Chaos, Solitons and Fractals 118 (2019) 35–40

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

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

Cryptocurrency forecasting with deep learning chaotic neural networks

Salim Lahmiri [a], Stelios Bekiros [b,c,d,*]

[a] ESCA School of Management, 7, Abou Youssef El Kindy Street, BD Moulay Youssef, Casablanca, Morocco
[b] Department of Economics, Villa La Fonte, Via delle Fontanelle, 18, I-50014, Florence, Italy
[c] Department of Acc. & Finance, 76 Patission Str. GR10434, Athens, Greece
[d] RCEA, LH3079, Wilfrid Laurier University, 75 University Ave W., ON, N2L3C5, Waterloo, Canada

ARTICLE INFO

Article history:
Received 31 October 2018
Accepted 9 November 2018
Available online 13 November 2018

JEL Classification:
90.01 Social phenomena

Keywords:
Digital currencies
Deep learning
Fractality
Neural networks
Chaos
Forecasting

ABSTRACT

We implement deep learning techniques to forecast the price of the three most widely traded digital currencies i.e., Bitcoin, Digital Cash and Ripple. To the best of our knowledge, this is the first work to make use of deep learning in cryptocurrency prediction. The results from testing the existence of nonlinearity revealed that the time series of all digital currencies exhibit fractal dynamics, long memory and self-similarity. The predictability of long-short term memory neural network topologies (LSTM) is significantly higher when compared to the generalized regression neural architecture, set forth as our benchmark system. The latter failed to approximate global nonlinear hidden patterns regardless of the degree of contamination with noise, as they are based on Gaussian kernels suitable only for local approximation of non-stationary signals. Although the computational burden of the LSTM model is higher as opposed to brute force in nonlinear pattern recognition, eventually deep learning was found to be highly efficient in forecasting the inherent chaotic dynamics of cryptocurrency markets.

1. Introduction

In contemporary econophysics literature, there is a growing interest in studying the general dynamics of Bitcoin and in general of digital currencies. For instance, long-memory was measured to assess the market efficiency of cryptocurrencies in [1–4], their volatilities were examined in [5–9], liquidity and microstructure was explored in [10–12], risk and speculative aspects were scrutinized in [13,14] whilst herding and optimal trading were investigated in [15,16].

Although stock price forecasting is a crucial step toward portfolio optimization and hedging, only a limited number of works have focused on this issue especially for the Bitcoin market. For instance, the authors in [17] used a set of technical indicators as inputs to deep learning architecture composed of seven layers to predict the future return trend of Bitcoin. A hybrid artificial neural network with generalized auto-regressive conditional heteroskedasticity (ANN-GARCH) trained with technical analysis indices, was proposed to forecast its price volatility in [18]. In another work, a deep learning based system was proposed to predict the fluctuation in the Bitcoin price and transactions based on user opinions and sentiments derived by online forums [19].

Artificial neural networks were widely employed to predict financial markets via the use of lags as inputs [20–25] or by utilizing technical indicators [17,18,26–30]. In other works textual analysis and sentiment indicators were measured as predictors for stock market forecasting [19,31,32]. However, generating accurate predictions in a complex and fast analytical framework still pertains as definitely a challenging problem.

The purpose of our current study is fundamentally twofold; firstly, we seek to assess the predictability of most active digital currencies by examining their inherent nonlinear dynamics including inherent chaoticity [33,34] and fractality [35]. Secondly, we aim at using deep learning [36] as the underlying dynamical system topology to automatically extract hidden patterns unveiling the nonlinear dynamics of their time series. Consequently, we contribute to the relevant literature in the following ways: as scrupulous investigations of the nonlinear statistical properties of the most active digital currencies are not present in recent works to the best of our knowledge, we attempt to delve into this matter from an econophysics perspective. Secondly, by examining whether chaoticity is inherent, we can shed light on the short-term predictability of cryptocurrencies. Thirdly, as opposed to previous works on currency forecasting [18,19] and financial market

* Corresponding author.
E-mail addresses: slahmiri@esca.ma (S. Lahmiri), stelios.bekiros@eui.eu (S. Bekiros).

https://doi.org/10.1016/j.chaos.2018.11.014
0960-0779/

prediction [20–32], we seek to build up a complex neural network based on deep learning to extract and exploit the hidden information represented by linear or nonlinear patterns within the raw data of digital currencies so as to simultaneously make both accurate and fast predictions, the latter of which is of utmost importance in modern trading practice. Considering that it is rather cumbersome or infeasible to set up analytical equations with time-varying parameters and coefficients in order to show the relationship among complex, unlabeled and high dimensional time series data, our analysis is evidently pragmatic. In addition, it is typically costly and time consuming to manually extract domain-specified patterns used to train a sophisticated and complex predictive system, including artificial neural networks. In sum, our results stemming from a nonlinear dynamics perspective are expected to indicate whether the most prominent digital currencies are predictable or not in the short-term depending on measured fractality and chaoticity, while our results from the introduction of deep learning will demonstrate the consistency and accuracy of their forecastability.

In this regard, we utilize the largest Lyapunov exponent (LLE) [33,34] and a detrended fluctuation analysis (DFA) based on the extracted Hurst exponent [35] of the time series to detect chaos and/or fractal characteristics of the underlying digital currencies. Specifically, the former allows checking the existence of nonlinear deterministic maps if present, whilst the later measures reveal the presence of long memory with no assumptions regarding stationarity. Finally, we deploy an intelligent signal data mining and forecasting system based on deep learning via a specific topology of neural networks named Long-Short Term Memory networks and denoted as LSTM hereafter. To test consistency and robustness, an experimental comparison between LSTM and benchmark well-known generalized regression neural networks (GRNN) will be undertaken on three major and liquid digital currencies.

The deep learning LSTM neural networks [36] overcome the problems with recurrent neural networks (RNN) related to vanishing gradients, by replacing nodes in the RNN with memory cells and gating mechanism [for details see 37]. In this regard, it is an attractive deep learning neural architecture mostly on the account of its efficacy in memorizing long and short-term temporal information simultaneously [36]. The LSTM topology is recurrent where connections between units form a directed cycle/loop allowing data input signals to flow both forwards and backwards within the network. Subsequently, past information is preserved for future processing. The GRNN [38] is a parallel and memory-based system that estimates the regression surface of a continuous variable while providing fast learning and convergence to the optimal regression surface as the number of sample becomes very large [39]. Due to those unique attractive features, LSTM and GRNN were successfully applied to various problems of data analysis and modeling [40–42].

To the best of our knowledge, this is the first work to make use of deep learning to forecast digital currency prices. The remainder of the paper is organized as follows. Section 2 describes the LLE and DFA methodologies to detect nonlinear dynamics while the LSTM and GRNN architectures are illustrated in detail. Section 3 exposes the protocol of conducted experiments, presents our data sets and reveals the empirical results. Finally, Section 4 concludes.

2. Methodology

2.1. "Nearest neighbourhood" estimation of largest Lyapunov exponent

We follow the approach of Rosenstein et al. [34] for the estimation of the largest Lyapunov exponent (LLE). The main advantages of the proposed algorithm are accuracy, robustness to small and noisy data sets, and fast computation [34]. The approach is briefly described next. We consider a digital currency signal $\{x_1,x_2,\ldots,x_N\}$ and assume that $X_i=(x_i,x_{i+J},\ldots,x_{i+(m-t)J})$ is the system state at discrete time i, where J is the lag or reconstruction delay, and m is the embedding dimension. Then, the reconstructed trajectory $\mathbf{X}$ becomes $\mathbf{X}=(X_1\ X_2\ \ldots\ X_M)^{\mathrm{T}}$. We then localize the nearest neighbor of each point on the reconstructed trajectory $\mathbf{X}$. For instance, the nearest neighbor $\mathbf{X}_j$ to $\mathbf{X}_i$ is found by finding the smallest distance $d_j(0)$ given by:

$$d_j(0) = \min_{\mathbf{X}_j} \left\|\mathbf{X}_j - \mathbf{X}_i\right\| \tag{1}$$

Then, the LLE denoted λ for the cryptocurrency signal is given in the following equation:

$$d(t) = Ce^{\lambda_1 t} \tag{2}$$

where $d(t)$ is the average divergence at time t of the underlying time series and C is a constant that normalizes the initial separation. Let's assume that the j-th pair of nearest neighbors diverges approximately at a rate given by the LLE as follows:

$$d_j(i) \approx C_j e^{\lambda_1 (i\Delta t)} \tag{3}$$

where Δt is the sampling period of the time series, and $d_j(i)$ the distance between the j-th pair of nearest neighbors after i discrete-time steps. By taking the logarithm of both sides in Eq. (3), the following is obtained:

$$\log d_j(t) \approx \log C_j + \lambda_1 (i\,\Delta t) \tag{4}$$

Indeed, Eq. (4) corresponds to a set of approximately parallel lines (for $j=1,2,\ldots,M$), each with a slope approximately proportional to λ_1. The LLE is estimated by using least-squares fit to the average line expressed as:

$$y(i) = \frac{1}{\Delta t}\left\langle \log d_j(i)\right\rangle \tag{5}$$

where $\langle\cdot\rangle$ denotes the average over all values of j [34]. Following Rosenstein et al. [34], the reconstruction delay J corresponds to the lag before the first decline of the autocorrelation function, and the embedding dimension m is determined based on the smallest value that allows convergence of results. This approach is optimal for fast computation. Overall, $\lambda \geq 0$ indicates that the signal possesses chaotic dynamics, while $\lambda < 0$ indicates convergence between close trajectories, i.e., existence of classic attractors.

2.2. Detrended fluctuation analysis and Hurst exponent

The DFA [35] is suitable for quantifying nonlinear dynamics and complexity in time series considering that it is less dependent on non-stationarity assumptions and noisy data. The computational steps for a signal x are described as follows:

a) Define the suite y_N of the cumulative series of the original signal x_i fluctuations about its mean as:

$$y_N = \sum_{i=1}^{N} (x_i - \bar{x}) \tag{6}$$

b) Divide y_N into boxes of equal length n.
c) In each box, fit the local trend of y_N by a polynomial $P(n,N)$ that represents the local trend of the box. In our study, we employ a polynomial of degree one.
d) For the given n box size, compute the root-mean-squared detrended fluctuation of the signal y_N as:

$$F(n,N) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - P(n,N))^2} \tag{7}$$

For each of the available n box size, the last step is repeated to obtain the empirical relationship between the overall fluctuation $F(n,N)$ and the box size n:

$$F(n, N) \propto n^H \tag{8}$$

At the end, the Hurst exponent HE is estimated by running a regression of log($F(n,N)$) upon the log(n). For $H = 0.5$, the dynamics of the original time series follow a random walk, whilst for $0 < H < 0.5$, the series are anti-persistent. On the contrary, the series are persistent when $0.5 < H < 1$ and lastly in case of $H \geq 1$, the autocorrelation exists but the series cease to follow a power-law form [35].

2.3. Long-short term memory deep learning neural networks

The LTSM neural networks [36,37] provide with a robust extension of the recurrent neural network (RNN) topology in terms of nonlinear modeling and more importantly forecasting. In this regard, deep learning LTSM neural networks systems not only keep adjacent temporal information in a spontaneous manner, but also control long-term (LT) information. In other words, the LSTM can preserve previous information, which can significantly help improving its ability to learn signal sequences and inherent nonlinear patterns.

Specifically, the main innovation of the LTSM is to introduce the notion of "*controlling gate*". For example, conditional on the inputs, the LSTM memory cell can remember or forget any cell state. The cell is supported by three gates: *input, forget* and *output*. The input gate determines how much current information should be treated as input in order to generate the current state, whilst the forget gate extracts how much information can be kept from the last prior state. Finally, the output gate filters the information that can be actually treated as significant and produces the output which basically in our context would be a forecast.

Let's denote the input to all cells as x_t and the previous time-step output as h_{t-1}. Then, the forget gate f_t computes the input for the cell state C_{t-1} using a sigmoid function, given by:

$$f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right) \tag{9}$$

The input gate i_t calculates the values to be updated to C_t as follows:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{10}$$

Subsequently, the output gate o_t controls the output values:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{11}$$

Finally, the output value of the LSTM memory cell is given by:

$$h_t = o_t \odot C_t \tag{12}$$

where,

$$C_t = f_i \odot C_{t-1} \oplus i_t \odot \bar{C}_{t-1} \tag{13}$$

with $\bar{C}_t$ symbolizing the output of the nonlinear hyperbolic tangent (*tanh*) function. In our paper historical sequences are processed as the inputs of the LSTM to extract hidden information, whereas the predicted digital currency price is regarded as the targeted output.

2.4. Generalized regression neural networks

The GRNNs [38] are parallel and memory-based systems that estimate the regression surface of a continuous variable. They afford fast learning and converge to the optimal regression surface when the sample size is large [38] as they comprise a one pass learning algorithm [38,39]. Suppose that the system input is a vector X, and the desired estimate of the system output is the vector Y. Also, we assume that the joint continuous probability density function of X and Y is represented by an unknown function $f(x,y)$. The regression of y on x is represented by:

$$E[y|X = x] = \frac{\int_{-\infty}^{+\infty} y\, f(x, y)dy}{\int_{-\infty}^{+\infty} f(x, y)dy} \tag{14}$$

The unknown joint continuous probability density function $f(x,y)$ is estimated in the following way:

$$\hat{f}(X, y) = \frac{1}{2\pi^{(p+1)/2\sigma^{(p+1)}}} \times n^{-1} \times \left\{ \sum_{i=1}^{n} \exp\left[-\frac{(X - X_i)^T (X - X_i)}{2\sigma^2} \right] \times \exp\left[-\frac{(Y - Y_i)^2}{2\sigma^2} \right] \right\} \tag{15}$$

where $\sigma < 1$ is the width of the Gaussian kernel, p is the dimension of X, and n is the sample size. Once $\hat{f}(X, y)$ is substituted in$E[y|X = x]$, the output function $Y(X)$ is expressed as follows:

$$Y(X) = \frac{\sum_{i=1}^{n} Y_i \exp\left[-\frac{D_i^2}{2\sigma^2} \right]}{\sum_{i=1}^{n} \exp\left[-\frac{D_i^2}{2\sigma^2} \right]} \tag{16}$$

where $D_i^2$is given by:

$$D_i^2 = (X - X_i)^T (X - X_i) \tag{17}$$

In general, the GRNN architecture consists of four layers: the input layer, pattern layer, summation layer, and output layer, the later of which yields the predicted value corresponding to an unknown input vector. In our work, the input to the GRNN is the past cryptocurrency price and the output the future price.

3. Experimental results

3.1. Protocol of experiments

Due to the utilization of the *tanh* function inside the deep learning LSTM topology and for better estimation of the optimal regression surface by the GRNN, the values of the raw (input) vector are converted into the range of [−1,1]. In addition, this scaling is ideal to allow deep learning LSTM networks and GRNNs handling variations in amplitudes. Since the available number of sample observations is limited because digital currencies are new cryptocurrencies or digital assets, the first 90% of the observations are used for training purposes and the remaining 10% most recent ones, for testing and out-of-sample forecasting. Moreover, the LLE estimation and the DFA-based HE calculation are used both on learning and testing sub-samples, to examine inherent chaoticity, fractality and any other nonlinear features throughout all time periods. Finally, the forecasting performance is evaluated by using the root mean squared error (RMSE) metric as widely employed in signal processing and prediction literature. The RMSE is given by:

$$RMSE = \sqrt{N^{-1} \sum_{i=1}^{N} \left(x_i - \hat{x}_i\right)^2} \tag{18}$$

where N is the number of observations used for testing, x is the true value, $\hat{x}$ is the forecasted value and t is time script.

3.2. Nonlinear forecasting

The dataset used in our study comprises daily prices in US dollars for the following three digital currencies: i) Bitcoin, from 16 July 2010 to 01 October 2018, Digital Cash spanning 8 February 2010 to 1 October 2018 and the Ripple from 21 January 2015 to 1 October 2018. These three cryptocurrencies are selected because of their high liquidity and data availability of at least more than one

Table 1
Estimated LLE and HE values.

	LLE		HE	
	Training sub-sample	Testing sub-sample	Training sub-sample	Testing sub-sample
Bitcoin	0.1250	−7.8711	1.0087	0.9776
Digital Cash	0.3205	−10.7333	0.9559	1.0901
Ripple	0.8181	−0.0065	1.0741	0.8715

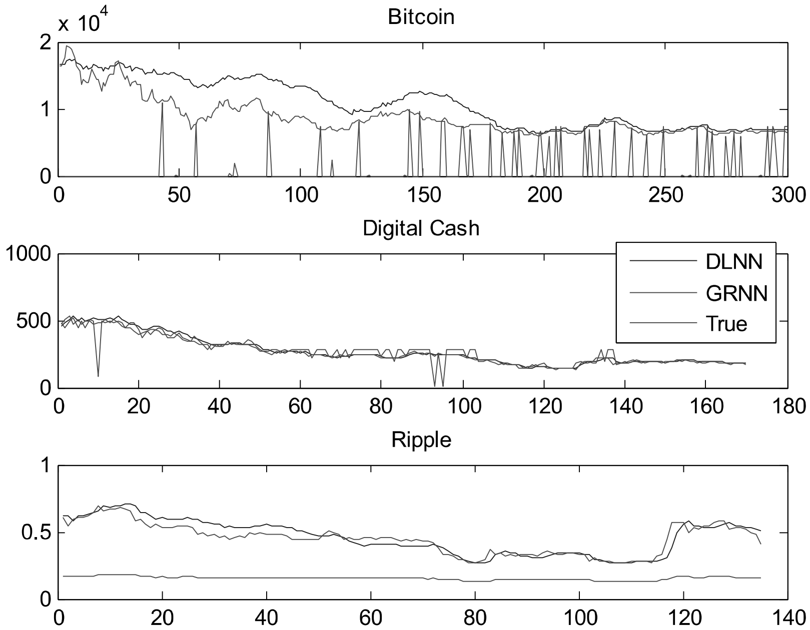

Fig. 1. Forecasted versus true (observed) values for the three major cryptocurrencies are presented. Time horizon is represented by *x*-axis while price values are displayed on the *y*-axis. Deep learning (DLNN) corresponds to the implementation of LTSM neural network architecture. GRNN represents the generalized regression neural topology.

thousand observations for each one. Hence, the number of samples obtained is 3006 for Bitcoin, 1704 for Digital Cash, and 1357 for Ripple.

The estimated values of the LLE and DFA-based HE for the learning and testing sub-samples of each digital currency are provided in Table 1. As illustrated, the LLE associated with training and testing is positive and negative, respectively. Accordingly, the price sample used in the learning phase exhibits chaotic dynamics, while the testing one reveals convergence to classic attractors. Therefore, we may conclude that each digital currency is governed by two different regimes temporally across time periods, whereby the first regime (learning phase) is characterized by a seemingly stochastic random-like behaviour yet driven by chaotic behaviour, whilst the second regime (testing phase) is characterized by relative convergence and stability. Besides, the estimated values of the HE all show that the three active cryptocurrencies possess long-memory features. In particular, prices eventually do not follow a random walk while they exhibit persistent dynamics during each single phase. Overall, the result from investigating chaos in the form of inherent fractality, presents strong evidence of short-term predictability of the price dynamics as would be the case for chaotic systems. In other words, a deep learning nonlinear pattern recognition system could be efficiently employed to model and predict their respective signal sequences.

The findings from forecasting via deep learning neural networks (DLNN) i.e., from implementing LSTMs, are presented in Fig. 1. The observed (true) and forecasted values derived from DLNN and RNNs are exhibited for each currency. As seen, for all digital currencies, DLNN-based forecasted values follow closely the observed ones compared to the GRNN topology. Accordingly, the computed RMSEs for DLNN and GRNN are respectively 2.75×10^3 and 8.80×10^3 for Bitcoin, 19.2923 and 50.2418 in case of Digital Currency, and 0.0499 and 0.3115 for Ripple (Table 2). Consequently, the RMSE scores confirm that deep learning LSTM neural networks perform better than GRNN in predicting the future price values of Bitcoin, Digital Cash and Ripple. Overall, deep learning LSTM systems were able to learn chaotic and self-similar patterns for the three major cryptocurrencies evidently better as well as robustly predict their future dynamics versus the generalized regression neural architecture. We thereby confirm the effectiveness of LSTM to model and forecast chaotic financial data structures in case of digital currency markets.

Table 2
RMSE scores for the three major cryptocurrencies.

	RMSE	
	DLNN	GRNN
Bitcoin	2.75×10^3	8.80×10^3
Digital cash	19.2923	50.2418
Ripple	0.0499	0.3115

4. Technical and economic implications

Our work, to the best of our knowledge, is the first to make use of deep learning to forecast digital currency prices. The investigation of chaotic and fractal dynamics in nonlinear systems is of paramount importance in terms of their predictability. An unstable or highly noisy system (signal) might present finite short- or long-term predictability, while the existence of fractality and self-similarity could enhance the possibility future sequences of such signals be accurately predicted, at least in the short-term. In the latter case, artificial neural networks are intelligent nonlinear systems with powerful approximation and self-adaptive data driven modelling capabilities that allow great flexibility in modelling non-stationary signals. As a result, over the last two decades, there has been a considerable interest developing gradually in the use of such computational intelligence systems, especially when highly nonlinear forecasting problems arise. Eventually more recently, there is a growing consensus that in particularly deep learning neural networks are more effective than conventional topologies used so far in analyzing and forecasting complex patterns in financial markets.

We contribute to the econophysics literature by investigating the chaotic properties of the three most widely traded digital currencies with sufficient historical data towards assessing their predictability. We presented a comparison analysis between the forecastability of deep learning Long-Short Term Memory (LSTM) neural networks versus the well established category of generalized regression neural networks (GRNN), as employed in cryptocurrency markets. We explored this path as interestingly the literature remains still very limited [17–19].

Initial results from investigating the nonlinear structure of the signals in question, showed that digital currencies exhibit chaotic characteristics, yet depending on the sample time period examined. Furthermore, all active cryptocurrencies revealed the existence of strong self-similarity in both training and testing subsamples, when this segmentation was performed during deep learning processing.

When we implemented advanced and conventional neural network architectures to three different and highly liquid digital currencies, we found that the LSTMs outperformed significantly the generalized regression neural networks, in terms of the root mean squared error. The reason is that deep learning neural systems are obviously effective in memorizing short- as well as longer-term temporal information simultaneously, which makes them capable of extracting hidden patterns from the underlying signals and generally modelling temporal sequences, with accuracy. Also, LSTMs were found significantly more successful in learning fractal patterns for the testing sample, which basically comprised the period of prediction.

On the contrary, the generalized regression neural networks failed overall to approximate global patterns including chaotic features, as they are based on Gaussian kernels to locally approximate non-stationary signals with or without a high level of contamination with noise. This result could be corroborated by the fact that the computational burden of the LSTMs appeared higher. Specifically, deep learning took between five to ten minutes of estimation time during the learning phase, whilst the generalized regression neural networks concluded training in less than a second. Although the difference is profound and at the expense of such elaborate techniques as deep learning, yet it was observed because as opposed to brute force, discovering and memorizing nonlinear hidden patterns bears a cost. That is LSTM yielded a much longer time required for convergence, than a system based on a one-pass algorithm utilizing a Gaussian kernel to achieve local-pattern recognition. More importantly, predictive accuracy was higher for the LSTMs even though training was more time consuming. However, the fact that LSTM neural networks outperformed the GRNNs should not lead to a generalization of their predictive ability as the latter may provide with better results if signal lengths and sample observations are very large. Overall, deep learning was found to be highly efficient in learning and forecasting inherent chaotic patterns for the most widely traded cryptocurrencies.

References

[1] Gajardo G, Kristjanpoller WD, Minutolo M. Does bitcoin exhibit the same asymmetric multifractal cross-correlations with crude oil, gold and DJIA as the Euro, Great British Pound and Yen. Chaos Solitons Fractals 2018;109:195–205.
[2] Takaishi T. Statistical properties and multifractality of bitcoin. Physica A 2018;506:507–19.
[3] Begušić S, Kostanjčar Z, Stanley HE, Podobnik B. Scaling properties of extreme price fluctuations in bitcoin markets. Physica A 2018;510:400–6.
[4] Lahmiri S, Bekiros S. Chaos, randomness and multi-fractality in bitcoin market. Chaos Solitons Fractals 2018;106:28–34.
[5] Lahmiri S, Bekiros S, Salvi A. Long-range memory, distributional variation and randomness of bitcoin volatility. Chaos Solitons Fractals 2018;107:43–8.
[6] Symitsi E, Chalvatzis KJ. Return, volatility and shock spillovers of bitcoin with energy and technology companies. Econ Lett 2018;170:127–30.
[7] Klein T, Thu HP, Walther T. Bitcoin is not the new gold – a comparison of volatility, correlation, and portfolio performance. Int Rev Financ Anal 2018;59:105–16.
[8] Yi S, Xu Z, Wang G-J. Volatility connectedness in the cryptocurrency market: is bitcoin a dominant cryptocurrency? Int Rev Financ Anal 2018;60:98–114.
[9] Katsiampa P. Volatility estimation for bitcoin: a comparison of GARCH models. Econ Lett 2017;158:3–6.
[10] Koutmos D. Liquidity uncertainty and bitcoin's market microstructure. Econ Lett 2018;172:97–101.
[11] Dyhrberg AH, Foley S, Svec J. How investible is bitcoin? analyzing the liquidity and transaction costs of bitcoin markets. Econ Lett 2018;171:140–3.
[12] Donier J, Bonart J. A million metaorder analysis of market impact on the bitcoin. Mark Microstruct Liquidity 2015;1:1550008.
[13] Osterrieder J, Lorenz J. A statistical risk assessment of bitcoin and its extreme tail behaviour. Ann Financ Econ 2017;12:1750003.
[14] Bouoiyour J, Selmi R, Tiwari AK. Is bitcoin business income or speculative foolery? new ideas through an improved frequency domain analysis. Ann Financ Econ 2015;10:1550002.
[15] Ajaz T, Kumar AS. Herding in crypto-currency markets. Ann Financ Econ 2018;13:1850006.
[16] Li TN, Tourin A. Optimal pairs trading with time-varying volatility. Int J Financ Eng 2016;3:1650023.
[17] Nakano M, Takahashi A, Takahashi S. Bitcoin technical trading with artificial neural network. Physica A 2018;510:587–609.
[18] Kristjanpoller W, Minutolo MC. A hybrid volatility forecasting framework integrating GARCH, artificial neural network, technical analysis and principal components analysis. Expert Syst Appl 2018;109:1–11.
[19] Kim YB, Lee J, Park N, Choo J, Kim J-H, Kim CH. When bitcoin encounters information in an online forum: using text mining to analyse user opinions and predict value fluctuation. PLoS ONE 2017;12:e0177630.
[20] Lahmiri S. Minute-ahead stock price forecasting based on singular spectrum analysis and support vector regression. Appl Math Comput 2018;320:444–51.
[21] Lei L. Wavelet neural network prediction method of stock price trend based on rough set attribute reduction. Appl Soft Comput 2018;62:923–32.
[22] Lahmiri S. Interest rate next-day variation prediction based on hybrid feedforward neural network, particle swarm optimization, and multiresolution techniques. Physica A 2016;444:388–96.
[23] Adhikari R, Agrawal RK. A combination of artificial neural network and random walk models for financial time series forecasting. Neural Comput Appl 2014;24:1441–9.
[24] Lahmiri S. Intraday stock price forecasting based on varitional mode decomposition. J Comput Sci 2016;12:23–7.
[25] Lahmiri S, Boukadoum M. Intelligent ensemble forecasting system of stock market fluctuations based on symmetric and asymmetric wavelet functions. Fluctuation Noise Lett 2015;14:1550033.
[26] Hsu C-M. A hybrid procedure with feature selection for resolving stock/futures price forecasting problems. Neural Comput Appl 2013;22:651–71.
[27] Das SR, Mishra D, Rout M. A hybridized ELM using self-adaptive multi-population-based Jaya algorithm for currency exchange prediction: an empirical assessment. *Neural Comput Appl* 2018:1–24. https://doi.org/10.1007/s00521-018-3552-8.
[28] Lahmiri S. A technical analysis information fusion approach for stock price analysis and modeling. Fluctuation Noise Lett 2018;17:1850007.
[29] Shynkevich Y, McGinnity TM, Coleman SA, Belatreche A, Li Y. Forecasting price movements using technical indicators: Investigating the impact of varying input window length. Neurocomputing 2017;264:71–88.
[30] Ticknor JL. A bayesian regularized artificial neural network for stock market forecasting. Expert Systems with Applications 2013;40:5501–6.
[31] Nguyen TH, Shirai K, Velcin J. Sentiment analysis on social media for stock movement prediction. Expert Syst Appl 2015;42:9603–11.
[32] Li B, Chan KCC, Ou C, Ruifeng S. Discovering public sentiment in social media for predicting stock movement of publicly listed companies. Inf Syst 2017;69:81–92.
[33] Peinke J, Parisi J, Rossler OE, Stoop R. Encounter with chaos. Springer-Verlag; 1992.
[34] Rosenstein MT, Collins JJ, De Luca CJ. A practical method for calculating largest Lyapunov exponents from small data sets. Physica D 1993;65:117–34.
[35] Peng C-K, Buldyrev SV, Havlin S, Simons M, Stanley HE, Goldberger AL. Mosaic organization of DNA nucleotides. Phys Rev E 1994;49:1685–9.
[36] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9:1735–80.

[37] Aungiers J. LSTM-Neural-Network-for-Time-Series-Prediction; 2016. http://www.jakob-aungiers.com/articles/a/LSTM-Neural-Network-for-Time-Series-Prediction.
[38] Specht DF. A general regression neural network. IEEE Trans Neural Netw 1991;2:568–76.
[39] Polat O, Yildirim T. Hand geometry identification without feature extraction by general regression neural network. Expert Syst Appl 2008;34:845–9.
[40] Zhao Z, Chen W, Wu X, Chen PCY, Liu J. LSTM network: a deep learning approach for short-term traffic forecast. IET Intell Transp Syst 2017;11:68–75.
[41] Soh P-W, Chang J-W, Huang J-W. Adaptive deep learning-based air quality prediction model using the most relevant spatial-temporal relations. IEEE Access 2018;6:38186–99.
[42] Lahmiri S. Comparing variational and empirical mode decomposition in forecasting day-ahead energy prices. IEEE Syst J 2017;11:1907–10.