IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.0322000

Autoencoder-Enhanced Clustering: A Dimensionality Reduction Approach to Financial Time Series

Daniel González Cortés [1], (Member, IEEE), Enrique Onieva [2], Iker Pastor López[2], Laura Trinchera [1], Jian Wu [1]

[1]NEOMA Business School, rue du Maréchal Juin, Mont Saint Aignan Cedex 76825, France
[2]Faculty of Engineering, University of Deusto, 48007, Bilbao, Spain

Corresponding author: Daniel González Cortés (e-mail: author@ daniel-alejandro.gonzalez-cortes.20@neoma-bs.com).

ABSTRACT While Machine Learning significantly boosts the performance of predictive models, its efficacy varies across different data dimensions. It is essential to cluster time series data of similar characteristics, particularly in the financial sector. However, clustering financial time series data poses considerable challenges due to the market's inherent complexity and multidimensionality. To address these issues, our study introduces a novel clustering framework that leverages autoencoders for a compressed yet informative representation of financial time series. We rigorously evaluate our approach through multiple dimensionality reduction and clustering algorithms, applying it to key financial indices, including IBEX-35, CAC-40, DAX-30, S&P 500, and FTSE 100. Our findings consistently demonstrate that incorporating autoencoders significantly enhances the granularity and quality of clustering, effectively isolating distinct categories of financial time series. Our findings carry significant ramifications for the financial industry. By refining clustering methodologies, we set the stage for increasingly accurate financial predictive models, offering valuable insights for optimizing investment strategies and enhancing risk management.

INDEX TERMS Clustering Methods, Data Compression, Financial Data Processing, Neural Network Applications, Time Series.

I. INTRODUCTION

Financial time series are challenging because they are inherently non-stationary, and it is common to find seasonal variations and long-term trends. In addition, price information derived from financial markets at any given time is often presented as a series of opening, high, low, and Closing Prices (CP) and transaction volume. This multidimensional nature hinders efficient examination of the data and impedes the proper use of clustering techniques [1].

The importance of clustering methods applied to time series is gaining traction due to the increase in technological applications that require, generate, and store information in real time. They play an essential role in different domains such as the internet of things [2], [3], autonomous vehicles [4], medicine [5], genetics [6] and finance [7], [8], where data usually is complex, non-stationary and high dimensional with temporal dependencies.

Consequently, clustering large time series with high dimensional data sets is complex, and correct dimensionality reduction and efficient extraction of the important features are essential for clustering. For example, a review by Aghabozorgi et al. [9] found that many authors focused on representing time series data in a lower dimension to be consistent with conventional clustering algorithms. On the other hand, Fawaz et al. [10] in an exhaustive review of deep learning for time series classification initially stated that the common attribute shared by those algorithms that outperform previous implementations is a transformation phase to convert the data series into a new feature space. This attribute raises the need for efficient transformation techniques since the most widely disseminated dimensionality-reduction method is the Principal Component Analysis (PCA); however, this technique is inherently linear and can only model linear interdependencies between the data set features [11].

This research intends to display a method to cluster multivariate financial data with a previous transformation stage based on a method for dimensionality reduction. We have proposed a reduction technique to compress multivariate time series in order to classify daily stock market activity and then use this compacted data representation in an efficient clustering process. This novel hybrid technique to cluster financial data can significantly improve the time and accuracy of classifying financial activity in an unsupervised manner. The first contribution made by the authors of this paper is to establish a concrete way of dealing with non-linear relationships to decrease the multi-dimensionality of a time series with opening, high, low and closing prices. The second contribution is implementing and comparing four different reduction techniques with six different technique algorithms to classify intraday activity from the IBEX-35 (IBEX), CAC-40 (CAC), DAX-30 (DAX), S&P 500 (SPX), and the FTSE 100 index (UKX). This review can help the decision-maker to better understand the daily stock market conditions or to use this method to classify time series, avoiding manual labelling.

The rest of the paper is organized as follows: Section II presents a brief review of research related to multidimensional data reduction. Next, section III presents a concise review of the work in the literature concerning time series clustering in different fields. Then, in section II, the long short-term memory autoencoder-based model will be introduced to reduce the overall dimensions of the financial time series, and then, in Section III, the classification algorithms will be presented. Next in Section IV, we present the proposal of this research with its respective settings and performance metrics. Then, in Section V, the results of each algorithm will be shown and discussed, and finally conclusions will be presented in Section VI.

II. DIMENSIONALITY REDUCTION TECHNIQUES

In this section we will discuss the different reduction techniques that have been used in this research. We will first briefly introduce the autoencoder and then go on to describe other commonly used reduction techniques.

A. AUTOENCODERS

An autoencoder (AE) is a form of an artificial neural network composed of two elements; an encoder and a decoder. It is possible to define it as a learning circuit with the primary goal of converting inputs into outputs with the lowest error or the least possible amount of distortion [12]. The function of the encoder is to reduce the raw input multidimensional data into a lower dimension. The decoder takes this transformed data and reconstructs it as accurately as possible by minimizing the reconstruction error using the chain rule to backpropagate error derivatives. The general structure of an AE is shown in Figure 1.

There are multiple forms of AE [12], such as those that are constructed using restricted Boltzmann machine [13] and deep learning structures [14] which change according to the complexity and arrangement of the data. Examples included fully connected deep AEs, convolutional neural networks AEs and recurrent neural network (RNN) AEs, which can be applied to different tasks and fields such as feature extraction for econometrics [15], fault diagnosis [16], fraud detection [17], genetics [18], image processing [19], language translation [20], remote sensing [21], robotics [22],and security [23] among others.

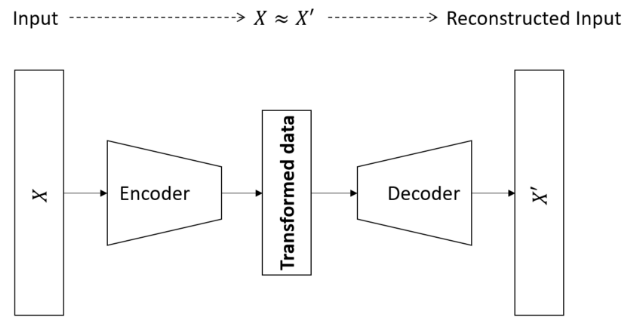

FIGURE 1. The structure of an AE

The RNN is a type of neural network that can deal efficiently with temporal data given that the output of the network is fed back into the network together with the new input through a sequential process. As such, an AE based on RNN can efficiently learn a vector representation of sequential data or time-series data because the architecture of this network is specially designed to deal with concatenations of data inputs. As shown in Figure 2, an RNN can be shown as a sequence of inputs, where A represents a layer of the network that contains vectors x with hidden state vectors h on time t.

There are different types of RNNs; one of them is the long short-term memory (LSTM) network created by Hochreiter and Schmidhuber [24] to improve some of the difficulties that classical RNN architecture encounters when trying to achieve an efficient learning process. This is due to the fact that the information from sequential data cannot be stored for a long time and can be affected by the vanishing gradient of the backpropagation used in artificial neural network training.

To overcome the difficulties of dealing with long-term data from classical RNN, LSTM networks augment explicit memory by using hidden units to recall short and long-term values. The different units of the LSTM are arranged to form a network with an input node and input, output, and forget gates to regulate the flow of information. The AE that uses LSTM has been successfully applied to various tasks. For example, Jung and Choi [25] use this method to forecast foreign exchange volatility. In addition, Maleki et al. [26] developed an enhanced LSTM AE for anomaly detection in sequential data.

B. PRINCIPAL COMPONENT ANALYSIS

The PCA is one of the most famous and widely used reduction techniques in multivariate statistical analysis. It can be applied to considerably diminish a large dataset's dimensions into a smaller one in an interpretable way, while maintaining as much statistical information as possible and efficiently

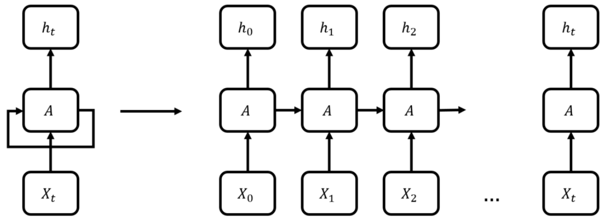

FIGURE 2. The structure of a RNN

dealing with data that might have multicollinearity or missing values.

This reduction technique can be defined in different ways, however there are two main descriptions that are frequently used [27]. The first one labels the PCA as an orthogonal projection of the information onto a principal subspace with a linear space of different dimensions to maximize the projected data. This method can also be described as the minimized squared distance between the project and the data points.

The maximum variance formulation considers a set of data $X = \{x_1, ..., x_N\}$, with m dimensions and $x_i \in \mathbb{R}^d$. The objective is to obtain a projection with a lower dimension d in $\mathbb{R}^m$, where $m < d$.

By considering $m = 1$, which needs a projection vector $u_i \in \mathbb{R}^d$, each point x_i projects to $u^T x_i$ and the variance of the projected data is defined as,

$$\frac{1}{N}\sum_{N}^{n=1}(u_1^T x_n - u_1^T \bar{x})^2 = u_1^T S u_1 \tag{1}$$

Where,

$$S = \frac{1}{N}\sum_{n=1}^{N}(x_n - \bar{x})(x_n - \bar{x})^T \tag{2}$$

$$\bar{x} = \frac{1}{N}\sum_{N}^{n=1} X_n \tag{3}$$

Then $u_1^T S u_1$, which is the projected variance that needs to be maximized with respect to u_1. The $\|u_1\| \to \infty$ maximization of the constraint u_1 also needs to be prevented with a normalization constrain $u_1^T u_1 = 1$.

This technique has been applied to reduce large datasets in multiple disciplines such as medicine [28], robotics [29], and statistical process control [30]. In financial markets, this technique is applied to predict the stock market in lower dimensions using multiple assets [31], [32] and to create models to select stocks [33]. In addition, Pasini [34] applied PCA to different subgroups of stocks efficiently deal with portfolio management; likewise, [35], using macroeconomic and institutional data from emerging markets using PCA, presented an equity asset pricing model to generate dynamic trading strategies.

The iPCA is a modification and substitution of the PCA to deal with large datasets that struggle with memory management. The iPCA uses an amount of memory that is independent of the number of input samples and constructs a low-rank approximation for the input data. This adaptation has the advantage of allowing sparse inputs and being more memory efficient than the traditional PCA model and has been used successfully in different fields such as medicine [36], chemistry [37], and biometric systems [38].

C. FAST FOURIER TRANSFORM

The Fast Fourier Transform (FFT), takes a signal from its original data domain to a different representation, decomposing a series of values into components of different frequencies and expressing it as a function that sums all the periodic components.

The FFT H_k of N points h_k is given by the formula

$$H_k = \sum_{n=0}^{N-1} \mathrm{e}^{-2\pi j \frac{kn}{N}} h_k \tag{4}$$

While the inverse transform is,

$$h_k = \frac{1}{N}\sum_{k=0}^{N-1} \mathrm{e}^{-2\pi j \frac{kn}{N}} H_n \tag{5}$$

as provided by [39].

Even though this method is effective, it is computationally expensive; however, the FFT algorithm can compute these transformations rapidly, reducing the complexity. As a result, the FFT has multiple applications and has been successfully applied to image processing [40].

III. CLUSTERING TECHNIQUES

Clustering is one of the so-called unsupervised learning methods in data mining. The objective of clustering methods is descriptive rather than predictive to group data instances into subsets to gather similar instances while different instances belong to different groups. To achieve efficient separation of groups, clustering methods rely on measuring distance between data samples and mechanisms to measure differences among data within similar clusters and differences with others [41].

In this section, we will describe the different cluster techniques used in this research. First, a brief introduction to agglomerative clustering (AGGLO) is given, followed by an overview of Balanced Iterative Reduction and Clustering by Hierarchy Clustering (BIRCH), and then k-means clustering algorithms with Euclidean distances (KNN-EUC) and Dynamic Temporal Warping (KNN-DTW). Finally, we examine the MiniBatch (MNBT) and Spectral (SPCT) techniques.

A. AGGLOMERATIVE CLUSTERING

Agglomerative clustering is one of the most common types of hierarchical clustering. This method initially assumes that each element represents an individual cluster and then, in an iterative procedure, combines the two most similar groups according to a criterion to measure distances between groups. The most common ways to measure the distance among groups are single, complete, average and ward [42]. In this experiment, the last mentioned distance will be used. The result of the agglomerative clustering is a tree-based representation of all the unions, called a dendrogram. After that, a cut of the dendrogram is performed to obtain the desired number of groups.

The formula of this technique as shown in [43] to express the ward distance $D(w, v)$ between any two clusters u and v as,

$$D(u,v) = \sqrt{\frac{|v| + [s]}{T} D(v,s)^2 + \frac{|v| + [t]}{T} D(v,s)^2 - \frac{|v|}{T} D(v,s)^2} \tag{6}$$

Given a dataset that holds N elements, while s and t are the new pair of joined clusters, where $T = |v| + |s| + |t|$

Agglomerative clustering has shown excellent performance in different tasks and has performed comparatively superior to other algorithms [44].

B. BIRCH CLUSTERING

The BIRCH technique is another type of hierarchical clustering that can deal quickly and efficiently with large repositories. This algorithm creates a more diminutive representation of the large dataset that summarizes the large group, holding as much information as possible.

The BIRCH cluster is formed by creating a clustering feature tree composed between different nodes that are useful for calculating inter and intra distances of two clusters i and j. The centroid Euclidean distance can be calculated as

$$DO_{if} = \sqrt{\left(\left(\frac{LS_i}{N_i} - \frac{LS_j}{N_j}\right)^2\right)} \tag{7}$$

$$LS_i = \sum_{N}^{i=1} x_i \tag{8}$$

With N_i number of points, where x_i is a point in the cluster feature i. To calculate LS_j, the same procedure given for Equation 8 needs to performed.

This clustering technique has been used successfully in big data analysis [45], [46]. In addition, it has been used in finance to cluster investment recommendations [47], [48], risk management [49], text data mining [50], stock data prediction model [51] and customer segmentation [52].

C. K-MEANS CLUSTERING

K-means clustering is one of the most popular methods in machine learning to create clusters. It is a vector quantization method that seeks to split a dataset into k clusters. The goal is to set each observation in a specific group with the nearest mean to a cluster centroid. One crucial factor to consider when building a cluster is using a proper metric to measure the distance between all the data points to secure regularity and similitude among all the observations. There are two essential distance measures to consider when dealing with k-means; Euclidean (EUC) and Dynamic Time Warping (DTW).

EUC is a distance function that is popularly used when using the k-means technique, which is based on measuring the distance of a line segment between two points in Euclidean space. This function is represented in Equation 9, as the square root of the sum of the distance between vectors x and y where the algorithm aims to split different sample sets into dissociated clusters.

$$\sqrt{\sum_{j=1}^{k} (x_j - y_j)^2} \tag{9}$$

However, one of the main problems with EUC is that it is not a normalized metric, and in high dimensional spaces, it tends to be augmented; therefore, a previous reduction technique is needed to avoid the curse of dimensionality. K-means clustering has been applied in finance to classify stock performance [53], hedging strategies [54], and financial time series forecasting [55], [56].

Time series should receive special treatment for clustering because of the need to adapt the clustering techniques to time shifts. The DTW is a suitable measure to address this because it breaks the limitation of other one-to-one alignment metrics. This method attempts to discover all the possible paths using the dynamic programming technique, picking the one that renders the minimum distance between two series and building a matrix with the cumulative distances.

The DTW formula measures the distance between two time series x and y as,

$$DTW(x,y) = \min_{\pi} \sqrt{\sum_{(i,j)\in\pi} d(x_i, y_i)^2} \tag{10}$$

Where the path π satisfies additional boundary, continuity, and monotonicity constraints. Since this algorithm has been developed primarily to deal with time series, applications are focused on classifying these datasets. This algorithm has been used for application in finance for financial time series clustering [57]–[59] and financial credit analysis [60].

D. MINIBATCH CLUSTERING

The MNBT clustering technique is an alternative approach to the k-means algorithm aimed at reducing its spatial and temporal cost. However, despite the popularity of k-means, it has not proven memory efficient. Therefore, the primary technique of MNBT is to reduce the memory used by creating fixed-size, tiny random batches from the whole dataset in an iterative process, where a new random sample is obtained to update the cluster at each step. Nevertheless, due to the computational efficiency of this technique, this algorithm has been used in different big data [61], [62] and industrial [63] projects.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3359413

E. SPECTRAL CLUSTERING

The SPCT clustering technique is based on algebraic graph theory using information from the eigenvalues of matrices constructed from the graph. The resulting Laplacian matrix is defined as $L = D - A$, where A_{ij} measures the affinity between a data points x_i and x_j and D is

$$D = \sum_{j} A_{ij} \tag{11}$$

This method performs better than others in non-convex sample spaces by not quickly falling into the local optimum. This clustering algorithm has attracted attention due to its remarkable performance and reliable theoretical foundation, and has been widely used in finance [64], [65].

IV. PROPOSAL

This paper uses AEs as a dimensionality reduction technique to study their impact on six clustering techniques: AGGLO, BIRCH, KNN-EUC, KNN-DTW, MNBT, and SPCT [1]. The data utilized in this research is intraday data from IBEX, CAC, DAX, SPX, and UKX. This research aims to reduce the dimension of the financial time series into one price instead of four (the open, high, low, and closing price) to avoid the curse of dimensionality of the clustered object. Another objective is to find an ideal method that allows multidimensional financial data to deal with traditional clustering techniques. The purpose of the technique proposed in this paper is to convert the original 10-second price series into a more extensive time series, in this case into 1, 5, 15 and 30 minute time series, and to compare the performance of the AE to other state-of-the-art reduction techniques such as PCA, iPCA, and FF. We also want to investigate if the reduction techniques are effective without losing information by comparing them to the closing price, which represents an input without any reduction. The general scheme of the proposal is shown in Figure 3.

A. PROCEDURE

The time series is obtained from the Bloomberg terminal with a frequency of 10 seconds, the lowest available among the previously mentioned indices. The chosen dataset ranges from 01 September 2020 to 16 March 2021. Adding the five indices together, this research amounted to 1,874,423 observations. The empty values are filled with the closest previous value, assuming that if no price is available, the last known value is the current value; subsequently, each data set is normalized. An LSTM structure with a sigmoid activation function for both the encoding and decoding neural networks is applied to create the AE. The input size of the encoder network is equivalent to a three-dimensional matrix, which reflects the total number of observations, including the total intraday prices per day and the number of prices per instrument. Similarly, the encoder aims to generate a compressed data representation characterized by a three-dimensional matrix. The dimensions of this matrix represent the number of observations, the frequency of the data, and the value of a single price. The decoder, on the other hand, uses as input the compressed representation of the information generated by the encoder; it also uses as output a matrix of identical dimensions to the one used as input by the encoder. Once the decoder finishes decompressing the information it was fed, it then compares this information with the initial input of the encoder to measure the error by means of the mean square error (MSE) function. The formula of the MSE is shown in Equation 12, where n represents the number of observations, while y_i and $\hat{y}_i$, represent the observed and the predicted value, respectively.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i) \tag{12}$$

B. METRICS OF PERFORMANCE

Once the information has been compressed, it is pass to the AGGLO, BIRCH, KNN-EUC, KNN-DTW, MNBT, and SPCT cluster algorithms. Then, two performance metrics are applied to measure the effectiveness of these fitting techniques.

The first metric is the SilHouette coefficient (SH), which measures how well defined a cluster is, and it is defined as:

$$SH = \frac{r - s}{max(s, r)} \tag{13}$$

Where s are the mean distances of a sample and r are all other points of the same class and the next cluster. It should be noted that the SH is framed between the values of -1 and 1, where a negative value means an inaccurate clustering process. Furthermore, a zero implies overlapping clusters, and a value equal to 1 indicates a highly dense and well-separated cluster. Therefore the higher the SC, the better the model's performance.

The second metric, called Calinski and Harabasz (CALI), represents the average similarity between clusters and measures the ratio between the dispersion within and between the clusters, where compact and properly separated groups should maximize this ratio. It is specified as:

$$CALI = \frac{n_d - nk}{nk - 1} \cdot \frac{tr(B_g)}{tr(W_g)} \tag{14}$$

With n_d representing the number of elements of a dataset d and nk the number of clusters. While $tr(W_g)$ and $tr(B_g)$ represents the trace of the with-in cluster and between-group dispersion matrices, denoted as:

$$W_g = \sum_{j=1}^{g} \sum_{x \epsilon M_j} (x - m_j)(x - m_j)^T \tag{15}$$

$$B_g = \sum_{j=1}^{g} n_j (m_j - m_D)(m_j - m_D)^T \tag{16}$$

[1]In this work, ward AGGLO, BIRCH, MNBT, and SPCT techniques are used within the proposed framework, implemented by scikit-learn [66], while the library Tslearn [67] is utilised for KNN-DTW and KNN-EUC algorithm.

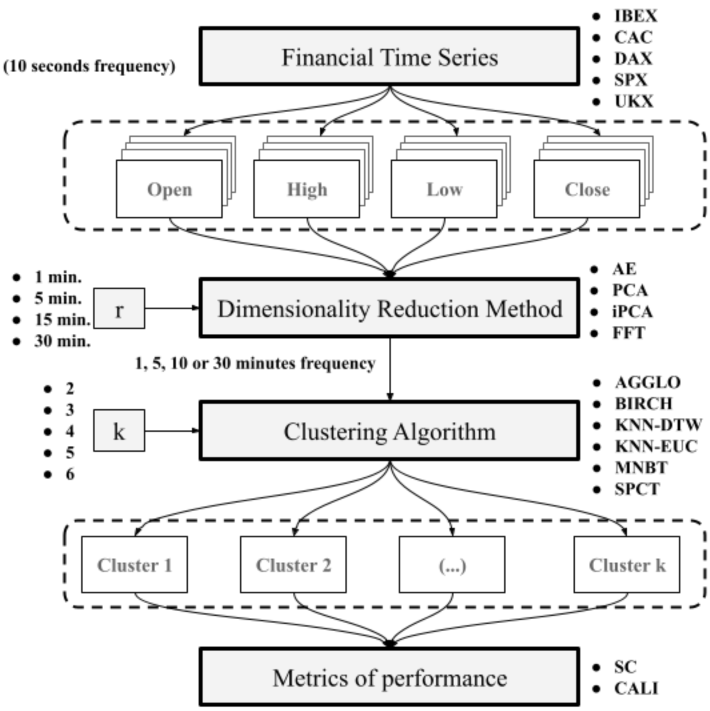

FIGURE 3. The general scheme of the proposal

The set of points in cluster j is defined as M_j and with cluster j and its center m_j. Where the center of D is defined as m_d and n_j is the number of observations in cluster j. A high CALI score implies a better performance of the cluster technique with well-separated and dense clusters.

V. EXPERIMENTATION

While the previous sections have described the techniques used to efficiently develop a reduction model for clustering data, this section will show our results on different data sets and measure them against the metrics described above. In Section IV, experimentation was carried out to validate the performance of the presented framework. We then analyzed the impact of autoencoders on six clustering techniques described below in Table 1, where each configuration has $k = \{2, 3, 4, 5, 6\}$ number of clusters. In addition, we compared this execution with CP, PCA, iPCA and FF, which are state-of-the-art reduction techniques previously explained in Section III.

Each reduction technique was tested on the following datasets: IBEX, CAC, DAX, SPX, and UKX, by converting the original 10-second price series into a reduced 1, 5, 15 and 30 minutes time series and was then tested by six different clustering algorithms. Each configuration was executed four times with different random seed numbers to validate the results statistically, with five different numbers of clusters and four epochs, obtaining a total number of 9600 experiments to analyze.

In Table 2, we can see the best results in terms of CALI and SH, given a particular index and a specific time frame. Moreover, it is possible to observe that the results vary according to the metric by which they are measured. On the one hand, we can see that, when using CALI, the reduction technique that concentrates the highest values is consistently given by the AE, except for the clustering of the UKX in 1 minute time series, where the reduction technique with better results turned out to be FF. In the same way, if we continue analyzing the results in terms of CALI, we see that the cluster techniques with the highest values do not focus specifically on specific algorithms but vary according to the index and the time frame, without a clear pattern. However, by observing the ideal number of clusters, denoted as k, we found that six clusters uniformly give the highest CALI values.

TABLE 1. Table with the principal configurations of the different reduction techniques

Cluster Technique	Parameters
AGGLO	Linkage = Ward Affinity = Euclidean
BIRCH	Threshold = 0.01 Branching factor = 50
KNN-DTW	Distance = DTW
KNN-EUC	Distance = Euclidean
MNBT	Initialization = k-means++
SPCT	Eigen solver = arpack

When analyzing the results in terms of SH, we do not necessarily see correlations with the best combinations measured by CALI. Instead, we find that the best-performing reduction

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3359413

TABLE 2. Results of different clustering techniques using multiple index data compressed by AE.

Exchange	Time Frame	Red_Cali	Cluster_Cali	k	CALI Score	Red_SH	Cluster_SH	k	SH Score
IBEX	1 minute	AE	MNBT	6	1673.31	PCA	AGGLO	2	0.8355
IBEX	5 minutes	AE	KNN-EUC	6	1871.78	PCA	AGGLO	2	0.8365
IBEX	15 minutes	AE	SPCT	6	1726.64	PCA	AGGLO	2	0.8388
IBEX	30 minutes	AE	KNN-EUC	6	1728.58	PCA	AGGLO	2	0.8418
CAC	1 minute	AE	SPCT	6	1163.60	PCA	KNN-EUC	2	0.8564
CAC	5 minutes	AE	MNBT	6	1182.68	PCA	KNN-EUC	2	0.8572
CAC	15 minutes	AE	SPCT	6	1185.16	PCA	KNN-EUC	2	0.8593
CAC	30 minutes	AE	KNN-EUC	6	1203.66	PCA	KNN-EUC	2	0.8620
DAX	1 minute	AE	MNBT	6	682.53	PCA	MNBT	2	0.7611
DAX	5 minutes	AE	MNBT	6	678.69	PCA	MNBT	2	0.7622
DAX	15 minutes	AE	SPCT	6	676.54	PCA	SPCT	2	0.7651
DAX	30 minutes	AE	MNBT	6	677.89	PCA	KNN-EUC	2	0.7736
SPX	1 minute	AE	SPCT	6	992.24	AE	MNBT	2	0.6591
SPX	5 minutes	AE	SPCT	6	1002.10	AE	KNN-EUC	2	0.6626
SPX	15 minutes	AE	SPCT	6	1008.60	AE	KNN-EUC	2	0.6628
SPX	30 minutes	AE	MNBT	6	1002.67	AE	KNN-EUC	2	0.6640
UKX	1 minute	FF	KNN-EUC	6	1167.09	AE	KNN-EUC	2	0.7549
UKX	5 minutes	AE	MNBT	6	1185.57	PCA	SPCT	2	0.7682
UKX	15 minutes	AE	SPCT	6	1213.87	AE	KNN-EUC	2	0.7605
UKX	30 minutes	AE	KNN-EUC	6	1232.10	AE	KNN-EUC	2	0.7680

and cluster techniques in a specific index are never the same if we compare the performance of the reduction and cluster combinations in terms of CALI and SH, with the exception of the cluster performed in a 30 minutes compressed representation of the UKX, where the highest values were found using AE and KKN-EUC.

Furthermore, by measuring the performance according to SH, it is possible to observe that in thirteen of the twenty combinations presented in Table 2, the highest value is given by PCA, followed by AE. The type of index seems to explain this change; for example, by clustering IBEX, CAC and DAX data, the highest values are found using PCA, while using SPX and UKX, the highest values are found by using AE. In the same way, the best cluster techniques are mainly differentiated by the type of data that they compress; for instance, the groups created in IBEX, with the highest values of SH, are formed using AGGLO techniques in the same way KNN-EUC is the predominant technique in those formed with CAC data. Additionally, similarly to those results measured with CALI, there is only one ideal value of k, and it takes the value of two.

To make an aggregate comparison of the different clusters, we have created Table 3, which counts the times that a given reduction technique performs better, the same and worse than the others by the results in percentage terms. In the lower part of Table 3, under the diagonal that represents the comparison of the same techniques, the results are presented in terms of CALI, while in the upper part, those are measured by SH. Table 4, which compares the different clustering techniques, is structured in the same way. In both tables, the first number represents the percentage by which the technique performed better or the same or worse than the others, respectively.

By revising the dissimilarities between the different reduction techniques shown in Table 3, we see that AE outperformed the others by a considerable percentage, especially in terms of CALI. Subsequently, the second-best reduction technique is iPCA, which performs better than CP, FFT and PCA, and finally, CP performs better than FFT and PCA in general percentage terms. Finally, when analyzing the results with respect to SH, we can see that they closely follow those presented above, where AE produces the best results, although to a lesser extent. Similarly, the second best technique is iPCA, followed by CP.

Table 4 shows a comparison similar to that of Table 3, however this table focuses on comparing the different cluster techniques instead of reduction techniques. It can be seen that AGGLO performs irregularly compared to the others; for example, 83 % of the time it is equal to BIRCH in terms of CALI and SH; however, it does not show any significant improvement compared to the other techniques. Also, BIRCH performs similarly to KNN-DTW but worse than KNN-EUC, MNBT and SPCT using CALI and to a lesser extent when using SH. On the other hand, the KNN-EUC technique outperforms all other techniques, both using CALI and to a lesser extent using SH; moreover, the MNBT technique ranks second as a clustering technique and, subsequently, the SPCT technique performs better than AGGLO, BIRCH and KNN-DTW in both metrics.

VI. CONCLUSIONS

Finding ways to reduce and classify the information in the financial markets is crucial. This research intends to contribute to these efforts to find a suitable technique that allows efficient classification of multidimensional financial data. In addition to proposing a novel hybrid technique for clustering financial time series using autoencoder-based compression and evaluating multiple clustering algorithms, the current paper contributes to the literature on unsupervised approaches

TABLE 3. Results of different clustering techniques using multiple index data compressed by AE.

	AE	CP	FFT	PCA	iPCA
AE	-	0.89, 0.0, 0.11	0.95, 0.0, 0.05	0.96, 0.0, 0.04	0.87, 0.0, 0.13
CP	0.24, 0.0, 0.76	-	0.95, 0.0, 0.05	0.95, 0.0, 0.05	0.17, 0.0, 0.83
FFT	0.0, 0.0, 1.0	0.0, 0.0, 1.0	-	0.59, 0.0, 0.41	0.05, 0.0, 0.95
PCA	0.0, 0.0, 1.0	0.0, 0.0, 1.0	0.31, 0.0, 0.69	-	0.05, 0.0, 0.95
iPCA	0.25, 0.0, 0.75	0.7, 0.0, 0.3	1.0, 0.0, 0.0	1.0, 0.0, 0.0	-

TABLE 4. Results of different clustering techniques using multiple index data compressed by AE.

	AGGLO	BIRCH	KNN-DTW	KNN-EUC	MNBT	SPCT
AGGLO	-	0.09, 0.83, 0.07	0.48, 0.1, 0.42	0.4, 0.09, 0.51	0.39, 0.09, 0.52	0.39, 0.1, 0.51
BIRCH	0.04, 0.83, 0.13	-	0.47, 0.1, 0.43	0.39, 0.1, 0.51	0.38, 0.1, 0.52	0.38, 0.1, 0.51
KNN-DTW	0.44, 0.1, 0.46	0.45, 0.1, 0.45	-	0.18, 0.3, 0.51	0.33, 0.14, 0.53	0.34, 0.14, 0.52
KNN-EUC	0.7, 0.09, 0.2	0.71, 0.1, 0.19	0.61, 0.3, 0.08	-	0.42, 0.18, 0.4	0.44, 0.18, 0.39
MNBT	0.67, 0.09, 0.23	0.68, 0.1, 0.22	0.6, 0.14, 0.26	0.31, 0.18, 0.51	-	0.43, 0.16, 0.41
SPCT	0.66, 0.1, 0.24	0.67, 0.1, 0.23	0.59, 0.14, 0.27	0.32, 0.18, 0.5	0.42, 0.16, 0.43	-

that deal with the difficulties of processing high-dimensional data [68], [69] and time series classification [70], [71]. Furthermore, it is pertinent and key to discover how algorithms can help stakeholders automate financial research that involves high volatility and large volumes of data, especially during short and fast periods, such as the intraday sessions of the stock market.

In this investigation, different reduction techniques were used; we tested the suitability of machine and deep learning algorithms, showing the efficiency of AE in intraday financial time series. In addition, this study demonstrated that AE outperformed other reduction techniques by allowing the creation of more compact and well-separated clusters by testing our techniques on five different indexes and four different time frames, contributing to the clustering intraday financial time series literature [72]. Thus, those interested in developing strategies based on information obtained from intraday time series should apply AE with another clustering technique rather than just using traditional techniques such as PCA, or relying only on the closing price.

Despite the proven efficiency and superiority of AE and KNN-EUC, it is possible to observe diffuse results when comparing the results of the clustering techniques with different exchanges and time frames. Furthermore, there may be specific cases where the maximum values found do not necessarily coincide with the averages; for example, the highest values in terms of SH for certain exchanges and time frames were found using PCA, even when, on average overall, EA proved to be the most successful technique by a wide margin. Nevertheless, this study establishes the suitability of AE to deal with multidimensional intraday data; therefore, future research can only focus on the search for the most suitable clustering technique for time series using compressed data from AE.

By measuring the performance of cluster techniques with reduced data, we can clearly understand the practical use of reduction techniques such as AE and its superiority over other methods. Furthermore, acknowledging these processes allows us to investigate larger volumes of information, expanding research toward models capable of integrating multiple and diverse variables, and creating the opportunity to expand the analysis capacity of investors and scholars. As such, AE can not only be used to decrease the dimensionality of intraday prices but also to reduce large datasets covering different prices from diverse instruments and markets. Based on our investigation, we can use reduction techniques to develop further investment strategies that determine different classifications to simulate more realistic scenarios, which is part of our future research.

REFERENCES

[1] H. Njah, S. Jamoussi, and W. Mahdi, "Breaking the curse of dimensionality: hierarchical bayesian network model for multi-view clustering," *Annals of Mathematics and Artificial Intelligence*, vol. 89, no. 10, pp. 1013–1033, 2021.

[2] X. Yao, J. Wang, M. Shen, H. Kong, and H. Ning, "An improved clustering algorithm and its application in iot data analysis," *Computer Networks*, vol. 159, pp. 63–72, 2019.

[3] C. Yin, S. Zhang, J. Wang, and N. N. Xiong, "Anomaly detection based on convolutional recurrent autoencoder for iot time series," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 1, pp. 112–122, 2022.

[4] W. Wang, A. Ramesh, J. Zhu, J. Li, and D. Zhao, "Clustering of driving encounter scenarios using connected vehicle trajectories," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 3, pp. 485–496, 2020.

[5] S. Kaplan Berkaya, A. K. Uysal, E. Sora Gunal, S. Ergin, S. Gunal, and M. B. Gulmezoglu, "A survey on ecg analysis," *Biomedical Signal Processing and Control*, vol. 43, pp. 216–235, 2018.

[6] J. Oyelade, I. Isewon, F. Oladipupo, O. Aromolaran, E. Uwoghiren, F. Ameh, M. Achas, and E. Adebiyi, "Clustering algorithms: Their application to gene expression data," *Bioinformatics and Biology Insights*, vol. 10, p. BBI.S38316, 2016.

[7] F. Cai, N.-A. Le-Khac, and T. Kechadi, "Clustering approaches for financial data analysis: a survey," 2016.

[8] K. Kim and J. W. Song, "Analyses on volatility clustering in financial time-series using clustering indices, asymmetry, and visibility graph," *IEEE Access*, vol. 8, pp. 208 779–208 795, 2020.

[9] S. Aghabozorgi, A. Seyed Shirkhorshidi, and T. Ying Wah, "Time-series clustering – a decade review," *Information Systems*, vol. 53, pp. 16–38, 2015.

[10] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, Jul. 2019.

[11] M. Qaraei, S. Abbaasi, and K. Ghiasi-Shirazi, "Randomized non-linear pca networks," *Information Sciences*, vol. 545, pp. 241–253, 2021.

[12] P. Baldi, "Autoencoders, unsupervised learning and deep architectures," in *Proceedings of the 2011 International Conference on Unsupervised and*

Transfer Learning Workshop - Volume 27, ser. UTLW'11. JMLR.org, 2011, p. 37–50.

[13] N. Le Roux and Y. Bengio, "Representational power of restricted boltzmann machines and deep belief networks," *Neural computation*, vol. 20, no. 6, pp. 1631–1649, 2008.

[14] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504 – 507, 2006.

[15] S. Gu, B. Kelly, and D. Xiu, "Autoencoder asset pricing models," *Journal of Econometrics*, 2020.

[16] R. Li, S. Li, K. Xu, X. Li, J. Lu, and M. Zeng, "A novel symmetric stacked autoencoder for adversarial domain adaptation under variable speed," *IEEE Access*, vol. 10, pp. 24 678–24 689, 2022.

[17] H. Fanai and H. Abbasimehr, "A novel combined approach based on deep autoencoder and deep classifiers for credit card fraud detection," *Expert Systems with Applications*, vol. 217, p. 119562, 2023.

[18] G. Eraslan, L. M. Simon, M. Mircea, N. S. Mueller, and F. J. Theis, "Single-cell rna-seq denoising using a deep count autoencoder," *Nature communications*, vol. 10, no. 1, pp. 1–14, 2019.

[19] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Artificial Neural Networks and Machine Learning – ICANN 2011*, T. Honkela, W. Duch, M. Girolami, and S. Kaski, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 52–59.

[20] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *CoRR*, vol. abs/1409.3215, 2014.

[21] G. Dong, G. Liao, H. Liu, and G. Kuang, "A review of the autoencoder and its variants: A comparative perspective from target recognition in synthetic-aperture radar images," *IEEE Geoscience and Remote Sensing Magazine*, vol. 6, no. 3, pp. 44–68, 2018.

[22] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1544–1551, 2018.

[23] M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "Deep learning approach combining sparse autoencoder with svm for network intrusion detection," *IEEE Access*, vol. 6, pp. 52 843–52 856, 2018.

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, p. 1735–1780, 1997.

[25] G. Jung and S.-Y. Choi, "Forecasting foreign exchange volatility using deep learning autoencoder-lstm techniques," *Complexity*, vol. 2021, 2021.

[26] S. Maleki, S. Maleki, and N. R. Jennings, "Unsupervised anomaly detection with lstm autoencoders using statistical data-filtering," *Applied Soft Computing*, vol. 108, p. 107443, 2021.

[27] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[28] A. Krishn, V. Bhateja, A. Sahu *et al.*, "Medical image fusion using combination of pca and wavelet analysis," in *2014 international conference on advances in computing, communications and informatics (ICACCI)*. IEEE, 2014, pp. 986–991.

[29] D. Zhang, X. Zhao, J. Han, and Y. Zhao, "A comparative study on pca and lda based emg pattern recognition for anthropomorphic robotic hand," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 4850–4855.

[30] M. Fuentes-García, G. Maciá-Fernández, and J. Camacho, "Evaluation of diagnosis methods in pca-based multivariate statistical process control," *Chemometrics and Intelligent Laboratory Systems*, vol. 172, pp. 194–210, 2018.

[31] M. Ghorbani and E. K. Chong, "Stock price prediction using principal components," *Plos one*, vol. 15, no. 3, p. e0230124, 2020.

[32] X. Zhong and D. Enke, "Forecasting daily stock market return using dimensionality reduction," *Expert Systems with Applications*, vol. 67, pp. 126–139, 2017.

[33] H. Yu, R. Chen, and G. Zhang, "A svm stock selection model within pca," *Procedia Computer Science*, vol. 31, pp. 406–412, 2014, 2nd International Conference on Information Technology and Quantitative Management, ITQM 2014.

[34] G. Pasini, "Principal component analysis for stock portfolio management," *International Journal of Pure and Applied Mathematics*, vol. 115, no. 1, pp. 153–167, 2017.

[35] P. K. Narayan, S. Narayan, and K. S. Thuraisamy, "Can institutions and macroeconomic factors predict stock returns in emerging markets?" *Emerging Markets Review*, vol. 19, pp. 77–95, 2014.

[36] V. Gupta and M. Mittal, "A comparison of ecg signal pre-processing using frft, frwt and ipca for improved analysis," *IRBM*, vol. 40, no. 3, pp. 145–156, 2019.

[37] J. Bouhlel, D. Jouan-Rimbaud Bouveresse, S. Abouelkaram, E. Baéza, C. Jondreville, A. Travel, J. Ratel, E. Engel, and D. N. Rutledge, "Comparison of common components analysis with principal components analysis and independent components analysis: Application to spme-gc-ms volatolomic signatures," *Talanta*, vol. 178, pp. 854–863, 2018.

[38] Y. Zhu, C. Zhu, and X. Li, "Improved principal component analysis and linear regression classification for face recognition," *Signal Processing*, vol. 145, pp. 175–182, 2018.

[39] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[40] A. Vyas, S. Yu, and J. Paik, "Fourier analysis and fourier transform," in *Multiscale Transforms with Application to Image Processing*. Springer, 2018, pp. 15–43.

[41] M. A. Mahdi, K. M. Hosny, and I. Elhenawy, "Scalable clustering algorithms for big data: A review," *IEEE Access*, vol. 9, pp. 80 015–80 027, 2021.

[42] D. Müllner, "fastcluster: Fast hierarchical, agglomerative clustering routines for r and python," *Journal of Statistical Software*, vol. 53, pp. 1–18, 2013.

[43] E. K. Tokuda, C. H. Comin, and L. da F. Costa, "Revisiting agglomerative clustering," *Physica A: Statistical Mechanics and its Applications*, vol. 585, p. 126433, 2022.

[44] X. Wu, T. Ma, J. Cao, Y. Tian, and A. Alabdulkarim, "A comparative study of clustering ensemble algorithms," *Computers Electrical Engineering*, vol. 68, pp. 603–615, 2018.

[45] F. Ramadhani, M. Zarlis, and S. Suwilo, "Improve birch algorithm for big data clustering," in *IOP Conference Series: Materials Science and Engineering*, vol. 725, no. 1. IOP Publishing, 2020, p. 012090.

[46] W. Li, H. Li, and Y. Luo, "Dynamic and static enhanced birch for functional data clustering," *IEEE Access*, vol. 11, pp. 111 448–111 465, 2023.

[47] W. Luo, "Application of improved clustering algorithm in investment recommendation in embedded system," *Microprocessors and Microsystems*, vol. 75, p. 103066, 2020.

[48] Y. Huang, "Financial investment recommendation in coastal areas based on improved clustering algorithm," *Journal of Coastal Research*, vol. 110, no. SI, pp. 215–218, 2020.

[49] H. Xiao, "Birch algorithm and data management in financial enterprises based on dynamic panel gmm test," *Cluster Computing*, vol. 22, no. 2, pp. 4231–4237, 2019.

[50] C. Wang, Z. Miao, Y. Lin, and J. Gao, "User and topic hybrid context embedding for finance-related text data mining," in *2019 International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2019, pp. 751–760.

[51] Y. Patil and M. Joshi, "Cluster driven candlestick method for stock market prediction," in *2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*. IEEE, 2020, pp. 1–5.

[52] G. Sengodan, "Customer segmentation using mobile phone usage data to reveal finance application user's behavior," in *2021 IEEE International Conference on Big Data (Big Data)*. IEEE, 2021, pp. 4578–4583.

[53] A. Phongmekin and P. Jarumaneeroj, "Classification models for stock's performance prediction: A case study of finance sector in the stock exchange of thailand," in *2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST)*. IEEE, 2018, pp. 1–4.

[54] M. Sun, Y. Zhang, R. Chen, Y. Wen, and P. Nie, "Hedging strategy for commodity futures based on svm-knn," *International Journal of Reasoning-based Intelligent Systems*, vol. 13, no. 3, pp. 139–146, 2021.

[55] H. Jiang, "Cryptocurrency price forecasting based on shortterm trend knn model," in *2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*. IEEE, 2021, pp. 1165–1169.

[56] G. Lin, A. Lin, and J. Cao, "Multidimensional knn algorithm based on eemd and complexity measures in financial time series forecasting," *Expert Systems with Applications*, vol. 168, p. 114443, 2021.

[57] P. E. Puspita *et al.*, "A practical evaluation of dynamic time warping in financial time series clustering," in *2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. IEEE, 2020, pp. 61–68.

[58] H. Ito, A. Murakami, N. Dutta, Y. Shirota, B. Chakraborty *et al.*, "Clustering of etf data for portfolio selection during early period of corona virus outbreak," *Gakushuin Hournal of Economics*, vol. 58, no. 1, pp. 99–114, 2021.

[59] Y. Shirota and A. Murakami, "Long-term time series data clustering of stock prices for portfolio selection," in *2021 IEEE International Confer-*

ence on Service Operations and Logistics, and Informatics (SOLI). IEEE, 2021, pp. 1–6.

[60] J. Sun, Y. Li, Q. Li, Y. Li, Y. Jia, and D. Xia, "Fine clustering analysis of internet financial credit investigation based on big data," *Big Data Research*, vol. 27, p. 100297, 2022.

[61] K. Peng, V. C. Leung, and Q. Huang, "Clustering approach based on mini batch kmeans for intrusion detection system over big data," *IEEE Access*, vol. 6, pp. 11 897–11 906, 2018.

[62] R. Tang and S. Fong, "Clustering big iot data by metaheuristic optimized mini-batch and parallel partition-based dgc in hadoop," *Future Generation Computer Systems*, vol. 86, pp. 1395–1412, 2018.

[63] S. Messaoud, A. Bradai, and E. Moulay, "Online gmm clustering and mini-batch gradient descent based optimization for industrial iot 4.0," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1427–1435, 2019.

[64] I. Aldridge and M. Avellaneda, *Big data science in finance*. John Wiley & Sons, 2021.

[65] R. E. Mansano, L. E. Allem, R. R. Del-Vecchio, and C. Hoppen, "Balanced portfolio via signed graphs and spectral clustering in the brazilian stock market," *Quality & Quantity*, pp. 1–16, 2021.

[66] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[67] R. Tavenard, J. Faouzi, G. Vandewiele, F. Divo, G. Androz, C. Holtz, M. Payne, R. Yurchak, M. Rußwurm, K. Kolar, and E. Woods, "Tslearn, a machine learning toolkit for time series data," *Journal of Machine Learning Research*, vol. 21, no. 118, pp. 1–6, 2020.

[68] L. Zhang, J. Lin, and R. Karim, "Sliding window-based fault detection from high-dimensional data streams," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 2, pp. 289–303, 2017.

[69] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 1, pp. 136–144, 2019.

[70] G. He, X. Xin, R. Peng, M. Han, J. Wang, and X. Wu, "Online rule-based classifier learning on dynamic unlabeled multivariate time series data," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 2, pp. 1121–1134, 2022.

[71] C.-L. Liu, W.-H. Hsaio, and Y.-C. Tu, "Time series classification with multivariate convolutional neural network," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 6, pp. 4788–4797, 2019.

[72] Y. Shi, B. Li, G. Du, and W. Dai, "Clustering framework based on multi-scale analysis of intraday financial time series," *Physica A: Statistical Mechanics and its Applications*, vol. 567, p. 125728, 2021.

DANIEL GONZÁLEZ CORTÉS is a Research Fellow at NEOMA Business School and a Ph.D. candidate at NEOMA and the University of Deusto. His research fuses Finance with Machine Learning and Artificial Intelligence, specializing in solving financial problems with Deep Learning, Reinforcement learning, and Explainable models. His interests are interdisciplinary research in Computer Science, Information Systems, Finance, and Management.

ENRIQUE ONIEVA received his B.E. degree in computer science engineering, his M.E. degree in soft computing and intelligent systems, and his Ph.D. in computer science from the University of Granada, Spain, in 2006, 2008, and 2011, respectively. From 2007 to 2012, he has been with the AUTOPIA Program at the Centre of Automation and Robotics, Consejo Superior de Investigaciones Científicas, Madrid, Spain. In 2012, he joined the Models of Decision and Optimization group at the University of Granada. Since 2013, he has been a professor in artificial intelligence at the University of Deusto and a Researcher of Intelligent Transportation Systems applications within the Deusto Smart Mobility Research Unit.

He has participated in more than 40 research projects. Among them are CYBERCARS-2 (FP6), ICSI (FP7), and PostLowCit (CEF-Transport). Research responsible for the Artificial Intelligence Work Package of Project TIMON (H2020) and Project Coordinator of the LOGISTAR Project (H2020). He has authored more than 100 scientific articles. From them, more than 40 are published in journals of the highest level. His research interest is based on applying Artificial Intelligence to Intelligent Transportation Systems, including fuzzy-logic-based decisions, evolutionary optimization, and machine learning.

IKER PASTOR LÓPEZ received a degree in computer engineering in 2007, a master's degree in information security in 2010, and a Ph.D. degree (cum laude) in computer science in 2013. He participated in the Program in Big Data and Business Intelligence, in 2016. He is with Deusto University and focuses its scientific interests on the areas of big data analytics, opinion mining, and computer vision. He is the author of several scientific articles reviewed by peers in conferences and indexed journals. He has participated in the gestation, scientific development, and technical development of numerous competitive projects and contracts with companies, the latter with several successful cases of knowledge transfer actions. In addition, he is a member of the Scientific Committee of several congresses, such as CISIS, SOCO, and ICEUTE. He is a Reviewer of many journals, included in the JCR as the magazine of Engineering and Industry–DYNA.

LAURA TRINCHERA is an Associate Professor of Statistics, NEOMA Business School. Holds a Master's degree in Economic and Business Sciences (2004) and a Ph.D. in Statistics (2008) from the University of Naples Federico II, Italy. Her research focuses on Data Sciences and Statistical Learning methods, with a focus on Structural Equation Modeling, PLS Methods, and classification algorithms. Her research has been published in internationally recognized journals such as Structural Equation Modeling: A Multidisciplinary Journal, Journal of Production Economics, Journal of Organizational Behavior, Recherche et Applications en Marketing, International Journal of Information Management and Management Decision. Also contributed to the Handbook of Partial Least Squares: Concepts, Methods and Applications. Visiting researcher at the University of California Santa Barbara, the University of Michigan at Ann Arbor, the University of Hamburg, the Charles University of Prague, and HEC Paris. Guest lecturer at ESSEC Business School, Sciences Po Paris, and Sorbonne University in Abu Dhabi.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3359413

JIAN WU holds a Ph.D. from the University of Paris Dauphine and has vast experience teaching Investments, Financial Risk Management, Sustainable Finance, Economic Environment, and its impact on Financial Markets in initial training and executive education courses. Additionally, she served over a decade as head of the Finance and Economics Department at NEOMA Business School. She has published her research work in journals such as Finance, International Review of Financial Analysis, Economic Bulletin, and Journal of Business Ethics. Her research interests are Financial Engineering, Corporate Governance, Banking Regulation, and Corporate Social Responsibility.