tuscia, Viterbo, VT 01100, Italy
cattani@unitus.it
[3] Ton Duc Thang University, Ho Chi Minh City, Vietnam

Abstract. This study is concerned with local Hölder exponent as a function regularity measure for time series regarding Finance data. The study examines 12 attributes in the Micro, Small and Medium Enterprises (MSME) data for 6 regions (2004–2018). The study has two different approaches. Firstly, (i) multi fractal methods and Brownian motion Hölder regularity functions were used for the identification of the significant and self-similar attributes in the Finance dataset. The steps are: (a) polynomial functions were applied on the Finance dataset and p_finance dataset was obtained. (b) exponential functions were applied and e_Finance dataset was obtained. Secondly, (ii) Artificial Neural Network (ANN) algorithms ((Feed Forward Back Propagation (FFBP) and Cascade Forward Back Propagation (CFBP)) were applied on both datasets in a and b. Finally, (iii) classification accuracy rates as per the outcomes obtained from the second stage by the ANN algorithms were compared. Our study has been conducted for the first time in the literature since with the ANN algorithms' application, it revealed how the most significant attributes are identified in the Finance dataset by Hölder functions (polynomial as well as exponential).

Keywords: Singularity · Hölder regularity · Finance data · Multifractals · ANN

1 Introduction

Feature selection aims to choose a minimal feature subset from a problem area while an appropriately high accuracy is retained for the representation of the original data, which is an essential method with regard to data preprocessing [1,2]. The main principle of fractal analysis is to seek a pattern which is not dependent on scale. A multifractal system is a generalization of a fractal system where a single exponent (the fractal dimension) would not suffice to perform the description of its dynamics. Rather, a continuous spectrum of exponents, namely

© Springer Nature Switzerland AG 2019
S. Misra et al. (Eds.): ICCSA 2019, LNCS 11620, pp. 270–284, 2019.
https://doi.org/10.1007/978-3-030-24296-1_23

singularity spectrum, is required (3). Multifractal Brownian Motion Synthesis 2D method is capable of determining the number of selected features. Therefore, it can be appropriately used for the solution of financial data set [4–6].

The Finance dataset of this study is one that was generated by International Finance Corporation (IFC) that carries out its works so as to develop solutions for bridging the Finance gap about the Micro, Small and Medium Enterprises MSME. IFC operates in more than 100 countries and 6 regions [7]. The data of the study is concerned with 6 regions (Europe and Central Asia, Latin America and the Caribbean, Middle East and North Africa, South Asia, East Asia and the Pacific and Sub-Saharan Africa). The MSME data was taken covering (2004–2018) [8].

In recent years, conventional theoretical approaches need to be expanded further since they explain the problems under ideal and ordinary circumstances. For this reason, alternative theories are required so that the complicated nature of financial markets can be analyzed and explained better taking into consideration the actual financial conditions.

There are various recent studies which use fractal and multifractal methods for Finance data. Their aim is to yield the most accurate results for future prediction. The study by Mulligan et al. [9] is concerned with the examination of maritime equity price series through the use of five self-affine fractal analysis techniques to explore the price series fractal properties by employing Hurst exponent, Mandelbrot-Lévy characteristic exponent, and fractal dimension. Another study by Mulligan [10] shows Hurst exponent signatures from time series of aggregate price indices for the US. The long-term study presented the fractal examination of the interplay between market prices along with added interest based on how those markets react to external shocks. Another study by the same author [11] examined capacity utilization rates and the index of industrial production for multifractality from 1972 to 2012. The study's Hurst signature analysis of capacity utilization allowed the examination as to whether capacity utilization stems from the Cantillon effects of monetary expansion or the organic changes in supply and demand. Concerning the use of Hölder, several other studies may be cited. For instance, Bianchi et al. [12] used Hurst-Hölder exponent for the quantification of the pointwise degree of efficiency or inefficiency, introducing-efficiency notion. Their analysis concerned four stock indexes that represented Asia, Europe as well as the U.S. Garcin's study [13] is also concerned with Hurst exponents to enhance the existing literature on the estimation of time-dependent Hurst exponents. The author proposed a smooth estimate which was obtained by variational calculus, which is said to be a method more accurate globally and easier compared to other non-parametric estimation techniques. In addition, Hurst Exponents used in the study allowed the forecasts that are reliant upon the estimated multifractional Brownian motion.

Regarding the use of artificial neural network algorithms on financial data, we may cite other studies. One study by Lin et al. [14] presents the use of a new recurrent neural network-echo state network (ESN) for the prediction of the next closing price in stock markets. Along with ESN, the Hurst exponent was also

applied to determine adaptively the initial transient and select sub-series which have the greatest predictability during training. Their results showed that ESN performed better than the other conventional neural networks in majority of the cases. The paper by Karaca et al. [15] examined the link between Turkey's international trade balance and economic growth, covering a period from 1960 to 2015. Artificial Neural Network methods, Feed Forward Back Propagation and Cascade Forward Back Propagation algorithms were used and classification was made as per international trade volume parameters. Accordingly, factors which influenced the development of economy as well as international trade in Turkey were classified.

The MSME Finance dataset handled in this study has been obtained from one formed by IFC based on 6 regions. The timeframe of the analysis is between the years of 2004 and 2018. In addition, 12 attributes were dealt with during the application, which are date, region, Mortgage Outstanding, Mortgage Outstanding USD($), Microloans Outstanding, Microloans Outstanding USD($), SME Loans Outstanding, SME Loans Outstanding USD ($), Micro Loans to Women-owned Firms Outstanding, Micro Loans to Women-owned Firms Outstanding ($), SME Loans to Women owned Firms Outstanding, SME Loans to Women owned Firms Outstanding($)). The steps in the study can be summarized as follows: In the first approach, multi fractal methods and Brownian motion Hölder regularity functions were employed to identify the self-similar and significant attributes in the dataset. In the related steps, (a) polynomial functions were applied to the Finance dataset, as a result of which p_finance data set was obtained. (b) exponential functions were applied, and as a result of this application, e_Finance dataset was obtained. In the second approach, FFBP and CFBP algorithms were applied on p_finance dataset and e_Finance dataset, respectively. And in the final step, for these two datasets, classification accuracy rates were compared. Having been conducted for the first time in the literature, our study showed how the most significant attributes are identified in the Finance dataset by the Hölder functions, namely polynomial and exponential, through the application of ANN algorithms, which are FFBP and CFBP. As can be seen from the outcomes of the analyses, two main contributions are at stake for this study. The first one is that Hölder regularity functions were applied on Finance data for the first time in literature, and secondly, significant and self-similar attributes were obtained for the first time as well for this particular dataset.

The organization of the paper is as such: Sect. 2 provides Materials and Methods. Methods of our approach are basic facts on Hölder Regularity, Multifractal Analysis, FFBP and CFBP algorithms. Finally, results and conclusions are presented in Sects. 3 and 4.

2 Materials and Methods

2.1 Material

This study is on the Finance dataset which includes 12 attributes with 6 regions covering the years (2004–2018) (for detailed information of data, see Table 1) (8).

Table 1. Attributes of the Finance dateset

Regions (between 2004-2018)	Attributes	Data Size
	Mortgage Outstanding ($)	
	Microloans Outstanding ($)	
East Asia and the Pacific	Microloans Outstanding ($)	
Europe and Central Asia	SME Loans Outstanding, ($)	
Latin America and the Caribbean	SME Loans Outstanding ($)	79x12
Middle East and North Africa	Micro Loans to Women owned Firms Outstanding ($)	
South Asia	...	
Sub-Saharan Africa	Micro Loans to Women owned Firms Outstanding ($)	
	SME Loans Women owned Firms Outstanding ($)	
	SME Loans Women owned Firms Outstanding ($)	

2.2 Methods

Our aim is to classify the financial data set through the Multifractal Brownian Motion Synthesis 2D multifractal analysis. Our method is based on the following procedures:

(i) Multifractal methods and Brownian motion Hölder regularity functions were used for the identification of the self-similar and significant attributes in the dataset. As the steps, (a) polynomial functions were applied to the Finance dataset, and p_finance data set was obtained. (b) exponential functions were applied, and we obtained the e_Finance dataset.
(ii) Two of the ANN algorithms, FFBP and CFBP, were applied on the p_finance dataset and e_Finance dataset, respectively.
(iii) Classification accuracy rates were compared with respect to these two datasets. Having been conducted for the first time in the literature, our study demonstrated how the most significant attributes are identified in the Finance dataset by the Hölder functions (polynomial as well as exponential functions) through ANN algorithms application. Eventually, Hölder regularity functions were applied on Finance data, and significant as well as self-similar attributes were obtained for the particular Finance dataset.

The results of the analyses were obtained by the FracLab [16] and Matlab environment.

2.2.1 2D Multifractal Brownian Motion Analysis

Since financial market behaviors are regarded as complicated systems, the status of the market movements can be explained with fractal structures which are capable of providing significant input related to the long-term memory of market behavior as the parts show complete similarity with the whole. Alternative theories are required so fractal analysis and fractal dimension approaches can be one alternative for risk assessment apart from conventional risk measure methods while addressing Finance data.

With Fractional Brownian motion Gauss process $\{B_H(t)\}$ zero mean and covariance function correspond to Eq. (1) [17,18].

$$cov\{B_H(s), B_H(t)\} = \frac{1}{2}\left\{|s|^{2H} + |t|^{2H} - |s-t|^{2H}\right\} \tag{1}$$

H Hurst exponent, $H = \alpha/2$ and $\alpha \in (0,1)$ [18,20]. Fractional Brownian motion is denoted according to Eq. (2) with stochastic integral.

$$\begin{aligned} B_H(t) = \tfrac{1}{\Gamma(H+(1/2))} \times &\left\{ \int_{-\infty}^{0} \left[(t-s)^{H-\left(\frac{1}{2}\right)} - (-s)(t-s)^{H-\left(\frac{1}{2}\right)}\right] \right\} dW(s) \\ &+ \int_{-\infty}^{0} \left[(t-s)^{H-\left(\frac{1}{2}\right)} dW(s)\right] \end{aligned} \tag{2}$$

W denotes the Wiener process that resembles itself in the signal and shows stable increases [18,19,21]. Hurst exponent (H) is significant both for the continuity of the signal and the dependency features of the signal [20]. Hurst exponent can denoted the Hölder function: $H(t)$ and $0 < H(t) < 1, \forall t$. Thus, stochastic integral is replaced by Hölder function. The upper limit of integral of integral of t is taken as mean square. For the Multifractional Brownian Motion, increases are not stable and signals from then on resemble themselves [18,21,22]. Hölder exponent gives information about the most important parts of the signals which are irregular and singular for 2-D analysis. Therefore, Hölder exponent measures the characterization variations in various domains of signal processing. For a positive measurement, represented by $X(t)$, for a signal, Hölder exponent characterizes the singularity power in $t = t(0)$ [18,19,21]. Further details about Hölder Regularity and Hölder Exponent can be obtained from references [18–22].

2.2.2 Artificial Neural Networks

Artificial neural networks (ANNs), which are computational models inspired by the working of the human brain, include a large number of connected nodes. Each node carries out a simple mathematical operation which determines the output of each node along with a set of parameters that are particular for that node. It is possible to learn and calculate very complex functions by connecting these nodes together and setting their parameters in a careful manner. having paved the way for many recent advances in artificial intelligence, artificial neural networks include image recognition, voice recognition, as well as robotics [18].

2.2.3 Feed Forward Back Propagation Algorithm

Feed Forward Neural Networks are one of the artificial neural networks in which the connections between units do not establish a cycle. Feedforward neural networks are the first type of artificial neural network devised and invented, being much simpler than their counterpart, namely the recurrent neural networks. The reason why they are named *feedforward* is that information travels only in forward direction in the network (with no loops). The course is first through the input nodes, subsequently through the hidden nodes (if they are present), and eventually through the output nodes [18].

When the m-dimensional input examples of the network structure are entered into the dataset, $x_i = [x_1, x_2, ..., x_m]^T$ is formed (see Fig. 3). Likewise, $d_k = [d_1, d_2, ..., d_n]^T$ denotes the n-dimensional output examples that are desired. If x_i values are the output values of the neurons in the i layer, the total input that will reach a neuron in j layer is calculated. The output of the neuron in the hidden layer is as shown by Eq. 2:

$$y_j = f_j(net_j),\ j = 1, 2, ..., J \tag{3}$$

Here, f_j is the transfer function of feedback. The total input that will correspond to the k neuron in the output layer is as shown in Eq. 3 [18,23,24].

$$net_k = \sum_{j=1}^{J} w_{kj}.\gamma_j \tag{4}$$

The non-linear output of a k neuron in the output layer is as provided in Eq. 4.

$$o_k = f_k\left(net_k\right),\ k = 1, 2, ..., n \tag{5}$$

The output obtained from the network is compared with the actual output and e_k error is calculated according to Eq. 5 [18,23,24]

$$e_k = \left(d_k - o_k\right) \tag{6}$$

Here, d_k corresponds to the target (desired) value of any k neuron in the output layer and ok shows what is obtained from the network (the actual one). One point that should be born in mind is that such a comparison is applicable only for the network's output layer. Hence, it will hold a privileged position in the identification of the weights for the connections that are related to the output layer. Total mean square is calculated according to Eq. 6 for each of the examples [18,23,24].

$$E = \frac{1}{2}\sum_{k}\left(d_k - o_k\right)^2 \tag{7}$$

In our study, Hölder regularity functions (polyinomial and exponential functions) were applied on the Finance dataset. The application of polyinomial functions yielded the p_Finance dataset and the application of exponential functions yielded the e_Finance dataset. For these two datasets, x inputs were taken as (8x8) for the classification to be performed for the p_Finance dataset and e_Finance dataset by the FFBP algorithm.

2.2.4 Cascade Forward Back Propagation Algorithm

Cascade Forward Back Propagation (CFBP) algorithm resembles the FFBP algorithm. Yet, there is one difference regarding the connection between the

neurons in the input layer, intermediate layer and in the output layer [27]. Cells which track one another are connected with the training performed in this way, andit is possible to apply the training process in two or more levels [25,26].

The algorithmic flow regarding training process is indicated as follows [25–27]:

Step 1: The initial values of the weightings are assigned, for each (x_k, d_k) data pattern x_k inputs are propagated forward through the neural network layers as shown in Eq. 7.

$$p_k = f_j\left(w_j a_{j-1} - b_j\right),\ j = 1, \ldots, J \tag{8}$$

Results and sensitivities are calculated in line with Eq. 10.

$$\delta_j = f_j(d_j - w_{j+1})^T \delta^{j+1},\ j = J-1, \ldots, 1 \tag{9}$$

Step 2: Should the error value be more than what has been estimated, weightings (weight and biases) are calculated once again, as indicated in Eqs. 9 and 10.

$$\Delta w^j = -\eta \delta^j (d^{j-1})^T,\ j = 1, \ldots, J \tag{10}$$

$$\Delta b^j = \eta \delta^j,\ j = 1, \ldots, J \tag{11}$$

Step 3: Once the estimated error value is attained, the training process is completed. If the estimated error value has not been attained, Step 2 is required to be reiterated. In our study, Hölder regularity functions (polinomial and exponential functions) were applied on the Finance dataset. The application of polinomial functions yielded the p_Finance dataset and the application of exponential functions yielded the e_Finance dataset. For these two datasets, x inputs were taken as (8x8) for the classification to be performed for the p_Finance dataset and e_Finance dataset by the CFBP algorithm.

3 Experimental Results and Discussion

Our aim is to make the classification of the Finance dataset by the Hölder exponent multifractal analysis [18]. Our method is based on the following steps:

I. Multifractal Brownian Motion Synthesis 2D is applied to the Finance dataset (see Fig. 1 (i)).
II. Brownian Motion Hölder regularity (polynomial, exponential) for analysis is applied to the data so that significant and self-similar attributes in the dataset can be identified (see Fig. 1(ii)).
III. Accuracy rates regarding classification as per the outcomes which were obtained from the application of the ANN algorithms (FFBP and CFBP) were compared. (see Fig. 1 (iii)).

Figure 1 provides a visual depiction of the steps applied to the Finance dataset (79x12) as specified in (i–iii).

The explanation for the steps is demonstrated in Fig. 1 and the steps for application on the Finance dataset (in Table 1).

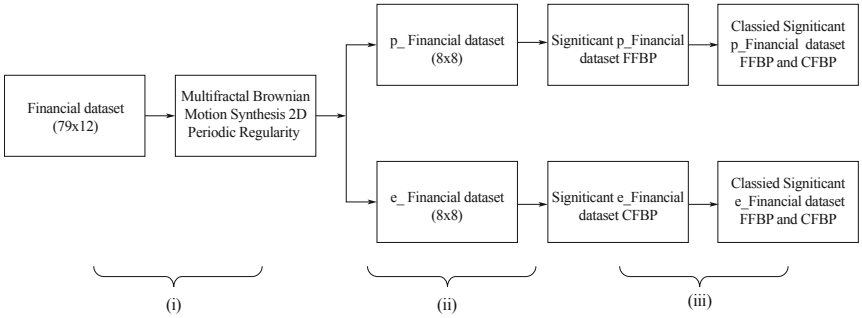

Fig. 1. Classifying the finance dataset with the application of multifractal brownian motion synthesis 2D by the (a) FFBP algorithm (b) CFBP algorithm.

3.1 Application of the FFBP and CFBP to the P_Finance Dataset

(I) Multifractal Brownian Motion Synthesis 2D is applied to the Finance dataset (see Fig. 1(i)).
Step (i): Application of Multifractal Brownian Motion Synthesis 2D to the Finance dataset:

In the Multifractal Brownian Motion Synthesis 2D, first the fractal dimension ($D = [x_1, x_2, ..., x_{79}]$) of the original Finance dataset is calculated. Then, the features which lead to the minimum change of current finance dataset's fractal dimension is dropped sequentially up till the remainder features' number is the upper bound of the fractal dimension D. The new Hölder exponent dataset generated is, $D^{new}(D^{new} = [x_1^{new}, x_2^{new}, ..., x_8^{new}])$.

(II) Brownian Motion Hölder regularity (polynomial) for analysis is applied to the data so that significant and self-similar attributes in the dataset can be identified (see Fig. 1(ii)) [18–20], .
Step (ii): Application of the Multifractal Brownian Motion Synthesis 2D to the data is presented in Fig. 2 below.

The steps of Fig. 2 are provided in detail with the following steps applied.
Step (1): Makes the estimation of the Hölder function, of the input Finance dataset, employing a least square regression [18–20]. The alpha and beta parameters are real values in (0,1) that describe the neighbourhood of each point where the exponent is computed. The Hölder exponent is estimated by using a neighbourhood of points for each point.
Step (2–5): The Averaging estimation utilizes a specific number of neighbours to make the computation of an average of oscillations [18–20]. Averaging enables to get more robust results. Yet, the estimation becomes smoother as the number of points increases. In the average, the gamma and delta parameters are real values in (0:1) that describe the neighbourhood of each point used for computing the average.

Algorithm 1. Application of the Multifractal Brownian Motion Synthesis 2D to the data
Input: Training set including m number of samples
$D = [x_1, x_2, \ldots, x_{79}](x_i \in R^d)$
Output: The new Hölder exponent dataset generated is
$D^{new}(D^{new} = [x_1^{new}, x_2^{new}, \ldots, x_8^{new}])$ (m > n)
Method:
(1) **while** (stopping condition is false)
(2) **for each** input vector $[x_1, x_2, \ldots, x_{79}]$
(3) //n = 8 and the neighborhood is defined by the set of points
V for exponential regularity $V(i) = (0.1 + 0.8 \times x \times y)$
(4) (Eq.1)
(5) }

Fig. 2. Application of the multifractal brownian motion synthesis 2D to the finance dataset to obtain polynomial regularity.

If (8, 256) is the size of the data x, then n=8. The neighbourhood size is defined by. If (8, 256) is the size of the datax, n= 8 and the neighbourhood is defined by the set of points V as: //exponential regularity $V(i) = (0.1 + 0.8 \times x \times y)$
$D^{new} = [x_1^{new}, x_2^{new}, \ldots, x_8^{new}]$ is the new Finance dataset generated (p_Finance dataset).

(III) FFBP and CFBP algorithms were applied on the data obtained (p_finance data)
Accuracy rates regarding classification as per the outcomes which were obtained from the application of the ANN algorithms (FFBP and CFBP) were compared. see Fig. 1(iii)) for p_Finance Data.
Step (iii): The singled out Finance dataset is finally classified by using the FFBP and CFBP algorithms.
The details of Multifractals analysis with the FFBP algorithm (**Algorithm 2**) and its steps regarding its application can be found in Reference [18].

Figure 3 provides and explains the steps of the CFBP Algorithm [18–20]:

Step (1): $\delta_j = f_j(d_j - w_{j+1})^T \delta^{j+1},\ j = J-1, \ldots, 1$
Step (2): If the error value is more than the estimated one, then weightings (weight and biases) are calculated as per Eqs. 9 and 10.
Step (3): $\Delta w^j = -\eta \delta^j (d^{j-1})^T,\ j = 1, \ldots, J$
Step (4): $\Delta b^j = \eta \delta^j,\ j = 1, \ldots, J$
Step (5): When the estimated error value is attained, the training process is brought to a termination. If this value is not attained, then step 2. is reiterated.

Algorithm 3. Multifractals Analysis with the CFBP Algorithm
Input: New training set that includes n number of samples obtained (from Fig. 3) ($D^{new} = [x_1^{new}, x_2^{new}, \ldots, x_8^{new}])(x_i \in R^d$)
Output:
label data that corresponds to m number of samples ($Y = [\mathrm{Re}gions]$)
Method:
(1) $\delta_j = f_j(d_j - w_{j+1})^T \delta^{j+1},\ j = J-1, \ldots, 1$
(2) Should the error value be higher than what is estimated, weightings (weight and biases) are calculated again according to Equation 9 and 10.
(3) $\Delta w^j = -\eta \delta^j (d^{j-1})^T,\ j = 1, \ldots, J$
(4) $\Delta b^j = \eta \delta^j, j = 1, \ldots, J$
(5) When the estimated error value is achieved, the training process comes to a termination. If this value is not attanined, step 2 is required to be reiterated.

Fig. 3. The application of the CFBP algorithm on the p_Finance dataset [18–20].

3.2 Accuracy Rates Obtained from the FFBP and CFBP Algorithm Classification of E_Finance Dataset

The application of the Exponential Regularity Function on the Finance dataset and its classification by the FFBP Algorithm and CFBP Algorithm.
(I) Application of Multifractal Brownian Motion Synthesis 2D to the Finance dataset: (see Fig. 1(i)).

Step (i): Application to Finance dataset Multifractal Brownian Motion Synthesis 2D to Finance dataset (e_Finance dataset):
Multifractal Brownian Motion Synthesis 2D can be explained as such: firstly, the fractal dimension $D(D = [x_1, x_2, \ldots, x_{79}])$ of the original Finance dataset is calculated. Subsequently, the features that lead to the minimum change of current Finance dataset's fractal dimension is dropped in a sequential manner up until the number of remainder features is the upper bound of the fractal dimension D. The new Hölder Exponential dataset generated is, $D^{new}(D^{new} = [x_1^{new}, x_2^{new}, \ldots, x_8^{new}])$.
(II) Brownian Motion Hölder regularity (exponential) [20–22] for analysis is applied to the data so that significant and self-similar attributes in the dataset can be identified (see Fig. 1(ii)).

Step (ii): Application of the Multifractal Brownian Motion Synthesis 2D on the new dataset (e_Finance dataset) can be seen from Fig. 4.

The steps of Fig. 4 are provided in detail with the following steps applied.
Step (1): Performs the estimation of the Hölder function, H of the input Finance dataset , by using a least square regression. The alpha and beta parameters are real values in (0,1) that illustrate the neighbourhood of each point in which the exponent is calculated. For each point, the estimation of Hölder exponent is done through using a neighbourhood of points.

Algorithm 4. Application of Multifractal Brownian Motion Synthesis 2D to the data
Input: Training set that includes m number of samples
$D = [x_1, x_2, \ldots, x_8]$ $(x_i \in R^d)$
Output: The new Hölder exponent dataset generated is
D^{new} $(D^{new} = [x_1^{new}, x_2^{new}, \ldots, x_8^{new}])$ $(m > n)$
Method:
(1) **while** (stopping condition is false) {
(2) **for each** input vector $[x_1, x_2, \ldots, x_{228}]$
(3) //n = 8 and the neighborhood is defined by the set of points V for logistic regularity
(4) $V(i) = 0.3 + \frac{0.3}{1+\exp(-100\times x-0.7)}$ (Eq.2)
(5) }

Fig. 4. Application of the multifractal brownian motion synthesis 2D to the finance dataset to obtain exponential regularity [18–20].

Step (2–5): The Averaging estimation employs a specific number of neighbours to calculate an average of oscillations. Averaging enables to get more robust results, yet as the number of points goes up, the estimation becomes smoother. In the average, the gamma and delta parameters are real values in (0:1) that illustrate the neighbourhood of each point used to make the computation of the average.

If (8,256) is the size of the data x, thus n=8. The size of the neighbourhood is defined by $2 \times m + 1$

If (8,256) is the size of the data x, n= 8 and the neighbourhood is defined by the set of points V as:
$V(i) = 0.3 + \frac{0.3}{1+\exp(-100\times x-0.7)}$

$D^{new} = [x_1^{new}, x_2^{new}, \ldots, x_8^{new}]$ is the new Finance dataset generated (e_Finance dataset).
The related procedural steps are explained below:

(III) Accuracy rates regarding classification as per the outcomes which were obtained from the application of the ANN algorithms (FFBP and CFBP) were compared. (see Fig. 1(iii)) for e_Finance data).

Step (iii): The singled out e_Finance dataset is finally classified by using the FFBP and CFBP algorithms.
The details of Multifractals analysis with the FFBP algorithm (**Algorithm 5**) and its steps regarding its application can be found in Reference [18].

As a result of the application of Exponential regularity function on the Finance dataset, e_Finance dataset was obtained. For the classification of the e_Finance dataset CFBP algorithm was used. The steps for the CFBP algorithm are provided below:

Algorithm 6. Multifractals Analysis with the CFBP Algorithm
Input: New training set that includes then number of samples obtained (Fig. 5) ($D^{new} = [x_1^{new}, x_2^{new}, \ldots, x_8^{new}]$) ($x_i \in R^d$)
Output:
label data that corresponds to the m number of samples ($Y = [Regions]$)
Method:
(1) $\delta_j = f_j(d_j - w_{j+1})^T \delta^{j+1},\ j = J-1, \ldots, 1$
(2) Should the error value be higher than what is estimated, weightings (weight and biases) are calculated again according to Equation 9 and 10.
(3) $\Delta w^j = -\eta \delta^j (d^{j-1})^T,\ j = 1, \ldots, J$
(4)
(5) $\Delta b^j = \eta \delta^j,\ j = 1, \ldots, J$
(6) When the estimated error value is attained, the training process is terminated. If this value is not achieved, step 2 needs to be reiterated.

Fig. 5. The Application of the CFBP algorithm on the e_Finance dataset [18–20].

Figure 5 provides and explains the steps of the CFBP Algorithm [18–20]:

Step (1): $\delta_j = f_j(d_j - w_{j+1})^T \delta^{j+1},\ j = J-1, \ldots, 1$
Step (2): Should the error value be higher than what is estimated, weightings (weight and biases) are calculated again in line with Eqs. 9 and 10.
Step (3): $\Delta w^j = -\eta \delta^j (d^{j-1})^T,\ j = 1, \ldots, J$
Step (4): $\Delta b^j = \eta \delta^j,\ j = 1, \ldots, J$
Step (5): When the estimated error value is attained, the training process is brought to a termination. If this value is not attained, then step 2. is reiterated.
Step (6): The condition could postulate a fixed number of iterations (i.e. executions of Step 2) or learning rate achieving a satisfactorily small value.

3.3 Accuracy Rates Obtained from the FFBP and CFBP Algorithm Classification of p_finance Dataset and e_Finance Dataset

Multifractal Brownian Motion Synthesis 2D functions in Finance dataset is provided below with the classification results obtained (Table 2 and Fig. 6).

For the classification of the e_Finance dataset and p_Finance dataset by the FFBP and CFBP algorithms, which are two of the ANN algorithms, the confusion matrix results are provided below:

Table 2 presents the accuracy rate results obtained as a result of the application of the FFBP and CFBP algorithms on the p_Finance dataset and e_Finance dataset.

As can be seen from Table 2, for both Polynomial (p_Finance dataset) and Exponential functions (e_Finance dataset), the application of the FFBP algorithm yielded higher accuracy rates in terms of classification compared to that of the CFBP algorithm.

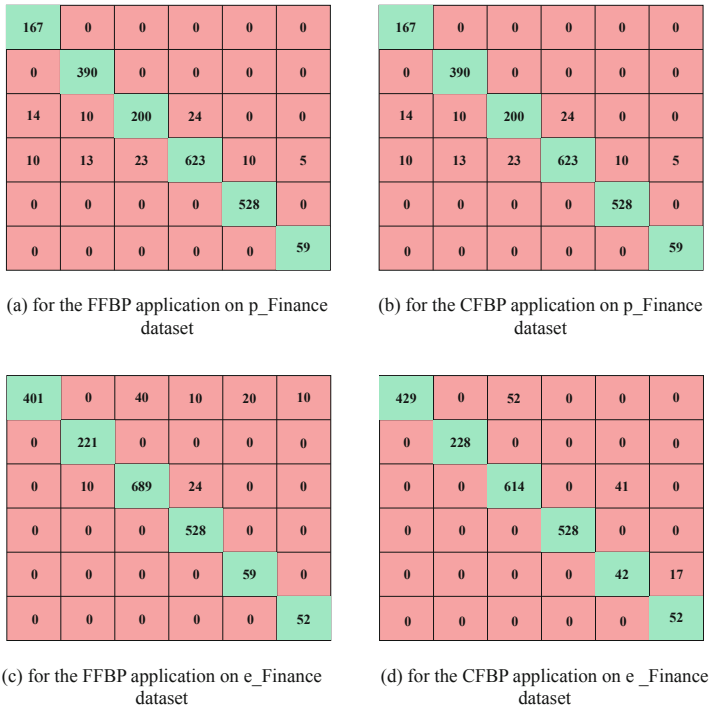

Fig. 6. Confusion matrix (a) for the FFBP application on p_Finance data (b) for the CFBP application on p_Finance data (c) for the FFBP application on e_Finance data (d) for the CFBP application on e_Finance data.

Table 2. Accuracy rates obtained from the FFBP and CFBP algorithm classification of p_Finance dataset and e_Finance dataset

Algorithms	p_Finance dataset (%)	e_Finance dataset (%)
FFBP	85.30%	82.02%
CFBP	80.30%	81.60%

4 Conclusion

Mainly concerned with local Hölder exponent as a function regularity measure for time series regarding Finance data, this study examined the financial data of Micro, Small and Medium Enterprises in six different regions. The timeframe of the study covers the years from 2004 to 2018. 2D multifractal Brownian Motion analysis was applied on the Finance data. This is the first contribution of the paper which yielded higher accuracy rates for Finance data when compared to other studies in the literature. For the highest level of accuracy rates, polynomial and exponential functions of Hölder functions were applied on the Finance dataset separately. As a second contribution, FFBP and CFBP algorithms, two

of the ANN algorithms, were applied on the new datasets (namely e_Finance dataset and p_Finance dataset) which were obtained as a result of the polynomial and exponential functions. For the first time, significant attributes were identified by the Hölder functions. Accordingly, the study has attempted to contribute to the literature by performing multifractal data processing on the Finance data as regards the Micro, Small and Medium Enterprises in six different regions along with the classification of the financial developments concerning the regions. To analyze the complicated nature of financial markets more accurately, conventional theoretical approaches need to be expanded and alternative theories are needed by taking into account the actual financial conditions. Based on this, this study has provided a framework with regard to the economic development of countries and their financial markets. It has also brought to the fore the significance of fractals in financial analyses and related predictions for the future.

References

1. Parveen, A.N., Inbarani, H.H., Kumar, E.S.: Performance analysis of unsupervised Feature selection methods. In: 2012 International Conference on Computing, Communication and Applications, pp. 1–7. IEEE (2012)
2. Kumar, P.S., Lopez, D.: A review on feature selection methodsfor high dimensional. Int. J. Eng. Technol. (IJET) **8**, 0975–4024 (2016)
3. Zunino, L., Tabak, B.M., Figliola, A., Pérez, D.G., Garavaglia, M., Rosso, O.A.: A multifractal approach for stock market inefficiency. Physical A: Stat. Mech. Appl. **387**(26), 6558–6566 (2008)
4. Mensi, W., Hamdi, A., Yoon, S.M.: Modelling multifractality and efficiency of GCC stock markets using the MF-DFA approach: a comparative analysis of global regional and islamic markets. Physical A. Stat. Mech. Appl. **503**, 1107–1116 (2018)
5. Rak, R., Grech, D.: Quantitative approach to multifractality induced by correlations and broad distribution of data. Physical A. Stat. Mech. Appl. **508**, 48–66 (2018)
6. Li, J.: An analytical model for multifractal systems. J. Appl. Math. Phys. **4**(07), 1192 (2016)
7. McCourtie, S.D.: Micro, small, and medium enterprise (MSME) Finance. The World Bank (2013)
8. https://finances.worldbank.org/widgets/2ppx-k958
9. Mulligan, R.F.: Fractal analysis of highly volatile markets: an application to technology equities. Q. Rev. Econ. Financ. **44**(1), 155–179 (2004)
10. Mulligan, R.F.: Multifractality of sectoral price indices: hurst signature analysis of Cantillon effects in disequilibrium factor markets. Physica A. Stat. Mech. Appl. **403**, 252–264 (2014)
11. Mulligan, R.F.: The multifractal character of capacity utilization over the business cycle: an application of Hurst signature analysis. Q. Rev. Econ. Financ. **63**, 147–152 (2017)
12. Bianchi, S., Pianese, A.: Time-varying hurst-hölder exponents and the dynamics of (in) efficiency in stock markets. Chaos, Solitons & Fractals **109**, 64–75 (2018)
13. Garcin, M.: Estimation of time-dependent Hurst exponents with variational smoothing and application to forecasting foreign exchange rates. Physical A: Stat. Mech. Appl. **483**, 462–479 (2017)

14. Lin, X., Yang, Z., Song, Y.: Short-term stock price prediction based on echo state networks. Expert Syst. Appl. **36**(3), 7313–7317 (2009)
15. Karaca, Y., Bayrak, Ş., Yetkin, E.F.: The classification of Turkish economic growth by artificial neural network algorithms, 115–126 (2017)
16. FracLab. https://project.inria.fr/fraclab/
17. Enriquez, N.: A simple construction of the fractional brownian motion. Stochast. Process. Appl. **109**(2), 203–223 (2004)
18. Karaca, Y., Cattani, C.: Computational Methods for Data Analysis. Walter de Gruyter GmbH Co KG, Berlin (2018)
19. Misawa, M.: Local hölderregularity of gradients fore volitional p-Laplaciansystems. Annalidi Matematica Puraed Applicata **181**(4), 389–405 (2002)
20. Karaca, Y., Cattani, C.: Clustering multiple sclerosis subgroups with multifractal methods and self-organizing map algorithm. Fractals **25**(04), 1740001 (2017)
21. Véhel, J.L., Legrand, P.: Signal and Image processingwith Frac Lab. Thinking in Patterns, 321–322 (2004)
22. Franchi, B., Lanconelli, E.: Hölderregularity theorem for a class of linearnon uniformlyel lipticoperators with measurable coefficients. Annalidella Scuola Normale SuperiorediPisa-Classedi Scienze **10**(4), 523–541 (1983)
23. Karaca, Y., Cattani, C., Karabudak, R.: ANN classification of MS subgroups with diffusion limited aggregation. In: Gervasi, O., et al. (eds.) ICCSA 2018. LNCS, vol. 10961, pp. 121–136. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95165-2_9
24. Karaca, Y., Moonis, M., Zhang, Y.D., Gezgez, C.: Mobile cloud computing based stroke healthcare system. Int. J. Inf. Manag. **45**, 250–261 (2019)
25. Warsito, B., Santoso, R., Yasin, H.: Cascade Forward Neural Network for Time Series Prediction. In: Journal of Physics: Conference Series, 1025(1) 012097. IOP Publishing (2018)
26. Abdul-Kadir, N.A., Sudirman, R., Mahmood, N.H., Ahmad, A.H.: Applications of cascade-forward neural networks for nasal, lateral and trill Arabic phonemes. In: 2012 8th International Conference on Information Science and Digital Content Technology (ICIDT2012), 3, 495–499. IEEE (2012)
27. Karaca, Y.: Case study on artificial neural networks and applications. Appl. Math. Sci. **10**(45), 2225–2237 (2016)