TEST Engineering & Management

January - February 2020
ISSN: 0193 - 4120 Page No. 2313 - 2322

An Empirical Study of Bitcoin Pricing using an Evolutionary Framework

Samuka Mohanty[1], Rajashree Dash[2]

[1,2] *Department of Computer Science and Engineering,*
Siksha O Anusandhan (Deemed to be University), Bhubaneswar, India
[1] *mohantysamuka@gmail.com*, [2] *rajashree_dash@yahoo.co.in*

Article Info
Volume 82
Page Number: 2313 - 2322
Publication Issue:
January-February 2020

Article History
Article Received: *14 March 2019*
Revised: *27 May 2019*
Accepted: *16 October 2019*
Publication: *18 January 2020*

Abstract

Cryptocurrency merchandising is growing as an attractive area of investment. Bitcoin is much preferred over other cryptocurrencies in the world and hence is becoming more popular. But, the bitcoin price is extremely volatile. So, the forecasting of its price is highly desirable. As nature inspired-machine learning is being used extensively for time series analysis and prediction, it can be explored for bitcoin prediction as well. Also, as bitcoin is gradually increasing as a promising virtual asset, its volatility needs to be measured. This paper unveils the consequence of using ChebyShev Ploynomial Neural Networks (CHPNN) for Bitcoin pricing process. The evolutionary algorithms: Particle Swarm Optimization (PSO) and Differential Evolution (DE) are utilized for training the model. This study analyses the performance of the model through three different error measures: Root Mean Square Error (RMSE), RRSE (Relative Root Square Error) and SSE (Sum of Squares Error). It shows that DE-CHPNN predicts better day-ahead price of bitcoin.

Keywords: *Cryptocurrency; Bitcoin; CHPNN; PSO; DE..*

1. Introduction

Bitcoin is the most favorable cryptocurrency in this era. In 2008, N.Santosi introduced the world with a new revolution in the financial market: an encrypted currency or more precisely an asset called Bitcoin[1].Since then its gaining popularity steadily. It is a cryptocurrency with no central governance. Rather, a bitcoin is possessed as an asset which has a decentralized control. The transaction is controlled by a Blockchain[2,3]. The blockchain is always dynamic with new nodes being attached and detached from the chain as per the transaction. The bitcoin prices are having very rapid growth and sometimes may fall back. This rapid fluctuation in the bitcoin price needs to be mapped to a nonlinear model. To aid financial investors in taking decisions for investing in bitcoins or purchasing some bitcoins, a more accurate model for prediction needs to be designed. The prediction of bitcoin price will boost up the financial market in all dimensions.

Financial Market Analysis using time-series models and machine learning techniques have been extensively investigated. But, very less work has been done in the field of bitcoin pricing. The bicoin market needs to be analyzed more and more so as to aid the recent smart users in gaining profit for a bitcoin transaction. A

SVM approach for bitcoin exchange rate forecasting along with a recurrent neural network and a tree classifier ensemble has been used in [4] for bitcoin price prediction. Also, in [5] a Bayesian Neural Network has been also demonstrated for forecasting the cipher currency. Various Neural Network Ensembles with Genetic Algorithm or other models have been proposed for predicting bitcoin prices[6-12].It has been observed that Bitcoin pricing has not been much analyzed by using a FLANN model. The fast convergent, computationally efficient and highly accurate in mapping nonlinear models FLANNs [13-14] needs to be investigated for mapping the high volatility of bitcoin. Also, a basic evolutionary algorithm needs to be implemented for such a scenario so as to get a thorough analysis of the bitcoin pricing. This motivation of designing a clear, simple, efficient and accurate predictor guided through this study.

The rest of the paper is organised as follows: Section 2 explains the related work. Section 3 introduces and describes the proposed work. Section 4 analyses the results. Section 5 concludes the work using the analyzed results.

2. Related Work

Design of CHPNN

CHPNN is a single hidden layered FLANN. The exclusion of multi-hidden layers is possible due to the increased dimensionality of the input by using an expansion block. CHPNN uses fast chebyshev polynomials for expansion and hence trains more rapidly than FLANN. CHPNN gives better efficiency than multi- layered networks and FLANN. [15, 16].The r-order Chebyshev polynomials are defined by $CH_r(i)$. These polynomials are recursively defined as follows:

$$CH_{j+1}(i) = 2iCH_j(i) - CH_{j-1}(i) \quad (1)$$

The zero and first order polynomials are defined as $CH_0(i) = 1$ and $CH_1(i) = i$.
Using the recursive equation (1) a n-dimensional input $I=[i_1, i_2 \ . \ . \ . \ i_n]^T$ is expanded to a d-dimensional pattern
CHI by the hidden block as $CHI = [1, CH_1(i_1), CH_2(i_1) \ . \ . \ . \ CH_r(i_1), \ . \ . \ . \ , CH_1(i_n), CH_2(i_n) \ . \ . \ . \ CH_r(i_n)]^T$, where $d= r*n+1$. The Weighted Sum (WS) is obtained as follows:

$$WS = \sum_{k=1}^{x} w_k CHI_k \quad (2)$$

The WS is used to predict an output by using any activation function. This forecasted output is compared to the actual output to generate a deviation. The deviation is used to generate updated weights so as to optimize the predicted output by using any learning method.

PSO Learning

PSO is a nature-inspired, population-based learning method developed by Dr. Eberhardt and Dr. Kennedy in 1995. In this learning method, individual bird positions represent the solution to the problem domain [17, 18]. The bird’s velocities are also used randomly as a way of depicting the birds' speed of changing their positions.
The new positions and velocities are calculated using the following equations:

TEST Engineering & Management
January - February 2020
ISSN: 0193 - 4120 Page No. 2313 - 2322

$$V^{i+1}_{p,q} = \alpha V^{i}_{p,q} + a_1 r_o (oBest^{i}_{p,q} - P^{i}_{p,q}) + a_2 r_s (sBest^{i}_{p,q} - P^{i}_{p,q}) \quad (3)$$

$$P^{i+1}_{p,q} = P^{i}_{p,q} + V^{i+1}_{p,q} \quad (4)$$

Where oBest is birds' own best position and sBest is the swarms' best position. r_o, r_s, a_1, a_2 and α are PSO parameters.

The pseudocode for PSO for a minimization problem: ***Input:*** *Bird Positions (P) and corresponding Velocities (V)* ***Output:*** *Optimized Position*

While max-iterations for each P_i

Calculate fitness (P_i)

if fitness(P_i) <= fitness(oBest)

Set oBest = P_i

if fitness(oBest) <= fitness(sBest) Set sBest = oBest

[End of for] for each P_i

Calculate updated V_i using equation (3)

Calculate updated P_i using equation (4)

[End of for] [End of while]

Return sBest as the optimized position after training the network.

DE Learning

DE is a learning method brought in by Storn and Price in 1996. In this method, a set of candidates form the solution space. Initially, the candidates are randomly generated. They iteratively mutate and crossover each other followed by a selection of a fit candidate for the next iteration [19, 20]. The mutation operation is defined as follows:

$$d_{c,k} = y_{c1,k} + F\,(y_{c2,k} - y_{c3,k})\ \text{where}\ c \neq c1 \neq c2 \neq c3 \quad (5)$$

Where $d_{c,k}$ is the mutated donor formed for each candidate c . F, c1, c2 and c3 are the DE parameters. The crossover operation is defined as follows:

$$t_{c,k} = d_{c,k} \ \text{if}\ r1 \leq rc \ \text{or}\ k = k_{rand} \ \text{else}\ t_{c,k} = y_{c,k} \quad (6)$$

Where $t_{c,k}$ is the trial candidate, r1 is a random number and rc is a DE parameter. The pseudocode for DE for a minimization problem:

Input: *Population of Candidate Solutions(C)*

Output: *Optimized Candidate While max-iterations*

for each C_i

Calculate fitness(C_i) for each C_i

d_i = mutate(C_i) using equation (5) for each C_i

t_i = crossover(C_i ,d_i) using equation (6) for each C_i

if(fitness(t_i)<= fitness(C_i)) Set C_i=t_i

[End of While]

Return C_i with minimum fitness as the optimized candidate after training the network.

3. Proposed Work

This work uses the CHPNN model along with the two basic evolutionary algorithms to predict the bitcoin prices. The pictorial representation of the proposed work is shown in figure 1.

The historical data of bitcoin prices for Europe, Japan and US is collected as

TEST Engineering & Management

January - February 2020
ISSN: 0193 - 4120 Page No. 2313 - 2322

BTC/EUR, BTC/JPY and BTC/USD datasets, respectively. These prices are collected for the time range of 8th August 2017 to 16th October 2019. Each dataset has 800 samples. The samples are normalized using Max-Min Technique to generate a uniform dataset of size 795. These normalized data are then passed through a sliding window of size 5. Using a 2:1 ratio, each dataset is further categorized into 530 samples for training and 265 samples for testing. The training samples are utilized to PSO-train and DE-train the CHPNN model for a population size of 10 for 60 iterations. The optimized weight generated after training is used for testing for the rest 265 samples for a prediction horizon 1 model.The proposed model is evaluated based on the following three error measures:

$$RMSE = \sqrt{\frac{1}{D}\sum_{i=1}^{D}\left(abp_i - pbp_i\right)^2} \qquad (7)$$

$$SSE = \sum_{i=1}^{D}\left(abp_i - pbp_i\right)^2$$

$$RRSE = \sqrt{\frac{\sum_{i=1}^{D}\left(pbp_i - abp_i\right)^2}{\sum_{i=1}^{D}\left(pbp_i - \left(\frac{1}{D}\right)\sum_{i=1}^{D} abp_i\right)^2}} \qquad (9)$$

Where abpi is the actual bitcoin price and pgpi is the predicted bitcoin price and D is the number of data specimen collected. The RMSE calculation is used as the fitness function which is being minimized in each iteration.

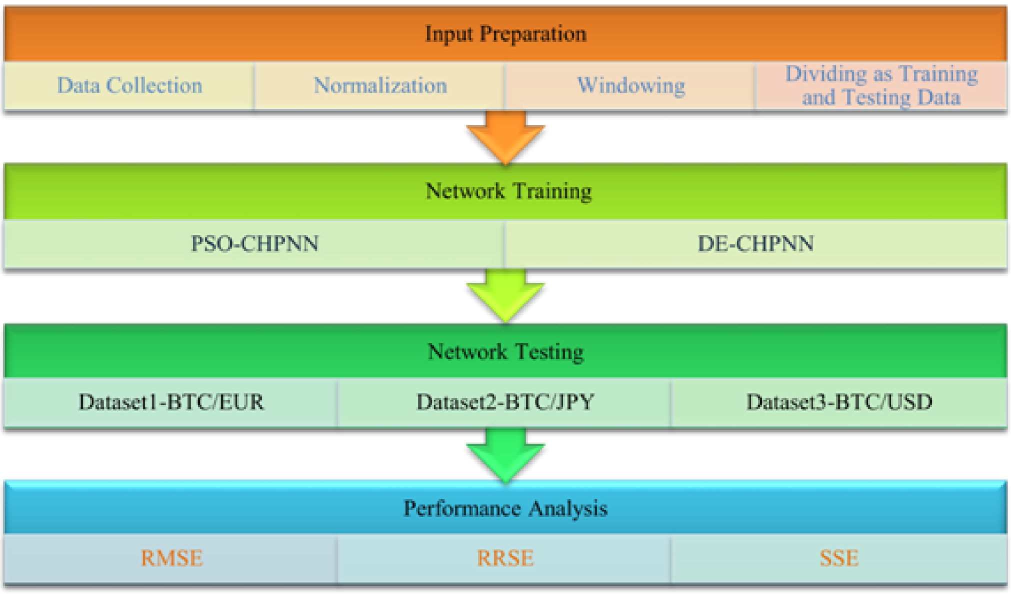

Fig. 1. Proposed Workflow

4. Result Analysis

The CHPNN model is trained and tested using three datasets and two evolutionary learning methods: PSO and DE. The training is done 10 times and the average errors are calculated. All the output values are recorded in the following tables to analyze the result. The mean for the three datasets are recorded respectively in the table 1, 2 and 3.

January - February 2020
ISSN: 0193 - 4120 Page No. 2313 - 2322

Table 1: Mean and Deviation for BTC/EUR Dataset

Expansion Order	Learning Algorithm	RMSE	SSE	RSSE
2	PSO	0.0671	1.3319	0.3091
	DE	**0.0412**	**0.5108**	**0.1972**
3	PSO	0.0756	1.6430	0.3405
	DE	0.0521	0.7940	0.2549
4	PSO	0.1233	4.6721	0.4783
	DE	0.0721	1.4411	0.3657

Table 2: Mean and Deviation for BTC/JPY Dataset

Expansion Order	Learning Algorithm	RMSE	SSE	RRSE
2	PSO	0.0426	0.5231	0.2617
	DE	**0.0331**	**0.3061**	**0.2004**
3	PSO	0.0961	2.6865	0.4498
	DE	0.0542	0.9815	0.3713
4	PSO	0.0858	2.0682	0.4715
	DE	0.0810	2.1151	0.5811

Table 3: Mean and Deviation for BTC/USD Dataset

Expansion Order	Learning Algorithm	RMSE	SSE	RRSE
2	PSO	0.0507	0.7597	0.2702
	DE	**0.0365**	**0.3671**	**0.2051**
3	PSO	0.0962	2.7669	0.4107
	DE	0.0487	0.6934	0.3032
4	PSO	0.0976	2.7087	0.4465
	DE	0.1098	5.4274	0.5040

The table shows that for most of the datasets DE outperforms PSO. Though the higher expansion order better maps the nonlinear relations, the prediction using the least expansion order, that is, order 2 is more promising. This work reveals that DE-CHPNN of order 2 is the best prediction model used in this work as the error measures are smallest for second order expansion. It is also signalling that when expansion order increases, PSO performs better than DE for the third dataset.

The price prediction and corresponding RMSE for each dataset using second order CHPNN is depicted in the figure 2 to figure 7. The minimum RMSE for all the three datasets for a second order CHPNN is shown in table 4.

Table 4. Errors for the Second Order CHPNN

Dataset	Learning Algorithm	Test RMSE	Test SSE	Test RRSE
BTC/EUR	PSO	0.044	0.5141	0.2092
	DE	**0.0264**	**0.1848**	**0.145**
BTC/JPY	PSO	0.205	0.1657	0.1518
	DE	**0.0227**	**0.1369**	**0.1351**
BTC/USD	PSO	**0.0239**	**0.1518**	**0.1327**
	DE	0.025	0.1651	0.1364

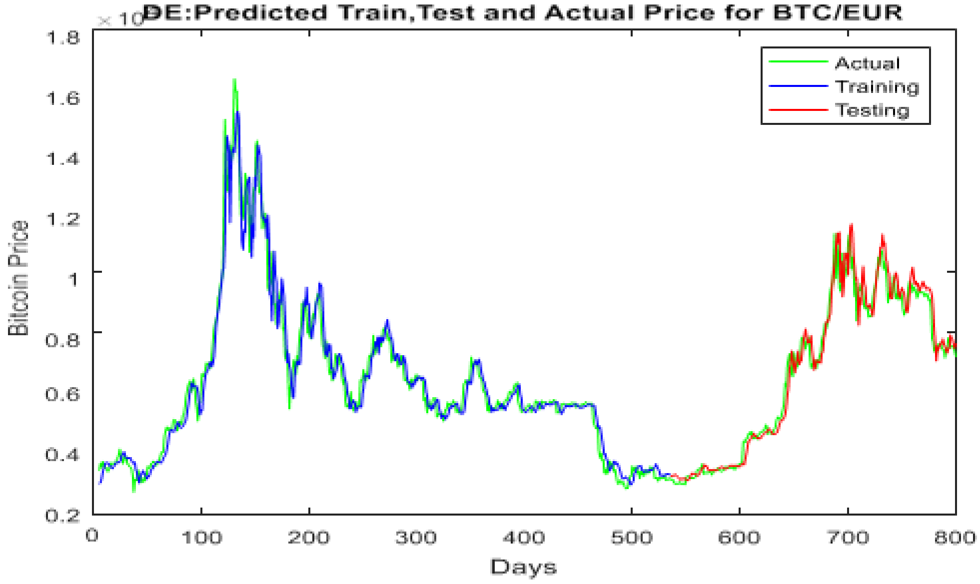

Fig. 2. Pricing for BTC/EUR Dataset

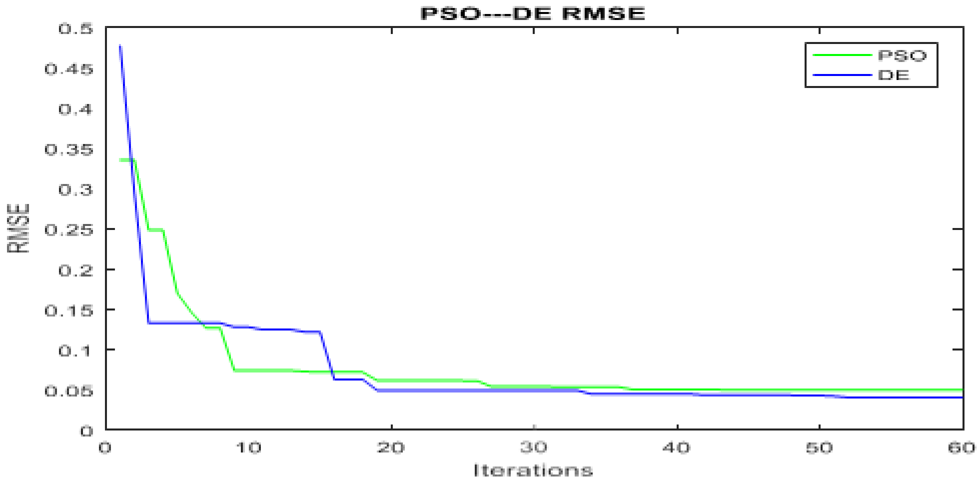

Fig.3 RMSE for BTC/EUR Dataset

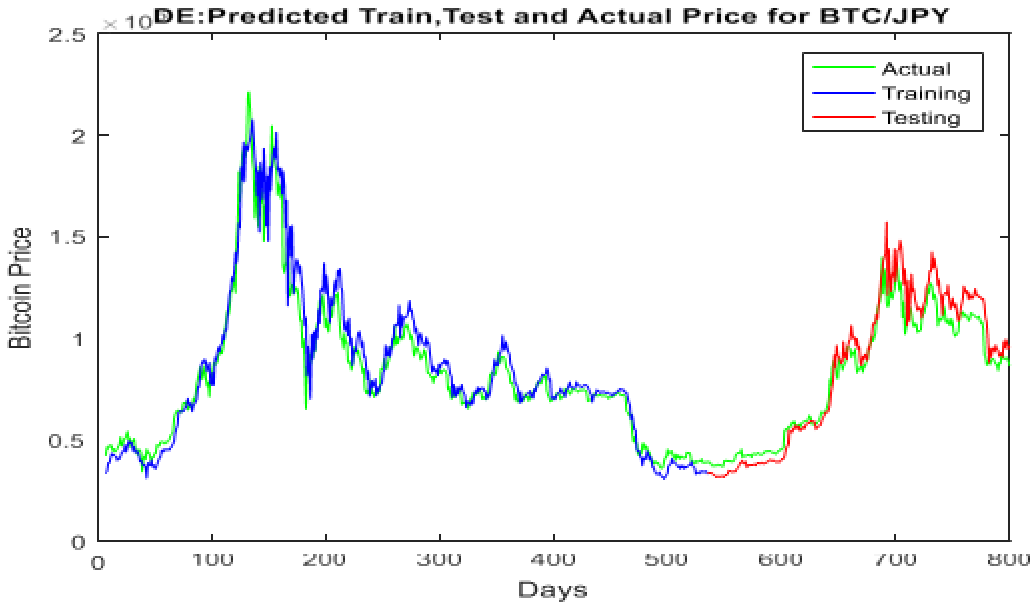

Fig. 4. Pricing for BTC/JPY Dataset

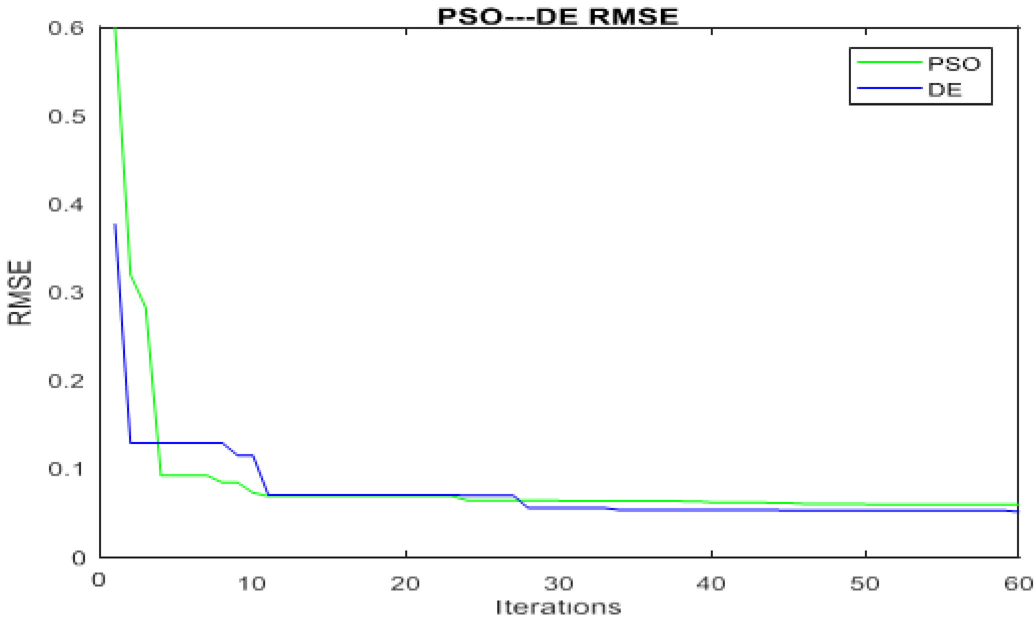

Fig. 5. RMSE for BTC/JPY Dataset

January - February 2020
ISSN: 0193 - 4120 Page No. 2313 - 2322

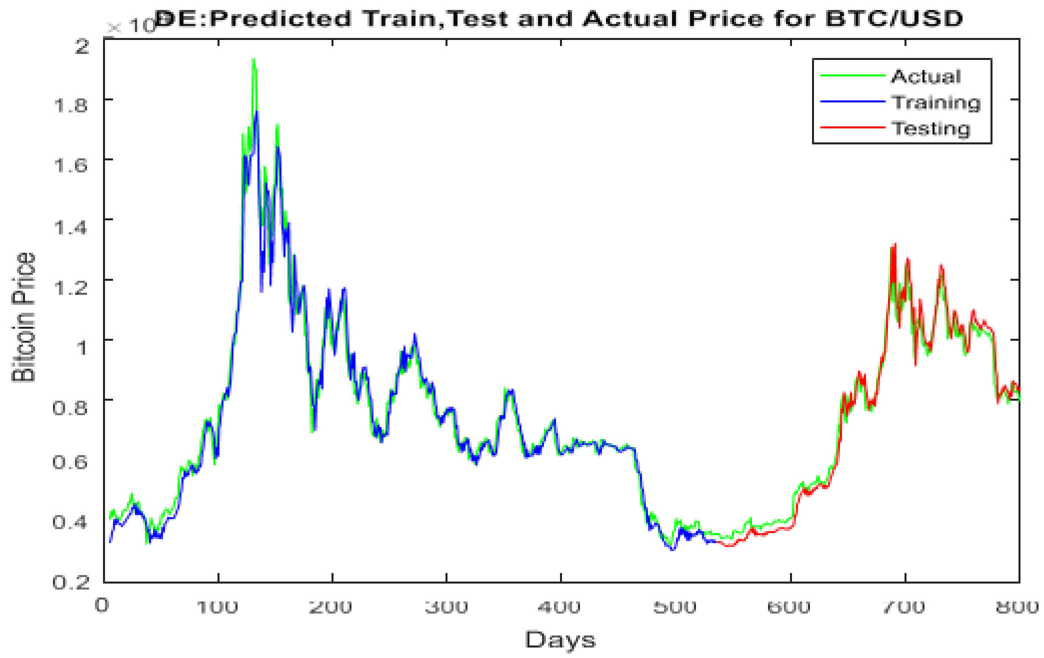

Fig. 6. Pricing for BTC/USD dataset

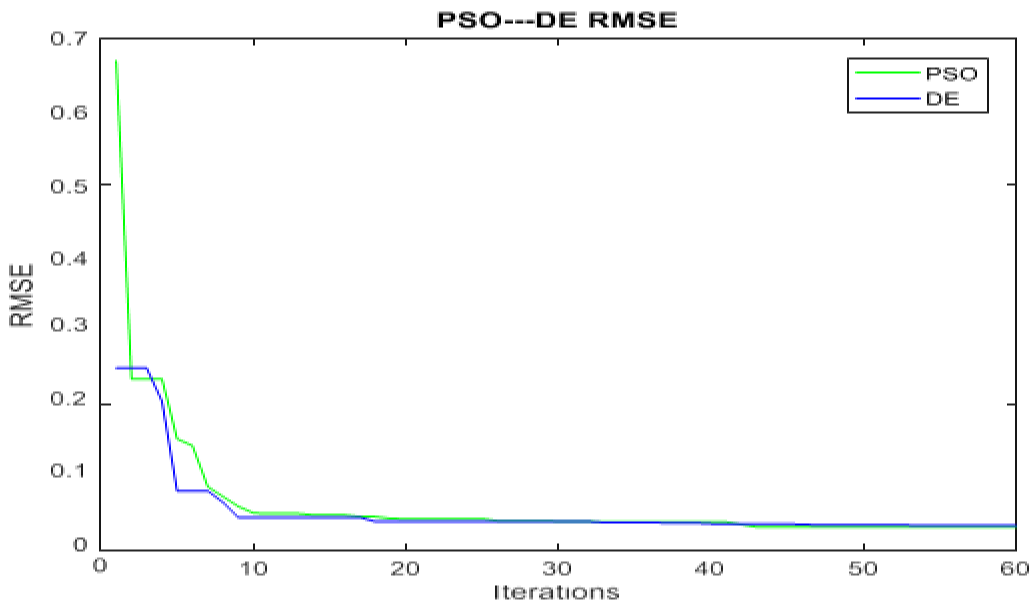

Fig. 7. RMSE for BTC/USD Dataset

5. Conclusion

In conclusion, this work reveals that the DE-CHPNN model is a better prediction model for bitcoin prediction. Also, the work shows that when the expansion order increases, prediction accuracy decreases. The study also reveals that when the expansion order increases PSO outperforms DE for some dataset. This comparative analysis suggests a second order CHPNN model with DE algorithm for bitcoin prediction. The high volatility of bitcoin is better captured by the DE-based model.

This work can be further extended for feature selection aspect of price prediction.

This study can be further extended to use various other regression models for bitcoin prediction.

References

1. Santosi, N. Bitcoin:A peer-to-peer electronic cash system. November. 2018.
2. Dong, H. et al. Virtual Currencies and Beyond: Initial Considerations. January 2016.
3. Michel, B., Oosterlinck, K. and Szafarz, A. Virtual currency, tangible return: Portfolio diversification with bitcoin. Journal of Asset Management. vol. 16. no. 6, pp. 365-373,2015.
4. Mallaqui, D.C.A. and Fernandes, R.A.S. Predicting the direction, maximum, minimum and closing prices of daily Bitcoin exchange rate using Machine Learning Techniques. Applied Soft Computing Journal, 75(2019) pp. 596-606.
5. Jang, H. and Lee, J. An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information. IEEE Access. November, 2017.
6. Sin, E. and Wang, L. Bitcoin Price Prediction Using Ensembles of Neural Networks. 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery.2017.
7. Guo, T. and Antulov-Fantulin, N. An experimental study of Bitcoin fluctuation using machine learning methods, in arXiv preprint, https://arxiv.org/abs/1802.04065, 2018.
8. .Katsiampa, T. Volatility estimation for Bitcoin: A comparison of GARCH models, Economics Letters, vol. 158, pp. 3-6, 2017/09/01/2017.
9. McNally, S., Roche, J. and Caton, S. Predicting the Price of Bitcoin Using Machine Learning. 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), 2018, pp. 339-343.
10. Karakoyun, K.N. and Çibikdiken, A.O. Comparison of ARIMA Time Series Model and LSTM Deep Learning Algorithm for Bitcoin Price Forecasting. The 13th Multidisciplinary Academic Conference in Prague 2018 (The 13th MAC 2018), ed. Czech Republic, 2018, pp.171-180.
11. Shah, D., Zhang, S. Bayesian regression and Bitcoin, Communication, Control, and Computing (Allerton). 52nd Annual Allerton Conference on IEEE.(2014).
12. Madan, I., Saluja, S. and Zhao. A. Automated Bitcoin Trading viaMachine Learning Algorithms.2015.
13. Mishra, M. K. and Dash, R. A Comparative Study of Chebyshev Functional Link Artificial Neural Network, Multi-layer Perceptron and Decision Tree for Credit Card Fraud Detection. IEEE International Conference in Information Technology (ICIT), (2014), 228-233.
14. Patra, J.C., Thanh,N.C. and Meher, P.K. Computationally efficient FLANN-based intelligent stock price prediction system. IEEE International Conference in Neural Networks, (IJCNN), (2009), 2431-2438.
15. Jiang, L. L., Maskell, L. L., & Patra, J. C. Chebyshev functional link neural network based modeling and experimental verification for photovoltaic arrays. IEEE International Conference in Neural Networks (IJCNN), (2012), 1-8.
16. Li, M., Liu, J., Jiang, Y., Feng, W. Complex-Chebyshev functional link neural network behavioral model for broadband wireless power amplifiers. IEEE Transactions on Microwave Theory and Techniques, 60(6), (2012), 1979-1989.
17. Dash, R., Dash, P. K. & Bisoi, R. A self adaptive differential harmony search based optimized extreme learning machine for financial time series

prediction. Swarm and Evolutionary Computation, 19, (2014), 25-42.
18. Dash, R., & Dash, P. K. Prediction of Financial Time Series Data using Hybrid Evolutionary Legendre Neural Network: Evolutionary LENN. International Journal of Applied Evolutionary Computation, Vol. 7(1), (2016). 16-32.
19. Rout, A. K., Dash, P. K., Dash, R., & Bisoi, R. Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach. Journal of King Saud University-Computer and Information Sciences, Vol. 29(4), (2017) 536-552.
20. Dash, R. DECPNN: A hybrid stock predictor model using Differential Evolution and Chebyshev Polynomial Neural Network, Intelligent Decision Technologies: an International Journal, 12(1), (2018), 93-104.