Computational Management Science (2023) 20:28
https://doi.org/10.1007/s10287-023-00460-4

ORIGINAL PAPER

Optimization of mixture models on time series networks encoded by visibility graphs: an analysis of the US electricity market

Carlo Mari[1] · Cristiano Baldassari[2]

Received: 14 December 2022 / Accepted: 22 May 2023

Abstract
We propose a fully unsupervised network-based methodology for estimating Gaussian Mixture Models on financial time series by maximum likelihood using the Expectation-Maximization algorithm. Visibility graph-structured information of observed data is used to initialize the algorithm. The proposed methodology is applied to the US wholesale electricity market. We will demonstrate that encoding time series through Visibility Graphs allows us to capture the behavior of the time series and the nonlinear interactions between observations well. The results reveal that the proposed methodology outperforms more established approaches.

Keywords Visibility graph · Markov transition field · Graph embedding · Graph machine learning · Topological data analysis

1 Introduction

Models based on complex networks have been extensively introduced in the literature to provide new insights into a wide range of natural and social phenomena (Vespignani 2018; Xie et al. 2021; Laengle et al. 2021). In particular, a plethora of network science-based methods have been published over the past decade to analyze the dynamic behavior of time series with the aim of understanding

Carlo Mari and Cristiano Baldassari contributed equally to this work.

✉ Cristiano Baldassari
cristiano.baldassari@unich.it

Carlo Mari
carlo.mari@unich.it

[1] Department of Economics, University of Chieti-Pescara, Viale Pindaro, 42, Pescara 65100, PE, Italy

[2] Department of Neuroscience, Imaging and Clinical Sciences, University of Chieti-Pescara, Via Luigi Polacchi, 11, Chieti 66100, CH, Italy

Published online: 30 May 2023

their fine and granular structure (Newman 2003, 2010). Indeed, converting a time series into a network increases the quality of the analysis, leading to the identification of non-trivial topological characteristics (Yang and Yang 2008; da Fontoura Costa et al. 2005) and providing fresh perspectives and ideas for investigating the probabilistic structure of time series (Zou et al. 2019; Silva et al. 2021). In this regard, network-based methods can be very useful in calibrating realistic models of time series dynamics on market data such as Gaussian Mixture Models (GMMs) (Mari and Baldassari 2022). The estimation of GMMs can be performed by maximum likelihood by using the Expectation-Maximization (EM) algorithm (Dempster et al. 1977). However, the existence of several solutions that are locally optimal requires the choice of appropriate initialization values of the parameters to obtain accurate results (Hipp and Bauer 2006; Shireman et al. 2017). Clustering methods are typically used in the literature to solve the EM initialization problem. Techniques based on *K*-means (Steinley and Brusco 2011) and Random algorithms (Biernacki et al. 2003) have recently garnered interest. Both of these approaches require the exogenous setting of the number of mixture components and, due to the large number of local solutions, the estimate technique must be re-initialized several times before a solution can be found. No method can determine how many initializations are needed to fully explore the likelihood function (Shireman et al. 2017).

Recently, a fully unsupervised framework based on mapping the time series into a complex network (or graph) has been proposed in the literature to determine the ideal number of mixture model components and the vector of initial parameters with the aim of improving the efficiency of the EM algorithm (Mari and Baldassari 2022). By using a Graph Approximation framework, hereinafter GA-framework, the suggested methodology effectively addresses the local-optima issue and provides reliable GMM parameter estimation.

Graph approximation techniques suffer from a loss of information from the observed time series that can affect the goodness of the estimation procedure. To overcome this difficulty, we propose a novel approach in this paper based on a different mapping scheme that can significantly reduce information loss and improve the estimation procedure. It is based on graph embedding techniques (Perozzi et al. 2014), and for we will therefore call it the Graph Embedding framework, hereinafter GE-framework. The relevance of the GE-framework over the GA-framework is due to the embedding techniques that enables us to work directly on a *not reduced* graph. In fact, graph embedding techniques convert the graph into a multidimensional Euclidean space using a data-driven approach and can significantly reduce information loss while maintaining the structural features of the graph (Cai et al. 2021). The time series encoding is performed using Visibility Graphs (Silva et al. 2021). These encoding techniques are based on the traditional visibility algorithms of computational geometry (Ghosh 2007), and allow univariate time series to be mapped into complex networks, thus focusing on the underlying structural features of the time series (Lacasa et al. 2008). We found them to be particularly suitable for identifying relevant features of the observed temporal dynamics. The proper combination of unsupervised techniques used to perform the initialization of the EM algorithm in the GE-framework is the main innovative aspect of the proposed methodology.

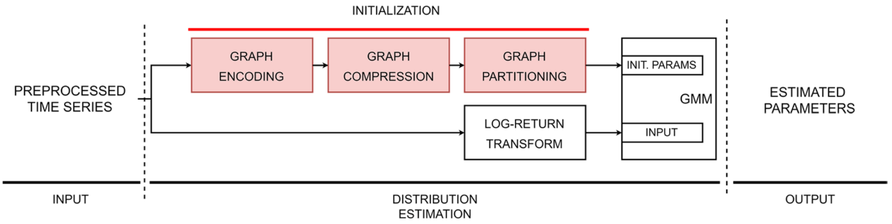

Fig. 1 End-to-end workflow

The performance of the GA and GE frameworks will be compared through GMM estimation on the time series of financial log-returns, which are calculated as the difference in log-prices between two subsequent observations. In fact, log-return time series display better statistical behavior than market price time series that are generally non-stationary (Voit 2013). In particular, we tested both the GA and GE frameworks in the US wholesale electricity market by analyzing the behavior of daily log-returns observed at Palo Verde (Southwest area), PJM (Northeast region), SP15 (Southern California) and Nepool (New England).

Due to its conservative pipeline (less information loss in the Graph Compression phase), superior data-driven and unsupervised Graph Machine Learning approaches, the GE-framework appears to be a powerful and flexible tool for analyzing the dynamics of time series. Although the GA-framework is able to outperform the *K*-means and Random-based initialization methods (Mari and Baldassari 2022), in this paper we will demonstrate that the GE-framework produces even better results. In fact, we will prove that the best-fitting model in the GE-framework is several times (from 2.14 for SP15 to 250.89 for PJM) more likely to be the best model than the best-fitting model in the GA-framework, and that, compared with the *K*-means approach, the results are far better in each market under consideration.

The rest of the present paper is structured as follows. The GE-framework is detailed in Sect. 2. Section 3 focuses on the empirical analysis. Some concluding remarks are provided in Sect. 4. The code needed to reproduce our results can be accessed on Github at https://bit.ly/3Hz1Dky.

2 The GE-framework

This section illustrates the Graph Machine Learning approach proposed in this paper to estimate GMMs by maximum likelihood by outlining the methods involved in the GE-framework in detail and highlighting the novelty of our study.

2.1 An overview of the methodology

Figure 1 shows a unified diagram of the entire methodology.

Three sections can be identified in the workflow: Input, Distribution Estimation and Output. The input consists of a preprocessed time series of log-prices. Very often, in fact, time series feature irregular sampling (lack of daily data points) as well as trend and seasonality. For these reasons, preprocessing, which consists of a gap-filling procedure and a seasonal-trend removal, is carried out first. Then, the preprocessed log-price time series enters the initialization blocks (represented in red in Fig. 1). Through a three-step sequence, namely Graph Encoding, Graph Compression and Graph Partitioning, the number of GMM components and the vector of initialization parameters are determined in a completely unsupervised manner. First, the graph associated with the preprocessed log-price time series is created in the graph encoding step; then, the complexity of the graph is appropriately reduced by compressing the graph in the graph compression step; finally the log-price communities and their associated log-return communities are discovered in the graph partitioning step. Thus, the EM algorithm can be the initialized in a completely unsupervised way. In fact, the number of GMM components is set equal to the number of discovered communities, and the initializing parameters of the EM algorithm can be computed by log-return community membership (Mari and Baldassari 2022). The GMM parameters are estimated in the output section.

The main innovative aspect of the proposed approach consists in the proper combination of unsupervised techniques used to perform the initialization of the EM algorithm. First, the graph encoding step is performed using Visibility Graphs (Silva et al. 2021). Second, the graph compression is performed by using graph embedding techniques belonging to the four classes of the Karate Club framework (Rozemberczki et al. 2020). We used 12 different embedding techniques to identify the one that best fits each market. Third, the community detection task is performed in the Graph Partitioning step directly on the network embedding, using a clustering approach based on Topological Data Analysis (TDA) (Skrlj et al. 2020). Embedding techniques and TDA methods improve the accuracy and efficiency of feature extraction (Cai et al. 2021).

2.2 Visibility graphs

Visibility graph mappings have been proposed in the literature as transformations from time series to complex networks based on the conventional visibility techniques of computational geometry. In the empirical analysis, we will use Natural Visibility algorithms (Lacasa et al. 2008) and Horizontal Visibility algorithms (Luque et al. 2009). These algorithms are easy to use, computationally fast and parameter-free.

The concept behind the Natural Visibility Graph (NVG) is that each observation in the time series is associated with a node in the graph and represented as a vertical bar with a height equal to the numerical value of the observation. The nodes in the graph are serially ordered since each one corresponds to a time stamp t of the time series. Two nodes are connected if a line of visibility that is not interrupted by the bar of a node between the two exists between the corresponding data bars. Stated mathematically, two nodes, v_i and v_j, are connected (have visibility) if any additional observation x_k with $i < k < j$ satisfies the following NVG algorithm,

$$x_k < x_j + \frac{j-k}{j-i}(x_i - x_j). \tag{1}$$

The NVG algorithm is simple to use and has a quadratic computational complexity $\mathcal{O}(n^2)$, making it rather slow for very long time series. However, a more efficient algorithm based on the *divide et conquer* technique (Lanet al. 2015) can be used and, in this case, the computational complexity is reduced to $\mathcal{O}(n \log n)$. Since visibility graphs are always undirected, and each node v_i can see at least two of its neighbors, namely v_{i-1} and v_{i+1}, they are always connected. NVGs take on many of the structural characteristics of time series and, for this reason, seem particularly suited to our study.

A simpler technique, known as the Horizontal Visibility Graph (HVG), has been presented in the literature to reduce the computing complexity associated with NVGs. The difference between the construction of NVGs and HVGs is that in the latter case the visibility lines are only horizontal. Stated mathematically, two nodes v_i and v_j are connected if the following condition is met,

$$x_k < x_i, x_j, \tag{2}$$

for all x_k such that $i < k < j$. In terms of computational complexity, the building of HVGs has a computational complexity of $\mathcal{O}(n \log n)$ (Yela et al. 2020). For a specific time series, the HVG is always a subgraph of the NVG, but the converse is not true. As a result, some quantitative information is lost in HVGs when compared to NVGs.

2.3 Graph embedding

Network embedding methods have made significant contributions to the use of Machine Learning (ML) in network science (Cui et al. 2018). Based on unsupervised feature extraction techniques from graph data, these methods yield information that can be used as input for link prediction, node and graph classification, and community detection. Therefore, they seem to be particularly suitable for our purposes. In particular, node embeddings map graph vertices to vectors in a Euclidean space, and the application of the Euclidean format, instead of the native graph, facilitates the use of conventional ML tools. In this paper we will adopt graph embedding methods belonging to the four classes of the Karate Club framework (Rozemberczki et al. 2020). To ensure generality and completeness, we will use one method chosen from each class: (i) Diff2Vec as a neighborhood preserving embedding; (ii) GraphWave as a structural embedding; (iii) Attributed Social Network Embedding (ASNE); (iv) Network Embedding Update (NEU) as a meta-embedding method. While neighborhood preserving embeddings preserve the closeness of graph nodes, structural embeddings preserve the structural roles of nodes in the embedding space, and attributed embeddings maintain the neighborhood, the structure and the generic attribute similarity of nodes. Meta-embeddings are designed to produce embeddings with a higher representation quality.

2.4 ToMATo clustering

Since the graph embedding step generates a high dimensional Euclidean representation of the graph, we chose a TDA-based approach, which is an emerging field of research, with the goal of providing mathematical and algorithmic tools for understanding the topological and geometric structure of data, that is particularly suitable for high-dimensional data (Zomorodian and Carlsson 2005). The community discovery process is performed directly on the graph embedding using the Topological Mode Analysis Tool (ToMATo) (Skrlj et al. 2020), an unsupervised TDA tool that allows us to identify clusters that are stable under small perturbations of the input (Chazal et al. 2013).

2.5 The GMM estimation method

Consider a possibly preprocessed log-price time series $\{x_t\}_{t=1}^{N}$ and its log-return transform $\{r_t\}_{t=1}^{N-1}$, where

$$r_i = x_{i+1} - x_i, \qquad i = 1, 2, \cdots, N-1. \tag{3}$$

Let us suppose that the values, r_i, $i = 1, 2, \cdots, N-1$, are extracted from a sequence of independent and identically distributed (iid) random variables with a probability density $p(x)$. We assume that $p(x)$ is a finite mixture distribution with C components,

$$p(r_i \mid \Theta) = \sum_{c=1}^{C} \alpha_c p_c(r_i \mid z_{ic} = 1, \theta_c). \tag{4}$$

In Eq. (4), $z_i = \{z_{i1}, z_{i2} \cdots, z_{iC}\}$ with $i = 1, 2, \cdots, N-1$, is a vector of C unobservable latent binary random variables that are mutually exclusive and exhaustive, i.e., one and only one of the z_{ic} is equal to 1, while the others are equal to 0. The vector z_i plays the role of an indicator random variable representing the identity of the mixture component responsible for the generation of the outcome r_i; $p_c(r_i \mid z_{ic} = 1, \theta_c)$ denotes the density distributions of the mixture components with parameters θ_c; finally, $\alpha_1, \alpha_2, \cdots, \alpha_C$ are the mixture weights, i.e., positive numbers such that

$$\sum_{c=1}^{C} \alpha_c = 1, \tag{5}$$

representing the probability that the value x_i was generated by the component c. The complete set of the mixture model parameters is denoted by $\Theta = \{\alpha_1, \cdots, \alpha_C, \theta_1, \cdots, \theta_C\}$. In the case of univariate Gaussian mixture models, the components of the model are described by univariate Gaussian densities,

$$p_c(r \mid \theta_c) = \frac{1}{\sigma_c \sqrt{2\pi}} \exp\left(\frac{-(r - \mu_c)^2}{2\sigma_c^2} \right), \tag{6}$$

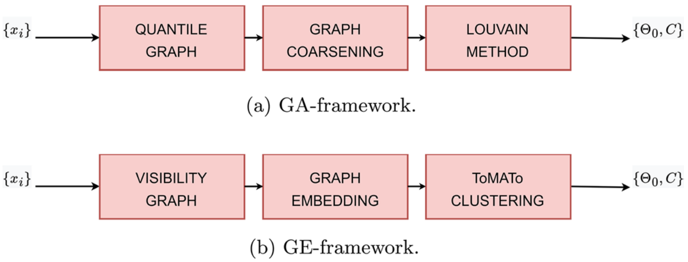

Fig. 2 The initialization blocks

with parameters $\theta_c = \{\mu_c, \sigma_c^2\}$ denoting, respectively, the mean and the variance of the single component distribution.

Mixture models can be efficiently estimated by maximum likelihood using the EM algorithm (Dempster et al. 1977). EM is an iterative procedure that begins with an initial estimate of parameters $\Theta = \Theta_0$ and then iteratively updates Θ until convergence is achieved (Smyth 2017).

The initial parameter vector, Θ_0, and the number of mixture components, C, are determined by performing the various processes involved in the three initialization blocks depicted in Fig. 1. The specifications of the methods used in the three steps, namely Graph Encoding, Graph Compression and Graph Partitioning, are depicted in Fig. 2 for both the GA and GE frameworks.

In the GA-framework (Mari and Baldassari 2022) the preprocessed log-price time series is encoded into a Quantile Graph. The graph encoding is carried out through the application of a Markov Transition Field (MTF) used as adjacency matrix (Campanharo et al. 2011). Then, the graph undergoes a compression phase through a coarsening process. The purpose of the graph coarsening is to find a smaller graph that well approximates the graph. In our case, the coarsening is performed by reducing the size of the MTF matrix averaging on its elements. The log-return community detection in the graph partitioning task is then performed by applying the Louvain method to the coarsened graph (Blondel et al. 2008). A comparative analysis of community detection algorithms reveals that, taking both accuracy and computational time into account, the Louvain algorithm outperforms other viable alternatives (Lancichinetti and Fortunato 2009). In general, communities can be classified as overlapping or non-overlapping, depending on whether a node can belong to several communities or only one. We adopted the latter option because the Louvain method for community detection is a non-overlapping method, which means that each node in the network can only belong to one community. In the Louvain algorithm, nodes are iteratively assigned to communities based on a modularity measure that indicates the quality of community structure in a network. The algorithm aims to maximize the modularity by moving nodes between communities until no further improvement can be achieved.

We recall that in the GE-framework the preprocessed log-price time series is encoded into a visibility graph. Then, the graph undergoes a compression phase through an embedding process. Graph embedding is an unsupervised Graph

Machine Learning technique that computes a representation Euclidean vector for each node in a graph. Community detection is performed through clustering techniques based on TDA applied to the embedded multi-dimensional Euclidean space. For comparative purposes, the non-overlapping communities option was also used in the GE-framework.

In both frameworks, the partitioning step serves as a feature extraction procedure for the distribution estimation to automatically generate the number of mixture components, C, and the initialization parameter vector, Θ_0. In fact, since each log-return value, $r_i = x_{i+1} - x_i$, can be uniquely associated to the value x_i, the detected communities can also be viewed as log-return communities. The number of the GMM components, C, is set equal to the number of the detected communities, and the initialization parameters that are contained in the vector Θ_0, are computed through community membership by using the following set of equations,

$$\alpha_{0,c} = \frac{N_c}{N}, \tag{7}$$

$$\mu_{0,c} = \frac{1}{N_c} \sum_{i=1}^{N} w_{ic} r_i, \tag{8}$$

$$\sigma^2_{0,c} = \frac{1}{N_c} \sum_{i=1}^{N} w_{ic}(r_i - \mu_{0,c})^2, \tag{9}$$

where $w_{ic} = 1$ if the observation r_i belongs to the cluster c, $w_{ic} = 0$ otherwise, and $N_c = \sum_{i=1}^{N} w_i$ represents the effective number of data points assigned to component c. We notice that both the mean and the variance are computed similarly to how standard empirical average and variance are computed, but with a weight w_{ic}.

Once initialized, the EM algorithm first computes the parameter values that maximizes the log-likelihood and determines the new membership weights accordingly, thus providing an overlay of the initial communities. The iterative procedure continues until convergence is achieved.

3 The experiment

In this section, we perform an experiment to evaluate the performance of GA and GE frameworks on US wholesale electricity prices. Our data collection consists of daily prices observed during the time interval from January 1, 2017 to December 31, 2021. Time series data can be freely downloaded from www.eia.gov/electricity/wholesale. Observed time series are illustrated in Fig. 3.

Volatility and extreme unpredictability characterize market price movements, which are frequently accompanied by sharp spikes and jumps generated by changes in the supply–demand balance. All these time series show irregular sampling (as a result of weekends, holidays and other missing data due to specific market reasons)

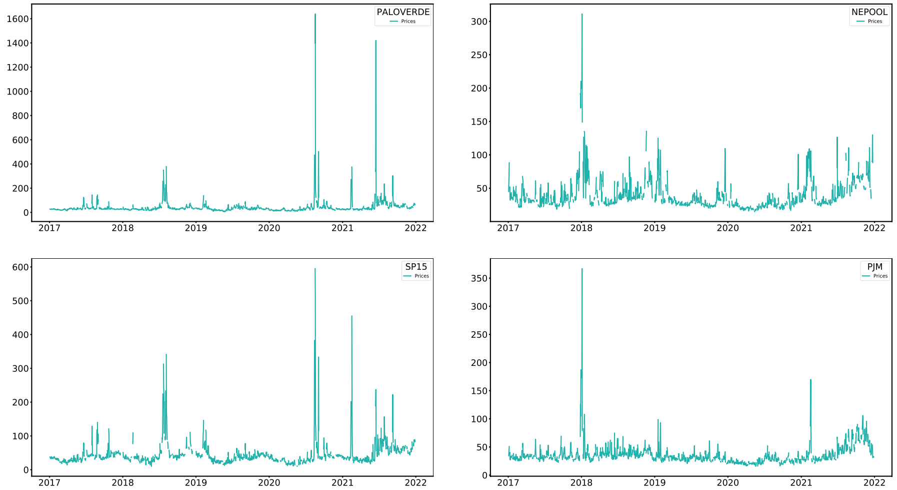

Fig. 3 Time series observed in the time interval from January 1, 2017 to December 31, 2021. Data are expressed in nominal dollars per megawatt-hour

as well as trend and seasonality. Preprocessing of the data was conducted first to fill in gaps in the time series and remove any trends and seasonality over time.

3.1 Data preprocessing

Let us denote with $\{y_{\bar{t}}^{ob}\}$ the time series of daily prices defined as a function of the incomplete observed raw time base $\{\bar{t}\}$, and $\{x_{\bar{t}}^{ob}\}$ its natural logarithm transformation, i.e., $x_{\bar{t}}^{ob} = \log y_{\bar{t}}^{ob}$. The time series $\{x_{\bar{t}}^{ob}\}$ is sampled irregularly (not evenly throughout the time interval) since weekends, holidays, and other sporadic missing days are not included. The imputation of missing data (gap filling) can convey important knowledge (Owen 2007), thus counteracting the phenomenon known as informative missingness. In the presence of missing data, different time periods may have different information content. In fact, even when the market is closed, new information that can influence price dynamics may emerge (French 1980; Mantegna , Stanley 1999).

The first step, therefore, involved the gap filling of the time series. To do this, we employed a complete daily grid $\{t\}$ and an imputation technique called missForest (Stekhoven and Bühlmann 2011), a ML algorithm for data imputation that is completely agnostic about data distribution. MissForest is used to compute a value for each missing point. In this way, following the same procedure proposed in (Mari , Baldassari 2021), we extended the raw time series $\{x_{\bar{t}}^{ob}\}$ to a complete daily time base $\{t\}$,

$$\{x_{\bar{t}}^{ob}\} \xrightarrow{\mathscr{F}} \{x_t^f\}, \tag{10}$$

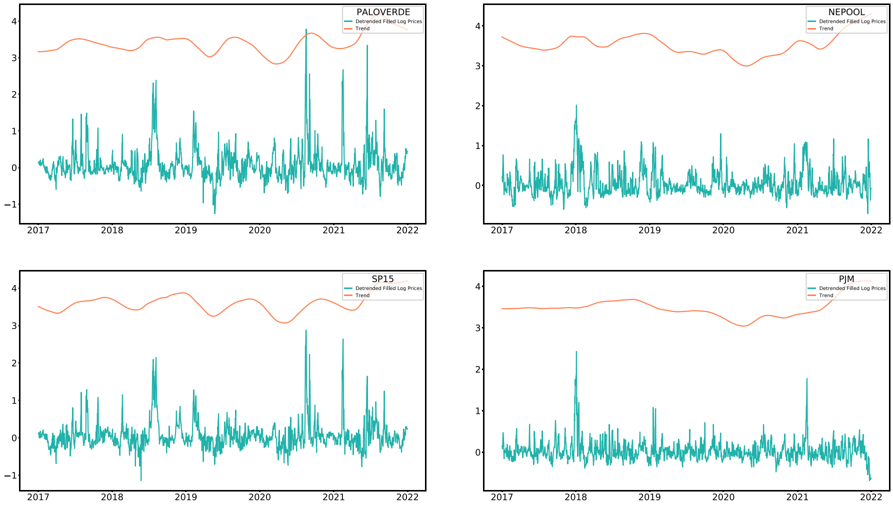

Fig. 4 Detrended log-price time series x_t (in green) and trends x_t^{tr} (in red)

where $\mathscr{F}$ is the application that transforms $\{\bar{t}\}$ in $\{t\}$ and computes missing values using the missForest algorithm to fill the daily-complete grid, mapping $\{x_{\bar{t}}^{ob}\}$ to $\{x_t^f\}$.

In the second step, we looked for temporal trend and seasonality, hereinafter trend, that must be eliminated in order to reveal the stochastic process driving market dynamics. The Locally Weighted Estimated Scatterplot Smoothing (LOWESS) technique was used to perform this task (Cleveland 1979; Cleveland and Devlin 1988). LOWESS is a versatile method for removing the trend by fitting basic polynomial models to restricted portions of data. The key benefit of LOWESS over other methods is that it does not need the specification of a global function or the assumption that the data must conform to a certain distribution shape (Dagum , Bianconcini 2016). Figure 4 shows detrended log-price time series, $\{x_t\}$, and the trend, $\{x_t^{tr}\}$, which is superimposed for each market under investigation.

Once detected, the trend can be eliminated from the filled time series, thus obtaining,

$$x_i = x_i^f - x_i^{tr}, \qquad i = 1, 2, \cdots, N, \tag{11}$$

where we assumed, without loss of generality, that the complete time grid $\{t\}$ is represented by the first N natural numbers, $\{1, 2, \cdots, N\}$. Then, log-returns are computed as the difference between two successive daily log-prices, i.e.,

$$r_i = x_{i+1} - x_i, \qquad i = 1, 2, \cdots, N-1. \tag{12}$$

Figure 5 depicts the log-return time series, $\{r_t\}$, for each market under investigation. Table 1 depicts the descriptive statistics of log-returns. In all four investigated markets, empirical log-return distributions exhibit heavy tails, as indicated by the high values of the kurtosis. The occurrence of jumps and spikes in electricity prices significantly increases the value of the fourth central moment of empirical log-return

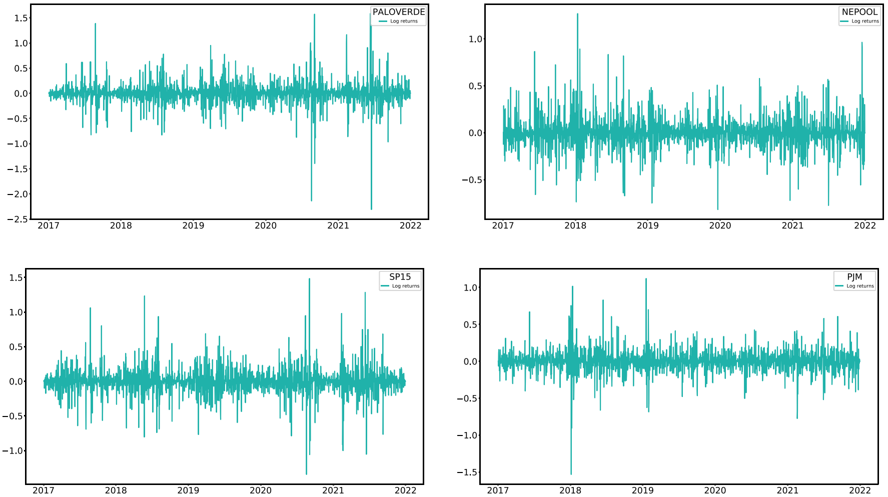

Fig. 5 Log-return time series

Table 1 Descriptive statistics of log-returns

	Mean	SD	Skew	Kurt
PALOVERDE	1.6e−04	0.23	− 0.38	19.84
NEPOOL	− 1.6e−04	0.17	0.51	9.22
SP15	6.9e−05	0.19	0.29	12.84
PJM	− 4.2e−04	0.15	− 0.20	16.02

distributions. From this perspective, it is crucial for a given probabilistic model to capture the first four central moments of the empirical distribution of log-returns, especially for financial applications to account for extreme events (Geman 2005; Geman and Roncoroni 2006).

3.2 Networks visualization

In this section, we constructed the visuals for the time series networks that were created in this analysis. It is very interesting to compare the two types of networks, NVGs and HVGs, described in Sect. 2. In particular, we used two distinct kinds of representations, namely the geometric representation and the Kamada–Kawai representation (Kamada and Kawai 1989) for both the NVG and HVG methods. In this regard, we note that graphs have an infinite number of possible representations, but the informative value of the graph depends on the ease with which it can be read (Di Battista et al. 1994). Figures 6 and 7 show NVGs and HVGs, respectively, in the geometric representation for the time series observed in the markets under investigation. In this geometric representation, the nodes are depicted at the same positions they occupy in the time series. The adopted geometric visibility criterion is

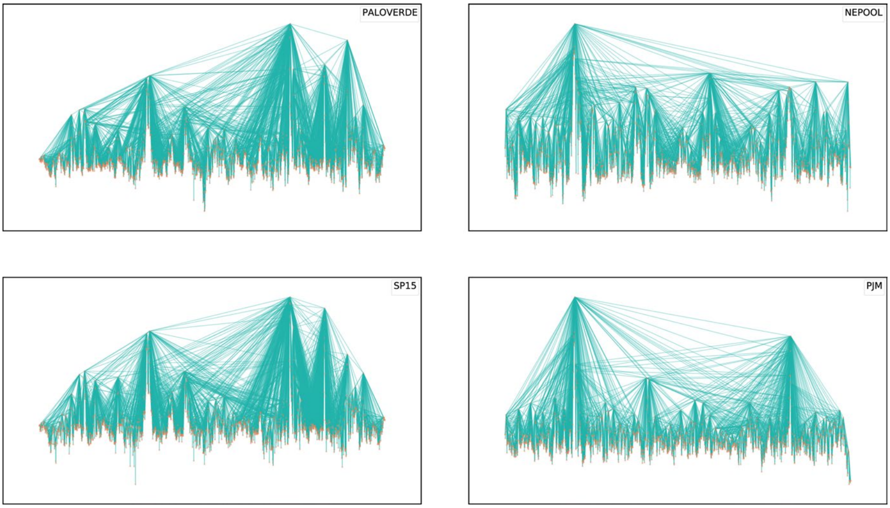

Fig. 6 NVGs: the geometric representation

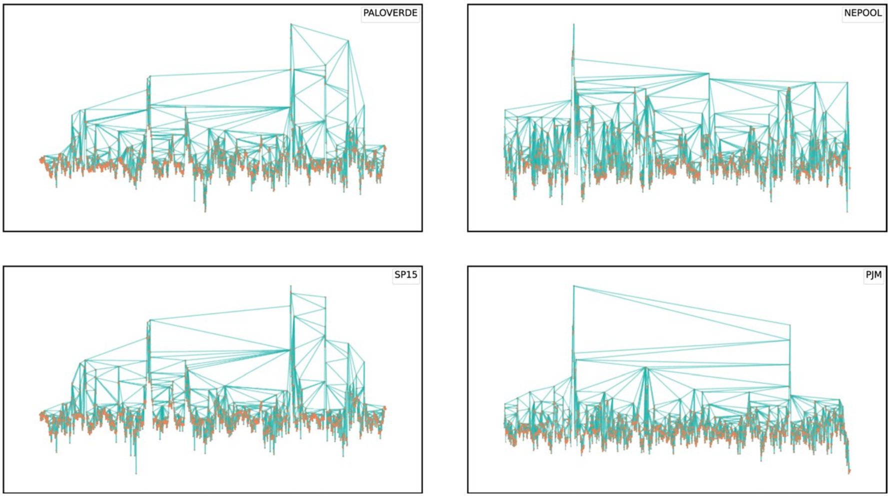

Fig. 7 HVGs: the geometric representation

self-evident. In fact, the nodes are connected by straight lines drawn according to Eq. (1) in Fig. 6 and according to Eq. (2) in Fig. 7.

Figures 8 and 9 illustrate NVGs and HVGs, respectively, in the Kamada–Kawai representation. The Kamada–Kawai representation provides a graph force-directed layout algorithm that positions the nodes of a graph in two-dimensional space, based on an optimization criterion that minimizes the total energy of the system (Kamada and Kawai 1989). It offers an effective and intuitive way to visualize complex graphs, and can help in the interpretation and

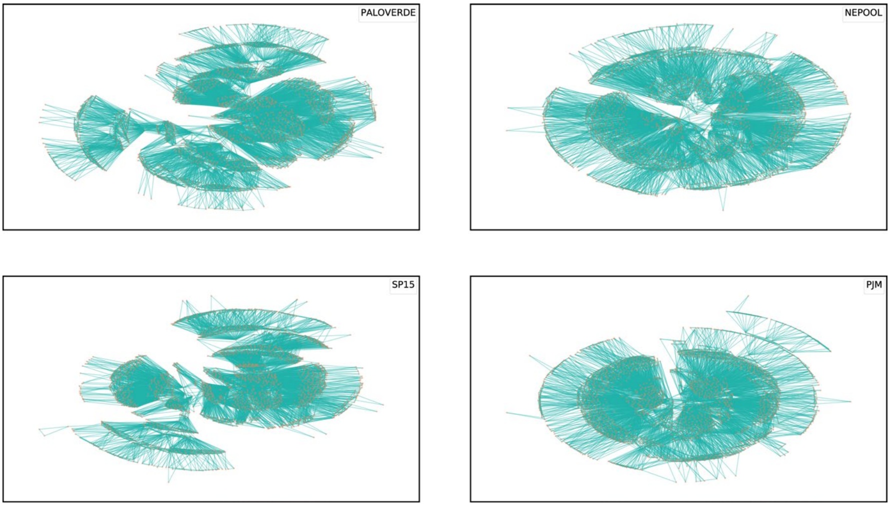

Fig. 8 NVGs: the Kamada–Kawai representation

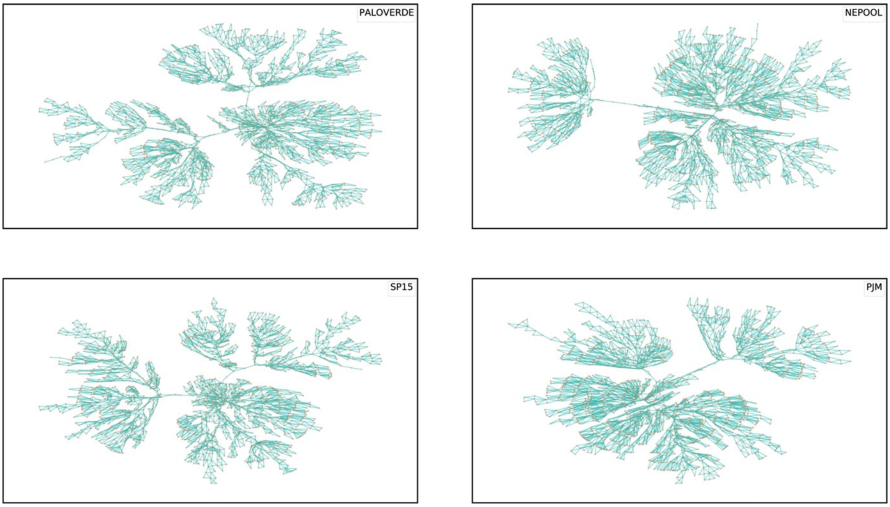

Fig. 9 HVGs: The Kamada–Kawai representation

understanding of graph structure and relationships between nodes. The method takes the relative distances between nodes and the edges that connect them into account, resulting in intuitive and visually appealing graph layouts that better attempt to highlight the initial communities that are identified.

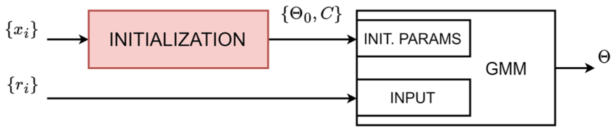

Fig. 10 The distribution estimation section

3.3 Distribution estimation

In this section we discuss the EM-based maximum likelihood estimation of GMMs on market log-returns time series. The initialization of the EM algorithm will be performed using both the GA and GE frameworks. Figure 10 shows a magnified version of the Distribution Estimation section and summarizes the calibration procedure. As discussed in the previous section, the initialization blocks produce both the number of the model component, C, and the initial parameter set, Θ_0. Indeed, the number of components is set equal to the number of the detected communities, and the initialization parameters are computed through community membership according to Eqs. (7)–(9). Then, the EM algorithm is used to estimate the GMM parameter set, Θ.

In the GA-framework, to generate the MTF adjacency matrix we need to quantize the time series in order to define the dynamic states. In our analysis, both Quantile binning and Normal binning will be used (Mari and Baldassari 2022). We perform the model estimation for each couple of values (Q, S) defined on a suitable grid, where Q is the number of bins, i.e., the dynamic states, and S is the dimension of the MTF coarsened adjacency matrix. The number of bins, Q, is made to vary from 2 to 100 in increments of 2 for both Quantile and Normal binning strategies; the parameter S is made to vary from 5 to 400 in increments of 5. In the GE-framework, the parameters of the various embedding methods that are employed in this study are the default parameters of the Karate Club framework. Moreover, the ToMATo clustering techniques needs to set three parameters, namely the neighborhood information of the point cloud, the density estimator, and the merging parameter. We used a ToMATo cluster function contained in the Python library tomaster[1] which needs only the neighborhood information as input and automatically calculates the other two parameters. Since ToMATo relies heavily on neighborhood information, a popular choice to model a neighborhood seems to be the K-nearest neighbors algorithm. As well documented in the literature, this ML algorithm performs well, recovering the correct clusters under an appropriate choice of the parameter K (Chazal et al. 2013). For the GE-framework the model estimation is, therefore, performed by exploring a suitable one-dimensional grid for the parameter K. In this grid, the parameter K varies from 2 to 100 in increments of 1.

[1] tomaster: Topological Mode Analysis on Steroids. Github repository at https://github.com/louisabraham/tomaster.

Table 2 Graph embedding experimental setup

Framework	Embedding method	Additional attributes	Embedding dimension
GE	Diff2Vec	–	128
GE	GraphWave	–	400
GE	ASNE	$\{r_t\}$	128
GE	NEU_Diff2Vec	–	128
GE	NEU_GraphWave	–	400
GE	NEU_ASNE	$\{r_t\}$	128

The embedding method, during the exploration of the grid, always operates on the complete graph without using an approximate form, as opposed to the coarsening method, which makes use of a reduced adjacency matrix of size S. This is one of the most important differences between the GA and GE frameworks. Table 2 reports the dimension of the Euclidean space for each embedding method. This parameter was chosen based on the default parameter provided by the library. Moreover, Table 2 shows the log-return time series in correspondence of two well defined embedding methods, namely the ASNE and NEU_ASNE methods.[2] As we will see in the next section, this feature can greatly improve the accuracy of the estimation procedure.

3.4 Results

This section outlines and discusses the empirical findings of the experiment. More specifically, we will compare the outcomes of six embedding methods for the GE-framework (see Table 2) and one coarsening method for the GA-framework. Due to the fact that the GA-framework includes two different binning strategies (Quantile and Normal), and the GE-framework also includes two different strategies (Natural and Horizontal), the comparison will involve 14 outcomes corresponding to 14 different initialization techniques. For comparative purposes, the estimation result obtained through a more traditional K-means-based initialization technique is also considered.

There exist information criteria for assessing the relative quality of statistical models, such as the Bayesian Information Criterion (BIC), and the Akaike Information Criterion (AIC) (Kuha 2004). Both the BIC and AIC criteria attempt to solve the model selection problem by introducing a penalty term for the number of parameters to account for the trade-off between the quality of the fit and the simplicity of the model. The penalty discourages issues related to overfitting because increasing the number of parameters in the model always improves the quality of the fit. AIC and BIC are expressed in terms of log-likelihood in the following way,

[2] We extended the applicability of the NEU meta-embedding to the ASNE approach as well, as additional attributes. The method is contained in our repository https://bit.ly/3zFtbQY.

Table 3 Paloverde

Model	Strategy	C	Grid parameters	BIC	$\varepsilon_4(\%)$
Diff2Vec	H	3	6	– 1244.98	35.53
Diff2Vec	N	3	29	– 1243.42	39.80
NEU_Diff2Vec	N	3	8	– 1243.33	34.78
GA-framework	n	3	(12, 265)	– 1241.43	44.66
GA-framework	q	3	(6, 395)	– 1241.23	44.17
NEU_ASNE	H	3	50	– 1240.20	33.93
ASNE	H	3	42	– 1239.12	47.36
K-means		3		– 1236.74	36.24
ASNE	H	3	45	– 1236.03	39.24
NEU_ASNE	N	3	41	– 1235.94	43.64
NEU_Diff2Vec	H	3	11	– 1235.52	38.56
NEU_GraphWave	N	3	8	– 1233.36	41.45
NEU_GraphWave	H	3	3	– 1229.43	50.60
GraphWave	H	5	87	– 1202.32	33.69
GraphWave	N	9	95	– 1088.18	28.62

$$AIC = -2\log(\mathcal{L}) + 2\lambda, \tag{13}$$

$$BIC = -2\log(\mathcal{L}) + \lambda\log(N), \tag{14}$$

where $\mathcal{L}$ is the likelihood, λ is the number of free parameters and N is the sample size. According to Eqs. (4)–(6), the number of free parameters is given by $\lambda = 3C - 1$. Lower AIC and BIC values indicate preferable statistical models. In most cases, the penalty term is larger in the BIC than in AIC criterion, because the penalty is 2λ for AIC, and $\lambda\log(N)$ for BIC. Therefore, BIC generally results in a smaller choice for C than AIC (Steele , Raftery 2010). For these reasons, we will apply the BIC criterion.

As a first result, we emphasize that in all four markets considered the most suitable model according to the BIC criterion is a three-component model in both the GA and GE frameworks. Tables 3, 4, 5, and 6 illustrate this result. Models are ordered by ascending BIC value, so that the first row of every table reports the best model according to the BIC criterion (the model with the minimum BIC value). For each model, the kurtosis absolute percentage error, ε_4, is also reported in order to measure the model ability to capture the heavy tail phenomenon as well. It is defined by

$$\varepsilon_4 = \frac{|\kappa_m - \kappa_o|}{\kappa_o}, \tag{15}$$

where κ_m denotes the model kurtosis and κ_o is the empirical kurtosis. We note that in the GMMs κ_m can be computed exactly. In addition, each table shows the binning strategy, which may be quantile (q) or normal (n) for the GA-framework, natural (N)

Table 4 Nepool

Model	Strategy	C	Grid parameters	BIC	ε_4 (%)
ASNE	N	3	95	– 1895.76	6.95
NEU_Diff2Vec	H	3	30	– 1895.15	1.22
NEU_GraphWave	H	3	92	– 1894.46	11.31
NEU_Diff2Vec	N	3	44	– 1892.56	7.37
GA-framework	n	3	(22, 95)	– 1891.19	16.15
GA-framework	q	3	(26, 175)	– 1889.73	18.91
NEU_ASNE	H	3	54	– 1889.82	19.55
Diff2Vec	N	3	20	– 1878.95	16.92
ASNE	H	3	24	– 1874.59	20.97
Diff2Vec	H	3	4	– 1874.45	8.77
NEU_ASNE	N	3	43	– 1874.26	22.47
K-means		2		– 1871.05	33.28
NEU_GraphWave	N	3	10	– 1856.92	18.55
GraphWave	H	5	89	– 1848.39	14.01
GraphWave	N	10	83	– 1727.04	20.69

Table 5 SP15

Model	Strategy	C	Grid parameters	BIC	ε_4 (%)
Diff2Vec	H	3	95	– 1669.10	19.36
Diff2Vec	N	3	22	– 1669.04	20.12
GA-framework	q	3	(34, 100)	– 1667.58	22.70
GraphWave	H	3	97	– 1666.51	25.26
GraphWave	N	3	9	– 1666.46	15.26
ASNE	H	3	40	– 1666.21	25.24
GA-framework	n	3	(16, 105)	– 1665.84	21.91
ASNE	N	3	100	– 1663.87	24.47
NEU_Diff2Vec	H	3	39	– 1661.79	10.78
NEU_Diff2Vec	N	3	10	– 1657.55	20.32
K-means		4		– 1638.10	3.31
NEU_GraphWave	H	4	6	– 1629.14	35.17
NEU_GraphWave	N	3	5	– 1626.38	35.75
NEU_ASNE	H	5	89	– 1622.14	19.87
NEU_ASNE	N	8	94	– 1559.93	4.78

or horizontal (H) for the GE-framework, and the best-fit model, along with the corresponding grid parameters.

The GE-framework outperforms the GA-framework in all the markets under investigation. Lower BIC values indicate better learning features. In each market, there is always an embedding method that produces better results than the GA-framework. In particular, we note that ASNE and Diff2Vec methods and their

Table 6 PJM

Model	Strategy	C	Grid parameters	BIC	ε_4 (%)
NEU_ASNE	N	3	28	– 2378.67	21.23
Diff2Vec	N	3	48	– 2374.49	18.43
NEU_Diff2Vec	H	3	67	– 2373.62	34.67
NEU_GraphWave	N	3	8	– 2372.28	29.10
GA-framework	n	3	(14, 255)	– 2367.62	3.71
GA-framework	q	3	(36, 385)	– 2361.95	43.88
Diff2Vec	H	3	95	– 2363.07	30.97
ASNE	N	3	14	– 2360.05	44.31
NEU_Diff2Vec	N	3	25	– 2358.55	29.22
ASNE	H	3	22	– 2345.36	51.48
NEU_ASNE	H	3	12	– 2344.04	50.03
K-means		4		– 2337.67	8.05
NEU_GraphWave	H	2	4	– 2319.21	47.50
GraphWave	H	5	93	– 2292.88	31.83
GraphWave	N	10	70	– 2204.89	29.54

meta-embeddings represent the best choice in all four power markets. The Diff2Vec method is the best choice in the Palo Verde and SP15 markets. The ASNE approach, which makes use of extra information carried on by log-return time series as an additional attribute, is the best choice in the Nepool market. Finally, we remark that the NEU approach performs very well, matching several rows in Table 8. In particular, its combination with the ASNE embedding method represents the best choice in the PJM market. Due to less information loss in the Graph Compression phase, superior data-driven and unsupervised Graph Machine Learning approaches and reduced computational complexity, the GE-framework seems to be a flexible and powerful tool for understanding the complex, highly nonlinear behavior of electricity market prices. Moreover, it requires less CPU time. In fact, the wall-clock time on the four markets amounted to about 120 h for the GA-framework and about 30 min for the GE-framework. This difference is primarily attributed to the optimization methods used in each framework. The GA-framework involves a two-parameter grid optimization process, whereas the GE-framework employs a simpler one-parameter optimization process. Additionally, the Louvain algorithm in the GA-framework requires a longer processing time compared to the GE-framework, which is designed to be more streamlined (a VM from Colab Pro was used with 2 cores Intel(R) Xeon(R) CPU 2.20GHz, RAM 13 GB). However, it is important to note that the GA-framework ranks in the top 5 of the 14 methods discussed in this paper in all four markets, despite using only one compression technique (graph coarsening). For comparison purposes, the estimation results obtained using a more traditional *K*-means-based initialization technique is also shown in Tables 3-6. As expected, our methodology outperforms this more conventional approach. In fact, for each market the BIC value obtained through this initialization technique is greater than the minimum BIC value obtained in both the GA and GE frameworks. The BIC

Table 7 BIC weights and weight ratios

Method	BIC	Weights	Weight ratios
PALOVERDE			
GE-framework	– 1244.98	0.8434	5.90
GA-framework	– 1241.43	0.1429	
K-means	– 1236.74	0.0137	
NEPOOL			
GE-framework	– 1895.76	0.9076	9.83
GA-framework	– 1891.19	0.0924	
K-means	– 1871.05	3.9e–06	
SP15			
GE-framework	– 1669.10	0.6814	2.14
GA-framework	– 1667.58	0.3186	
K-means	– 1638.10	1.3e–07	
PJM			
GE-framework	– 2378.67	0.9960	250.89
GA-framework	– 2367.62	0.0040	
K-means	– 2337.67	1.2e–09	

weights obtained by comparing the best performing models of the GE-framework, the GA-framework and the *K*-means approach are shown in Table 7.

The BIC weights are computed according to

$$w_i = \frac{\exp(-0.5\Delta_i)}{\sum_{k=1}^{3} \exp(-0.5\Delta_i)} \quad i = 1, 2, 3, \tag{16}$$

where

$$\Delta_i = \mathrm{BIC}_i - \mathrm{BIC}_{min}, \tag{17}$$

with BIC_{min} is the minimum BIC value (Wagenmakers and Farrell 2004). The weight w_i can be interpreted as the probability that the *i*-th model is the best model in terms of Kullback-Leibler discrepancy (Anderson , Burnham 2004). The weight ratio w_i/w_j is calculated between the two models with the highest probability. From Table 7 it can be inferred that the best-fitting model in the GE-framework is several times (from 2.14 for SP15 to 250.89 for PJM) more likely to be the best model than the best-fitting model in the GA-framework. The weight ratios between the best GE-framework model and the best *K*-means model are much higher for each market under consideration. The same line of reasoning proves that the GA-framework also far outperforms the *K*-means approach.

Finally, Table 8 shows the models of both GA and GE frameworks with three Gaussian components that minimize the value of the absolute percent error of kurtosis, ε_4.

Overall, the methodology proposed for both the GE and GA frameworks, in addition to showing a high degree of learning as highlighted by the BIC values, allows for remarkable replication of the first four moments of the empirical distribution, as

Table 8 Three-component models with minimum ε_4

Method	Strategy	Grid params	BIC	ε_4 (%)	Mean	Standard deviation	Skewness	Kurtosis
PALOVERDE								
NEU_Dif-f2Vec	N	15	– 1216.97	14.71	1.6e–04	0.23	0.30	16.92
GA-frame-work	n	(6, 335)	– 1212.52	2.38	1.6e–04	0.23	0.39	19.37
NEPOOL								
NEU_Dif-f2Vec	H	29	– 1894.97	0.18	– 1.6e–04	0.17	0.43	9.24
GA-frame-work	q	(12, 15)	– 1886.26	1.53	– 1.6e–04	0.17	0.35	9.37
SP15								
NEU_Dif-f2Vec	N	20	– 1665.79	2.72	6.9e–05	0.19	0.06	13.19
GA-frame-work	n	(22, 345)	– 1648.84	0.53	6.9e–05	0.19	0.03	12.91
PJM								
ASNE	H	14	– 2363.76	2.48	– 4.2e04	0.15	0.11	15.62
GA-frame-work	q	(12, 390)	– 2362.55	0.15	– 4.2e–04	0.15	0.14	15.99

can be directly verified by comparing the model results with the empirical results reported in Table 1.

4 Concluding remarks

Based on Graph Machine Learning techniques, we provided a new methodology for estimating Gaussian mixture models on financial time series by maximum-likelihood using the EM algorithm. We conducted an experiment on US electricity market price time series by employing two different schemes of analysis, namely the GA-framework and the GE-framework. The graph encoding is performed by using quantile graphs with a Markov Transition Field as the adjacency matrix in the GA-framework and visibility graphs in the GE-framework.

The electricity market price structure is a good example of a system that defies straightforward modeling. To take advantage of the graph-structured information included in electricity market data, we used complex networks techniques that link time series and graphs. In the literature, the GA-framework has been shown to outperform more traditional initialization approaches such as *K*-means and random techniques (Mari and Baldassari 2022). In this paper, we demonstrated that the GE-framework provides even better results. In fact, we showed that the best-fitting model in the GE-framework is several times (from 2.14 for SP15 to 250.89 for PJM) more likely to be the best model than the best-fitting model in the GA-framework. Compared with the GA-framework, the GE-framework, which uses data-driven and unsupervised graph embedding methods along with TDA-based clustering, has a more conservative approach and therefore loses less information in the process. Moreover, it requires less CPU time.

The proposed approach allows for a high degree of generalization and flexibility and could enable the creation of a super framework adaptable, in principle, to any empirical situation. In fact, referring to Fig. 1, the super framework could accommodate the methods used in this study and admit new ones. The ability of its functional blocks to integrate a wide range of methods and techniques makes it particularly flexible to a wide range of empirical analyses. Within this super framework, both the GA-framework and the GE-framework can be considered as two different configurations of a very flexible overall scheme. This topic will benefit from future investigation. In particular, our future research will focus on two main issues: (i) combining embedding and coarsening procedures in the graph compression block and testing the accuracy of the results; (ii) using initialization blocks as a self-supervised learning framework to discover communities and assign them to different dynamics.

Funding Open access funding provided by Università degli Studi G. D'Annunzio Chieti Pescara within the CRUI-CARE Agreement.

Data Availibility Statement The time series data used in this study are available in the wholesale electricity and natural gas market data archive https://www.eia.gov/electricity/wholesale.

Declarations

Conflicts of interest The authors declare no conflict of interest.

References

Anderson D, Burnham K (2004) Model selection and multi-model inference. Springer, New York. https://cds.cern.ch/record/1608735/files/9780387953649_TOC.pdf

Biernacki C, Celeux G, Govaert G (2003) Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models. Comput Stat Data Anal 41(3–4):561–575. https://doi.org/10.1016/S0167-9473(02)00163-9

Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech Theory Exp 2008(10):P10008. https://doi.org/10.1088/1742-5468/2008/10/p10008

Cai C, Wang D, Wang Y (2021) Graph coarsening with neural networks. arXiv preprint arXiv:2102.01350. https://doi.org/10.48550/arXiv.2102.01350

Campanharo A, Sirer M, Malmgren R, Ramos F, Amaral L (2011) Duality between time series and networks. PLoS ONE 6:e23378. https://doi.org/10.1371/journal.pone.0023378

Chazal F, Guibas LJ, Oudot SY, Skraba P (2013) Persistence-based clustering in riemannian manifolds. J ACM. https://doi.org/10.1145/2535927

Cleveland WS (1979) Robust locally weighted regression and smoothing scatterplots. J Am Stat Assoc 74(368):829–836. https://doi.org/10.1080/01621459.1979.10481038

Cleveland WS, Devlin SJ (1988) Locally weighted regression: an approach to regression analysis by local fitting. J Am Stat Assoc 83:596–610. https://doi.org/10.1080/01621459.1988.10478639

Cui P, Wang X, Pei J, Zhu W (2018) A survey on network embedding. IEEE Trans Knowl Data Eng 31(5):833–852. https://doi.org/10.1109/TKDE.2018.2849727

da Fontoura Costa L, Rodrigues FA, Travieso G, Boas PRV (2005) Characterization of complex networks: a survey of measurements. Adv Phys 56:167–242. https://doi.org/10.1080/00018730601170527

Dagum E, Bianconcini S (2016) Seasonal adjustment methods and real time trend-cycle estimation. Springer International Publishing, Berlin. https://doi.org/10.1007/978-3-319-31822-6

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the em algorithm. J R Stat Soc Ser B (Methodol) 39(1):1–22. https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

Di Battista G, Eades P, Tamassia R, Tollis IG (1994) Algorithms for drawing graphs: an annotated bibliography. Comput Geom 4(5):235–282. https://doi.org/10.1016/0925-7721(94)00014-X

French KR (1980) Stock returns and the weekend effect. J Financ Econ 8(1):55–69. https://doi.org/10.1016/0304-405X(80)90021-5

Geman H (2005) Commodities and commodity derivatives: modeling and pricing for agriculturals, metals and energy. Wiley, Hoboken https://shs.hal.science/halshs-00144182

Geman H, Roncoroni A (2006) Understanding the fine structure of electricity prices. J Bus 79:1225–1261. https://doi.org/10.1086/500675

Ghosh SK (2007) Visibility algorithms in the plane. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511543340

Hipp JR, Bauer DJ (2006) Local solutions in the estimation of growth mixture models: correction to hipp and bauer (2006). Psychol Methods 11:305–305. https://doi.org/10.1037/1082-989X.11.3.305

Kamada T, Kawai S (1989) An algorithm for drawing general undirected graphs. Inf Process Lett 31(1):7–15. https://doi.org/10.1016/0020-0190(89)90102-6

Kuha J (2004) Aic and bic: comparisons of assumptions and performance. Sociol Methods Res 33(2):188–229. https://doi.org/10.1177/0049124103262065

Lacasa L, Luque B, Ballesteros F, Luque J, Nuño JC (2008) From time series to complex networks: the visibility graph. Proc Natl Acad Sci 105(13):4972–4975. https://doi.org/10.1073/pnas.0709247105

Laengle S, Lobos V, Merigó JM, Herrera-Viedma E, Cobo MJ, De Baets B (2021) Forty years of fuzzy sets and systems: a bibliometric analysis. Fuzzy Sets Syst 402:155–183. https://doi.org/10.1016/j.fss.2020.03.012

Lan X, Mo H, Chen S, Liu Q, Deng Y (2015) Fast transformation from time series to visibility graphs. Chaos Interdiscipl J Nonlinear Sci 25(8):083105. https://doi.org/10.1063/1.4927835

Lancichinetti A, Fortunato S (2009) Community detection algorithms: a comparative analysis. Phys Rev E 80:056117. https://doi.org/10.1103/PhysRevE.80.056117

Luque B, Lacasa L, Ballesteros F, Luque J (2009) Horizontal visibility graphs: exact results for random time series. Phys Rev E 80:046103. https://doi.org/10.1103/PhysRevE.80.046103

Mantegna R, Stanley H (1999) Introduction to econophysics: correlations and complexity in finance. Cambridge University Press, Cambridge

Mari C, Baldassari C (2021) Ensemble methods for jump-diffusion models of power prices. Energies. https://doi.org/10.3390/en14082084

Mari C, Baldassari C (2022) Unsupervised expectation-maximization algorithm initialization for mixture models: A complex network-driven approach for modeling financial time series. Inf Sci 617:1–16. https://doi.org/10.1016/j.ins.2022.10.073

Newman M (2010) Networks: an introduction. Oxford University Press, Oxford

Newman ME (2003) The structure and function of complex networks. SIAM Rev 45(2):167–256. https://doi.org/10.1137/S003614450342480

Owen M (2007) Practical signal processing. Cambridge University Press, Cambridge. https://www.cambridge.org/it/academic/subjects/engineering/communications-and-signal-processing/practical-signal-processing?format=HB &isbn=9780521854788

Perozzi B, Al-Rfou R, Skiena S (2014) Deepwalk: online learning of social representations. pp. 701–710. https://doi.org/10.1145/2623330.2623732

Rozemberczki B, Kiss O, Sarkar R (2020) Karate club: An api oriented open-source python framework for unsupervised learning on graphs. pp 3125–3132. https://doi.org/10.1145/3340531.3412757

Shireman E, Steinley D, Brusco MJ (2017) Examining the effect of initialization strategies on the performance of gaussian mixture modeling. Behav Res Methods 49:282–293. https://doi.org/10.3758/s13428-015-0697-6

Silva VF, Silva ME, Ribeiro P, Silva F (2021) Time series analysis via network science: concepts and algorithms. WIREs Data Min Knowl Discov 11(3):e1404. https://doi.org/10.1002/widm.1404

Skrlj B, Kralj J, Lavrac N (2020) Embedding-based silhouette community detection. Mach Learn. https://doi.org/10.1007/s10994-020-05882-8

Smyth P (2017) Mixture models and the em algorithm. Department of Computer Science, University of California, Irvine. https://www.ics.uci.edu/~smyth/courses/cs274/notes/Notes7_Mixtures_and_EM.pdf

Steele RJ, Raftery AE (2010) Performance of bayesian model selection criteria for gaussian mixture models. https://citeseerx.ist.psu.edu/document?repid=rep1 &type=pdf &doi=76b7913597dfa46891c349adf62247e5fadf4f02

Steinley D, Brusco M (2011) Evaluating mixture modeling for clustering: Recommendations and cautions. Psychol Methods 16:63–79. https://doi.org/10.1037/a0022673

Stekhoven DJ, Bühlmann P (2011) MissForest-non-parametric missing value imputation for mixed-type data. Bioinformatics 28(1):112–118. https://doi.org/10.1093/bioinformatics/btr597

Vespignani A (2018) Twenty years of network science. Nature 558:528–529. https://doi.org/10.1038/d41586-018-05444-y

Voit J (2013) The statistical mechanics of financial markets. Springer, Berlin

Wagenmakers EJ, Farrell S (2004) Aic model selection using akaike weights. Psychon Bull Rev 11:192–196. https://doi.org/10.3758/BF03206482

Xie Y, Chen C, Gong M, Li D, Qin A (2021) Graph embedding via multi-scale graph representations. Inf Sci 578:102–115. https://doi.org/10.1016/j.ins.2021.07.026

Yang Y, Yang H (2008) Complex network-based time series analysis. Physica A 387(5–6):1381–1386. https://doi.org/10.1016/j.physa.2007.10.055

Yela DF, Thalmann F, Nicosia V, Stowell D, Sandler M (2020) Online visibility graphs: encoding visibility in a binary search tree. Phys Rev Res 2(2):023069. https://doi.org/10.1103/PhysRevResearch.2.023069

Zomorodian A, Carlsson G (2005) Computing persistent homology. Discrete Comput Geom 33(2):249–274. https://doi.org/10.1007/s00454-004-1146-y

Zou Y, Donner RV, Marwan N, Donges JF, Kurths J (2019) Complex network approaches to nonlinear time series analysis. Phys Rep 787:1–97. https://doi.org/10.1016/j.physrep.2018.10.005

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.