entropy

MDPI

*Article*

# Dynamical Analysis of the Dow Jones Index Using Dimensionality Reduction and Visualization

**António M. Lopes [1,*,†] and Jóse A. Tenreiro Machado [2,†]**

[1] LAETA/INEGI, Faculty of Engineering, University of Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
[2] Department of Electrical Engineering, Institute of Engineering, Polytechnic of Porto, Rua Dr. António Bernardino de Almeida, 431, 4249-015 Porto, Portugal; jtm@isep.ipp.pt
* Correspondence: aml@fe.up.pt
† These authors contributed equally to this work.

**Abstract:** Time-series generated by complex systems (CS) are often characterized by phenomena such as chaoticity, fractality and memory effects, which pose difficulties in their analysis. The paper explores the dynamics of multidimensional data generated by a CS. The Dow Jones Industrial Average (DJIA) index is selected as a test-bed. The DJIA time-series is normalized and segmented into several time window vectors. These vectors are treated as objects that characterize the DJIA dynamical behavior. The objects are then compared by means of different distances to generate proper inputs to dimensionality reduction and information visualization algorithms. These computational techniques produce meaningful representations of the original dataset according to the (dis)similarities between the objects. The time is displayed as a parametric variable and the non-locality can be visualized by the corresponding evolution of points and the formation of clusters. The generated portraits reveal a complex nature, which is further analyzed in terms of the emerging patterns. The results show that the adoption of dimensionality reduction and visualization tools for processing complex data is a key modeling option with the current computational resources.

**Keywords:** dimensionality reduction; data visualization; clustering; time-series; complex systems



**Citation:** Lopes, A.M.; Tenreiro Machado, J.A. Dynamical Analysis of the Dow Jones Index Using Dimensionality Reduction and Visualization. *Entropy* **2021**, *23*, 600. https://doi.org/10.3390/e23050600

Academic Editor: Vasily E. Tarasov

Received: 7 April 2021
Accepted: 9 May 2021
Published: 13 May 2021



## 1. Introduction

Complex systems (CS) are composed of several autonomous entities, described by simple rules, that interact with each other and their environment. The CS give rise to a collective behavior that exhibits a much richer dynamical phenomena than the one presented by the individual elements. Often, CS exhibit evolution, adaptation, self-organization, emergence of new orders and structures, long-range correlations in the time–space domain, chaoticity, fractality, and memory effects [1–4]. The CS are not only pervasive in nature, but also in human-related activities, and include molecular dynamics, living organisms, ecosystems, celestial mechanics, financial markets, computational systems, transportation and social networks, and world and country economies, as well as many others [5–8].

Time-series analysis has been successfully adopted to study CS [9,10]. The CS outputs are measured over time and the data collected are interpreted as manifestations of the CS dynamics. Therefore, the study of the time-series allows for conclusions about the CS behavior to be reached [11,12]. Nonetheless, real-word time-series may be affected by noise, distortion and incompleteness, requiring advanced processing methods for the extraction of significant information from the data [13]. Information visualization plays a key role in time-series analysis, as it provides an insight into the data characteristics. Information visualization corresponds to the computer generation of dataset visual representations. Its main goal is to expose features hidden in the data, in order to understand the system that generated such data [14,15]. Dimensionality reduction [16] plays a key role in information visualization, since the numerical data are often multidimensional. Dimensionality

reduction-based schemes try to preserve, in lower-dimensional representations, the information present in the original datasets. They include linear methods, such as classic multidimensional scaling (MDS) [17], principal component [18], canonical correlation [19], linear discriminant [20] and factor analysis [21], as well as nonlinear approaches, such as non-classic MDS, or Sammon's projection [22], isomap [23], Laplacian eigenmap [24], diffusion map [25], t-distributed stochastic neighbor embedding (t-SNE) [26] and uniform manifold approximation and projection (UMAP) [27].

Financial time series have a complex nature and their dynamic characterization is challenging. The Dow Jones Industrial Average (DJIA) is an important financial index and is adopted in this paper as a dataset generated by a CS. The paper explores an alternative strategy to the classical time-domain analysis, by combining the concepts of distance and dimensionality reduction with computational visualization tools. The DJIA time-series of daily close values is normalized and segmented, yielding a number of objects that characterize the DJIA dynamics. These objects are vectors, whose time-length and partial time-overlap represent a compromise between time resolution and memory length. The objects are compared using various distances and their dissimilarities are used as the input to different dimensionality reduction and information visualization algorithms, namely hierarchical clustering (HC), MDS, t-SNE and UMAP. The aforementioned algorithms construct representations of the original dataset, where time is a parametric variable. The structure of the plots is further analyzed in terms of the emerging patterns. The formation of clusters and the evolution of the patterns over time maps a dynamical behavior with discontinuities for periods where the memory is somehow lost. Numerical experiments illustrate the feasibility and effectiveness of the method for the processing of complex data.

The paper organization is summarized as follows. Section 2 reviews mathematical fundamental concepts, namely the distances and the algorithms adopted in the study for processing and visualizing data. Section 3 introduces the DJIA dataset. Section 4 analyses the data and interprets the results in the light of the distances used. Section 5 assesses the effect of the time-length and overlap of the segmenting window. Section 6 presents the conclusions.

## 2. Mathematical Concepts and Tools

### *2.1. Distances*

Given two points $\mathbf{v}_i$ and $\mathbf{v}_j$ in a set $\mathcal{X}$, the function $d(\mathbf{v}_i, \mathbf{v}_j) : \mathcal{X} \times \mathcal{X} \to [0, +\infty]$ represents a distance between the points if, and only if, it satisfies the conditions: identity of indiscernibles, symmetry and triangle inequality [28].

In this paper, the distances $\{\text{Arccosine, Canberra, Dice, Divergence, Euclidean, Jaccard, Lorentzian, Manhattan, Sørenson, Generalized}\} = \{d_1, \ldots, d_{10}\}$ are considered. Therefore, given $\mathbf{v}_i = (v_{i1}, \cdots, v_{iP})$ and $\mathbf{v}_j = (v_{j1}, \cdots, v_{jP})$ in a $P$-dimensional space, $\mathcal{P}$, the 10 distances are given by [28]:

$$\text{Arccosine}: \quad d_1(\mathbf{v}_i, \mathbf{v}_j) = \arccos\left(\frac{\sum_{k=1}^{P} v_{ik} \cdot v_{jk}}{\sqrt{\sum_{k=1}^{P} v_{ik}^2}\sqrt{\sum_{k=1}^{P} v_{jk}^2}}\right), \tag{1}$$

$$\text{Canberra}: \quad d_2(\mathbf{v}_i, \mathbf{v}_j) = \sum_{k=1}^{P} \frac{|v_{ik} - v_{jk}|}{|v_{ik}| + |v_{jk}|}, \tag{2}$$

$$\text{Dice}: \quad d_3(\mathbf{v}_i, \mathbf{v}_j) = \frac{\sum_{k=1}^{P} (v_{ik} - v_{jk})^2}{\sum_{k=1}^{P} v_{ik}^2 + \sum_{k=1}^{P} v_{jk}^2}, \tag{3}$$

$$\text{Divergence}: \quad d_4(\mathbf{v}_i, \mathbf{v}_j) = 2 \sum_{k=1}^{P} \frac{(v_{ik} - v_{jk})^2}{(v_{ik} + v_{jk})^2}, \tag{4}$$

$$\text{Euclidean}: \quad d_5(\mathbf{v}_i, \mathbf{v}_j) = \sqrt{\sum_{k=1}^{P} (v_{ik} - v_{jk})^2}, \tag{5}$$

$$\text{Jaccard}: \quad d_6(\mathbf{v}_i, \mathbf{v}_j) = \frac{\sum_{k=1}^{P} (v_{ik} - v_{jk})^2}{\sum_{k=1}^{P} v_{ik}^2 + \sum_{k=1}^{P} v_{jk}^2 - \sum_{k=1}^{P} v_{ik} v_{jk}}, \tag{6}$$

$$\text{Lorentzian}: \quad d_7(\mathbf{v}_i, \mathbf{v}_j) = \sum_{k=1}^{P} \ln\Big(1 + |v_{ik} - v_{jk}|\Big), \tag{7}$$

$$\text{Manhattan}: \quad d_8(\mathbf{v}_i, \mathbf{v}_j) = \sum_{k=1}^{P} |v_{ik} - v_{jk}|, \tag{8}$$

$$\text{Sørenson}: \quad d_9(\mathbf{v}_i, \mathbf{v}_j) = \frac{\sum_{k=1}^{P} |v_{ik} - v_{jk}|}{\sum_{k=1}^{P} |v_{ik}| + |v_{jk}|}, \tag{9}$$

$$\text{Generalized}: \quad d_{10}(\mathbf{v}_i, \mathbf{v}_j) = \sum_{r=1}^{9} \lambda_r \frac{d_r(\mathbf{v}_i, \mathbf{v}_j)}{\max[d_r(\mathbf{v}_i, \mathbf{v}_j)]}, \tag{10}$$

where $\lambda_r \in \mathbb{R}$, $\sum_{i=1}^{9} \lambda_r = 1$.

The distances (1)–(9) have advantages and disadvantages, meaning that they unravel specific features embedded in the data, while neglecting others. Therefore, the 'generalized' distance $d_{10}$ may eventually capture a multi-perspective information by combining (1)–(9) in a complementary form.

Other techniques [29] and distances [28] can also be adopted to compare the data. However, a more extensive overview and utilization of a larger number of alternatives is out of the scope of the paper.

### *2.2. Dimensionality Reduction and Visualization*

In the next subsections, the dimensionality reduction and visualization techniques that are adopted for data processing are presented, namely the HC, MDS, t-SNE and UMAP.

Given a set of $N$ objects, $\mathbf{v}_i$, $i = 1, \ldots, N$, in space $\mathcal{P}$, all methods require the definition of a distance $d(\mathbf{v}_i, \mathbf{v}_j)$, $i, j = 1, \ldots, N$, between the objects $i$ and $j$.

#### 2.2.1. The Hierarchical Clustering

The HC groups similar objects and represents them in a 2-dim locus. The algorithm involves two main steps [30]. In the first, the HC constructs a matrix of distances, $\mathbf{D} = [d(\mathbf{v}_i, \mathbf{v}_j)]$, of dimension $N \times N$, where $d(\mathbf{v}_i, \mathbf{v}_j) = d(\mathbf{v}_j, \mathbf{v}_i)$. In the second step, the algorithm arranges the objects in a hierarchical structure and depicts them in a graphical portrait, namely, a hierarchical tree or a dendrogram. This is achieved by means of two alternative techniques: the divisive and the agglomerative procedures. In the divisive scheme, all objects start in one single cluster and end in separate clusters. This is done by iteratively removing the 'outsiders' from the least cohesive cluster. In the agglomerative scheme, each object starts in its own cluster and all end in one single cluster. This is accomplished by successive iterations that join the most similar clusters. The HC requires the specification of a linkage criterion for measuring the dissimilarity between clusters. Often, the average-linkage, $d_{av}(R, S)$, is adopted [31], where $R$ and $S$ represent two clusters. Therefore, denoting $d(\mathbf{v}_R, \mathbf{v}_S)$ the distance between a pair of objects $\mathbf{v}_R \in R$ and $\mathbf{v}_S \in S$, in the clusters $R$ and $S$, respectively, we have:

$$d_{av}(R, S) = \frac{1}{\|R\| \|S\|} \sum_{\mathbf{v}_R \in R, \mathbf{v}_S \in S} d(\mathbf{v}_R, \mathbf{v}_S). \tag{11}$$

The reliability of the clustering can be assessed by the cophenetic coefficient *cc* [32]

$$cc = \frac{\sum_{i<j}\left[d(\mathbf{v}_i,\mathbf{v}_j) - \mathrm{av}\left(d(\mathbf{v}_i,\mathbf{v}_j)\right)\right]\left[\hat{d}(\mathbf{t}_i,\mathbf{t}_j) - \mathrm{av}\left(\hat{d}(\mathbf{t}_i,\mathbf{t}_j)\right)\right]}{\sqrt{\left[\sum_{i<j}\left[d(\mathbf{v}_i,\mathbf{v}_j) - \mathrm{av}\left(d(\mathbf{v}_i,\mathbf{v}_j)\right)\right]^2\right]\left[\sum_{i<j}\left[\hat{d}(\mathbf{t}_i,\mathbf{t}_j) - \mathrm{av}\left(\hat{d}(\mathbf{t}_i,\mathbf{t}_j)\right)\right]^2\right]}}, \tag{12}$$

where $\{\mathbf{v}_i,\mathbf{v}_j\}$ and $\{\mathbf{t}_i,\mathbf{t}_j\}$ stand for the original objects and their HC representations, respectively, $\mathrm{av}(\cdot)$ denotes the average of the input argument, and $\hat{d}(\mathbf{t}_i,\mathbf{t}_j)$ represents the cophenetic distance between $\mathbf{t}_i$ and $\mathbf{t}_j$. We always obtain $0 \leq cc \leq 1$, with the limits corresponding to bad and good clustering, respectively. Additionally, the original and the cophenetic distances can be represented in a scatter plot denoted by Shepard diagram. A good clustering corresponds to points located close to a 45° line.

#### 2.2.2. The Multidimensional Scaling

The MDS includes a class of methods that represent high-dimensional data in a lower dimensional space, while preserving the inter-point distances as much as possible. The matrix $\mathbf{D} = [d(\mathbf{v}_i,\mathbf{v}_j)]$ feeds the MDS dimensionality reduction and visualization algorithm. The MDS tries to find the positions of $Q$-dimensional objects, $\mathbf{t}_i$, with $i = 1,\ldots,N$, represented by points in space $\mathcal{Q}$, so that $Q \leq P$, while producing a matrix $\mathbf{T} = [\hat{d}(\mathbf{t}_i,\mathbf{t}_j)]$ that approximates $\mathbf{D}$. This is accomplished by means of an optimization procedure that tries to minimize a fitness function. Usually, the stress cost function, $\mathcal{S}$, is adopted

$$\mathcal{S} = \left[\sum_{i<j}\left[d(\mathbf{v}_i,\mathbf{v}_j) - \hat{d}(\mathbf{t}_i,\mathbf{t}_j)\right]^2\right]^{\frac{1}{2}}. \tag{13}$$

The Sammon criterion is an alternative, yielding

$$\mathcal{S} = \left[\frac{\sum_{i<j}\left[d(\mathbf{v}_i,\mathbf{v}_j) - \hat{d}(\mathbf{t}_i,\mathbf{t}_j)\right]^2}{\sum_{i<j}\left[d(\mathbf{v}_i,\mathbf{v}_j)\right]^2}\right]^{\frac{1}{2}}. \tag{14}$$

The 'quality' of the MDS is assessed by comparing the original and the reproduced information. This can be accomplished by means of the Shepard diagram, which depicts $d(\mathbf{v}_i,\mathbf{v}_j)$ versus $\hat{d}(\mathbf{t}_i,\mathbf{t}_j)$. Additionally, since the stress $S$ decreases monotonically with the dimension $Q$, the user can establish a compromise between the two variables. Often, the practical choice is $Q = 2$ or $Q = 3$, since those values yield a direct graphical representation in the embedding space. Nevertheless, if the MDS locus is unclear, then the user must adopt another measure $d(\mathbf{v}_i,\mathbf{v}_j)$ until a suitable representation is obtained.

#### 2.2.3. The t-Distributed Stochastic Neighbor Embedding

The t-SNE [26] is a technique for visualizing high-dimensional datasets, with applications including computer security [33], music analysis [34], bioinformatics [35] and other areas [36,37].

The algorithm comprises two main stages. In the first, for each pair of objects $(\mathbf{v}_i,\mathbf{v}_j)$, $i,j = 1,\ldots,N$, the t-SNE constructs a joint probability distribution $p_{ij}$ measuring the similarity between $\mathbf{v}_i$ and $\mathbf{v}_j$, in such a way that similar (dissimilar) objects are assigned a higher (lower) probability

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}, \tag{15}$$

$$p_{j|i} = \begin{cases} \dfrac{\exp[-d(\mathbf{v}_i, \mathbf{v}_j)^2/(2\sigma_i^2)]}{\sum_{k \neq i} \exp[-d(\mathbf{v}_i, \mathbf{v}_k)^2/(2\sigma_i^2)]}, & j \neq i \\ 0, & j = i \end{cases}, \tag{16}$$

where $p_{ij} = p_{ji}$, $p_{ii} = 0$, $\sum_{i,j} p_{ij} = 1$ and $\sum_j p_{j|i} = 1, \forall i, j$. The parameter $\sigma_i^2$ represents the variance of the Gaussian kernel that is centered on $\mathbf{v}_i$. A particular value of $\sigma_i$ induces a probability distribution $P_i$, over all of the other datapoints. In other words, $P_i$ represents the conditional probability distribution over all other datapoints given the datapoint $\mathbf{v}_i$. The t-SNE searches for the value of $\sigma_i$ that generates a distribution $P_i$ with the value of perplexity specified by

$$\text{perplexity}(P_i) = 2^{H(P_i)}, \tag{17}$$

where $H(P_i)$ is the Shannon entropy of $P_i$

$$H(P_i) = \sum_j p_{j|i} \log_2(p_{j|i}). \tag{18}$$

As a result, the variation in the Gaussian kernel is adapted to the density of the data, meaning that smaller (larger) values of $\sigma_i$ are used in denser (sparser) parts of the data space. The perplexity can be interpreted as a smooth measure of the effective number of $\mathbf{v}_i$ neighbors. Typical values of perplexity$(P_i)$ are in the interval $[5, 50]$.

In the second stage, the t-SNE calculates the similarities between pairs of points in $\mathcal{Q}$

$$q_{ij} = \frac{q_{j|i} + q_{i|j}}{2N}, \tag{19}$$

$$q_{ij} = \begin{cases} \dfrac{(1 + ||\mathbf{t}_i - \mathbf{t}_j||^2)^{-1}}{\sum_{k \neq l} (1 + ||\mathbf{t}_k - \mathbf{t}_l||^2)^{-1}}, & j \neq i \\ 0, & j = i \end{cases}, \tag{20}$$

where the symbol $||\cdot||$ denotes the 2-norm of the argument, $q_{ij} = q_{ji}$, $q_{ii} = 0$, $\sum_{i,j} q_{ij} = 1$ and $\sum_j q_{j|i} = 1, \forall i, j$.

The t-SNE performs an optimization, while attempting to minimize the Kullback–Leibler ($KL$) divergence between the Gaussian distribution of the points in space $\mathcal{P}$ and the Student $t$-distribution of the points in the embedding space $\mathcal{Q}$:

$$KL = \sum_{i \neq j} p_{ij} \ln \frac{p_{ij}}{q_{ij}}. \tag{21}$$

The minimization scheme starts with a given initial set of points in $\mathcal{Q}$, and the algorithm uses the gradient descent

$$\frac{\partial KL}{\partial \mathbf{t}_i} = 4 \sum_j (p_{ij} - q_{ij})(\mathbf{t}_i - \mathbf{t}_j)(1 + ||\mathbf{t}_i - \mathbf{t}_j||^2)^{-1}. \tag{22}$$

The $KL$ divergence between the modeled input and output distributions is often used as a measure of the quality of the results.

#### 2.2.4. The Uniform Manifold Approximation and Projection

The UMAP is a recent technique [27] for clustering and visualizing high-dimensional datasets, which seeks to accurately represent both the local and global structures embedded in the data [38,39].

Given a distance, $d(\mathbf{v}_i, \mathbf{v}_j)$, between pairs of objects $\mathbf{v}_i$ and $\mathbf{v}_j$, $i, j = 1, \ldots, N$, and the number of neighbors to consider, $k$, the UMAP starts by computing the $k$-nearest neighbors of $\mathbf{v}_i$, $\mathcal{N}_i$, with respect to $d(\mathbf{v}_i, \mathbf{v}_j)$. Then, the algorithm calculates the parameters $\rho_i$ and $\sigma_i$,

for each datapoint $\mathbf{v}_i$. The parameter $\rho_i$ represents a nonzero distance from $\mathbf{v}_i$ to its nearest neighbor and is given by

$$\rho_i = \min_{j \in \mathcal{N}_i} \{ d(\mathbf{v}_i, \mathbf{v}_j) | d(\mathbf{v}_i, \mathbf{v}_j) > 0 \}. \tag{23}$$

The parameter $\rho_i$ is important to ensure the local connectivity of the manifold. This means that it yields a locally adaptive exponential kernel for each point.

The constant $\sigma_i$ must satisfy the condition

$$\log_2 k = \sum_{j \in \mathcal{N}_i} \exp\left[ \frac{-\max(0, d(\mathbf{v}_i, \mathbf{v}_j) - \rho_i)}{\sigma_i} \right], \tag{24}$$

determined using binary search.

The algorithm constructs a joint probability distribution $p_{ij}$ measuring the similarity between $\mathbf{v}_i$ and $\mathbf{v}_j$, in such a way that similar (dissimilar) objects are assigned a higher (lower) probability

$$p_{ij} = p_{j|i} + p_{i|j} - p_{j|i} p_{i|j}, \tag{25}$$

$$p_{j|i} = \begin{cases} \exp\left[ \frac{-\max(0, d(\mathbf{v}_i, \mathbf{v}_j) - \rho_i)}{\sigma_i} \right], & j \neq i \\ 0, & j = i \end{cases}, \tag{26}$$

where $p_{ij} = p_{ji}$, $p_{ii} = 0$, $\sum_{i,j} p_{ij} = 1$ and $\sum_j p_{j|i} = 1, \forall i, j$.

In the second stage, the UMAP computes the similarities between each pair of points in the space $\mathcal{Q}$

$$q_{ij} = q_{j|i} + q_{i|j} - q_{j|i} q_{i|j}, \tag{27}$$

$$q_{ij} = \begin{cases} \left[ 1 + a || \mathbf{t}_i - \mathbf{t}_j ||^{2b} \right]^{-1}, & j \neq i \\ 0, & j = i \end{cases}, \tag{28}$$

where $q_{ij} = q_{ji}$, $q_{ii} = 0$, $\sum_{i,j} q_{ij} = 1$ and $\sum_j q_{j|i} = 1, \forall i, j$. The constants $a, b \in \mathbb{R}$ are either user-defined or are determined by the algorithm given the desired separation between close points, $\delta \in \mathbb{R}^+$, in the embedding space $\mathcal{Q}$

$$\left[ 1 + a || \mathbf{t}_i - \mathbf{t}_j ||^{2b} \right]^{-1} \approx \begin{cases} 1, & \mathbf{t}_i - \mathbf{t}_j \leq \delta \\ \exp[-(\mathbf{t}_i - \mathbf{t}_j) - \delta], & \mathbf{t}_i - \mathbf{t}_j > \delta \end{cases}. \tag{29}$$

The UMAP performs an optimization while attempting to minimize the cross-entropy $CE$ between the distribution of points in $\mathcal{P}$ and $\mathcal{Q}$

$$CE = \sum_{i \neq j} \left[ p_{ij} \ln \frac{p_{ij}}{q_{ij}} - (1 - p_{ij}) \ln \frac{1 - p_{ij}}{1 - q_{ij}} \right]. \tag{30}$$

The minimization scheme starts with a given initial set of points in $\mathcal{Q}$. The UMAP uses the Graph Laplacian to assign initial low-dimensional coordinates, and then proceeds with the optimization using the gradient descent

$$\frac{\partial CE}{\partial \mathbf{t}_i} = \sum_j \left[ \frac{2ab[d(\mathbf{t}_i, \mathbf{t}_j)]^{2(b-1)}}{1 + a[d(\mathbf{t}_i, \mathbf{t}_j)]^{2b}} p_{ij} - \frac{2b}{[d(\mathbf{t}_i, \mathbf{t}_j)]^2 (1 + a[d(\mathbf{t}_i, \mathbf{t}_j)]^{2b})} (1 - p_{ij}) \right] (\mathbf{t}_i - \mathbf{t}_j). \tag{31}$$

## 3. Description of the Dataset

The prototype dataset representative of a given CS corresponds to the DJIA daily closing values from 28 December 1959 up to 12 March 2021. Each week includes 5 working days. Occasional missing data are obtained by means of linear interpolation. The resulting

time series $\mathbf{x} = \{x_k : k = 1, \ldots, L\}$ comprises $L = 15970$ values, $x_k$, covering approximately half a century.

Often, we pre-process $\mathbf{x}$ in order to reduce the sensitivity to a high variation in the numerical values, yielding $\tilde{\mathbf{x}} = \{\Phi_q(x_k) : k = 1, \ldots, L\}$. Functions $\Phi_q(\cdot)$, which are commonly adopted, are the logarithm of the values, the logarithm of the returns and the normalization by the arithmetic mean, $\mathrm{av}(\mathbf{x})$, and the standard deviation, $\sigma(\mathbf{x})$, given by

$$\Phi_1(x_k) = \ln x_k, \tag{32}$$

$$\Phi_2(x_k) = \begin{cases} \ln \frac{x_{k+1}}{x_k}, & k = 1, \ldots, L-1 \\ 0, & k = L \end{cases}, \tag{33}$$

$$\Phi_3(x_k) = \frac{x_k - \mathrm{av}(\mathbf{x})}{\sigma(\mathbf{x})}. \tag{34}$$

Figure 1 depicts the evolution of $\mathbf{x}$, as well as the logarithm of the returns $\tilde{\mathbf{x}} = \{\Phi_2(x_k) : k = 1, \ldots, L\}$, which reveals a fractal nature. We verify the existence of 13 main periods denoted from $\mathcal{A}$ to $\mathcal{M}$. For $k \in [1, 640]$, corresponding to the periods $\mathcal{A}$ and $\mathcal{B}$, the values of $x_k$ are small, starting with a decrease, followed by a recovering trend. This behavior is followed by a sustainable increase in the DJIA during $k \in [640, 1555]$, period $\mathcal{C}$. The interval $k \in [1555, 5890]$ corresponds to the periods $\mathcal{D}$, $\mathcal{E}$ and $\mathcal{F}$, which are characterized by an overall stagnation in the between of severe crises. For $k \in [5890, 7237]$, that is, period $\mathcal{G}$, we have an important rising trend, interrupted abruptly, but rapidly recovered, marking the beginning of period $\mathcal{H}$ for $k \in [7237, 10{,}340]$. For $k \in [10{,}340, 11{,}240]$, corresponding to period $\mathcal{I}$, the DJIA reveals a decreasing trend. This behavior is followed by the period $\mathcal{J}$, during the interval $k \in [11{,}240, 12{,}500]$, characterized by a sustained increase in the DJIA values. For $k \in [12{,}500, 12{,}840]$, the period $\mathcal{K}$ reveals a strong falling trend. Then, recovery initiates and a rising trend is verified during the period $\mathcal{L}$, that is, for $k \in [12{,}840, 15{,}690]$. This period is interrupted suddenly, but rapidly, recovered, signaling the beginning of $\mathcal{M}$, corresponding to $k \in [15{,}690, 15{,}970]$. Table 1 summarizes the DJIA main periods and some historical events occurred during 28 December 1959 up to 12 March 2021.

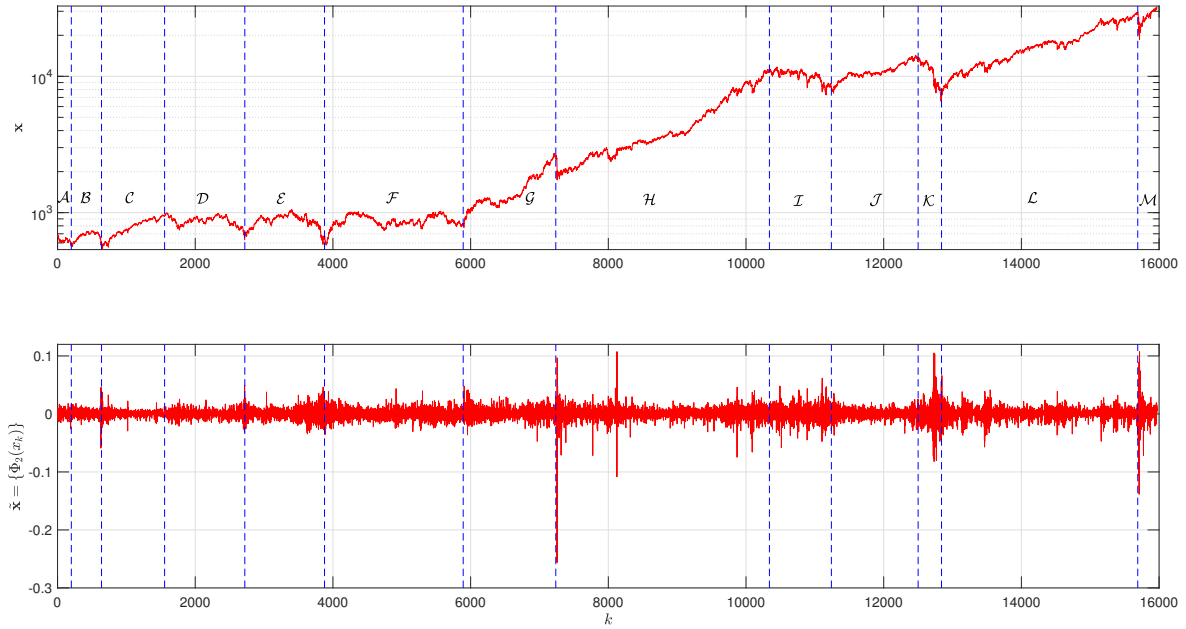


**Figure 1.** The evolution of the time-series $\mathbf{x}$ and $\tilde{\mathbf{x}} = \{\Phi_2(x_k) : k = 1, \ldots, L\}$, in the period from 28 December 1959 up to 12 March 2021.

**Table 1.** The DJIA main periods and some historical events occurred during 28 December 1959 up to 12 March 2021.

| Period | Interval, $k$ | Start Date | End Date | Main Events |
|---|---|---|---|---|
| $\mathcal{A}$ | [1, 200] | 28 December 1959 | 30 September 1960 | 1961 Berlin Wall; Bay of Pigs |
| $\mathcal{B}$ | [200, 640] | 30 September 1960 | 8 June 1962 | |
| $\mathcal{C}$ | [640, 1555] | 8 June 1962 | 10 December 1965 | 1962 Cuban Missile Crisis; 1963 John F. Kennedy Assassination; 1964 Vietnam War Begins; 1965 The Great Inflation Begins |
| $\mathcal{D}$ | [1555, 2720] | 10 December 1965 | 29 May 1970 | 1967 The Six Day War |
| $\mathcal{E}$ | [2720, 3878] | 29 May 1970 | 6 November 1974 | 1972 Watergate; Munich Olympics Massacre; 1973 U.S. Involvement in Vietnam Ends; Arab Oil Embargo; 1974 President Nixon Resigns |
| $\mathcal{F}$ | [3878, 5890] | 6 November 1974 | 23 July 1982 | 1977 Panama Canal Treaty; 1979 Iran Hostage Crisis; 1980 Iraq - Iran War; 1981 President Reagan Shot; 1982 Falkland Islands War |
| $\mathcal{G}$ | [5890, 7237] | 23 July 1982 | 22 September 1987 | 1983 Grenada Invasion; 1986 U.S. Attacks Libya; Chernobyl Accident; 1987 Financial Panic; Stock Market Crash |
| $\mathcal{H}$ | [7237, 10,340] | 22 September 1987 | 13 August 1999 | 1989 U.S. Invades Panama; German Unification; 1991 The Golf War; Soviet Union Collapse; 1992 Civil War in Bosnia; 1993 World Trade Center Terrorist Attack; 1995 Oklahoma Terrorist Attack; 1997 Asian Currency Crisis; Global Stock Market Rout |
| $\mathcal{I}$ | [10,340, 11,240] | 13 August 1999 | 24 January 2003 | 2000 Bush - Gore Election Crisis; 2001 Terrorist Attack on World Trade Center & Pentagon; Enron Crisis; 2003 War in Iraq |
| $\mathcal{J}$ | [11,240, 12,500] | 24 January 2003 | 23 November 2007 | 2004 Global War on Terror; 2005 Record High Oil Prices; 2007 Subprime Mortgage; Credit Debacle |
| $\mathcal{K}$ | [12,500, 12,840] | 23 November 2007 | 13 March 2009 | 2008 Credit Crisis; Financial institution Failures |
| $\mathcal{L}$ | [12,840, 15,690] | 13 March 2009 | 14 February 2020 | 2010 European Union Crisis; Massive Debt; 2011 U.S. Credit Downgrade; 2012 European Debt; 2013 U.S. Government Shutdown; 2014 Oil Price Decline; 2015 Refugee Crisis; 2016 Brexit Referendum; 2017 Trump Administration; 2018 Warnings About Climate Change; U.S. - China Trade War; President Trump Impeachment Process |
| $\mathcal{M}$ | [15,690, 15,970] | 14 February 2020 | 12 March 2021 | 2020 COVID19 Pandemics; Black Lives Matter |

To assess the dynamics of the DJIA, the time-series $\tilde{\mathbf{x}}$ is segmented into $N = \lfloor 1 + \frac{L-W}{(1-\alpha)W} \rfloor$, where $W$ is the window length, $\alpha \in [0, 1]$ stands for the window overlapping factor, and $\lfloor \cdot \rfloor$ denotes the floor function. Therefore, the $i$th, $i = 1, \ldots, N$, window consists of the vector $\mathbf{v}_i = \{\Phi_q(x_p) : p = (i-1)(1-\alpha)W + 1, \ldots, (i-1)(1-\alpha)W + W\}$.

Figure 2 portraits the histogram of $\tilde{\mathbf{x}} = \{\Phi_2(x_k) : k = 1, \ldots, L\}$ for consecutive disjoint windows ($\alpha = 0$) and $W = 60$. We verify the existence of fat tails in the statistical distribution, as well as a 'noisy' behavior, which are also verified for other functions $\Phi_q$ and values of $\alpha$ and $W$.

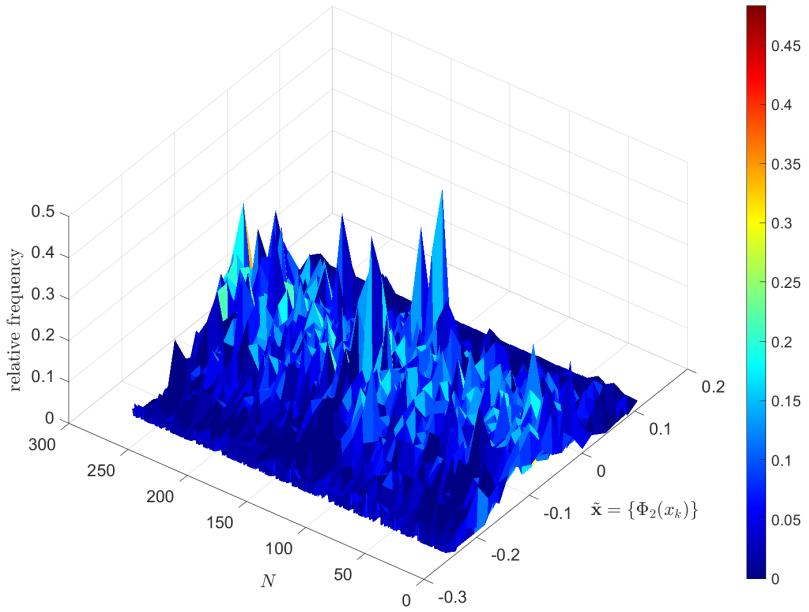


**Figure 2.** The histogram of $\tilde{\mathbf{x}} = \{\Phi_2(x_k) : k = 1, \ldots, L\}$ for consecutive disjoint time windows ($\alpha = 0$) and $W = 60$.

## 4. Analysis and Visualization of the DJIA

The DJIA time-series $\mathbf{x}$ is normalized using expression (34), yielding $\tilde{\mathbf{x}} = \{\Phi_3(x_k) : k = 1, \ldots, L\}$. Naturally, other types of pre-processing are possible, but the linear transform (34) is common in signal processing [40] and several experiments showed that it yields good results.

In the next subsections, $\tilde{\mathbf{x}}$ is segmented using consecutive disjoint ($\alpha = 0$) time windows of length $W = 60$ days, which yield $N = 266$ objects, $\mathbf{v}_i$, with $i = 1, \ldots, N$. These objects are processed by the dimensionality reduction and visualization methods, while adopting different distances (1)–(9) to quantify the dissimilarities between objects. For the generalized distance $d_{10}$, given by expression (10), since no a priori preference for a given formula is set, we adopt identical weights, that is, $\lambda_r = \frac{1}{9}$, $r = 1, \ldots, 9$. The values of $\alpha$ and $W$ were chosen experimentally. Obviously, other values could have been adopted, but those used lead to a good compromise between time resolution and suitable visualization.

### *4.1. The HC Analysis and Visualization of the DJIA*

The neighbor-joining method [41] and the successive (agglomerative) clustering using average-linkage are adopted, as implemented by the software `Phylip` [42] with the option `neighbor`. Figure 3 depicts the HC trees with the distances $d_2$, $d_3$, $d_5$ and $d_{10}$. The circular marks correspond to objects (window vectors) and the colormap represents the arrow of time. We verify that the HC has difficulty in separating the periods $\mathcal{A}$-$\mathcal{F}$ and, for distance $d_5$, this difficulty is also observed for the periods $\mathcal{H}$-$\mathcal{J}$. For other distances, we obtain loci of the same type.

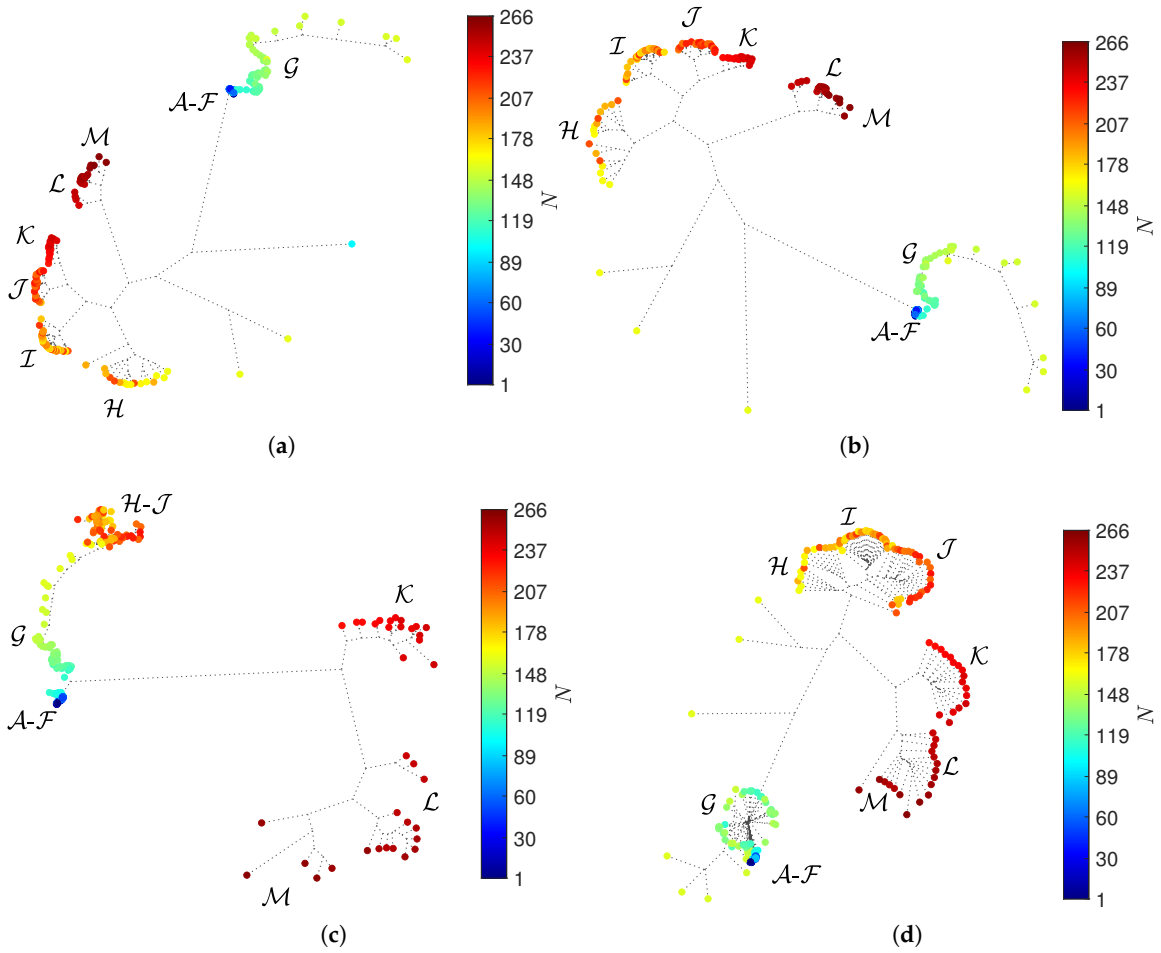


**Figure 3.** The hierarchical trees obtained by the HC for $\alpha = 0$ and $W = 60$ ($N = 266$) with four distances: (**a**) $d_2$; (**b**) $d_3$; (**c**) $d_5$; (**d**) $d_{10}$. The circular marks correspond to objects (window vectors) and the colormap represents the arrow of time.

The HC loci reflect the relationships between objects, but the interpretation of such loci is difficult due to the presence of many objects and because we are constrained to 2-dim visual representations. The reliability of the clustering, that is, how well the hierarchical trees reproduce the original dissimilarities of the original objects in the dataset, was verified. Nevertheless, we do not include the Shepard diagrams for the sake of parsimony.

### *4.2. The MDS Analysis and Visualization of the DJIA*

We now visualize the DJIA behavior using the MDS. The `Matlab` function `mdscale` with the Sammon nonlinear mapping criterion is adopted. Figure 4 depicts the 3-dim loci obtained for $\alpha = 0$ and $W = 60$ ($N = 266$) with the distances $d_2$, $d_3$, $d_5$ and $d_{10}$.

The reliability of the 3-dim loci was verified through the standard Shepard and stress plots, which showed that the objects in the embedding space $\mathcal{Q}$ reproduce those in the original space $\mathcal{P}$. Those diagrams are not depicted for the sake of parsimony. We verify that the MDS unravels patterns compatible with the DJIA 13 periods $\mathcal{A}$-$\mathcal{M}$. However, the algorithm cannot discriminate between them. The patterns are composed by two 'segments' formed by objects that reveal an almost continuous and smooth evolution in time. Each segment translates into a DJIA dynamics exhibiting strong memory effects that are captured by the visualization technique with the adopted distance. The transition between segments corresponds to some discontinuity where the memory of past values is somehow lost.

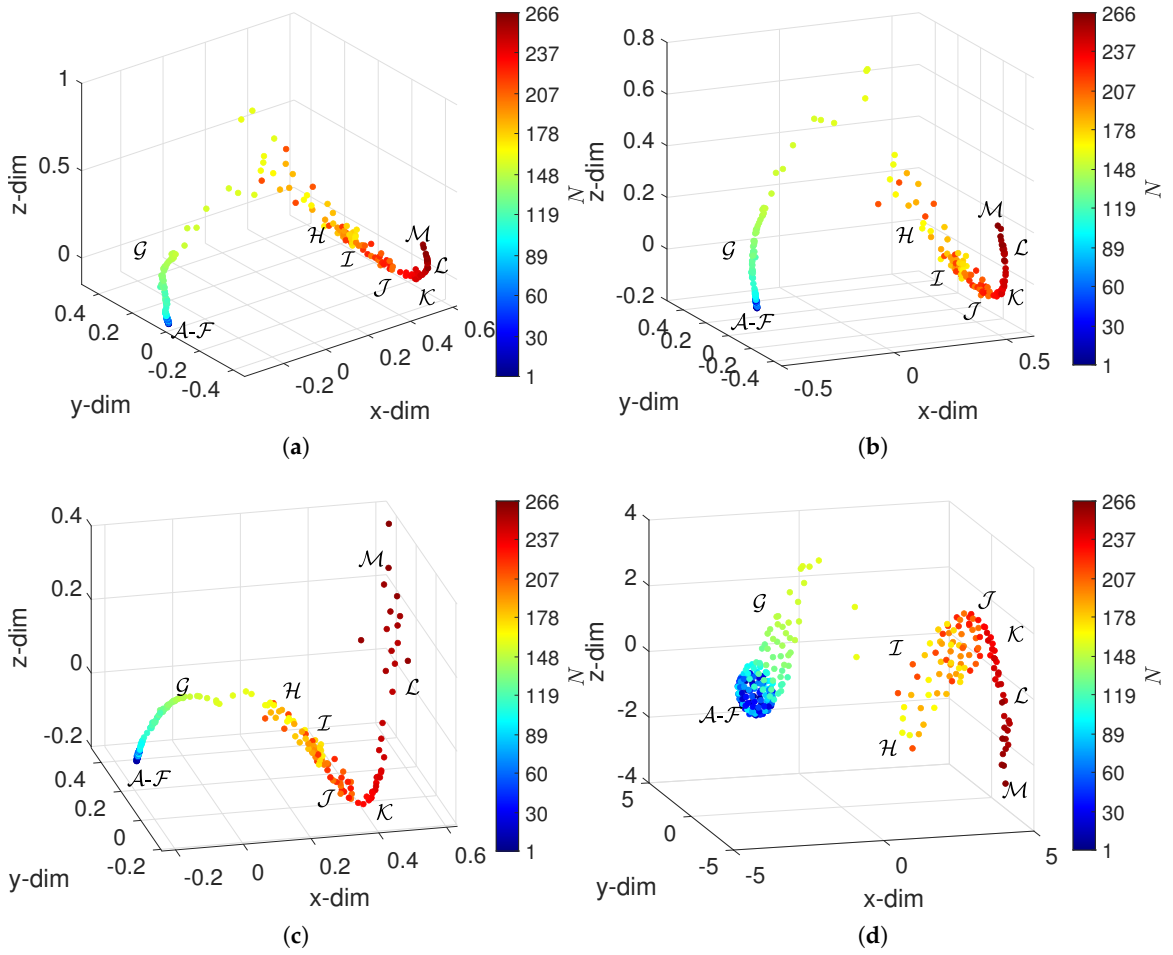


**Figure 4.** The 3-dim loci obtained by the MDS for $\alpha = 0$ and $W = 60$ ($N = 266$) with four distances: (**a**) $d_2$; (**b**) $d_3$; (**c**) $d_5$; (**d**) $d_{10}$. The circular marks correspond to objects (window vectors) and the colormap represents the arrow of time.

For other distances, we obtain loci of several types. However, it should be noted that often the definition of an adequate distance (in the sense of assessing the dynamical effects) necessitates some numerical trials. Different distances can lead to valid visual representations, but may be unable to capture the features of interest. For example, the correlation distance, $d_{11}$, given by

$$\text{correlation} : d_{12}(\mathbf{v}_i, \mathbf{v}_j) = \left( 1 - \frac{\sum_{k=1}^{P} [v_{ik} - \text{av}(\mathbf{v}_i)][v_{jk} - \text{av}(\mathbf{v}_j)]}{\sqrt{\sum_{k=1}^{P} [v_{ik} - \text{av}(\mathbf{v}_i)]^2} \sqrt{\sum_{k=1}^{P} [v_{jk} - \text{av}(\mathbf{v}_j)]^2}} \right)^{\frac{1}{2}}, \tag{35}$$

leads to the loci shown in Figure 5, revealing that neither the HC nor the MDS can capture the memory effects embedded in the dataset.

### *4.3. The t-SNE Analysis and Visualization of the DJIA*

The `Matlab` function `tsne` was adopted to visualize the dataset $\tilde{\mathbf{x}} = \{\Phi_3(x_k) : k = 1, \ldots, L\}$. The `algorithm` was set to `exact` and the value 5 was given to the `Exaggeration` and the `Perplexity`. These values were adjusted by trial in order to obtain good visualization. The `Exaggeration` corresponds to the size of natural clusters in data. A large exaggeration creates relatively more space between clusters in the embedding space $\mathcal{Q}$.

The `Perplexity` is related to the number of local neighbors of each point. All other parameters kept their default values. Figure 6 depicts the 3-dim loci obtained for the distances $d_2$, $d_3$, $d_5$ and $d_{10}$. The loci reveal that the t-SNE can arrange objects according to their the periods $\mathcal{A}$-$\mathcal{M}$ and that the plots generated with the different distances are similar.

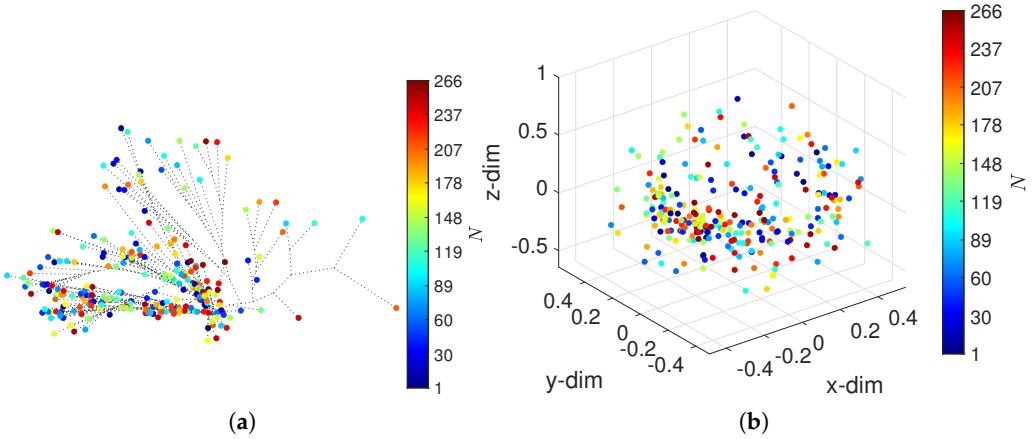


**Figure 5.** The loci obtained for $\alpha = 0$ and $W = 60$ ($N = 266$) with the correlation distance $d_{11}$: (**a**) hierarchical tree; (**b**) MDS locus.

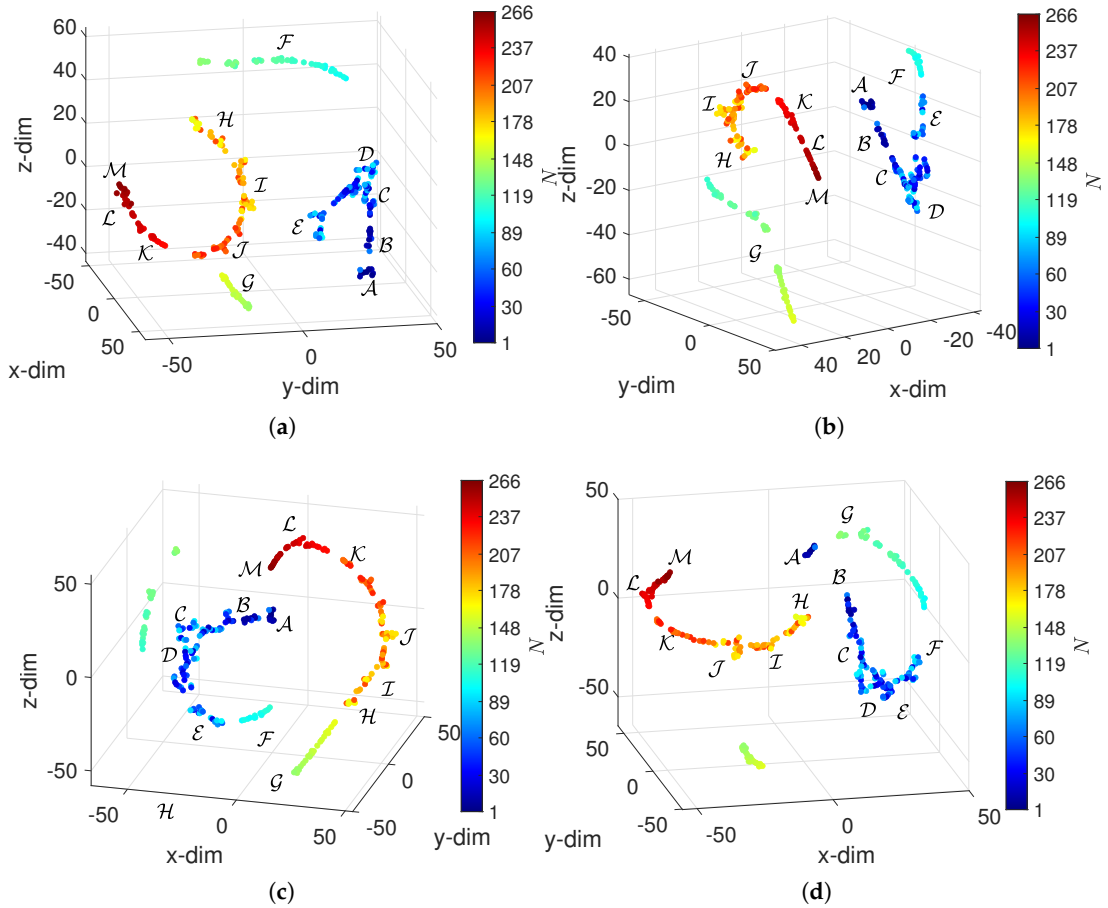


**Figure 6.** The 3-dim maps obtained by the t-SNE for $\alpha = 0$ and $W = 60$ ($N = 266$) with four distances: (**a**) $d_2$; (**b**) $d_3$; (**c**) $d_5$; (**d**) $d_{10}$. The circular marks correspond to objects (window vectors) and the colormap represents the arrow of time.

### *4.4. The UMAP Analysis and Visualization of the DJIA*

For implementing the UMAP dimensionality reduction and visualization, we adopted the `Matlab UMAP code, version 2.1.3`, developed by Stephen Meehan et al. [43]. The function `run_umap` was used with parameters `n_neighbors` and `min_dist` set to 5 and 0.2, respectively, adjusted by trial and error in order to obtain good visualization. These parameters correspond directly to $k$ and $\delta$ introduced in Section 2.2.4. All other parameters are set to their default values. Figure 7 depicts the 3-dim loci obtained for the distances $d_2$, $d_3$, $d_5$ and $d_{10}$.

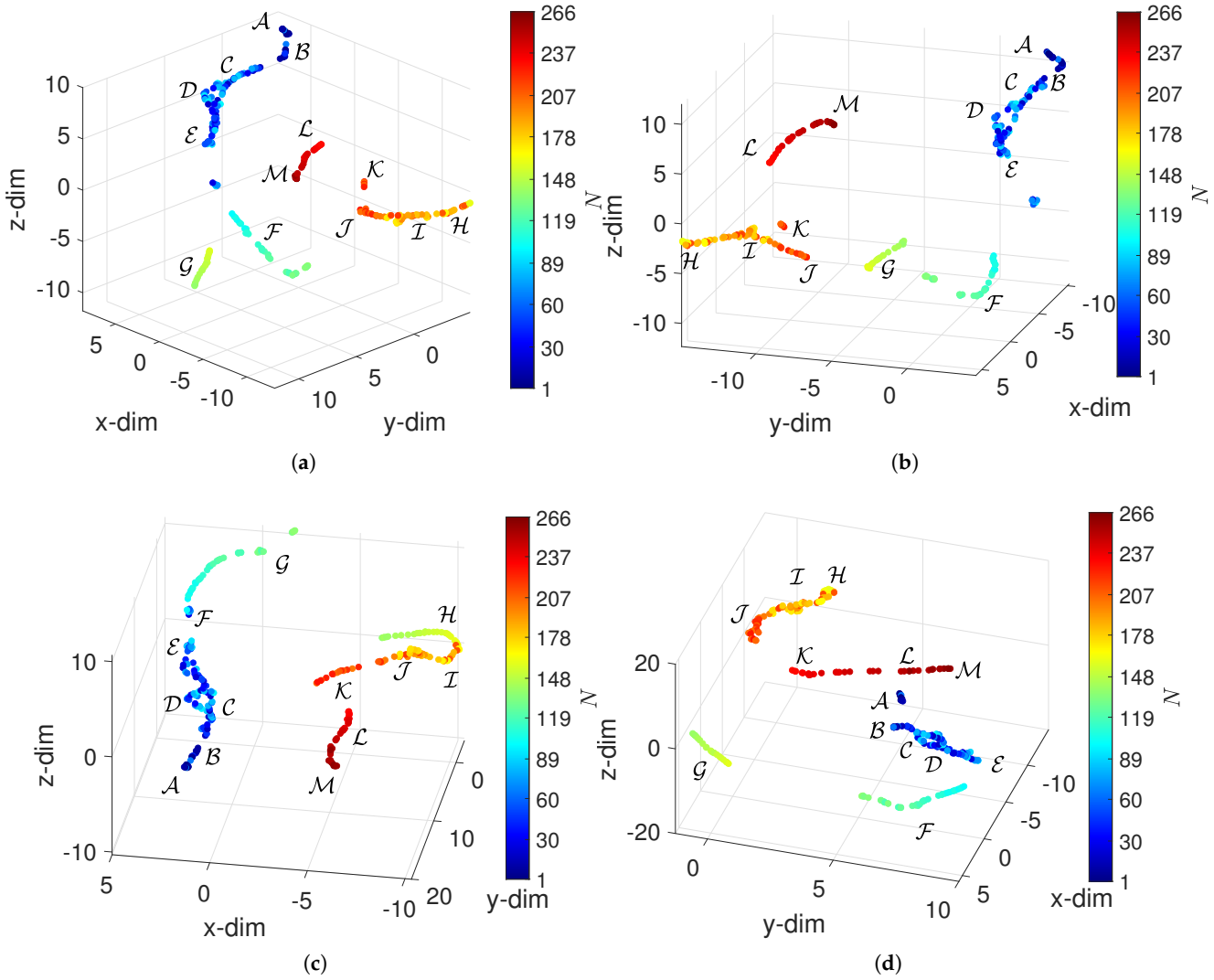


**Figure 7.** The 3-dim loci obtained by the UMAP for $\alpha = 0$ and $W = 60$ ($N = 266$) with four distances: (**a**) $d_2$; (**b**) $d_3$; (**c**) $d_5$; (**d**) $d_{10}$. The circular marks correspond to objects (window vectors) and the colormap represents the arrow of time.

The UMAP can organize objects in $\mathcal{Q}$ according to their characteristics, identifying well the periods $\mathcal{A}$-$\mathcal{M}$, independently of the adopted distance. Therefore, we conclude that both the t-SNE and the UMAP perform better than the MDS in representing the DJIA dynamics. The visualization has only slight variations with the distance adopted to compare objects.

## 5. Assessing the Effect of $W$ and $\alpha$ in the Visualization of the DJIA Dynamics

The window width and overlap, $W$ and $\alpha$, represent a compromise between time resolution and memory length. In this section, we study the effect of these parameters on the patterns generated by the HC, MDS, t-SNE and UMAP. The analysis was performed for all distances and several combinations of $W$ and $\alpha$. The results are presented for the Canberra distance, $d_2$, and the cases summarized in Table 2, where $W = \{90, 60, 30, 10\}$ and $\alpha = \{0, 0.2, 0.5\}$. For other distances, we obtain similar conclusions.

**Table 2.** List of experiments varying $W$ and $\alpha$.

| | $W$ | $\alpha$ | $N$ | | $W$ | $\alpha$ | $N$ |
|---|---|---|---|---|---|---|---|
| $E_1$ | 90 | 0 | 177 | $E_7$ | 30 | 0 | 532 |
| $E_2$ | 90 | 0.2 | 221 | $E_8$ | 30 | 0.2 | 665 |
| $E_3$ | 90 | 0.5 | 353 | $E_9$ | 30 | 0.5 | 1063 |
| $E_4$ | 60 | 0 | 266 | $E_{10}$ | 10 | 0 | 1597 |
| $E_5$ | 60 | 0.2 | 332 | $E_{11}$ | 10 | 0.2 | 1996 |
| $E_6$ | 60 | 0.5 | 531 | $E_{12}$ | 10 | 0.5 | 3193 |

Figures 8–11 depict the loci generated. Regarding the HC, we verify that the loci are quite insensitive to the parameter $W$, with the exception of $W = 10$. For this value of window length, the HC can discriminate objects in the periods $\mathcal{A}$-$\mathcal{F}$, despite the fact that capability depends on the overlap $\alpha$. For $W = 10$ and $\alpha = 0.5$, the objects in $\mathcal{A}$-$\mathcal{F}$ spread out in space, but their clusters are still unclear. Concerning the MDS, besides the density of objects, which, naturally, varies with $N$, the 3-dim loci are almost invariant with respect to the parameters $W$ and $\alpha$. The t-SNE and UMAP reveal a superior ability to generate patterns that correspond to dissimilarities between objects and, therefore, are able to identify the 13 periods $\mathcal{A}$-$\mathcal{M}$. However, for the t-SNE, this ability is weakened as the number of objects increases, $N$, meaning small values of $W$ and high values of $\alpha$. For such cases, the generated loci are difficult to interpret. The UMAP reveals the 13 periods $\mathcal{A}$-$\mathcal{M}$ for all combinations of $W$ and $\alpha$. Moreover, for small values of $W$ several sub-periods are unraveled, which directly relate to the time evolution of the DJIA.

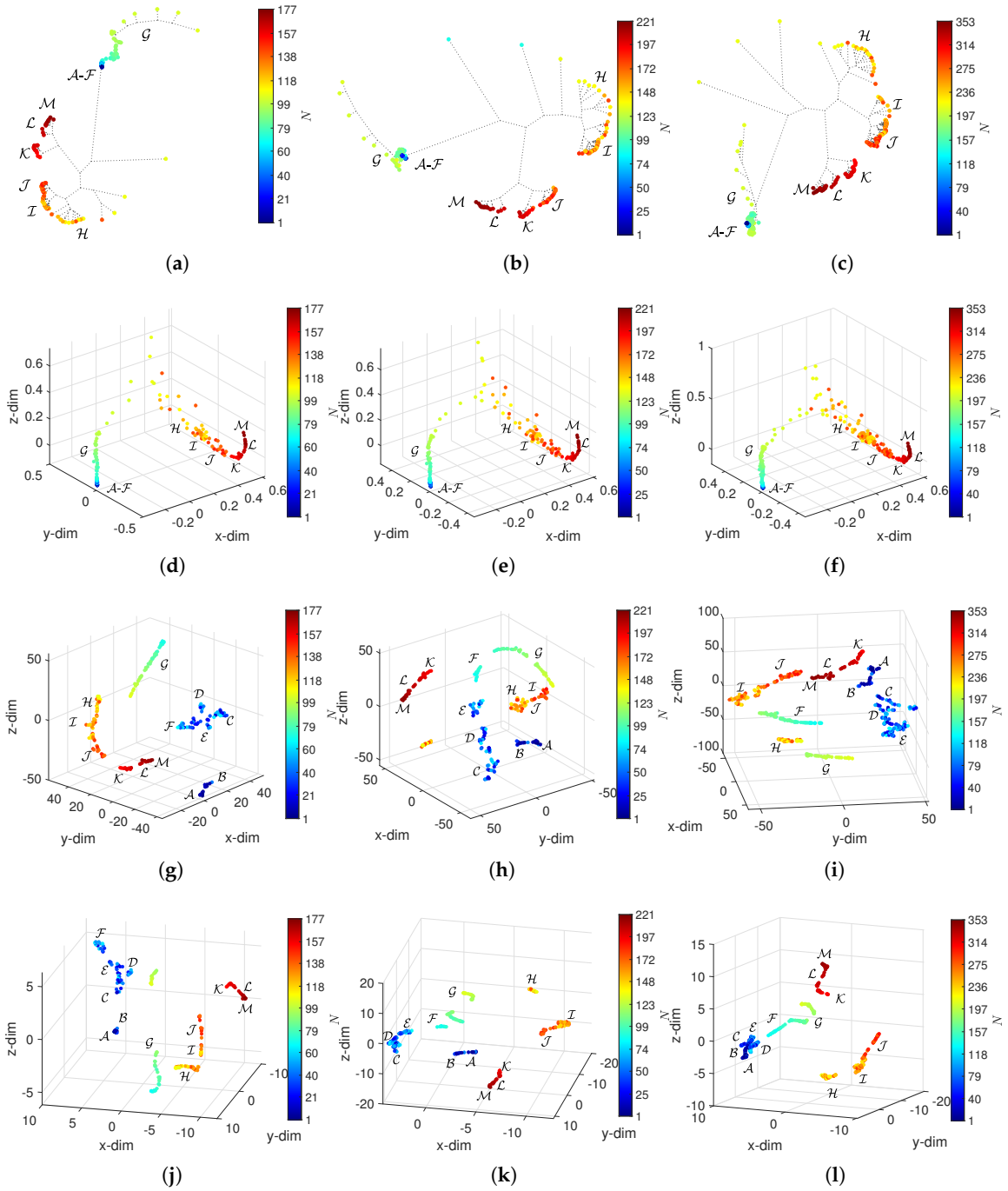


**Figure 8.** The 3-dim loci obtained for $d_2$ and $W = 90$: (**a**) HC and $\alpha = 0$ ($E_1$); (**b**) HC and $\alpha = 0.2$ ($E_2$); (**c**) HC and $\alpha = 0.5$ ($E_3$); (**d**) MDS and $\alpha = 0$ ($E_1$); (**e**) MDS and $\alpha = 0.2$ ($E_2$); (**f**) MDS and $\alpha = 0.5$ ($E_3$); (**g**) t-SNE and $\alpha = 0$ ($E_1$); (**h**) t-SNE and $\alpha = 0.2$ ($E_2$); (**i**) t-SNE and $\alpha = 0.5$ ($E_3$); (**j**) UMAP and $\alpha = 0$ ($E_1$); (**k**) UMAP and $\alpha = 0.2$ ($E_2$); (**l**) UMAP and $\alpha = 0.5$ ($E_3$).

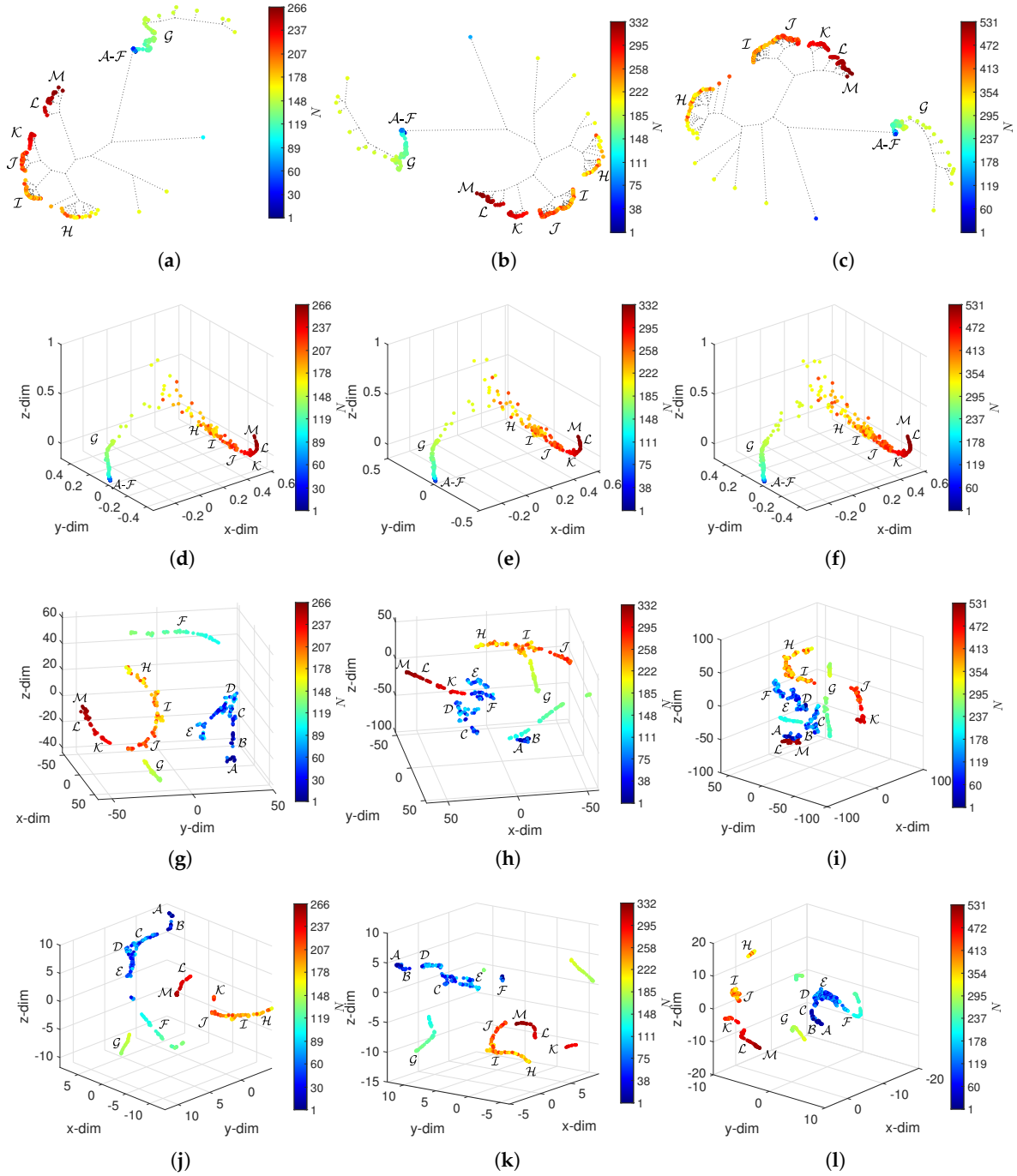


**Figure 9.** The 3-dim loci obtained for $d_2$ and $W = 60$: (**a**) HC and $\alpha = 0$ ($E_4$); (**b**) HC and $\alpha = 0.2$ ($E_5$); (**c**) HC and $\alpha = 0.5$ ($E_6$); (**d**) MDS and $\alpha = 0$ ($E_4$); (**e**) MDS and $\alpha = 0.2$ ($E_5$); (**f**) MDS and $\alpha = 0.5$ ($E_6$); (**g**) t-SNE and $\alpha = 0$ ($E_4$); (**h**) t-SNE and $\alpha = 0.2$ ($E_5$); (**i**) t-SNE and $\alpha = 0.5$ ($E_6$); (**j**) UMAP and $\alpha = 0$ ($E_4$); (**k**) UMAP and $\alpha = 0.2$ ($E_5$); (**l**) UMAP and $\alpha = 0.5$ ($E_6$).

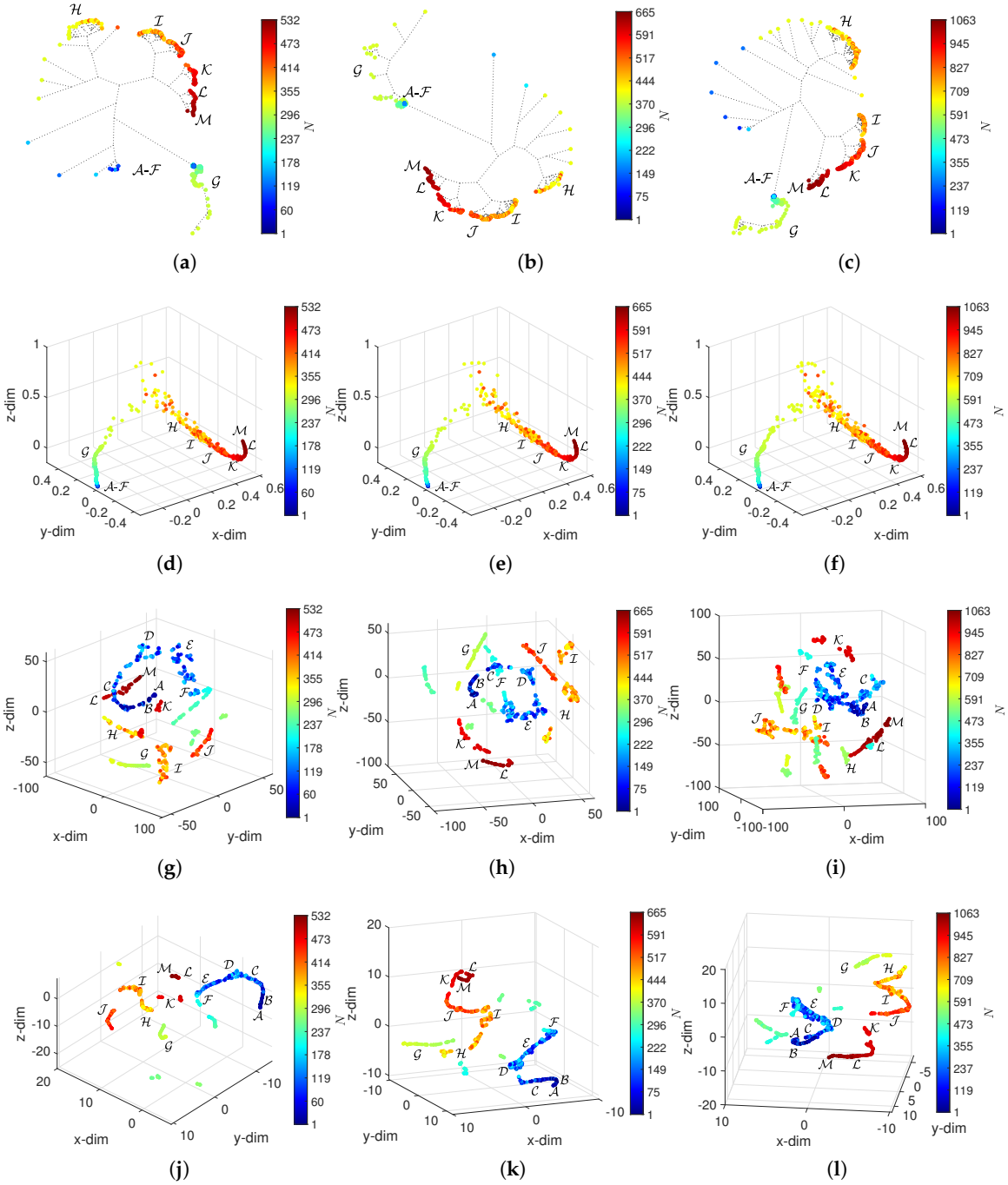


**Figure 10.** The 3-dim loci obtained for $d_2$ and $W = 30$: (**a**) HC and $\alpha = 0$ ($E_7$); (**b**) HC and $\alpha = 0.2$ ($E_8$); (**c**) HC and $\alpha = 0.5$ ($E_9$); (**d**) MDS and $\alpha = 0$ ($E_7$); (**e**) MDS and $\alpha = 0.2$ ($E_8$); (**f**) MDS and $\alpha = 0.5$ ($E_9$); (**g**) t-SNE and $\alpha = 0$ ($E_7$); (**h**) t-SNE and $\alpha = 0.2$ ($E_8$); (**i**) t-SNE and $\alpha = 0.5$ ($E_9$); (**j**) UMAP and $\alpha = 0$ ($E_7$); (**k**) UMAP and $\alpha = 0.2$ ($E_8$); (**l**) UMAP and $\alpha = 0.5$ ($E_9$).

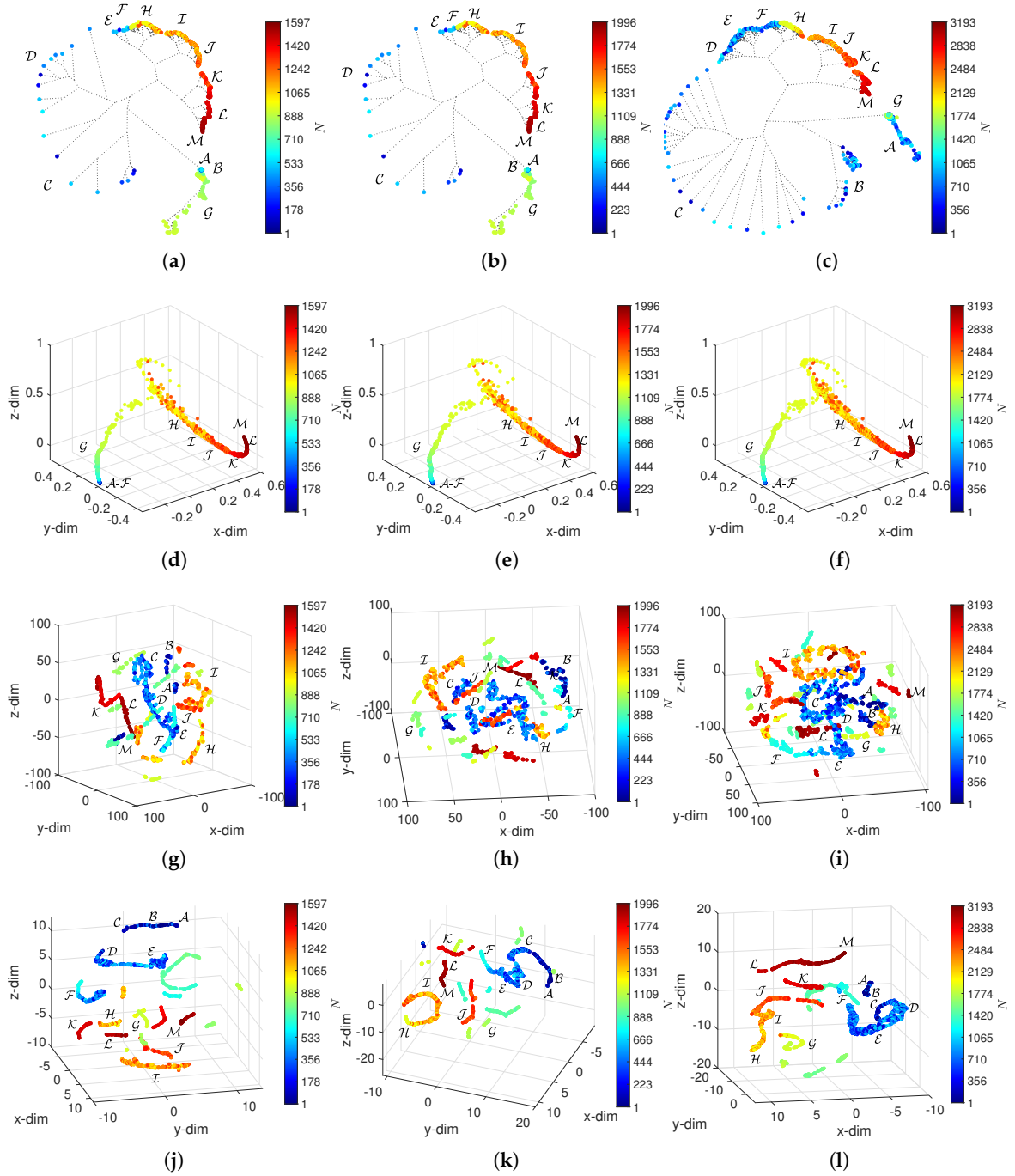


**Figure 11.** The 3-dim loci obtained for $d_2$ and $W = 10$: (**a**) HC and $\alpha = 0$ ($E_{10}$); (**b**) HC and $\alpha = 0.2$ ($E_{11}$); (**c**) HC and $\alpha = 0.5$ ($E_{12}$); (**d**) MDS and $\alpha = 0$ ($E_{10}$); (**e**) MDS and $\alpha = 0.2$ ($E_{11}$); (**f**) MDS and $\alpha = 0.5$ ($E_{12}$); (**g**) t-SNE and $\alpha = 0$ ($E_{10}$); (**h**) t-SNE and $\alpha = 0.2$ ($E_{11}$); (**i**) t-SNE and $\alpha = 0.5$ ($E_{12}$); (**j**) UMAP and $\alpha = 0$ ($E_{10}$); (**k**) UMAP and $\alpha = 0.2$ ($E_{11}$); (**l**) UMAP and $\alpha = 0.5$ ($E_{12}$).

## 6. Conclusions

This paper explored a strategy representing an alternative to the classical time analysis in the study multidimensional data generated by CS. The DJIA index of daily closing values from 28 December 1959 up to 12 March 2021 was adopted for the numerical experiments. In the proposed scheme, the original time-series was normalized and segmented, yielding a number of objects. These objects are vectors, whose dimension and overlap represent a compromise between time resolution and memory length. The objects were compared using various distances and their dissimilarities are used as the input to the four dimen-

sionality reduction and information visualization algorithms, namely, HC, MDS, t-SNE and UMAP. These algorithms construct representations of the original dataset, where time is a parametric variable, with no a priori requirements. The algorithms are based on the minimization of the difference between the original and approximated data. The plots were analyzed in terms of the emerging patterns. Those graphical representations are composed of a number of 'segments', formed by objects with an almost continuous evolution in time, interlaid, eventually, by some discontinuities. This translates into the DJIA dynamics that depicts phases with visible correlation. Consequently, memory effects and transitions corresponding to some discontinuities where the memory of past values is not present. Numerical experiments illustrated the feasibility and effectiveness of the method for processing complex data. The approach can be easily extended to deal with more features and richer descriptions of the data involving a higher number of dimensions.

**Author Contributions:** A.M.L. and J.A.T.M. conceived, designed and performed the experiments, analyzed the data and wrote the paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Pinto, C.; Mendes Lopes, A.; Machado, J. A review of power laws in real life phenomena. *Commun. Nonlinear Sci. Numer. Simul.* **2012**, *17*, 3558–3578. [CrossRef]
2. Tarasova, V.V.; Tarasov, V.E. Concept of dynamic memory in economics. *Commun. Nonlinear Sci. Numer. Simul.* **2018**, *55*, 127–145. [CrossRef]
3. Tarasov, V.E. Fractional econophysics: Market price dynamics with memory effects. *Phys. A Stat. Mech. Its Appl.* **2020**, *557*, 124865. [CrossRef]
4. Tarasov, V.E.; Tarasova, V.V. *Economic Dynamics with Memory: Fractional Calculus Approach*; Walter de Gruyter GmbH & Co KG: Berlin, Germany; Boston, MA, USA, 2021; Volume 8.
5. Lopes, A.M.; Tenreiro Machado, J.; Huffstot, J.S.; Mata, M.E. Dynamical analysis of the global business-cycle synchronization. *PLoS ONE* **2018**, *13*, e0191491. [CrossRef] [PubMed]
6. Lopes, A.M.; Tenreiro Machado, J.; Galhano, A.M. Multidimensional scaling visualization using parametric entropy. *Int. J. Bifurc. Chaos* **2015**, *25*, 1540017. [CrossRef]
7. Meyers, R.A. *Complex Systems in Finance and Econometrics*; Springer Science & Business Media: New York, NY, USA, 2010.
8. Xia, P.; Lopes, A.M.; Restivo, M.T. A review of virtual reality and haptics for product assembly: From rigid parts to soft cables. *Assem. Autom.* **2013**. [CrossRef]
9. Li, J.; Shang, P.; Zhang, X. Financial time series analysis based on fractional and multiscale permutation entropy. *Commun. Nonlinear Sci. Numer. Simul.* **2019**, *78*, 104880. [CrossRef]
10. Machado, J.T.; Lopes, A.M. Fractional state space analysis of temperature time series. *Fract. Calc. Appl. Anal.* **2015**, *18*, 1518. [CrossRef]
11. Lopes, A.M.; Machado, J.T. Dynamical analysis and visualization of tornadoes time series. *PLoS ONE* **2015**, *10*, e0120260. [CrossRef]
12. Lopes, A.M.; Tenreiro Machado, J. Power law behavior and self-similarity in modern industrial accidents. *Int. J. Bifurc. Chaos* **2015**, *25*, 1550004. [CrossRef]
13. Nigmatullin, R.R.; Lino, P.; Maione, G. *New Digital Signal Processing Methods: Applications to Measurement and Diagnostics*; Springer Nature: Cham, Switzerland, 2020.
14. Ware, C. *Information Visualization: Perception for Design*; Elsevier: Waltham, MA, USA, 2012.
15. Spence, R. *Information Visualization: An Introduction*; Springer: Cham, Switzerland, 2001; Volume 1.
16. Van Der Maaten, L.; Postma, E.; Van den Herik, J. Dimensionality reduction: A comparative. *J. Mach. Learn. Res.* **2009**, *10*, 66–71.
17. Tenreiro Machado, J.; Lopes, A.M.; Galhano, A.M. Multidimensional scaling visualization using parametric similarity indices. *Entropy* **2015**, *17*, 1775–1794. [CrossRef]
18. Dunteman, G.H. *Principal Components Analysis*; Number 69; Sage: Newbury Park, CA, USA, 1989.

19. Thompson, B. Canonical correlation analysis. In *Encyclopedia of Statistics in Behavioral Science*; John Wiley & Sons: Chichester, UK, 2005.
20. Tharwat, A.; Gaber, T.; Ibrahim, A.; Hassanien, A.E. Linear discriminant analysis: A detailed tutorial. *AI Commun.* **2017**, *30*, 169–190. [CrossRef]
21. Child, D. *The Essentials of Factor Analysis*; Cassell Educational: London, UK, 1990.
22. France, S.L.; Carroll, J.D. Two-way multidimensional scaling: A review. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2010**, *41*, 644–661. [CrossRef]
23. Lee, J.A.; Lendasse, A.; Verleysen, M. Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis. *Neurocomputing* **2004**, *57*, 49–76. [CrossRef]
24. Belkin, M.; Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* **2003**, *15*, 1373–1396. [CrossRef]
25. Coifman, R.R.; Lafon, S. Diffusion maps. *Appl. Comput. Harmon. Anal.* **2006**, *21*, 5–30. [CrossRef]
26. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
27. McInnes, L.; Healy, J.; Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv* **2018**, arXiv:1802.03426.
28. Deza, M.M.; Deza, E. *Encyclopedia of Distances*; Springer: New York, NY, USA, 2009.
29. Nigmatullin, R.R. Discrete Geometrical Invariants in 3D Space: How Three Random Sequences Can Be Compared in Terms of "Universal" Statistical Parameters. *Front. Phys.* **2020**, *8*, 76. [CrossRef]
30. Hartigan, J.A. *Clustering Algorithms*; John Wiley & Sons: New York, NY, USA, 1975.
31. Aggarwal, C.C.; Hinneburg, A.; Keim, D.A. *On the Surprising Behavior of Distance Metrics in High Dimensional Space*; Springer: Berlin, Germany 2001.
32. Sokal, R.R.; Rohlf, F.J. The comparison of dendrograms by objective methods. *Taxon* **1962**, *11*, 33–40. [CrossRef]
33. Hamid, Y.; Sugumaran, M. A t-SNE based non linear dimension reduction for network intrusion detection. *Int. J. Inf. Technol.* **2020**, *12*, 125–134. [CrossRef]
34. Rao, A.; Aditya, A.; Adarsh, B.; Tripathi, S. Supervised Feature Learning for Music Recommendation. In *Communications in Computer and Information Science, Proceedings of the International Symposium on Signal Processing and Intelligent Recognition Systems, Chennai, India, 14–17 October 2020*; Springer: Singapore, 2020; pp. 122–130.
35. Li, W.; Cerise, J.E.; Yang, Y.; Han, H. Application of t-SNE to human genetic data. *J. Bioinform. Comput. Biol.* **2017**, *15*, 1750017. [CrossRef]
36. Kobak, D.; Berens, P. The art of using t-SNE for single-cell transcriptomics. *Nat. Commun.* **2019**, *10*, 1–14. [CrossRef] [PubMed]
37. Cieslak, M.C.; Castelfranco, A.M.; Roncalli, V.; Lenz, P.H.; Hartline, D.K. t-Distributed Stochastic Neighbor Embedding (t-SNE): A tool for eco-physiological transcriptomic analysis. *Mar. Genom.* **2020**, *51*, 100723. [CrossRef] [PubMed]
38. Becht, E.; McInnes, L.; Healy, J.; Dutertre, C.A.; Kwok, I.W.; Ng, L.G.; Ginhoux, F.; Newell, E.W. Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.* **2019**, *37*, 38–44. [CrossRef] [PubMed]
39. Dorrity, M.W.; Saunders, L.M.; Queitsch, C.; Fields, S.; Trapnell, C. Dimensionality reduction by UMAP to visualize physical and genetic interactions. *Nat. Commun.* **2020**, *11*, 1–6. [CrossRef] [PubMed]
40. Papoulis, A. *Signal Analysis*; McGraw-Hill: New York, NY, USA, 1977.
41. Saitou, N.; Nei, M. The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.* **1987**, *4*, 406–425.
42. Felsenstein, J. *PHYLIP (Phylogeny Inference Package), Version 3.5 c*; University of Washington: Seattle, WA, USA, 1993.
43. Meehan, C.; Ebrahimian, J.; Moore, W.; Meehan, S. Uniform Manifold Approximation and Projection (UMAP). 2021. Available online: https://www.mathworks.com/matlabcentral/fileexchange/71902 (accessed on 12 February 2021) .