Representation of Financial Time Series Data in a Three-Dimensional Space Using a Geographic Model

Kambiz Borna, Duaa Alshadli

Abstract

This paper introduces a new approach to visualising and interpolating financial time series data, e.g., Bitcoin prices, in a spatial domain using the notion of spatialization: forming a spatial representation of non-spatial phenomena. The proposed algorithm first utilises temporal components of the observations, i.e., date and time, to build a 2D map-like space. It then uses the coordinates of the observations in the 2D map along with the Bitcoin prices to construct a 3D topographic map. We use this map to create 30-minute frequency data, and compare it with the actual observed Bitcoin prices. The results show the reliability and effectiveness of the proposed method as a new graphical tool in analysing time series data.

1. Introduction

The vast majority of the algorithms in financial markets are established based on using time series data observed at equal time intervals. In this structure, an observation at time interval t, x_t, is formally defined via a n-dimensional vector, where $n \in \mathbb{N}$. A typical case of time series data is when n=1, e.g., using the closing prices of Bitcoin [1,2]. In this case, there is one observation at each time interval that changes with time. These changes are usually represented and modelled using an irregular line or curve in a one-dimensional space, where the x-axis represents time with a fixed interval, and the y-axis represents the variable of interest, e.g., the daily closing price of Bitcoin.

In the case that n>1, e.g., using the opening, high, low and closing prices of Bitcoin [3,4], the algorithms use a set of observations at each time step for modelling time series data. In this scenario, the changes in the observations are typically represented via a combination of regular and irregular lines, e.g., candlestick or bar charts. The irregular horizontal line is formed along the x-axis using the scalar value observed at equal time, e.g., the daily closing price. The straight vertical lines are applied to represent the opening, high, low and closing prices along the y-axis. The lengths of vertical lines are adjusted based on the difference between the values of the observation vector at each time interval.

The above geometric structures have two main characteristics in common. First, they use lines and curves as main geometric elements to model the dependencies between observations in the time series. Second, they can only represent observations in a one-dimensional cross section view. In this paper, we propose a novel approach for analysing and visualising the time series data in a 3D map-like environment. This approach is formulated based on spatialization: modelling a non-spatial phenomenon in a spatial domain [5,6]. The use of this model gives the power to time series representation algorithms to simultaneously link observations gathered at different times and represent them in a 3D space. This enables traders or financial analysers to use new graphical tools, e.g., 3D projections, in analysing and representing time series data in a simple and meaningful way.

In Section 2, the methodology of the proposed method will be described. Section 3 discusses the results of the proposed method. Section 4 concludes the main outputs of the paper and presents other areas for the development of the proposed method in future research.

2. Data and methodology

To interpolate data and assess the proposed approach, Bitcoin data from August 1, 2021, to August 13, 2021, was downloaded from https://firstratedata.com/i/crypto/BTC and applied. We used one-minute frequency data to identify each day's opening, high, low, and closing price and the times assigned to these prices. The proposed method is implemented in two steps: creating the 2D spatialized map and constructing the topographic map. MATLAB2022a and ArcGIS 10.5 are applied to create, analyse, and render the 2D and 3D maps in a 3D environment.

2.1. Creation of the 2D spatialized map

To transfer the temporal data of Bitcoin prices into a 2D map, we use analytic geometry, where the location of each point in a two-dimensional space is defined by an ordered pair of numbers (*x*, *y*). Let $B = \{b_1, b_2, \ldots, b_t\}$, *t* = 1, 2, . . ., *N* represents a set of Bitcoin observations where *N>1,* and specifies the total number of observation days. b_t values include a pair of numbers (p_i, h_i), *i* = 1, . . ., *4.*, where p_i is the price and h_i is the time at which this price is recorded on day *t*. Therefore, each dataset's price has two temporal components: event time and event day. Event times repeat themselves every 24 hours via the opening and closing prices of each day, like the day/night cycle in the real world. We use event time values, h, to define a *y* value for each price in the 2D map. Event day, *t,* determines the *x*-coordinates for each price in the 2D map. In the time-space, they are defined based on the notion of linear time, moving in one direction without repetition. We apply these two components (*t*, *h*) to place data samples (Bitcoin price) in a 2D map (Figure 1 (a)).

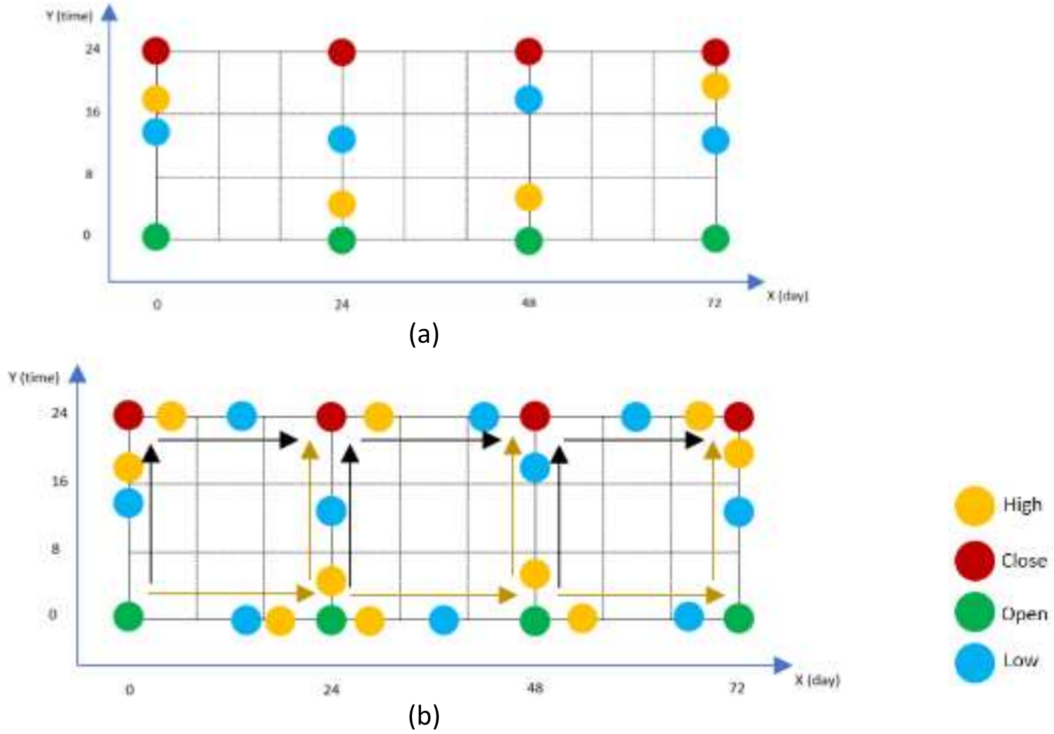


Figure 1. (a) and (b) display the events (prices) along the x-axis and y-axis, and (b) the black and brown arrows show different directions in each square to link the open price in one day to the close price in the next day. (b) creates a symmetric 2D space along the southwest to the northeast axis every 48 hours.

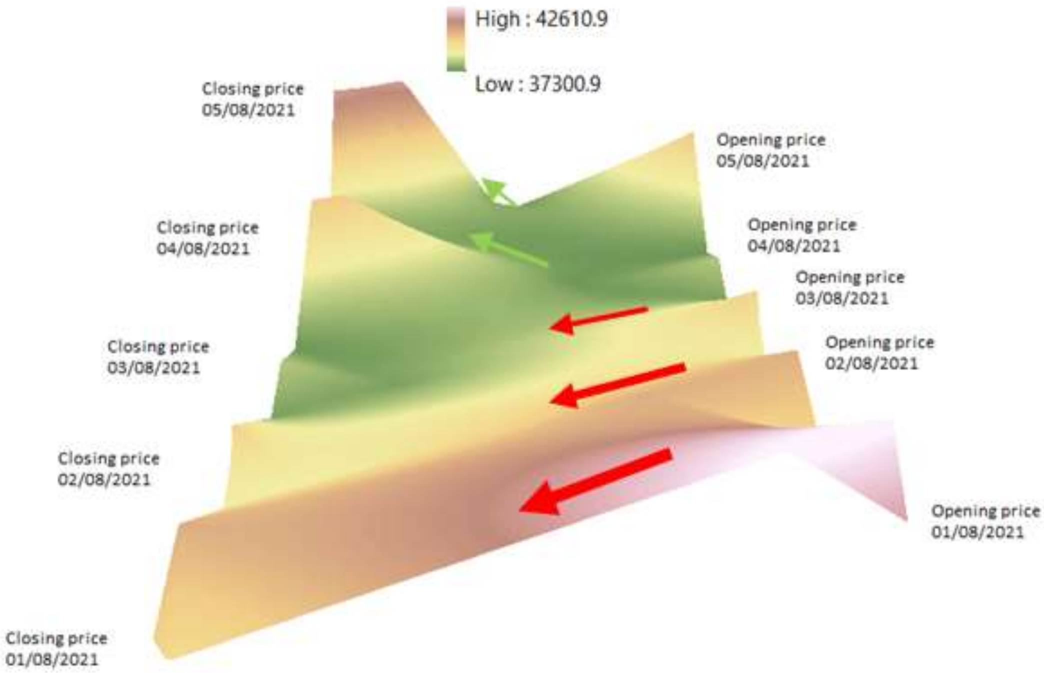
To convert data from a linear structure to a polygonal structure, the data points are also added to the horizontal lines at y=0 (0 minutes) and y=24 (1440 minutes) (Figure 1 (b)). This figure illustrates a 2D coordinate system that includes a new set of observations $F = \{f_1, f_2, \ldots, f_k\}$, k = 1, 2, . . ., S, where S= $4\times(2N\text{-}1)$ and defines the number of point features in the 2D vector map in Figure 1(b). Each point feature f has the coordinate attributes x, y that determine the location of each point (price) on the map. As shown in Figure 1(b), there are two routes in each tile that link opening, high, low, and closing prices of each day to the prices the next day. These two routes form a square of size 24×24 every 24 hours on the x-axis, where time moves from one day to the next. The 2D map in Figure 1(b) provides a 2D planar framework that allows the prediction algorithms to use spatial tools and functions to analyse crypto data.

2.2. Construction of the topographic map

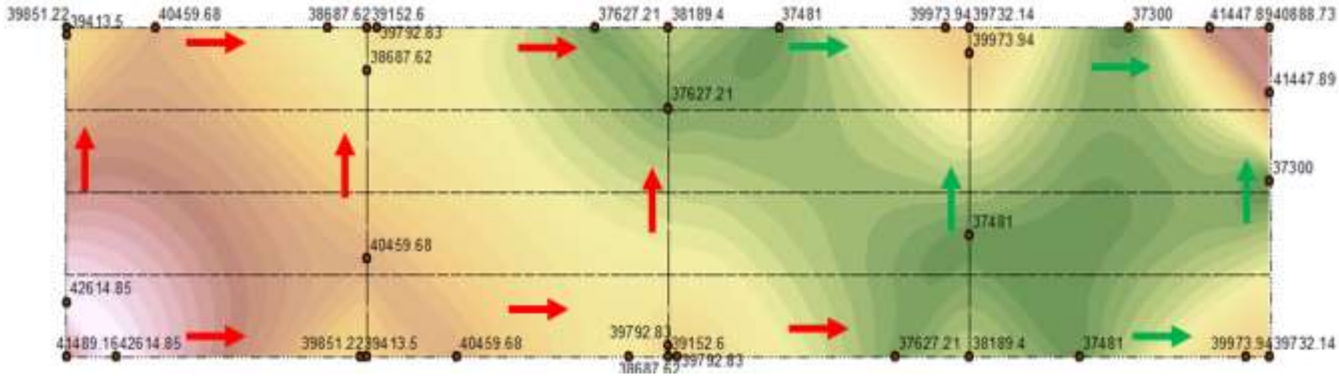
Let $P = \{p_1, p_2, \ldots, p_m\}$ represent a set of m unknown values where m is specified based on the number of cells in the above 2D map. Each p has coordinates determined by (x, y). To calculate the z value of p in the 3D map, the method uses the following equation:

$$p(x, y) = \sum_{i=1}^{n} w_i \times f(x_i, y_i)\ , \tag{1}$$

where $p(x, y)$ is the estimation at (x, y), and n is the number of nearest neighbors used for interpolation. $f(x_i, y_i)$ is the observed data from set $F = \{f_1, f_2, \ldots, f_k\}$, and w_i is the weight associated with $f(x_i, y_i)$. The Natural Neighbor Interpolation (NNI) algorithm is applied to calculate w_i[7]. The weights are determined via a set of irregular polygons formed based on the location of observation and unknown data in the 2D map. In this algorithm, the closest sample points to an unknown point have the highest influence on that point's value in the spatial domain. The method, in fact, indirectly applies Tobler's first law of geography *"everything is usually related to all else, but those which are near to each other are more related when compared to those that are further away"* [8]. This is similar to the rules that are generally applied in a time domain by conventional prediction algorithms to estimate an unknown price. Figure 2 shows a Bitcoin map generated using Equation 1 via an iterative process. In Figure 2, the green arrows indicate when the closing price is higher than the opening price, and the red arrows illustrate when the closing price is lower than the opening price.



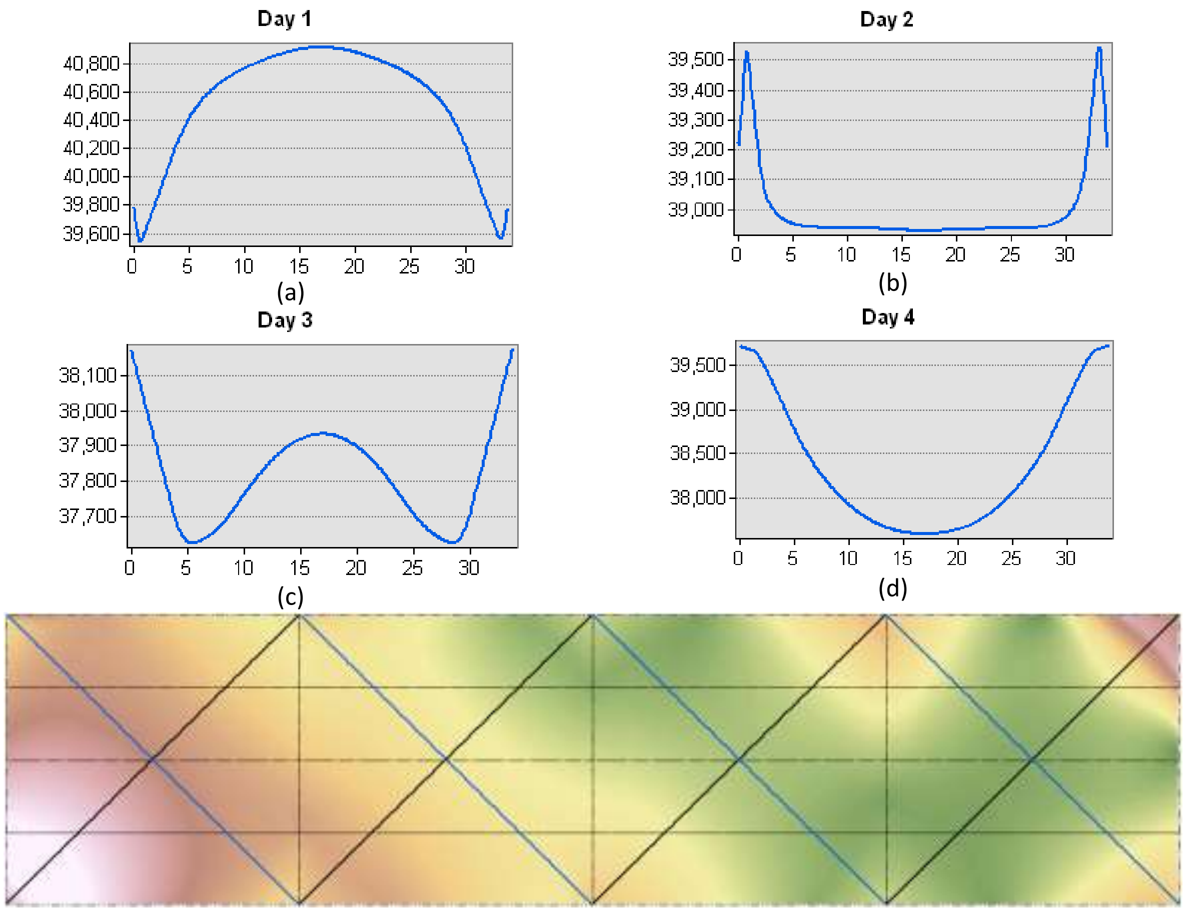
(a)



(b)

Figure 2. (a) 3D cross-section view of Bitcoin price. (b) The produced Bitcoin map using the NNI algorithm. Point features are applied to show the location of each price on the map.

We use the 3D maps to extract Bitcoin prices each day to assess the behaviour of Bitcoin prices. Figure 3 (a-d) illustrates the behaviour of bitcoin prices using the proposed method between the closing price of one day and the opening price of the next day, which have the same value. These graphs have a symmetric shape based on line x=34, $\frac{\sqrt{(24^2+24^2)}}{2}$. This is because sample points are repeated along the lines y=0 and y=24, as the line y=0 includes the data from one day and the line y=24 contains samples of the next day. Figure 3(b) shows that the opening and closing prices are close in day 2. In Figure 3(a), a pit exists close to the endpoint in the graph, which means that the lowest price on day 1 is similar to the closing price on day 1.



(e)

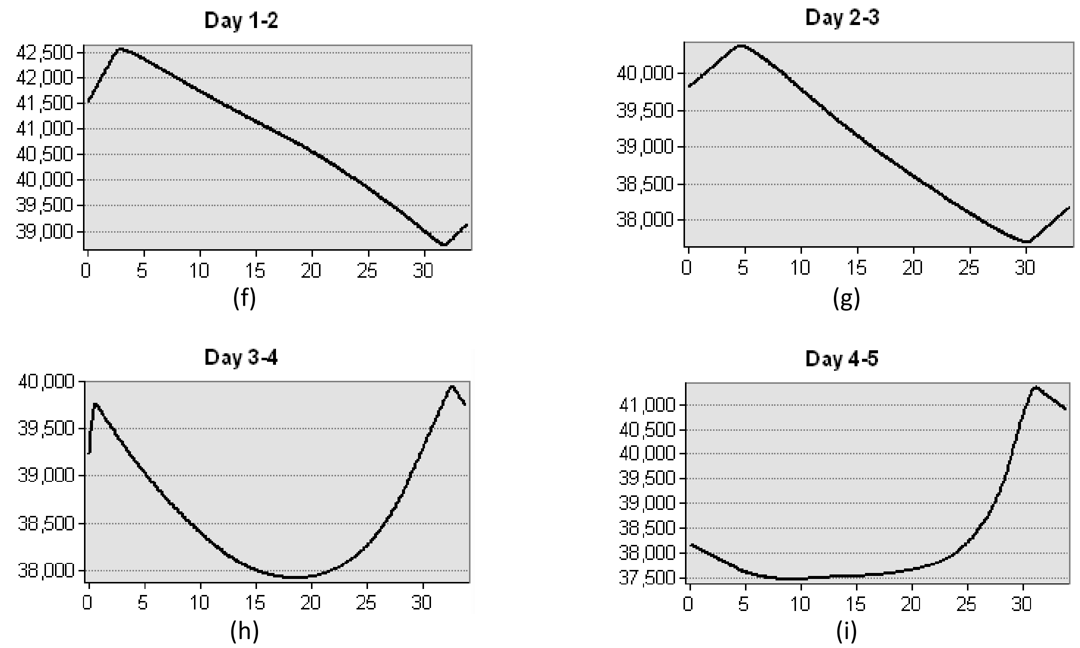


Figure 3. (a-d) profile views of the map (e) along the northwest-southeast line (blue). (f-i) profile views of the map (e) along the southwest-northeast line (black)

Profiles in Figure 3(f-i) are created using the southwest-northeast line. That means these profiles show the change in exchange price from one day to the next. They are a scaled representation of 48 hours of data between two closing prices. Figures 3(f) and 3(g), show a steep downward slope, which means the prices are falling. Figure 3(h) shows a pit close to the line x=17 and two peaks around the end points. This illustrates a transition from a red market to a green market. Figure 3(i) displays a pit around the starting point and a peak near the ending point, indicating that the prices in day 5 are higher than the prices in day 3.

3. Results

To quantitively assess the results of the proposed method, we use the 30-minute frequency Bitcoin prices, which are divided into two groups: Dataset 1 (1/8/2021-4/8/2021) and Dataset 2 (8/8/2021-12/8/2021). The following metrics: RMSE (root mean squared error), MAPE (mean absolute percentage error), and DA (directional accuracy) are applied to assess the accuracy and dynamic behaviour of the interpolated prices via the 3D maps [2].

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y(t) - \hat{y}(t))^2} \tag{2}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y(t) - \hat{y}(t)}{y(t)}\right| \tag{3}$$

$$DA = \frac{1}{n}\sum_{t=1}^{n} a(t) \times 100\% \tag{4}$$

Where the $y(t)$ and $\hat{y}(t)$ are the actual and interpolated prices at time *t*, respectively, and *n* is the number of observations. In Equation 4, $a(t)$=0 if $(y(t+1) - y(t)) \times \left(\hat{y}(t+1) - y(t)\right) < 0$, otherwise $a(t)$=1. To transform the irregularly spaced time series prices into regularly spaced time series prices, we also use the 8th degree polynomial, and the Hermite interpolation function, which is denoted as irregular data interpolation (IRDInterpolation) in Table 1. This allows us to compare the performance of the proposed method against these two conventional methods, which are usually applied to create regular time series data from irregular time series data. We also apply the Hermite interpolation function to transform the 24-hour frequency prices into 30-minute frequency prices, denoted as regular data interpolation (RDInterpolation) in Table 1. This enables us to assess the effect of using multiple observations, namely high and low prices, in the interpolation process. Table 1 shows the results of these four methods in terms of the RMSE, MAPE and DA values. The best and worst values are highlighted in bold and underlined, respectively.

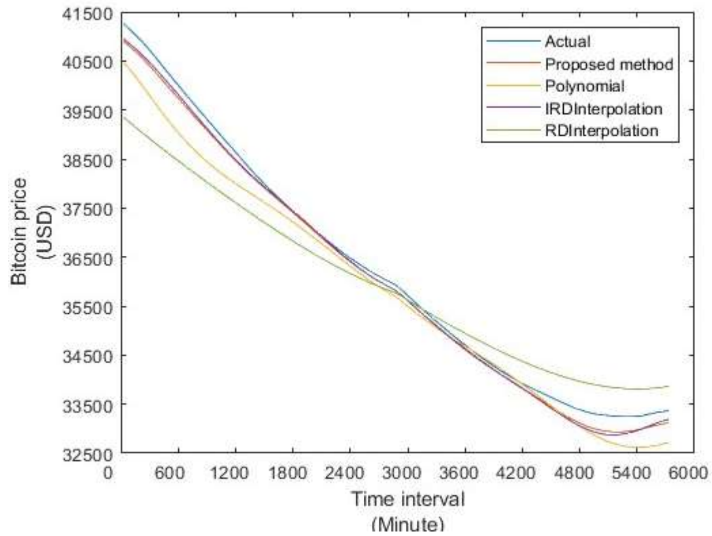
Table 1. shows the RMSE, MAPE, and DA values of the interpolation data created using four different methods: the proposed method (3D maps), IRDInterpolation, RDInterpolation, and the 8th degree polynomial function.

Data	Method	RMSE	MAPE	DA
	Proposed method	**484.75**	**0.91**	**56.25**
Dataset 1	IRDInterpolation	594.70	1.12	<u>54.17</u>
	Polynomial	<u>724.45</u>	<u>1.19</u>	55.73
	RDInterpolation	552.30	0.99	55.21
	Proposed method	**429.14**	**0.75**	**58.33**
Dataset 2	IRDInterpolation	485.76	0.89	<u>53.33</u>
	Polynomial	<u>658.99</u>	<u>1.10</u>	**58.33**
	RDInterpolation	519.98	0.91	55.00

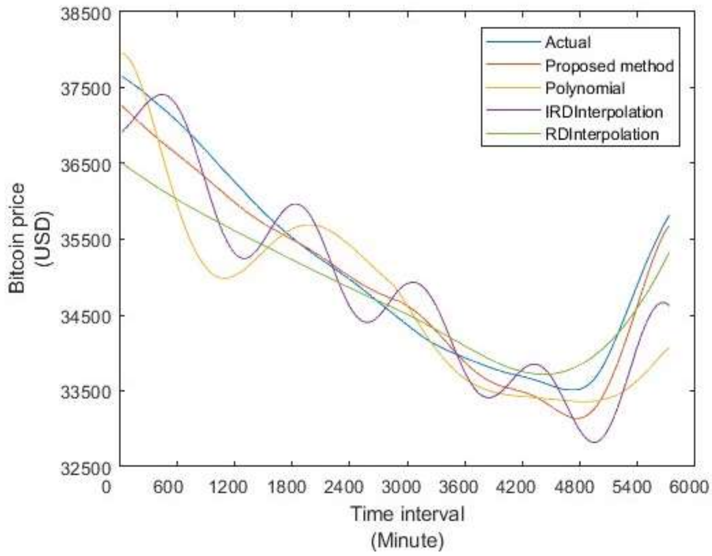
As shown in Table 1, the proposed method provides better results compared with the other three methods for both datasets. In Dataset 1, the proposed method improves the RMSE values by more than 18%, 33%, and 12% compared with the IRDInterpolation, Polynomial, and RDInterpolation methods, respectively. For the MAEP values, an improvement of 17%, 23%, and 8% can be seen using the proposed method compared to the other methods. The highest DA value for Dataset 1 is 56.25, which belongs to the proposed method. This indicates that the 3D maps not only improve the level of accuracy of the interpolated data but also model the dynamic behaviour of time series data better in comparison with the other methods.

The RMSE and MAPE values confirm the better performance of the IRDInterpolation and RDInterpolation algorithms compared with the polynomial method. However, the DA values in Table 1 indicate that the polynomial algorithm performs better than the interpolation function in modelling the dynamic behaviour of the interpolated time series prices. This is because the interpolating function must pass exactly through all the observed prices. Therefore, if there is a sudden jump or drop rise in the Bitcoin prices, the algorithm might create unwanted undulations by decreasing or increasing slopes of interpolant at the interpolation points. This can change the dynamic behaviour of the time series data. The results in Table 1 for Dataset 1 also show the better performance of the RDInterpolation method compared with the IRDInterpolation, while both methods use the same interpolation function to create the data samples. The possible reason for the better performance of RDInterpolation is that the method is formed based solely on the closing price observed with the 24-hour interval. Therefore, the low and high prices cannot affect the slopes of the interpolant. While the RDInterpolation algorithm is formed based on multiple observations. This can limit the accuracy of the interpolation function when there is a sharp rise or fall in the Bitcoin prices.

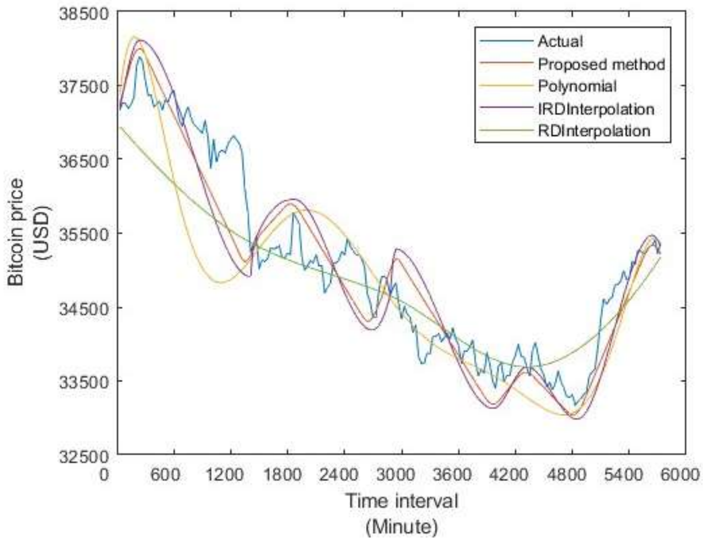
Similar to Dataset 1, the results of the proposed method show a better performance in generating the interpolated prices based on Dataset 2 compared with the other three methods. For example, the use of the 3D map generated data shows an improvement of more than 12% for the RMSE and MAPE values compared with the second-best RMSE and MAPE values in Table 1. For Dataset 2, the highest DR value is 58.33, which belongs to the polynomial and the proposed algorithm. However, the RMSE and MAPE values show that the polynomial function performs poorly compared to the proposed method. The poor results of the polynomial function can be due to the fact the function is formed by fitting a mathematical model to the prices. Therefore, it is not necessary for the function to pass precisely through the observed prices. This can reduce the model's accuracy, especially when the volatility of the Bitcoin prices is high. For both datasets, a comparison between the results of the 3D maps and RDInterpolation confirms that the use of the closing, low and high prices to interpolate the time series data can significantly improve the accuracy of the interpolated data. Figure 4 shows the time series data and its corresponding trend components for Dataset 1 and Dataset 2, which are drawn using the actual data and the interpolated data.



(a)

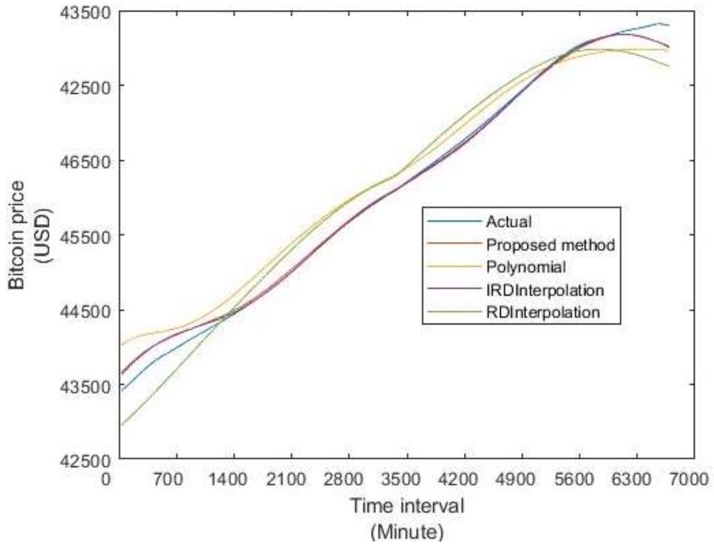


(b)



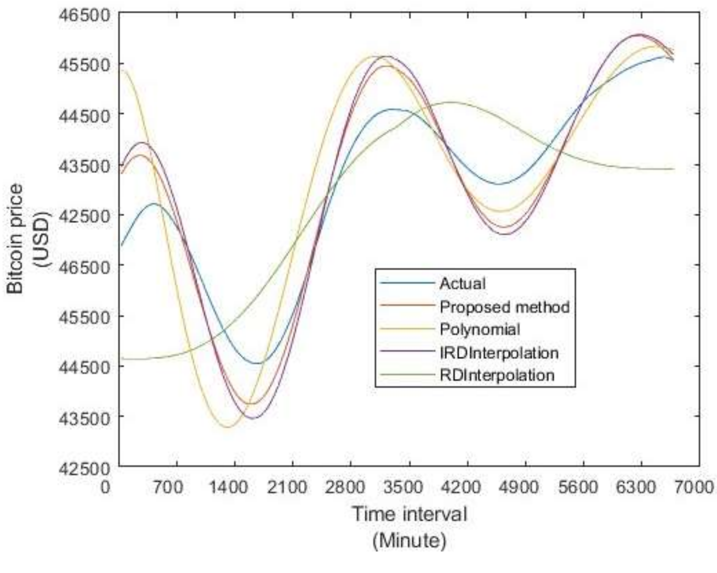
Actual
Proposed method
Polynomial
IRDInterpolation
RDInterpolation
Bitcoin price
(USD)
38500
37500
36500
35500
34500
33500
32500
0
600
1200
1800
2400
3000
3600
4200
4800
5400
6000
Time interval
(Minute)

(c)



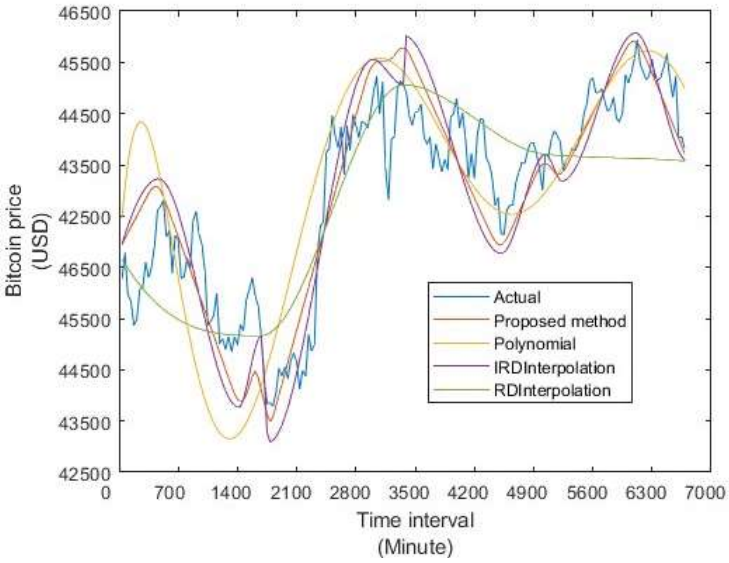
Actual
Proposed method
Polynomial
IRDInterpolation
RDInterpolation
Bitcoin price
(USD)
43500
42500
46500
45500
44500
43500
42500
0
700
1400
2100
2800
3500
4200
4900
5600
6300
7000
Time interval
(Minute)

(d)



Actual
Proposed method
Polynomial
IRDInterpolation
RDInterpolation
Bitcoin price
(USD)
46500
45500
44500
43500
42500
46500
45500
44500
43500
42500
0
700
1400
2100
2800
3500
4200
4900
5600
6300
7000
Time interval
(Minute)

(e)



(f)

Figure 4. (a), (b), and (c) illustrate the trend component of the time series data, the combination of seasonal and trend component of the time series data, the time series data, respectively, for Dataset 1 using the actual data (blue), the proposed method (red), the polynomial function (yellow), the IRDInterpolation method (purple), and the RDInterpolation method (green). (d), (e), and (f) display the trend, combined seasonal and trend, and time-series data for Dataset 2 and using different methods, according to the colour code shown in Figure 4, respectively.

Figure 4(a) illustrates that the geometric behaviour of the trend created by IRDInterpolation highly coincides with the trend generated using the 3D maps. This is because, in both methods, the model is formed by passing through the observed prices. Figure 4(a) also displays that there is a gap between the trend created by the RDInterpolation and IRDInterpolation method from time 0 to time 1800, where there is a sharp drop in the Bitcoin price, while both methods use the same interpolation function. This can be explained by the fact that the RDInterpolation method uses more information to interpolate the data. The graphs in Figure 4 (b) are formed by adding the seasonal component to the trend of the time series data. It can be seen that the proposed method shows better performance than the other methods. A comparison between graphs in Figure 4(c) and Figure 4(b) also confirms the better performance of the 3D map in estimating the stochastic component of the time series data compared with the other methods.

In contrast to Dataset 1 that there is a downward trend, Figure 4(d) shows an upward trend for the Bitcoin prices based on Dataset 2. Similar to the graphs in the first dataset, the trend of the proposed method is the closest trend to the actual data trend. This indicates the ability of the 3D maps to accurately interpolate prices under different circumstances. Figure 4(e) displays that the graphs of the 3D maps and the IRDInterpolation are similar. However, Figure 4(f) shows the created graph by the proposed method is closer to the actual data than the graph generated by the IRDInterpolation method. This means that the 3D map estimates the stochastic component of the time series data better than the IRDInterpolation method. The results in Table 1 and Figure 4 confirm the high capability of the proposed model to analyse time series data and also interpolate time series data without setting any polynomial functions, which are usually applied by the conventional interpolation methods.

4. Conclusions and future works

This paper proposed a novel approach for visualizing and analyzing financial time series data in a three-dimensional space using spatialization. The method used the temporal elements of Bitcoin prices first to transfer data into a 2D map and then to create a 3D surface. In this study, we used the NNI algorithm to estimate the price of unknown points in 3D maps based on spatial relationships between known and unknown feature points.

The results indicate the reliability of the proposed method in analyzing and visualizing time series data in a straightforward manner. It also shows the capability of the method to interpolate the time series data without setting any polynomial functions. Using spatial tools, such as hillshade mapping or visibility mapping to implement financial concepts, e.g., modelling volatility or predicting return prices, would be an exciting area for future research.

References

1. Sebastião, H., Godinho, P. Forecasting and trading cryptocurrencies with machine learning under changing market conditions. Financ Innov 7, 3 (2021).
2. Mingxi Liu, Guowen Li, Jianping Li, Xiaoqian Zhu, Yinhong Yao, Forecasting the price of Bitcoin using deep learning, Finance Research Letters, Volume 40, 2021,101755.
3. Xiaolei Sun, Mingxi Liu, Zeqian Sima, A novel cryptocurrency price trend forecasting model based on LightGBM, Finance Research Letters, Volume 32, 2020.
4. Cohen, G. Forecasting Bitcoin Trends Using Algorithmic Learning Systems. Entropy 2020, 22, 838. https://doi.org/10.3390/e22080838.
5. Skupin, A., and Fabrikant, S. Spatialization Methods: A Cartographic Research Agenda for Non-Geographic Information Visualization. *Cartography and Geographic Information Science*. 2003, 30 (2): 99-119.
6. Moore A. B. & Bricker M. ‘Mountains of work’: spatialization of work projects in a virtual geographic environment, *Annals of GIS*, 21:4, 2015, 313-323.
7. Sibson, R. A Brief Description of Natural Neighbor Interpolation, chapter 2 in Interpolating Multivariate Data. New York: John Wiley & Sons, 1981. 21–36.
8. Tobler W., (1970) “A computer movie simulating urban growth in the Detroit region”. Economic Geography, 46(2): 234-240.