Physica A 554 (2020) 124702

Contents lists available at ScienceDirect

Physica A

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

Correlation dynamics in the cryptocurrency market based on dimensionality reduction analysis

Chun-Xiao Nie

School of Statistics and Mathematics, Zhejiang Gongshang University, Hangzhou 310018, China

ARTICLE INFO

Article history:
Received 6 November 2019
Received in revised form 27 February 2020
Available online 14 May 2020

Keywords:
$t-SNE$
Correlation matrix
Cryptocurrency
Multidimensional scaling analysis

ABSTRACT

In this study, dimensionality reduction algorithms are applied to examine the correlation dynamics in the cryptocurrency market in which the time-dependent correlation matrix sequence is transformed into a distance matrix. The results based on multidimensional scaling (*MDS*) analysis show that information related to dynamic structural changes in the correlation coefficient matrix can be reconstructed well in low-dimensional Euclidean space. In particular, we found that the t-distributed stochastic neighbor embedding ($t-SNE$) algorithm can effectively exhibit the correlation dynamics in a two-dimensional (2D) space, thus providing a visualization method for analyzing dynamics in the market. Finally, we extract the clusters generated by the $t-SNE$ algorithm using the k nearest neighbor (*kNN*) network, and study the difference in the yield distribution of the periods corresponding to different clusters. Based on a comparison with the CCI 30 index, we determined that the components in the *kNN* network correspond well to different states in the market. The results show that dramatic changes in the correlation matrix suggest that significant changes may occur in the distribution of yields, such as a decrease in the average return.

1. Introduction

In recent years, the cryptocurrency market has experienced rapid growth, and new cryptocurrencies are constantly being added to the market. Currently, one of the hot topics of researchers is the quantitative characteristics in the cryptocurrency market. Many studies have conducted empirical analyses of cryptocurrencies, such as studying typical facts and discussing the characteristics of fractals [1–8].

The existence of multiple currencies in the cryptocurrency market enables discussion of the correlation between different cryptocurrencies. Thus, exploring correlation dynamics is a major issue. Similar topics for the stock market have been widely discussed in recent years. Currently, random matrix theory (*RMT*) and network methods have been effectively applied to the study of the dynamics of correlation matrices, where *RMT* focuses on the spectrum of correlation matrices, whereas network-based methods focus on extracting the main structures in matrices [9–12]. In addition to correlation matrices, researchers have also analyzed financial markets based on partial correlation matrices. For example, Kenett and Tumminello et al. proposed network analysis based on partial correlation [13]. Kenett and Huang et al. constructed a quantitative index based on partial correlation to characterize the influences of economic sectors [14]. One research interest is to analyze the characteristics of eigenvalues and network structures, particularly the dynamics of correlation matrices during financial crises [15–18].

E-mail addresses: niechunxiao2009@163.com, niechunxiao@zjgsu.edu.cn.

https://doi.org/10.1016/j.physa.2020.124702
0378-4371/

In addition to *RMT* and network-based methods, a method of directly studying the dynamics of matrices in the market is to assign a metric to the correlation matrices sequence and analyze the patterns in the distance matrix. For example, Münnix et al. applied the sliding window method to generate a matrix sequence and calculated the distance matrix of the sequence [19]. The resulting distance matrix showed dramatic changes in the matrix sequence during the 2008 crisis [19]. Another related work identifies market states using a distance matrix between correlation matrices, where cluster analysis is used to extract levels of similarity between different market states [20]. In addition, Jurczyk et al. defined the similarity between the correlation coefficient matrices based on the 1-norm and used it to analyze the critical transitions of the market [21]. In summary, these studies provide a means to extract global information in the market directly from the correlation matrix sequence, such as when measuring the market state.

Currently, several methods can be used to analyze the correlation of cryptocurrencies. For example, researchers have found that the correlation matrix in the cryptocurrency market exhibits a non-trivial hierarchical structure, and its spectrum is different from that predicted by *RMT* [22]. In addition, previous research has shown that extreme correlation is related to market trends but not to market fluctuations [23]. An analysis based on the detrended cross-correlation (*DCCA*) shows that the cryptocurrency market is different from the stock market in terms of effectiveness [24]. In addition, analysis based on *DCCA* also shows that contagious effects exist within the cryptocurrency market [25]. Finally, some studies have applied dynamic models to identify significant positive correlations in the cryptocurrency market [26,27]. In short, correlation analytical tools provide multiple perspectives. However, we found that the dynamics based on correlation matrices in the cryptocurrency market have not been extensively studied. In this article, we discuss the dynamics of cryptocurrencies from the perspective of dimensionality reduction analysis.

In our research, four key methods were combined for data analysis. First, we applied existing methods to characterize the differences in correlation structures, which were characterized by matrix distances [19–21]. Second, we applied multidimensional scaling (*MDS*) analysis to specify a vector for each correlation matrix and to explore the hidden geometry in a set of correlation matrices [28]. In previous studies, multidimensional scaling analysis was proven useful in extracting structures in financial dependence matrices [29]. In addition to *MDS*, principal component analysis can also be applied to dimensionality reduction analysis of market data, including visualizing the stock market in three-dimensional Euclidean space based on principal components [30]. Third, based on multidimensional scaling analysis, we used the t-distributed stochastic neighbor embedding ($t - SNE$) algorithm to show changes in the correlation structure in a 2D space [31]. Fourth, to analyze the structural differences in greater detail, we applied the *kNN* network to extract correlation change information from a low-dimensional space [32,33]. Based on the aforementioned steps, this study provides a means of visualizing the evolution of the market, thus enabling us to explore changes in the correlation structure in a low-dimensional space.

2. Data and method

2.1. Data

2.1.1. The details of the data

The data used in this study were extracted from the Internet https://coinmarketcap.com/. For a currency, multiple trading markets may exist. Therefore, the website provides data that has been pre-processed based on transaction data for multiple markets. Because the price data studied here were adjusted based on data from multiple trading markets, the price series globally depicted the price level of a cryptocurrency. Please see the relevant website for more details https://coinmarketcap.com/methodology/.

Currently, the cryptocurrency market includes many new currencies, which prevents long-term data from being extracted. Therefore, we selected only a few of the top 100 currencies based on market capitalization, where the time interval of the data were 2018/1/1–2019/7/1. thus providing us with a total of 546 trading days. After we removed some of the missing data currency, 63 cryptocurrencies were extracted, as shown in Table 1.

To describe the cryptocurrency market globally, we used the data of the CCI30 index, which is an index that includes 30 cryptocurrencies with top market capitalization. Detailed information on this index can be found on the website https://cci30.com/.

2.1.2. Correlation coefficient matrix

For convenience, we specify a label for each selected cryptocurrency, that is, a total of n cryptocurrencies $\{i, i = 1, 2, \ldots, n\}$. From the raw data, the cryptocurrency i we extracted is the price series $P_i = \{P_i(t)\}$, which needs to be converted into a logarithmic return series $R_i = \{R_i(t)\}$, where $R_i(t) = log(\frac{P_i(t+1)}{P_i(t)})$. Here we discuss the sequence of correlation coefficient matrices based on the returns of cryptocurrencies.

For n cryptocurrencies, the element $C(i, j)$ (Eq. (1)) in the matrix $C = [C(i, j)]$ describes the correlation between cryptocurrencies i and j. The symbol $[t_1, t_2]$ indicates that the start time and the end time of the time series are t_1 and t_2, respectively, where $t_2 - t_1 + 1 = L$ means that the length of the calculation window is L. The symbol $\overline{R_i}$ represents the average value of the time series. Thus, a matrix sequence $\{C_l\}$ is generated as we move the calculation window from the

Table 1
The list of cryptocurrencies.

Currency	Currency	Currency	Currency	Currency	Currency
Bitcoin	Ethereum	XRP	Litecoin	Bitcoin Cash	EOS
BNB	Tether	Cardano	TRON	Stellar	Monero
Dash	Chainlink	NEO	IOTA	ETC	NEM
Zcash	Maker	Tezos	Qtum	Bitcoin Gold	BAT
Dogecoin	OmiseGO	Decred	Lisk	BCD	HyperCash
Waves	0x	Augur	Nano	BitShares	MonaCoin
Bytecoin	Komodo	Bytom	ICON	KCS	DigiByte
Siacoin	ETP	Verge	Aeternity	GXChain	Steem
Ardor	Dent	aelf	MAID	Enjin Coin	Crypto.com
Status	Zcoin	Golem	Stratis	Dai	WAX
Nebulas	SAN	NULS	–	–	–

BNB: Binance Coin. BAT: Basic Attention Token. ETP: Metaverse ETP. SAN: Santiment Network Token. MAID: MaidSafeCoin. ETC: Ethereum Classic. BCD: Bitcoin Diamond. KCS: KuCoin Shares.

endpoint and calculate the correlation coefficient matrices. Here, we ignore the time interval and mark the correlation coefficient matrix only with the label l. In this paper, we mainly study the hidden patterns in this matrix sequence.

$$C(i,j)_{[t_1,t_2]} = \frac{\sum_t (R_i(t) - \overline{R_i})(R_j(t) - \overline{R_j})}{\sqrt{\sum_t (R_i(t) - \overline{R_i})^2}\sqrt{\sum_t (R_j(t) - \overline{R_j})^2}} \tag{1}$$

2.2. Method

2.2.1. Frobenius distance

Here we apply an existing method to characterize the dynamics of the market using a distance matrix based on the correlation matrix. Previous studies have applied different norms, such as using 1-norm and 2-norm to define the distance between two correlation matrices [19–21]. Here, we use the Fröbenius norm, which is equivalent to 2-norm in this paper [34]. Based on the Fröbenius norm, the distance between the matrices $C_{l_1} = [C_{l_1}(i,j)]$ and $C_{l_2} = [C_{l_2}(i,j)]$ is defined as Eq. (2).

$$D_F(l_1, l_2) = [\sum_{i=1}^{n}\sum_{j=1}^{n}(C_{l_1}(i,j) - C_{l_2}(i,j))^2]^{1/2} \tag{2}$$

The matrix sequence $\{C_l\}$ corresponds to a Fröbenius distance matrix D_F, that is, the level of the difference between the matrices is quantized, so that the evolution information is included in the matrix D_F.

2.2.2. Dimensionality reduction algorithm

Since the correlation matrix includes n^2 elements, in our study, the Frobenius distance can be considered as the distance between two high dimensional vectors. In order to study the information included in the matrix D_F in detail, we apply dimensionality reduction algorithms to analyze whether the changes in the correlation structure can be displayed in a low-dimensional Euclidean space. Here, we use the *MDS* algorithm and the $t-SNE$ algorithm for dimensionality reduction analysis, where the former algorithm is used to specify one vector for each correlation matrix $C_l \in \{C_l\}$, and the latter is used to show changes in the correlation structure in two-dimensional Euclidean space. We only briefly describe the two algorithms as follows.

Since the Fröbenius distance of the matrix can here essentially be regarded as the Euclidean distance between the vectors, the *MDS* algorithm can construct a vector set $\{v_l^E\}$ in Euclidean space based on D_F such that the vector v_l^E corresponds to the matrix C_l. We need to calculate several related matrices when solving the classical solution of *MDS*. Matrix $A = [A(i,j)]$ is constructed from D_F, where $A(i,j) = -\frac{1}{2}D_F^2$. Further, matrix B is generated by $B = HAH$, where $H = I_n - \frac{1}{n}1_n 1_n^t$. Here I_n is an identity matrix, and 1_n is a column vector where each element is 1. Then, we construct a solution to the *MDS* algorithm as follows [28].

- We calculate the set of eigenvalues $\{\lambda_i, i = 1, 2\cdots, n\}$ of B ($\lambda_1 \geq \lambda_2 \geq \cdots \lambda_n$). The eigenvector $\overline{x_i}$ corresponds to the eigenvalue λ_i, and $\overline{x_i} \perp \overline{x_j}$ ($i \neq j$).
- Finally, we construct the matrix $\overline{X} = [\overline{x_1}, \overline{x_2}, \ldots, \overline{x_s}]$, where the ith row is the coordinate vector corresponding to point i. Here s is a positive integer we specify, which is less than n. Correspondingly, the distance matrix based on the Euclidean metric of this vector set is denoted as D_E^s. If the first dim column is selected, an dim-dimensional vector representation of $\{C_l\}$ is generated.

We use stress-1 to quantitatively describe the difference between a distance matrix D_E^{dim} generated based on a reconstructed vector set and the original distance matrix, which is defined as Eq. (3) [28].

$$Stress-1=\sqrt{\frac{\sum(D_E^{dim}(i,j)-D_F(i,j))^2}{\sum D_F(i,j)^2}} \tag{3}$$

The $t-SNE$ algorithm is an excellent dimensionality reduction algorithm that has been widely used in various problems. The $t-SNE$ algorithm is an extension of the SNE (Stochastic Neighbor Embedding) algorithm that solves the crowding problem that exists in SNE. We briefly describe the algorithm as follows; the details of the algorithm can be found in the original paper [31].

It is assumed here that the original high-dimensional data is $v=\{v_i^E, i=1,2,\ldots,m\}$, and the data generated by dimensionality reduction is $v'=\{v_i', i=1,2,\ldots,m\}$. First, this algorithm needs to calculate the joint probability density p_{ij} (Eq. (4)) and q_{ij} (Eq. (5)). The variance σ_i here changes as the data point changes, depending on the perplexity. Here, we use the symbol $\overline{p}$ to represent perplexity. The concept perplexity is defined by Shannon entropy and the $t-SNE$ algorithm is robust to this parameter. In our study the perplexity was set to 30 ($\overline{p}=30$). Second, the function C is defined (Eq. (6)), and its gradient is as shown in Eq. (7). The output of the algorithm is generated by an optimization method based on the sample initial solution satisfying the Gaussian distribution and C.

$$p_{ij}=\frac{e^{-\|v_i^E-v_j^E\|^2/2\sigma_i^2}}{\sum_{k\neq l}e^{-\|v_k^E-v_l^E\|^2/2\sigma^2}} \tag{4}$$

$$q_{ij}=\frac{(1+\|v_i'-v_j'\|^2)^{-1}}{\sum_{k\neq l}(1+\|v_k'-v_l'\|^2)^{-1}} \tag{5}$$

$$C=\sum_i\sum_j p_{ij}log\frac{p_{ij}}{q_{ij}} \tag{6}$$

$$\frac{\partial C}{\partial v_i'}=4\sum_j(p_{ij}-q_{ij})(1+\|v_i'-v_j'\|^2)^{-1}(v_i'-v_j') \tag{7}$$

2.2.3. kNN network

In previous studies, k nearest neighbor graphs or networks are often used in machine learning problems, which mainly include two construction methods [32,33]. We assume that point i corresponds to node v_i in the kNN network $W_{\overline{p}}^k(V,E_{\overline{p}}^k)$ ($V=\{v_i\}$), $E_{\overline{p}}^k$ is the edge set, and the set of the nearest k neighbors of point i is $V_{neighbor}^i$. Here, the symbol $\overline{p}$ represents the value of perplexity. The two methods are as follows. In the first method, for any point i and j, if at least one point belongs to the set of nearest neighbors of another point, there is an undirected edge between points v_i and v_j in the network [32]. In the second method, if point j belongs to $V_{neighbor}^i$, a directed edge is generated, from v_i to v_j, such that a directed network is generated [33]. This paper uses the first method, which transforms the correlation matrix into an undirected network, where the degree of each node is greater than or equal to k.

In addition, we use Jaccard similarity to characterize the similarity between networks with different parameters to analyze the robustness of the method. The Jaccard similarity between $W_{\overline{p}_1}^k(V,E_{\overline{p}_1}^k)$ and $W_{\overline{p}_2}^k(V,E_{\overline{p}_2}^k)$ is defined on the sets $E_{\overline{p}_1}^k$ and $E_{\overline{p}_2}^k$ (Eq. (8)), where "Card" represents the cardinality [35–37].

$$J(W_{\overline{p}_1}^k(V,E_{\overline{p}_1}^k),W_{\overline{p}_2}^k(V,E_{\overline{p}_2}^k))=\frac{Card(E_{\overline{p}_1}^k\cap E_{\overline{p}_2}^k)}{Card(E_{\overline{p}_1}^k\cup E_{\overline{p}_2}^k)} \tag{8}$$

2.2.4. The calculation steps in this article

This paper mainly analyzes the evolution of the correlation matrix in the cryptocurrency market through the dimensionality reduction algorithms. The main steps are as follows.

- Step 1. We assume that each price series has been converted to a yield series. We set the length of the calculation window and then slide the window to calculate a series of correlation coefficient matrices.
- Step 2. We compute the Frobenius distance matrix D_F of the matrix sequence $\{C_l\}$ such that the features of the correlation matrix that change over time are characterized by the distance matrix.
- Step 3. Based on the distance matrix D_F, we apply the MDS algorithm to specify a vector v_l^E for each correlation coefficient matrix C_l, so that the matrix sequence is converted into a vector sequence.
- Step 4. The vector v_l^E generated in step 3 is a high dimensional vector. In order to visually show the structure in the D_F, we apply the $t-SNE$ algorithm to convert the high-dimensional vector set $\{v_l^E\}$ to a two-dimensional vector set $\{v_l'\}$.

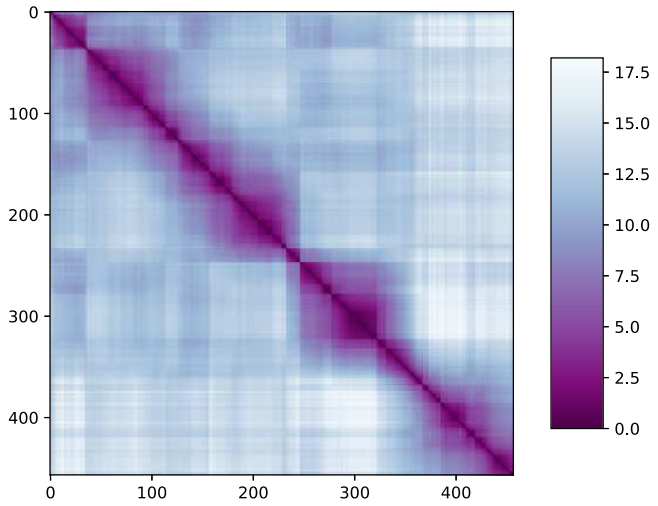


Fig. 1. The figure shows the F-distance matrix generated based on the correlation coefficient matrices in the cryptocurrency market. Intuitively, the matrix includes modules with smaller distance elements.

3. Results

3.1. Correlation dynamics based on Frobenius distance in the cryptocurrency market

In this section, we calculate the price-based correlation matrix sequence $\{C_l\}$ and calculate the Frobenius distance matrix D_F. Here, we set the calculation window to be 90 days, a total of 457 price-based correlation matrices are generated, and correspondingly, D_F is a 457×457 matrix.

Fig. 1 shows the dynamics of the correlation matrix from the perspective of the Frobenius distance matrix. We find similar phenomena in the cryptocurrency market as in the stock market, as shown in previous studies [19–21]. It can be seen that some modules are shown in Fig. 1. The distance between the correlation matrices in the module is smaller, and the distance between the correlation matrices in different modules is larger. This indicates that the correlation matrix changes steadily over some periods, but there are also periods of drastic changes that result in the generation of modules.

Intuitively, Fig. 1 shows the hidden patterns in the sequence $\{C_l\}$, suggesting that the correlation dynamics in the cryptocurrency market is not trivial, that is, it includes some critical time, corresponding to dramatic changes in the correlation matrix.

3.2. Multidimensional scaling analysis of the Frobenius distance matrix

Since the matrix D_F studied in this paper can be considered as being defined on a set of high-dimensional vectors, we can directly embed $\{C_l\}$ into the Euclidean space using the *MDS* algorithm. That is, we can get a geometric representation of $\{C_l\}$.

In our study, there are a total of 457 correlation matrices, so that each matrix is corresponding to a 456-dimensional vector by the *MDS* algorithm. However, our calculations show that a subspace much smaller than 456 dimensions contains most of the distance information. This can be inferred from the eigenvalues of B and stress.

We sorted the top 100 eigenvalues in descending order in Fig. 2. The largest eigenvalue is 11095.6, which is much larger than the second largest eigenvalue (6297.5). The first eight eigenvalues are 11095.6, 6297.5, 5874.6, 2247.5, 1695.2, 1516.4, 774.1, and 473.9, respectively. This implies that a subspace with a dimension much smaller than 456 dimensions can well include most information of the original distance matrix D_F.

Below we calculate the stress values (Eq. (3)) corresponding to different dimensions (*dim*). Here, the smaller the stress value indicates that the corresponding subspace can include more information in the original matrix. We only show the results of calculations with a dimension less than or equal to 30 (Fig. 3). The stress value is 0.0510 when $dim = 8$, and the stress value decreases to 0.0089 when *dim* increases to 30, indicating that most of the information in the D_F is stored in a subspace with a dimension much lower than n.

3.3. A low-dimensional representation of the correlation structure

We have found that the set of correlation matrices assigned to the Frobenius distance can be well embedded in a low-dimensional Euclidean subspace. A derivative topic is how to meaningfully display a sequence of correlation matrices

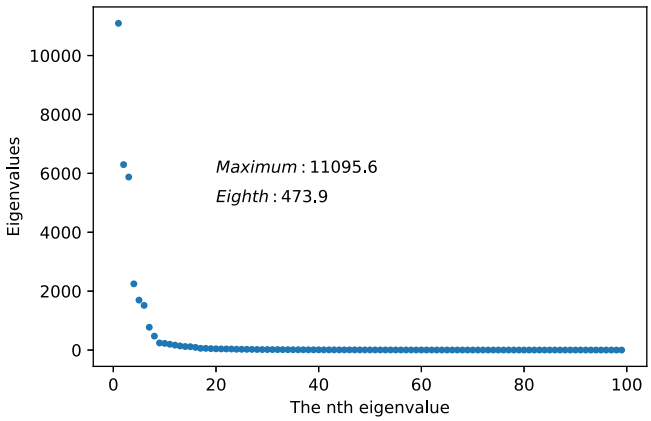


Fig. 2. The figure shows the top 100 eigenvalues generated in the *MDS* algorithm, where the values are sorted in descending order. It can be found that the eigenvalues decay rapidly.

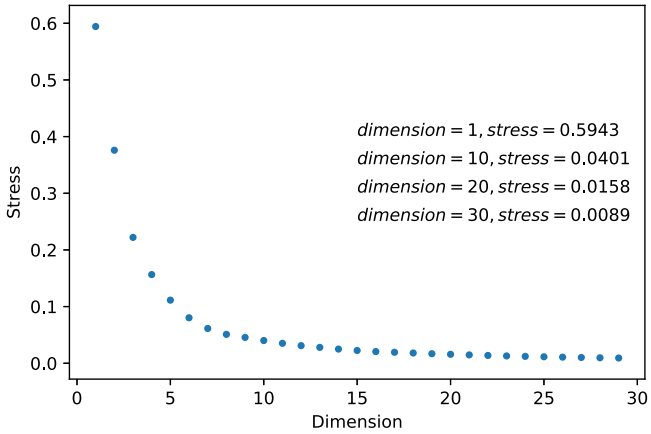


Fig. 3. The relationship between stress and dimension. When the dimension is less than 5, the stress decreases significantly as the dimension increases. When the dimension is greater than 20, the stress value is less than 0.0158, indicating that the distance element in D_F can be reconstructed well with a vector set in a subspace with a dimension much smaller than 457.

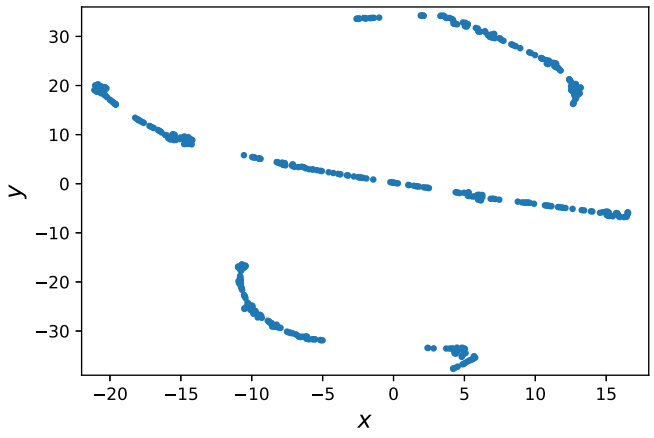
in a 2D space. In this section, we further demonstrate the set of correlation matrices in a 2D space using the $t-SNE$ algorithm. We show the correlation matrices calculated in the previous section in Fig. 4, each matrix corresponding to a two-dimensional vector. It can be found that some separate earthworm-like clusters are included.

In Fig. 4(b), we report a three-dimensional example with the same parameter $\overline{p}$, where each matrix corresponds to a three-dimensional vector. Intuitively, we can find that there are no clear earthworm-like clusters. In this article, we only analyze the two-dimensional projection of the cryptocurrency market.

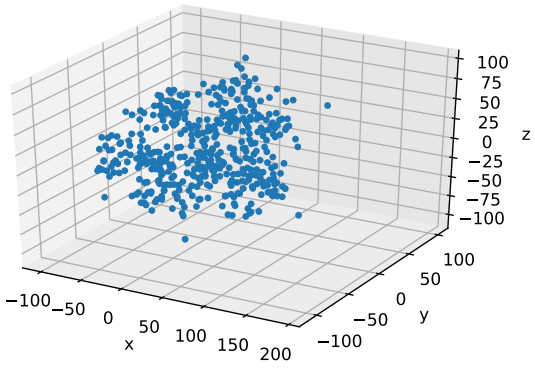
Since the correlation matrix typically changes very little during a period, the matrix with the smallest distance of a correlation matrix is usually its neighbor in the sense of time. Thus, in the sequence $\{C_l\}$, the distance between the matrix C_{l_0} and C_{l_0+1} or C_{l_0-1} is usually small. However, the correlation matrix changes significantly during some special periods, resulting in a significantly larger distance. For the above reasons, the set of correlation matrices exhibits earthworm-like clusters on a two-dimensional plane. Since the $t-SNE$ algorithm distorts the metrics in the original distance matrix, we use the *kNN* network to extract the local information of each correlation matrix, thereby extracting the earthworm-like clusters as shown in Fig. 4(a). Next, we compute different *kNN* networks based on the two-dimensional vector and then compute the connected components. Here we use the two-dimensional vector set generated by the $t-SNE$ algorithm to calculate the distance matrix, and then calculate the *kNN* networks.

Fig. 5 shows the number of connected components (N_c) corresponding to different k values. When $k=1$, there are a total of 160 connected components, but the N_c value drops rapidly as k increases. Here we choose the k value to make the N_c change stable, thus extracting the main structure. A suitable choice is $k=14$, which consists of five connected components.

We use different colors to mark the corresponding earthworm-like clusters of different components on the plane (Fig. 6). It can be clearly seen that different earthworm-like clusters are well distinguished. Here, different components



(a)



(b)

Fig. 4. Figure (a) shows the projection of a matrix sequence on a 2D space, which includes some separated earthworm-like clusters. Figure (b) shows an example of a three-dimensional projection without a clear cluster structure.

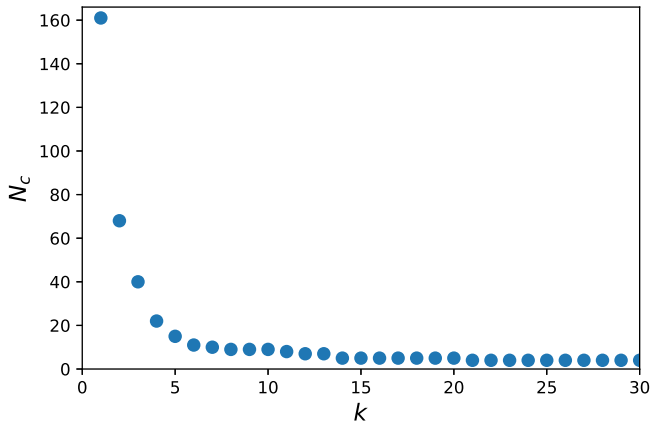


Fig. 5. The number of components included in different *kNN* networks, where N_c decreases as k increases, N_c is always equal to 5 when $14 \leq k \leq 20$.

are marked with symbols such as No.1. Table 2 shows the time interval corresponding to each component, where t_c^1 and t_c^2 are the endpoints of the time interval.

Table 2
The period corresponding to the component.

Comp	t_c^1	t_c^2
No.1	2018/4/1	2018/5/8
No.2	2018/5/9	2018/8/5
No.3	2018/8/6	2018/12/3
No.4	2018/12/4	2019/2/16
No.5	2019/2/17	2019/7/1

Comp: Component.

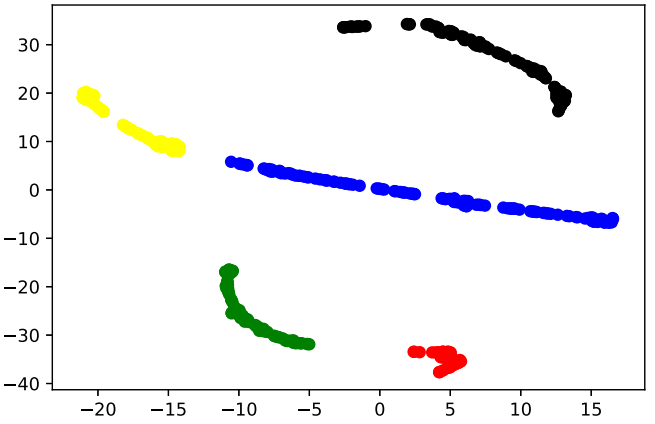


Fig. 6. The different earthworm-like clusters in Fig. 3 identified by the *kNN* network are labeled with different colors. No.1: red. No.2: green. No.3: black. No.4: yellow. No.5: blue.

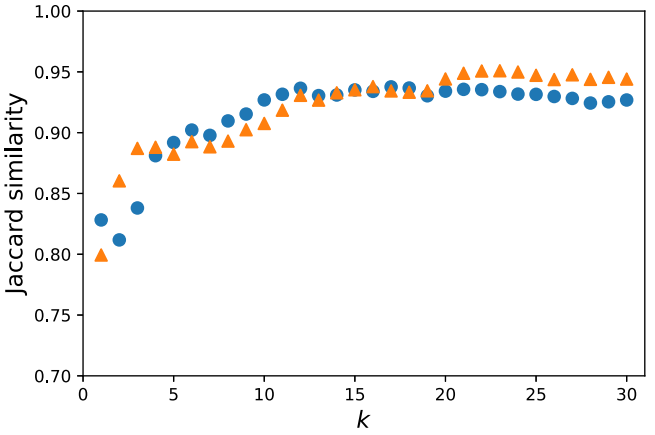


Fig. 7. The figure shows the sequences $\{J(W_{20}^k, W_{30}^k)\}$ (circle) and $\{J(W_{30}^k, W_{40}^k)\}$ (triangle), where $J(W_{20}^{14}, W_{30}^{14}) = 0.9310$ and $J(W_{30}^{14}, W_{40}^{14}) = 0.9328$.

3.4. Robustness analysis of parameters

The value of $\overline{p}$ needs to be set in the calculation. Below, the calculation shows that the $\overline{p}$ value has little effect on the calculation result. Since the analysis of the components is based on *kNN* network, we only need to analyze the differences between the networks corresponding to different $\overline{p}$ values. Next, we set the $\overline{p}$ values to 20 and 40, and calculate the coordinates. For each $\overline{p}$ value, we compute the *kNN* network sequence $\{W_{\overline{p}}^k(V, E_{\overline{p}}^k), k = 1, 2, \ldots, 30\}$. Fig. 7 shows the Jaccard similarity between different networks. When $k \geq 9$, all Jaccard similarity values are greater than 0.9, indicating that the parameter $\overline{p}$ has little effect on the structure of the *kNN* network.

3.5. Analysis of components in kNN network

We have found that there are modules in the distance matrix D_F that correspond to earthworm-like clusters on a two-dimensional plane. Further, earthworm-like clusters can be extracted by components in the *kNN* network. Thus these

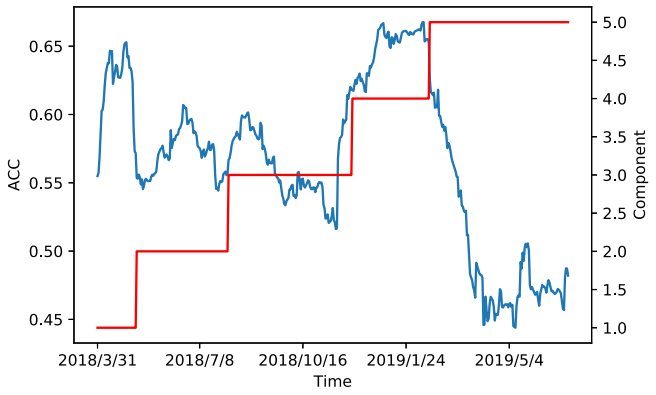


Fig. 8. The figure shows the relationship between the component label and the average correlation coefficient (*ACC*).

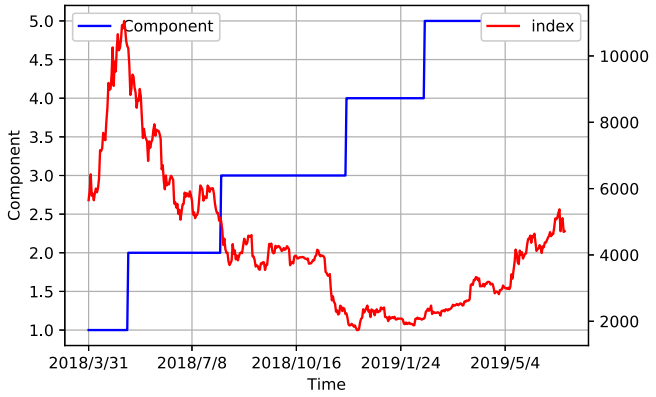


Fig. 9. The relationship between component labels (blue line) and the CCI 30 index (red line).

separate components are derived from significant changes in the correlation matrix. Next, we analyze the relationship between the components and the states of the market, and analyze the basic statistics of the returns corresponding to the components.

First, we calculate the average correlation coefficient of the matrix included in each component, that is, the average of the elements of the upper triangular matrix. Then, we compare the different components with the average Pearson correlation coefficient (*ACC*) as shown in Fig. 8. It can be found that there is a significant difference in the average correlation of different components, especially in the marginal regions between different components. For example, the *ACC* value in the marginal region between the component No.1 and the component No.2 drops significantly, and in addition, the marginal region between the component No.4 and the component No.5 has a similar phenomenon. Conversely, the region between component No.3 and component No.4 corresponds to a rapid increase in the *ACC* value, while there is a small increase in the *ACC* value in the marginal region of component No.2 and component No.3. In addition, it is important to note that the matrices in each component are continuous in time sense, which also indicates that, in general, the change in the correlation matrix is not so significant that new components cannot be generated.

Second, Fig. 9 shows the index and component labels. The CCI 30 index reached its maximum on 5/5/2018 (11055.7387), while t_c^2 of component No.1 corresponds to 5/18/2018. The CCI 300 index decreased sharply after May 5, 2018. Based on Fig. 9, it can be found that component No.2 includes this period of drastic change. Component No.3 includes the stable period of the index and a short-term shock, where t_c^2 is 2018/12/3. The minimum value of the CCI30 index corresponds to 2018/12/14 (1734.3486), which is close to t_c^2 of component No.3. Fig. 9 also shows that component No.4 includes a stable period, and component No.5 includes a growth period. In summary, we find that the cryptocurrency market switches between different phases. The components generated by the two-dimensional projection of the correlation structure are related to the transition of states.

Globally, Fig. 8 shows a relationship between components and changes in the average correlation coefficient. Third, we analyze the relationship between the statistics of the return series and the components in more detail. We consider the statistics of the return distribution of all cryptocurrencies. For the time interval corresponding to a component, we

Table 3
The statistics of the average of the returns.

Component	Min	Max	Med	ACC
No.1	−0.0000	0.0366	0.0154	0.0164
No.2	−0.0186	0.0094	−0.0102	−0.0094
No.3	−0.0138	0.0017	−0.0076	−0.0069
No.4	−0.0094	0.0080	−0.0027	−0.0020
No.5	−0.0028	0.0154	0.0047	0.0048

Min: minimum. Max: maximum. Med: median. Ave: average.

Table 4
The statistics of the standard deviation of the returns.

Component	Min	Max	Median	Ave
No.1	0.0023	0.1097	0.0705	0.0713
No.2	0.0043	0.1937	0.0576	0.0607
No.3	0.0058	0.1023	0.0612	0.0611
No.4	0.0050	0.1834	0.0547	0.0569
No.5	0.0057	0.1106	0.0507	0.0528

Min: minimum. Max: maximum. Med: median. Ave: average.

calculate the mean and standard deviation of the return series for each cryptocurrency in the interval. Then, we plot the frequency histogram of the mean, as shown in Fig. 10. Similarly, the frequency histogram of the standard deviation is shown in Fig. 11. It can be intuitively found that there is a difference between the histograms corresponding to different components, especially the large differences between the intervals to which the values belong. For example, the values in Fig. 10(a) are all greater than 0, but most of the values in Fig. 10(b) are less than zero. That is, in the time interval corresponding to component No.1, the average return of each cryptocurrency is greater than 0, but the average return of most currencies in component No.2 is less than zero.

In order to carefully compare the statistical characteristics of different components, we list the maximum and minimum values, as well as the median, in Tables 3 and 4. Based on these two tables, we found that, first of all, the maximum and median of different components vary greatly. This indicates that there is a large difference in the distribution in the time interval corresponding to different components. Second, although the standard deviation distribution also changes with time, the relative change in the median and mean of the standard deviations of the different components is small. In addition, significant changes in the maximum and minimum values indicate heterogeneity in the volatility of cryptocurrency from a microscopic perspective.

The calculations in this section show that the generation of a component implies a significant change in the average correlation, and there may be significant changes in the return distribution.

4. Discussion and conclusion

4.1. Discussion

In this paper, we construct a low-dimensional vector representation based on the Frobenius distance matrix of the correlation matrix to visually study the correlation structure. The dimensionality reduction analysis here depends directly on the definition of the distance matrix. Therefore, for other measures of correlation, correspondingly, the Frobenius distance matrix can also be directly defined and further used for dimensionality reduction analysis. In addition, different distances can be assigned between matrices, such as 1-norm-based metrics and so on.

4.2. Conclusion

We constructed the Frobenius distance matrix of the correlation coefficient matrix in the cryptocurrency market and transformed the correlation structure into a vector set in Euclidean space. In terms of abstract space, the correlation coefficient matrix can be considered a high-dimensional vector. However, analysis based on the *MDS* algorithm showed that the correlation coefficient matrix could be well embedded in a subspace, where the dimension of the subspace is much smaller than the number of matrix elements or the number of cryptocurrencies. In addition, we used the $t-SNE$ algorithm to reduce the dimension of the correlation coefficient matrix assigned to the metric and found that several earthworm-like clusters were generated in the 2D space. These earthworm-like clusters corresponded to some correlation matrices that are close together in time. We used *kNN* networks to analyze these clusters further. Based on the appropriate parameter k, corresponding to the connected components in the *kNN* network, the set of correlation matrices is divided into several subsets, which respectively correspond to different time intervals. In particular, based on a as comparison with the CCI30 index, we determined that a relationship existed between the components in the *kNN* network and the state of the market. We found that the regions between the components corresponded to significant changes in the average

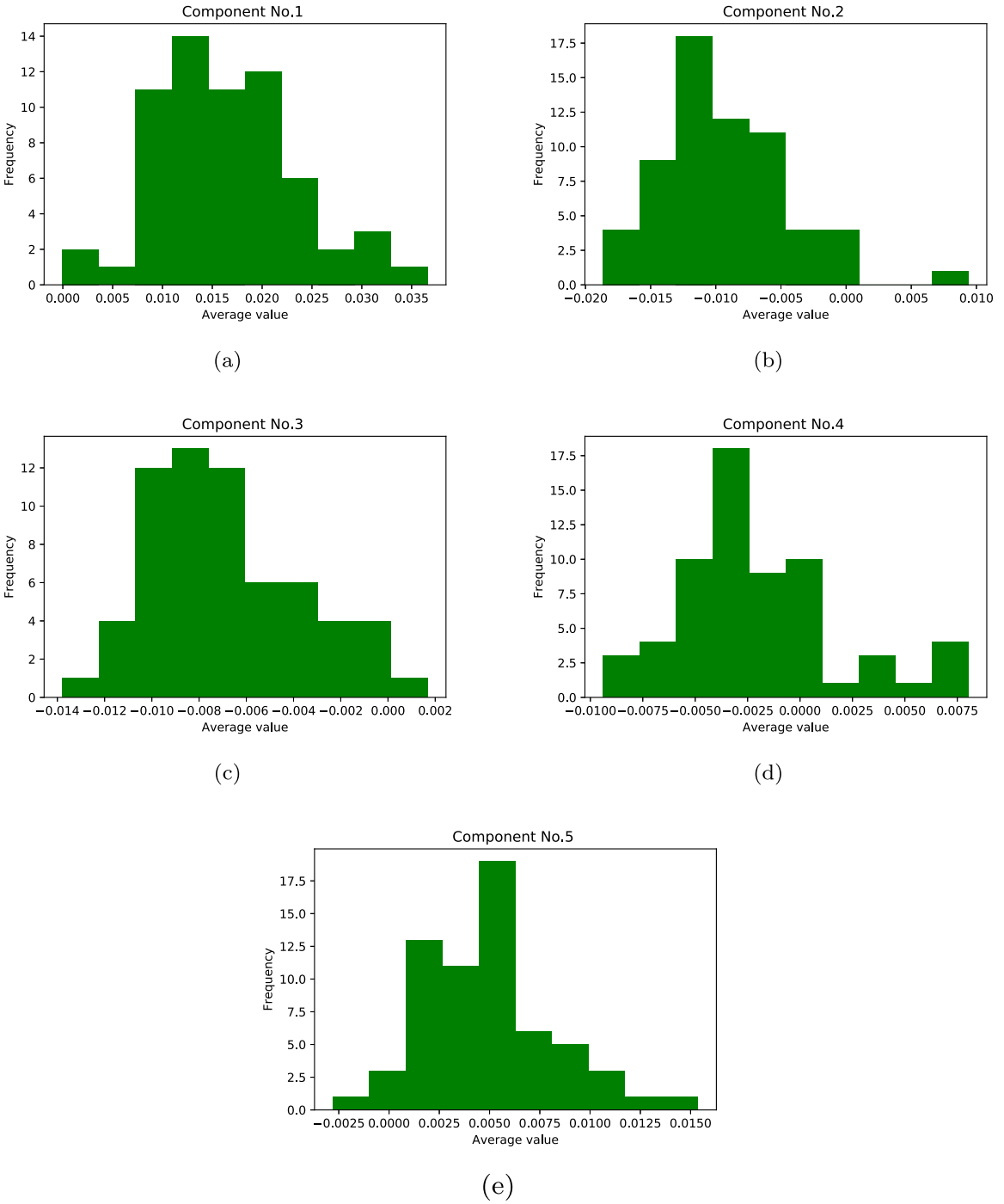


Fig. 10. The figure shows the distribution of average returns for different components.

correlation coefficient. Finally, we analyzed the distribution of yields in the interval corresponding to different components and found that the distribution of yields corresponding to different components showed significant differences.

Based on the existing algorithms, this study provided a new geometric perspective for analyzing correlation dynamics, thereby enabling market dynamics with high-dimensional characteristics to be examined in low-dimensional space. Our analysis showed that this method is effective and robust and enables analyzing the state of the market in a low-dimensional space.

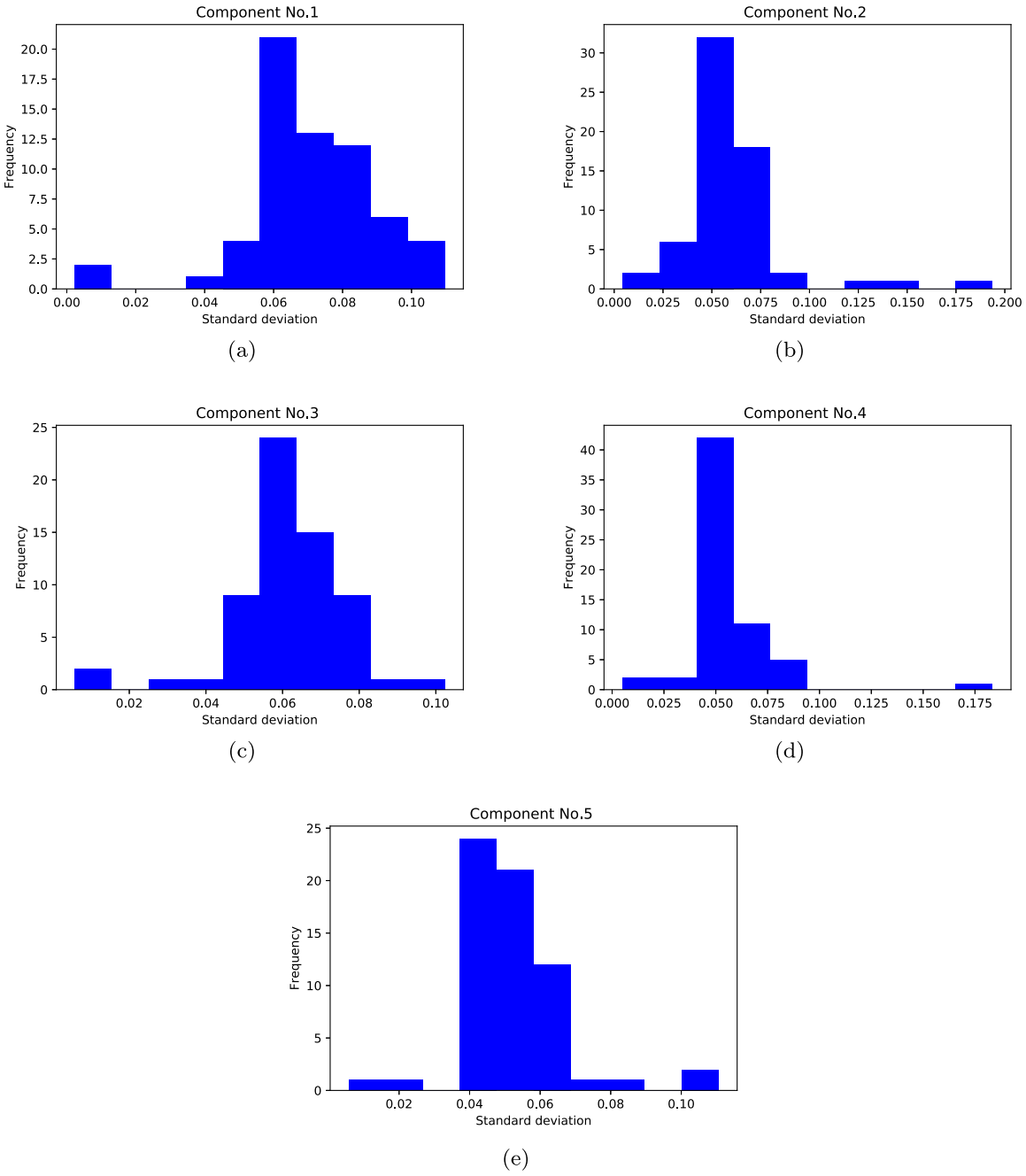


Fig. 11. The figure shows the distribution of the standard deviations for the different components.

CRediT authorship contribution statement

Chun-Xiao Nie: Conceptualization, Methodology, Writing - original draft, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A.F. Bariviera, M.J. Basgall, W. Hasperué, M. Naiouf, Some stylized facts of the Bitcoin market, Physica A 484 (2017) 82–90.
[2] W. Zhang, P. Wang, X. Li, D. Shen, Some stylized facts of the cryptocurrency market, Appl. Econ. 50 (55) (2018) 5950–5965.
[3] Y. Zhang, S. Chan, J. Chu, S. Nadarajah, Stylised facts for high frequency cryptocurrency data, Physica A 513 (2019) 598–612.
[4] J. Alvarez-Ramirez, E. Rodriguez, C. Ibarra-Valdez, Long-range correlations and asymmetry in the Bitcoin market, Physica A 492 (2018) 948–955.
[5] L. Kristoufek, On Bitcoin markets (in) efficiency and its evolution, Physica A 503 (2018) 257–262.
[6] A.C. da Silva Filho, N.D. Maganini, E.F. de Almeida, Multifractal analysis of Bitcoin market, Physica A 512 (2018) 954–967.
[7] D. Stosic, D. Stosic, T.B. Ludermir, T. Stosic, Multifractal behavior of price and volume changes in the cryptocurrency market, Physica A 520 (2019) 54–61.
[8] G.M. Caporale, L. Gil-Alana, A. Plastun, Persistence in the cryptocurrency market, Res. Int. Bus. Finance 46 (2018) 141–148.
[9] L. Laloux, P. Cizeau, J.-P. Bouchaud, M. Potters, Noise dressing of financial correlation matrices, Phys. Rev. Lett. 83 (7) (1999) 1467.
[10] V. Plerou, P. Gopikrishnan, B. Rosenow, L.A.N. Amaral, H.E. Stanley, Universal and nonuniversal properties of cross correlations in financial time series, Phys. Rev. Lett. 83 (7) (1999) 1471.
[11] R.N. Mantegna, Hierarchical structure in financial markets, Eur. Phys. J. B 11 (1) (1999) 193–197.
[12] M. Tumminello, T. Aste, T. Di Matteo, R.N. Mantegna, A tool for filtering information in complex systems, Proc. Natl. Acad. Sci. 102 (30) (2005) 10421–10426.
[13] D.Y. Kenett, M. Tumminello, A. Madi, G. Gur-Gershgoren, R.N. Mantegna, E. Ben-Jacob, Dominating clasp of the financial sector revealed by partial correlation analysis of the stock market, PLoS One 5 (12) (2010).
[14] D.Y. Kenett, X. Huang, I. Vodenska, S. Havlin, H.E. Stanley, Partial correlation analysis: Applications for financial markets, Quant. Finance 15 (4) (2015) 569–578.
[15] L.S. Junior, I.D.P. Franca, Correlation of financial markets in times of crisis, Physica A 391 (1–2) (2012) 187–208.
[16] T. Conlon, H.J. Ruskin, M. Crane, Cross-correlation dynamics in financial time series, Physica A 388 (5) (2009) 705–714.
[17] G. Zumbach, Empirical properties of large covariance matrices, Quant. Finance 11 (7) (2011) 1091–1102.
[18] A. Sensoy, S. Yuksel, M. Erturk, Analysis of cross-correlations between financial markets after the 2008 crisis, Physica A 392 (20) (2013) 5027–5045.
[19] M.C. Münnix, R. Schäfer, O. Grothe, Estimating correlation and covariance matrices by weighting of market similarity, Quant. Finance 14 (5) (2014) 931–939.
[20] M.C. Münnix, T. Shimada, R. Schäfer, F. Leyvraz, T.H. Seligman, T. Guhr, H.E. Stanley, Identifying states of a financial market, Sci. Rep. 2 (2012) 644.
[21] J. Jurczyk, T. Rehberg, A. Eckrot, I. Morgenstern, Measuring critical transitions in financial markets, Sci. Rep. 7 (1) (2017) 11564.
[22] D. Stosic, D. Stosic, T.B. Ludermir, T. Stosic, Collective behavior of cryptocurrency price changes, Physica A 507 (2018) 499–509.
[23] K. Gkillas, S. Bekiros, C. Siriopoulos, Extreme correlation in cryptocurrency markets, 2018, Available at SSRN 3180934.
[24] P. Ferreira, L. Kristoufek, E.J.d.A.L. Pereira, DCCA And DMCA correlations of cryptocurrency markets, Physica A (2019) 123803.
[25] P. Ferreira, É. Pereira, Contagion effect in cryptocurrency market, J. Risk Financ. Manag. 12 (3) (2019) 115.
[26] P. Katsiampa, S. Corbet, B. Lucey, High frequency volatility co-movements in cryptocurrency markets, J. Int. Financ. Mark. Inst. Money 62 (2019) 35–52.
[27] N.P. Canh, U. Wongchoti, S.D. Thanh, N.T. Thong, Systematic risk in cryptocurrency market: Evidence from DCC-MGARCH model, Finance Res. Lett. 29 (2019) 90–100.
[28] I. Borg, P. Groenen, Modern multidimensional scaling: Theory and applications, J. Educ. Meas. 40 (3) (2003) 277–280.
[29] L.S. Junior, A. Mullokandov, D.Y. Kenett, Dependency relations among international stock market indices, J. Risk Financ. Manag. 8 (2) (2015) 227–265.
[30] Y. Shapira, D.Y. Kenett, E. Ben-Jacob, The index cohesive effect on stock market correlations, Eur. Phys. J. B 72 (4) (2009) 657.
[31] L.v.d. Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (Nov) (2008) 2579–2605.
[32] M. Brito, E. Chavez, A. Quiroz, J. Yukich, Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection, Statist. Probab. Lett. 35 (1) (1997) 33–42.
[33] V. Hautamaki, I. Karkkainen, P. Franti, Outlier detection using k-nearest neighbour graph, in: Proceedings of the 17th International Conference on Pattern Recognition, 2004, Vol. 3, ICPR 2004, IEEE, 2004, pp. 430–433.
[34] S.J. Leon, Linear Algebra with Applications, eighth ed., Pearson Education, Inc, 2010, p. 235.
[35] M. Levandowsky, D. Winter, Distances between sets, Nature 234 (5) (1971) 34–35.
[36] C. Donnat, S. Holmes, Tracking network dynamics: a survey of distances and similarity metrics, 2018, arXiv preprint arXiv:1801.07351.
[37] A. Rawashdeh, A.L. Ralescu, Similarity measure for social networks-a brief survey, Maics (2015) 153–159.