IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 24, 2020, accepted August 31, 2020, date of publication September 4, 2020, date of current version September 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3021898

Cluster-Specific Latent Factor Estimation in High-Dimensional Financial Time Series

STJEPAN BEGUŠIĆ, (Member, IEEE), AND ZVONKO KOSTANJČAR, (Member, IEEE)
Laboratory for Financial and Risk Analytics, Faculty of Electrical Engineering and Computing, University of Zagreb, 10000 Zagreb, Croatia

Corresponding author: Stjepan Begušić (stjepan.begusic@fer.hr)

This work was supported in part by the Croatian Science Foundation under Project 5241, and in part by the European Regional Development Fund under Grant KK.01.1.1.01.0009 (DATACROSS).

ABSTRACT Unsupervised learning methods have been increasingly used for detecting latent factors in high-dimensional time series, with many applications, especially in financial risk modelling. Most latent factor models assume that the factors are pervasive and affect all of the time series. However, some factors may affect only certain assets in financial markets, due to their clustering within countries, asset classes, or sector classifications. In this paper we consider high-dimensional financial time series with pervasive and cluster-specific latent factors, and propose a clustering and latent factor estimation method. We also develop a model selection algorithm, based on the spectral properties of asset correlation matrices and asset graphs. A simulation study with known data generating processes demonstrates that the proposed method outperforms other clustering methods and provides estimates with a high degree of accuracy. Moreover, the model selection procedure is also shown to provide stable and accurate estimates for the number of clusters and latent factors. We apply the proposed methods to datasets of asset returns from global financial markets using a backtesting approach. The results demonstrate that the clustering approach and estimated latent factors yield relevant information, improve risk modelling and reduce volatility in optimal minimum variance portfolios.

INDEX TERMS Latent factor models, high-dimensional data analysis, financial risk modeling.

I. INTRODUCTION

With the rise of data driven decision making in risk management, statistical and machine learning methods are becoming increasingly important as their ability to uncover meaningful information and perform well out-of-sample is put to the test in real world scenarios. This field has recently attracted a fair amount of interdisciplinary research, bringing together mathematical, physical, econometric and computer science approaches [1]–[3]. These methods are of critical importance in financial risk modelling, where the dynamics of asset return time series are driven by underlying risk factors [4]. To estimate the effects that these underlying factors have on observed asset returns, traditional modelling approaches use observable macroeconomic time series (such as GDP growth, interest rates, or market returns) as model inputs [5], while others focus on finding proxies for unobservable factors (known as size, value, or momentum) using economic firm-level data [6], [7]. However, this information is not always available for every security (i.e. derivatives or certain ETFs or indices), meaning that these standard approaches may not be universally applicable [8]. Moreover, recent empirical results have been challenging some of these models, giving advantage to more agnostic statistical approaches [9].

Today, with the advances in financial technology and the globalization of financial markets, the number of investable securities and their diversity in terms of asset classes and country of origin is larger than ever. Throughout the past decades, these developments motivated the increased focus on statistical and unsupervised learning techniques for uncovering latent risk factors in asset return data [10], [11]. However, even though the number of assets continues to grow, the observable time period used to estimate these models must remain short. This is due to the fact that financial markets are known to exhibit sudden changes in dynamics and stationarity can not be assumed over long time periods – asset return volatilities and correlations change over time, especially in the presence of financial bubbles and crashes [12]–[14]. Therefore, these unsupervised learning methods must be

The associate editor coordinating the review of this manuscript and approving it for publication was Juan Wang.

able to perform well on high-dimensional datasets when the number of time series N is commensurate or even larger than their length T [15]. The out-of-sample performance of these estimates is crucial for many portfolio optimization or risk management applications [1], [16]. Since deep learning techniques (such as autoencoders, restricted Boltzmann machines, or GANs) use nonlinear models with a large number of parameters, they require large amounts of data for training [17] and may perform poorly in high-dimensional settings - instead, more restricted and parsimonious methods are required [18].

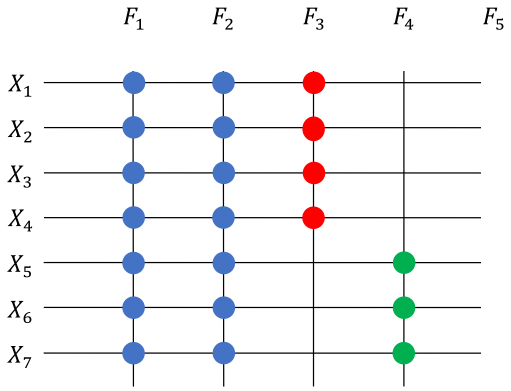


FIGURE 1. A grid view of a setting with time series $X_1, \ldots, X_7$ affected by factors $F_1, \ldots, F_5$, such that F_1 and F_2 are pervasive factors, F_3 is specific to time series $X_1, \ldots, X_4$, while F_4 and F_5 are specific to time series $X_5, \ldots, X_7$.

In this search for tractable and plausible latent factor estimation methods, it is crucial to take advantage of the structural specifics and statistical stylized facts of financial markets [19]. Since global markets consist of assets from different exchanges and various asset classes, certain risk factors will be specific only to a subset of assets [20]. For instance, in a global set of financial assets, pervasive global factors may affect all time series (such as the global macroeconomic and market shocks), and cluster-specific factor related to certain countries will affect only specific clusters of assets (for instance, European stocks will be affected by their own set of factors and may not be affected by some Asian market factors, after controlling for the common global component). Such a setting is shown in Figure 1, where the assets $X_1, \ldots, X_7$ are exposed to pervasive factors F_1 and F_2 and only certain clusters of assets are exposed to the cluster specific factors F_3 (affecting cluster of assets $X_1, \ldots, X_4$) and F_4, F_5 (affecting cluster of assets $X_5, \ldots, X_7$). However, the majority of modelling approaches consider only pervasive latent factors, decomposing the asset return variability into the variability explained by pervasive factors (affecting all assets) and idiosyncratic components (individual asset risk) [5], [21]. In this paper we consider the existence of cluster-specific latent factors, and propose a clustering and latent factor estimation method which simultaneously estimates the unknown cluster structures with the pervasive and cluster-specific latent factors. We consider an *approximate factor model*[1] which belongs to a class of models proposed by Ando and Bai [22], [23], who consider panel data with observable pervasive and unobserved pervasive and cluster-specific factors. The variability of asset returns is decomposed into the variability explained by pervasive factors, cluster-specific factors and idiosyncratic components. The pervasive factors affect all asset return time series, and these assets are divided into clusters in which a certain number of cluster-specific latent factors (the number of which may vary between clusters) affect the assets within that cluster. Since the clustering procedure may be biased towards the clusters with a larger number of cluster-specific factors (due to the fact that more factors will always be able to explain more variability in the data), the algorithm is divided into two main phases: the clustering phase which uses a fixed number of cluster-specific factors for all clusters, and the latent factor estimation phase based on the estimated asset clusters. We also propose a computational approach to model selection which detects the number of pervasive factors, the number of clusters and the number of cluster-specific factors in each cluster.

The main contributions of this paper are: (i) a new method for clustering and estimation of pervasive and cluster-specific latent factors in high-dimensional financial time series; (ii) a new model selection procedure based on spectral properties of the time series correlations and asset graphs. Since there is no "ground truth" in financial data (the number of factors, the factors themselves, as well as the clusters are all unknown), we also develop a simulation framework based on data generating processes (DGPs) which feature heavy-tailed distributed returns and correlated residuals (thus replicating statistical properties of asset returns), in which the ground truth is known - allowing us to measure the performance of the estimation procedure and the model selection method. Furthermore, we consider two datasets covering global financial markets, and apply the proposed method to the weekly return data. To measure the quality of the estimates, we develop a backtesting framework which enables us to obtain cross-validation results for the out-of-sample performance of the estimated models. We also construct a portfolio optimization scenario based on mean-variance optimization and perform backtests on financial market data to demonstrate the value of the proposed approach and the ability of the method to reduce risk in portfolio optimization scenarios.

The rest of the paper is organized as follows. In Section II we provide a deeper look into the latent factor model approach and its relation with dimensionality reduction techniques. Section III defines the clustering and latent factor estimation algorithm, as well as the model selection approach. In Section IV a detailed description of the DGPs used to obtain simulation results is given. Section V provides

[1] Approximate factor models, as opposed to strict factor modes, allow for correlated residuals, thus relaxing the strict assumption of a diagonal residual covariance and allowing for off-diagonal non-zero covariance elements, providing a more realistic assumption on the data.

our results on both simulation data and financial market data, and Section VI ends with a conclusion.

II. LATENT FACTORS IN HIGH-DIMENSIONAL FINANCIAL TIME SERIES

Dimensionality reduction techniques are commonly applied to obtain lower-dimensional representations of high-dimensional data, such that these representations maintain some key properties of the original data [24], [25]. This is a crucial step in coping with the so-called *curse of dimensionality* which manifests itself through computational issues, such as sparse samples in high dimensions [26] or the rank deficiency of sample covariance estimates and the difficulties in estimating sample distributions [15]. Feature selection algorithms primarily focus on finding a function $z = g(x)$ transforming the original high-dimensional data x (which may have irrelevant or redundant information) to a lower-dimensional set of variables z which aggregate the relevant information for a certain modelling task [27]. The function g is found by optimization of certain properties, which may be assisted by the class labels or target variables, depending on the modelling task. When the class labels or target variables are not available, unsupervised feature selection techniques focus on finding features which best preserve the clusters in the data [28], remove redundancy [29], [30] or optimize certain spectral properties of the underlying data graphs [31], [32] – either in the original data space or new subspaces [33]. Generally, unsupervised feature selection approaches have been found to yield relevant results in many machine learning tasks, including sequence analysis in bioinformatics [34], text classification [35], and other applications [36].

As opposed to feature selection, the latent factor model approach is focused primarily on finding a function $x = h(f)$ which explains the high-dimensional observed data x by a lower-dimensional set of factors f. The task of estimating factor models in high-dimensional data may be reduced to a regression task when these factors are known and observed – such cases may be common in biometric, psychometric or economic applications, where factor models are used to investigate the driving factors underlying the dynamics of some phenomena or processes [37]. However, these factors can often be unknown and unobserved, meaning that they must be estimated as latent variables from the data [38], requiring an unsupervised learning approach. The primary task is still to estimate the function $x = h(f)$, but now the factors need to be estimated from the data $f = g(x)$. Evidently, autoencoder-type approaches can be used to estimate the encoder ($f = g(x)$) and decoder ($x = h(f)$) parts of the model, offering a large range of architectures and the ability to model non-linear relationships [39], [40]. However, in the presence of high-dimensional data with the number of samples being small in comparison to the number of features/variables, nonlinear models often fail to generalize due to the large number of parameters – this turns the attention of recent research to high-dimensional latent factor estimation based on robust and regularized statistical methods [18], [41].

In this paper we consider high-dimensional financial time series of asset returns, with the goal of modelling the asset return time series by associating the assets with a lower-dimensional set of underlying factors. Since risk in finance is most commonly proxied by the variability of asset returns, the goal is to explain the variability of asset return time series by their exposure to latent factors. In addition to explaining risk, the estimated latent factor models are often used to obtain better estimates of the high-dimensional covariance matrices, which are ultimately a key component in portfolio optimization [15]. Traditionally, latent factor models in finance assume that the factors are pervasive (they affect all assets) and thus can be found as common components in high-dimensional asset return time series [18], [21]. On the other hand, some recent results suggest that assets indeed tend to form clusters and communities which can be observed in their dependence network structures (modelled either by correlation or other measures of connectedness) [42], [43]. Assuming a strict hierarchical clustering structure, Tumminello *et al.* [44] form a hierarchical latent factor model and propose an estimation method based on the minimum spanning tree of the underlying assets. Clusters of assets are also known to emerge in stocks of single equity markets (for instance, clusters of stocks belonging to the same sectors) - Kakushadze *et al.* [45] consider clustering techniques for estimating these groups from the asset return time series. Verma *et al.* [20] proposed a cluster-specific factor model for the log-volatility with the goal of studying the heteroskedastic properties of volatility in financial assets returns. Other clustering approaches were also shown to improve high-dimensional covariance matrix estimates, which ultimately reduces risk in optimized portfolios [46]–[49]. However, while the evidence on the existence of asset clusters is compelling, certain latent factors may still be pervasive and affect all assets - for instance, global macroeconomic shocks or the market factor [50], [51]. These may not be omitted in the search for asset clusters. To fully exploit the structural properties and obtain better latent factor models, both the asset clustering as well as latent pervasive and clusters-specific factors need to be estimated from the data.

III. METHODOLOGY

A. MODEL

Let X_{ti} denote the return[2] of asset i at time step t, calculated as the percentage change in prices between periods $t-1$ and t. Each asset i is associated with one of K clusters where $g_i \in \{1, \ldots, K\}$ denotes the cluster index for asset i. We assume a latent factor model in which asset returns depend on the realizations of pervasive (common) factors f_{tp} and

[2]In this paper we consider the periodic (also known as arithmetic) returns $X_t = (S_t - S_{t-1})/S_{t-1}$, where S_t is the asset price at time t. Even though log-returns may have more elegant statistical properties, periodic returns allow for efficient matrix operations to be used in cross-sectional and portfolio return calculations. For more details on this topic see [52].

cluster-specific factors ϕ_{tq}:

$$X_{ti} = \sum_{p=1}^{P} f_{tp} b_{ip} + \sum_{q=1}^{C_k} \phi_{tq} \lambda_{iq}^{(k)} + e_{ti}, \quad g_i = k, \tag{1}$$

where $t = 1, \ldots, T$ is the time index, $i = 1, \ldots, N$ is the asset index, $p = 1, \ldots, P$ is the pervasive factor index, and $q = 1, \ldots, C_k$ is the cluster-specific factor index for cluster $k = 1, \ldots, K$. Each of the K clusters is allowed a different number of factors C_k - thus, the total number of cluster-specific factors is $Q = \sum_k C_k$. The residual term e_{ti} (also called the idiosyncratic component) represents the sources of risk which are individual to each asset and are not explained by the common factors. The model (1) can also be written in matrix notation as:

$$\mathbf{X} = \mathbf{F}\mathbf{B}^\intercal + \mathbf{\Phi}\mathbf{\Lambda}^\intercal + \boldsymbol{e}, \tag{2}$$

where $\mathbf{X} \in \mathbb{R}^{T \times N}$ contains N asset return time series of length T, $\mathbf{F} \in \mathbb{R}^{T \times P}$ are the realizations and $\mathbf{B} \in \mathbb{R}^{N \times P}$ are the loadings for P pervasive factors. The realizations of Q cluster-specific factors for all K clusters are $\mathbf{\Phi} = \left[\mathbf{\Phi}^{(1)}, \ldots, \mathbf{\Phi}^{(K)}\right]$ and the cluster-specific factor loadings are $\mathbf{\Lambda} = \left[\mathbf{\Lambda}^{(1)}, \ldots, \mathbf{\Lambda}^{(K)}\right]$, where $\mathbf{\Phi}^{(k)} \in \mathbb{R}^{T \times C_k}$ and $\mathbf{\Lambda}^{(k)} \in \mathbb{R}^{N \times C_k}$ denote the C_k columns of $\mathbf{\Phi}$ and $\mathbf{\Lambda}$ corresponding to factor realizations and loadings associated with cluster k. The term $\boldsymbol{e} \in \mathbb{R}^{T \times N}$ contains all of the N individual idiosyncratic components.

Since the pervasive factors affect all time series, the pervasive factor loading matrix $\mathbf{B}$ is full, whereas the cluster-specific loading matrix $\mathbf{\Lambda}$ is non-zero only for the elements which correspond to assets and factors associated with the same cluster:

$$\Lambda_i^{(k)} = 0, \quad g_i \neq k. \tag{3}$$

The pervasive and cluster-specific factors are assumed to be uncorrelated: $\mathrm{Cov}(\boldsymbol{f}_p, \boldsymbol{\phi}_q) = 0, \ \forall p, q$. The factor covariance matrices $\mathrm{Cov}(\mathbf{F})$ and $\mathrm{Cov}(\mathbf{\Phi})$ are assumed to be positive definite, thus allowing for some correlations between cluster-specific factors.

The assumed factor model is *approximate*, meaning that the error terms $\mathbf{e}$, also known as idiosyncratic components (since they represent individual sources of risk for each asset), are zero-mean but are allowed cross-sectional correlations and heteroskedasticity. This implies that residual covariance $\mathrm{Cov}(\mathbf{e})$ is not necessarily diagonal, but it needs to be sparse (the cross-correlations in the idiosyncratic components can not be a consequence of common factors in the data) [18].

The factors are latent (unobservable), the clustering is unknown, as well as the numbers of factors, clusters, and cluster-specific factors - all of these need to be estimated from the data. Given the model (2) and the assumptions, in the following we propose an approach to estimate all of the above. First an iterative method clusters the data assuming a fixed number of cluster-specific factors in each cluster. Then the numbers of cluster-specific factors inferred from the data using the estimated clusters. To estimate the number of pervasive factors and clusters, we propose a model selection method based on the spectral properties of the asset correlation matrix and the asset graph estimated from the return time series.

B. CLUSTERING AND LATENT FACTOR ESTIMATION

Let $||\mathbf{A}||_F^2 = \sum_i \sum_j A_{ij}^2$ denote the Frobenius norm of a matrix $\mathbf{A}$. Given a data matrix $\mathbf{X}$, and assuming a known number of pervasive factors P, number of clusters K and number of cluster-specific factors in each cluster C_k, consider the following loss function:

$$l(\mathbf{X}; \mathbf{F}, \mathbf{B}, \mathbf{\Phi}, \mathbf{\Lambda}) = \frac{1}{NT} ||\mathbf{X} - \mathbf{F}\mathbf{B}^\intercal - \mathbf{\Phi}\mathbf{\Lambda}^\intercal||_F^2. \tag{4}$$

The loss function is the error of unexplained variation in the data. According to the Eckart–Young–Mirsky theorem, if all factors are pervasive the optimal low-rank approximation is given by the principal components (PC) estimator [5], [18], [53], based on the eigenvalue decomposition of the matrix $\frac{1}{T}\mathbf{X}^\intercal\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{U}^\intercal$. The pervasive factors and loadings are then estimated as:

$$\begin{aligned} \hat{\mathbf{B}} &= \mathbf{U}_P \sqrt{\mathbf{D}_P}, \\ \hat{\mathbf{F}} &= \mathbf{X}\hat{\mathbf{B}}\mathbf{D}_P^{-1}, \end{aligned} \tag{5}$$

where $\mathbf{U}_P$ are the P eigenvectors corresponding to the largest P eigenvalues, contained in the diagonal matrix $\mathbf{D}_P$. The principal components estimator is in the focus of many high-dimensional statistical applications [18], [21] - however, it is not applicable in the presence of cluster-specific factors. Moreover, for the assumed model, such a direct analytic estimation is not obtainable, since the loss function (4) needs to be optimized subject to the cluster-specific factor condition (3), given the clustering $G = [g_1, \ldots, g_N]$. The estimates of the pervasive factors, cluster memberships, and cluster-specific factors all depend on each other, and thus require an iterative approach - in which the PC estimator will prove useful.

1) CLUSTER ASSIGNMENT

If the pervasive factors $\mathbf{F}$ with loadings $\mathbf{B}$ and cluster specific factors $\mathbf{\Phi}$ are known, each asset can be assigned to the cluster which minimizes its value of the loss function (4). To do so, we define $\mathbf{Y} = \mathbf{X} - \mathbf{F}\mathbf{B}^\intercal$ and find the candidate cluster-specific loadings for cluster k as:

$$\tilde{\mathbf{\Lambda}}^{(k)} = \mathbf{Y}^\intercal \mathbf{\Phi}^{(k)} (\mathbf{\Phi}^{(k)\intercal} \mathbf{\Phi}^{(k)})^{-1}, \tag{6}$$

where $\mathbf{\Phi}^{(k)}$ are the cluster-specific factor realizations for cluster k, as defined previously. Using the estimates we calculate the loss matrix $L_{ik} = l(X_i; \mathbf{F}, \mathbf{B}, \mathbf{\Phi}^{(k)}, \tilde{\mathbf{\Lambda}}^{(k)})$ for each combination of assets $i = 1, \ldots, N$ and clusters $k = 1, \ldots, K$. The clusters are then directly assigned as:

$$\hat{g}_i = \underset{k}{\operatorname{argmin}} \, L_{ik}, \tag{7}$$

meaning that each asset belongs to the cluster whose factors minimize the loss function (4) for that asset. This step can

also be interpreted as a generalization of the assignment step in Lloyd's algorithm for k-means clustering, with C_k cluster-specific factors instead of centroids, and the loss function (4) instead of the Euclidean distance.

2) ESTIMATION OF CLUSTER-SPECIFIC FACTORS

For a given clustering $\mathbf{g} = [g_1, \ldots, g_N]$ and known pervasive factors $\mathbf{F}$ with loadings $\mathbf{B}$, all assets within cluster k are exposed to the cluster-specific factors $\mathbf{\Phi}^{(k)}$ – for that subset of assets, these factors could be considered pervasive. This enables the estimation of the factors using the subset of asset return time series $\mathbf{Y}^{(k)} \in \mathbb{R}^{T \times N_k}$ containing only the N_k time series in cluster k. Following the logic in (5), the factor loadings $\hat{\mathbf{\Lambda}}^{(k)}$ for cluster k are then estimated from the eigenvectors of the largest C_k eigenvalues of the $N_k \times N_k$ matrix $\frac{1}{T}\mathbf{Y}^{(k)\intercal}\mathbf{Y}^{(k)}$. The cluster-specific factor realizations are calculated as $\hat{\mathbf{\Phi}}^{(k)} = \mathbf{Y}^{(k)}\hat{\mathbf{\Lambda}}^{(k)}$.

3) ESTIMATION OF PERVASIVE FACTORS

Given the clustering $\mathbf{g}$ and cluster-specific factors $\mathbf{\Phi}$ with loadings $\mathbf{\Lambda}$, we define $\mathbf{Z} = \mathbf{X} - \mathbf{\Phi}\mathbf{\Lambda}^\intercal$. The pervasive factor loadings $\hat{\mathbf{B}}$ are estimated from the eigenvectors of the largest P eigenvalues of $\frac{1}{T}\mathbf{Z}^\intercal\mathbf{Z}$, and the factor realizations are $\hat{\mathbf{F}} = \mathbf{Z}\hat{\mathbf{B}}(\hat{\mathbf{B}}^\intercal\hat{\mathbf{B}})^{-1}$.

C. MODEL SELECTION

1) ESTIMATING THE NUMBER OF PERVASIVE FACTORS AND CLUSTERS

To estimate the number of pervasive factors P and the number of clusters K from the data, we apply the Ahn-Horenstein eigenvalue ratio (ER) test [54] and propose a method for estimating the number of clusters using a graph (network) of assets. Since the estimates depend on each other, we propose a method in which P and K are estimated from several considered candidates, based on a joint criterion.

The ER approach sorts the eigenvalues of the data correlation matrix in a descending order and defines the eigenvalue ratio:

$$\eta_i^{(p)} = \xi_i / \xi_{i+1}, \tag{8}$$

where ξ_i is the i-th largest eigenvalue. The test detects the shift from the common factor part of the spectrum to the idiosyncratic part [54], as seen in Figure 2. The larger the ER ratio $\eta_i^{(p)}$, the more evidence in favor of i being the correct number of pervasive factors. Therefore, in the ER test the estimated number of factors is $\hat{P} = \text{argmax}_i \eta_i^{(p)}$. However, in this case, the shift will be between the pervasive factor part and the cluster-specific factor part (since the cluster-specific factors affect less assets, the eigenvalues corresponding to them will be lower than those representing pervasive factors). Moreover, instead of just picking the maximum value of ER, to obtain a more robust final estimate and avoid discarding potentially better solutions, we select a number of candidates for the the number of pervasive factors $\tilde{P}_1, \ldots, \tilde{P}_n$, corresponding to the n largest ratios $\tilde{\eta}_1^{(p)}, \ldots, \tilde{\eta}_n^{(p)}$.

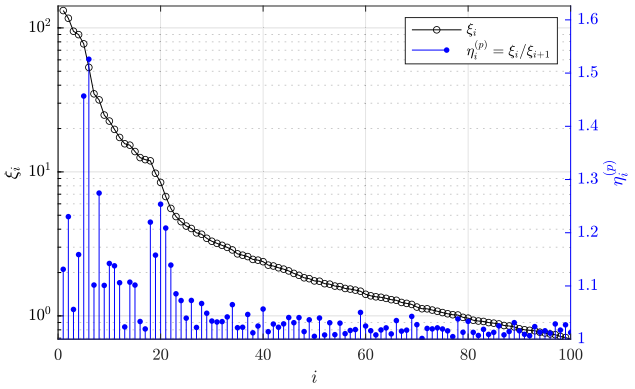


FIGURE 2. The first 100 eigenvalues and eigenvalue ratios of a sample correlation matrix. The best candidates for *P* in this case are 5 and 6, as seen in the eigenvalue ratios.

To detect the clusters of data, for each $\tilde{P}_i$ we form an asset graph from the time series $\mathbf{Y} = \mathbf{X} - \hat{\mathbf{F}}\hat{\mathbf{B}}^\intercal$, where $\hat{\mathbf{F}}$ and $\hat{\mathbf{B}}$ are estimated from the data using the PC estimator. Each of the N nodes in the graph represents an asset and the edges between them depend on a similarity measure $w_{ij} = |\rho(Y_i, Y_j)|$. In order to obtain accurate and robust estimates, the asset graph needs to reflect the following properties:

i) assets which are very close (having a high w_{ij}) should be connected,
ii) assets in the same cluster should have a short path between them (high connectivity clusters),
iii) the spectral properties of the graph need to be stable, since the estimation depends on the Laplacian spectrum.

The first property is found in the ϵ-neighborhood (ϵ-N) graph, constructed simply by keeping only the edges $w_{ij} > \epsilon$ which are above a certain threshold ϵ. The second property is found in the k-nearest neighbors (kNN) graph constructed by keeping the k edges with highest values of w_{ij} for each node $i = 1, \ldots N$, commonly used in spectral clustering [55]. Finally, since the ϵ-N and kNN graphs may not always be connected graphs (they may contain multiple connected components), their spectral properties may differ depending on the number of connected components, we also consider the *maximum spanning tree* (MST) graph, which always consists of one connected component. To obtain the maximum spanning tree, the edges w_{ij} are multiplied by -1 and Kruskal's algorithm for minimum spanning tree construction is applied. The final asset graph is a combination of the three approaches:

$$\mathbf{W} = \mathbf{W}^{(\epsilon N)} \cup \mathbf{W}^{(kNN)} \cup \mathbf{W}^{(MST)}, \tag{9}$$

with $\mathbf{W}^{(\epsilon N)}$, $\mathbf{W}^{(kNN)}$, and $\mathbf{W}^{(MST)}$ being the adjacency matrices of the ϵ-N, kNN and MST graphs. The asset graph has favorable properties from all three methods combined, resulting in a structure shown in Figure 3.

Given the graph adjacency matrix $\mathbf{W}$, the number of clusters can be estimated based on the spectral properties of the

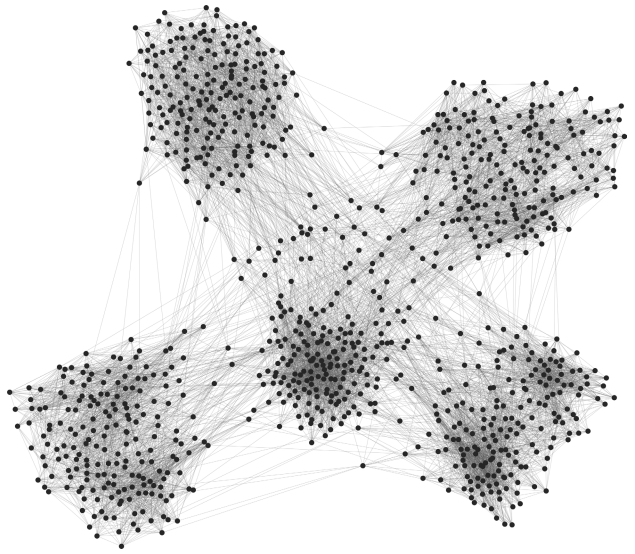


FIGURE 3. An example of the asset graph containing $N = 1000$ nodes, estimated from a sample with exactly 5 clusters – all of which are clearly visible in the graph structure.

asset graph Laplacian:

$$\mathbf{L} = \mathbf{D} - \mathbf{W}, \tag{10}$$

where $\mathbf{D}$ is the diagonal node degree matrix $\mathbf{D} = \text{diag}(d_1, \ldots, d_N)$. The number of zero valued eigenvalues in the spectrum of the Laplacian matrix is equal to the number of connected components in the graph. Since the proposed graph $\mathbf{W}$ contains the MST, it will always have one connected component – thus the Laplacian $\mathbf{L}$ will have exactly one eigenvalue equal to zero. The $K-1$ eigenvalues $\psi_2, \ldots, \psi_K$ will be close to zero for a graph containing K clusters (the end case being a graph divided into K connected components which will have exactly K eigenvalues equal to zero).[3] To find the number of clusters in the graph, eigenvalues ψ_i of the Laplacian are sorted in an ascending order and, in analogy with (8), we define the the Laplacian eigenvalue ratio (LER) $\eta_i^{(c)} = \psi_i/\psi_{i+1}$, as seen in Figure 4. Just like in the ER test, the number of clusters is estimated as the i which maximizes the LER: $\hat{K} = \text{argmax}_i \eta_i^{(c)}$.

Combining the two approaches, for each $\tilde{P}_i$ with its corresponding ER $\tilde{\eta}_i^{(p)}$, we have a $\tilde{K}_i$ with its LER $\tilde{\eta}_i^{(c)}$. To decide between the candidate numbers of pervasive factors and clusters, we focus on the ER and LER values - the higher the ER and LER, the more evidence towards them representing the correct P and K. Thus, taking both into account, we propose a heuristic rule to select $\hat{P} = \tilde{P}_j$ and $\hat{K} = \tilde{K}_j$, where $j = \text{argmax}_i \ \tilde{\eta}_i^{(c)} \cdot \tilde{\eta}_i^{(p)}$. By doing so, among the similar candidates for P, we select those which yield better resolution for selecting K. The overview of the proposed model selection algorithm is given in Algorithm 1.

[3] In spectral graph theory, the second smallest eigenvalue of the Laplacian (also called the Fiedler eigenvalue), and the corresponding eigenvector (also called the algebraic connectivity) are in the focus of research on the bisection of graphs – here instead of bisecting graphs into two components, we consider dividing graphs into a number of clusters, and thus consider the $K-1$ smallest non-zero eigenvalues.

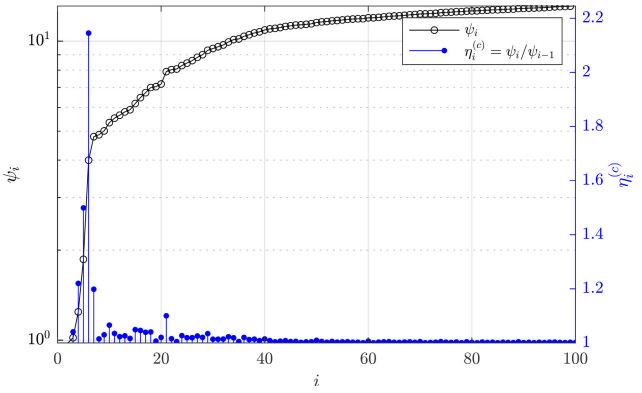


FIGURE 4. The first 100 eigenvalues and Laplacian eigenvalue ratios (LER) of the Laplacian matrix of a sample asset graph. The first eigenvalue and LER are omitted since the first eigenvalue is zero (the graph has one connected component). The best candidate for K in this case is 5, as seen by the LER.

Algorithm 1 Model Selection

estimate candidates $\tilde{P}$ and $\tilde{\eta}^{(p)}$ from $\mathbf{X}$
foreach $\tilde{P}_i$ **do**
  estimate $\tilde{P}_i$ factors $\hat{\mathbf{F}}$ and loadings $\hat{\mathbf{B}}$ from $\mathbf{X}$
  construct the asset graph from $\mathbf{Y} = \mathbf{X} - \hat{\mathbf{F}}\hat{\mathbf{B}}^\intercal$
  estimate $\tilde{K}_i$ and $\tilde{\eta}_i^{(c)}$ from the graph Laplacian
end
$j \leftarrow \text{argmax}_i \ \tilde{\eta}_i^{(c)} \cdot \tilde{\eta}_i^{(p)}$
$\hat{P} \leftarrow \tilde{P}_j$
$\hat{K} \leftarrow \tilde{K}_j$

2) ESTIMATING THE NUMBER OF CLUSTER-SPECIFIC FACTORS

During the cluster assignment step, the clusters with a larger number of cluster-specific factors C_k will naturally attract more assets (since the time series in clusters with more cluster-specific factors will tend to have a lower value of L_{ik}), and the cluster membership estimates will be biased towards them. Even knowing the right number of cluster-specific factors in each cluster will not guarantee that the assets will be associated with the correct clusters. Our algorithm resolves this issue by having the number of clusters equal for all clusters $C_k = C_0, \forall k$ during the entire iterative clustering procedure. Given the estimated clustering $\hat{\mathbf{g}}$, the N_k time series $\mathbf{Y}^{(k)} = \mathbf{X}^{(k)} - \mathbf{F}\mathbf{B}^{(k)\intercal}$ will have a pure factor structure, containing C_k factors, and C_k can be estimated using the ER estimator. After C_k is estimated for each cluster, another phase of the iterative procedure is run, containing only the update step for the cluster-specific factor estimates and the pervasive factor estimates. An overview of the entire procedure, including clustering, factor estimation and the estimation of the number of cluster-specific factors is given in Algorithm 2.

Algorithm 2 Clustering and Estimation of Pervasive and Cluster-Specific Factors

initialize $\hat{\mathbf{F}}, \hat{\mathbf{B}}, \hat{\mathbf{\Phi}}, \hat{\mathbf{\Lambda}}, \hat{\mathbf{g}}$
set $C_k = C_0$ for all clusters $k = 1, \ldots, K$
while *clustering convergence criteria not met* **do**
  update cluster membership:
    given $\hat{\mathbf{F}}, \hat{\mathbf{B}}, \hat{\mathbf{\Phi}}$, estimate $\tilde{\mathbf{\Lambda}}$ from $\mathbf{Y} = \mathbf{X} - \hat{\mathbf{F}}\hat{\mathbf{B}}^\intercal$
    calculate $L_{ik} = l(X_i; \hat{\mathbf{F}}, \hat{\mathbf{B}}, \hat{\mathbf{\Phi}}^{(k)}, \tilde{\mathbf{\Lambda}}^{(k)})$
    set $\hat{g}_i \leftarrow \underset{k}{\operatorname{argmin}}\, L_{ik}$
  update cluster-specific factors:
    given $\hat{\mathbf{F}}, \hat{\mathbf{B}}, \hat{\mathbf{g}}$, calculate $\mathbf{Y}^{(k)} = \mathbf{X}^{(k)} - \hat{\mathbf{F}}\hat{\mathbf{B}}^{(k)\intercal}$
    estimate $\hat{\mathbf{\Phi}}^{(k)}, \hat{\mathbf{\Lambda}}^{(k)}$ for all clusters $k = 1, \ldots, K$
    set $\hat{\mathbf{\Phi}} \leftarrow [\hat{\mathbf{\Phi}}^{(1)}, \ldots, \hat{\mathbf{\Phi}}^{(K)}]$,
  $\hat{\mathbf{\Lambda}} \leftarrow [\hat{\mathbf{\Lambda}}^{(1)}, \ldots, \hat{\mathbf{\Lambda}}^{(K)}]$
  update pervasive factors:
    given $\hat{\mathbf{\Phi}}, \hat{\mathbf{\Lambda}}$, calculate $\mathbf{Z} = \mathbf{X} - \hat{\mathbf{\Phi}}\hat{\mathbf{\Lambda}}^\intercal$
    estimate and set $\hat{\mathbf{F}}, \hat{\mathbf{B}}$ from $\mathbf{Z}$
end
given $\hat{\mathbf{F}}, \hat{\mathbf{B}}, \hat{\mathbf{g}}$ update C_k for all clusters $k = 1, \ldots, K$
while *error convergence criteria not met* **do**
  update cluster-specific factors:
    given $\hat{\mathbf{F}}, \hat{\mathbf{B}}, \hat{\mathbf{g}}$, calculate $\mathbf{Y}^{(k)} = \mathbf{X}^{(k)} - \hat{\mathbf{F}}\hat{\mathbf{B}}^{(k)\intercal}$
    estimate $\hat{\mathbf{\Phi}}^{(k)}, \hat{\mathbf{\Lambda}}^{(k)}$ for all clusters $k = 1, \ldots, K$
    set $\hat{\mathbf{\Phi}} \leftarrow [\hat{\mathbf{\Phi}}^{(1)}, \ldots, \hat{\mathbf{\Phi}}^{(K)}]$, $\hat{\mathbf{\Lambda}} \leftarrow [\hat{\mathbf{\Lambda}}^{(1)}, \ldots, \hat{\mathbf{\Lambda}}^{(K)}]$
  update pervasive factors:
    given $\hat{\mathbf{\Phi}}, \hat{\mathbf{\Lambda}}$, calculate $\mathbf{Z} = \mathbf{X} - \hat{\mathbf{\Phi}}\hat{\mathbf{\Lambda}}^\intercal$
    estimate and set $\hat{\mathbf{F}}, \hat{\mathbf{B}}$ from $\mathbf{Z}$
end

D. INITIALIZATION AND CONVERGENCE CRITERION

For the initialization, the P pervasive factors $\mathbf{F}$ and loadings $\mathbf{B}$ are estimated from the data $\mathbf{X}$ first, then the asset graph is constructed from $\mathbf{Y} = \mathbf{X} - \mathbf{F}\mathbf{B}^\intercal$, based on which a spectral clustering method is used to obtain the initial clustering. Finally, for the given clustering $\mathbf{g}$ and pervasive factors $\mathbf{F}$ with loadings $\mathbf{B}$, we estimate the cluster-specific factors using the data $\mathbf{Y}^{(k)}$, for each cluster $k = 1, \ldots, K$. In both phases (the clustering and the cluster-specific factor estimation), the algorithm stops when there are no cluster changes and the reduction in the loss function $l^{(i)} - l^{(i-1)}$ is less than $10^{-5} \cdot \sigma_m^2$, where σ_m^2 is the median variance of all time series $\mathbf{X}$.

IV. SIMULATIONS AND DATA

To verify the validity of our proposed approach and test the estimation algorithm, we define several data-generating processes (DGP) which correspond to the assumed factor model structures. To obtain a model in the form of 2, we generate random factor loadings. The elements of the pervasive loadings matrix $\mathbf{B}$ are drawn from a uniform random distribution with mean 0 and variance 1. For the cluster-specific loadings matrix $\mathbf{\Lambda}$, the elements $\Lambda_i^{(k)}$ are random (also uniform with mean 0 and variance 1) if asset i belongs to cluster k, and are zero otherwise. We form clusters which are all of equal size $N_k = N/K$. Since the approximate factor model allows for some off-diagonal elements in the covariance of residuals, we also generate random sparse covariance matrices with a given idiosyncratic variance σ_e^2 on the diagonal (for the details on the procedure for generating positive semi-definite sparse covariance matrices, see the Appendix VI). Given the factor loadings and the idiosyncratic components, the asset mean and covariance can then be calculated as

$$\begin{aligned} \boldsymbol{\mu}_X &= \boldsymbol{\mu}_F \mathbf{B}^\intercal + \boldsymbol{\mu}_\Phi \mathbf{\Lambda}^\intercal, \\ \mathbf{\Sigma}_X &= \mathbf{B}\mathbf{\Sigma}_F \mathbf{B}^\intercal + \mathbf{\Lambda}\mathbf{\Sigma}_\Phi \mathbf{\Lambda}^\intercal + \mathbf{\Sigma}_e, \end{aligned} \tag{11}$$

where $\boldsymbol{\mu}_F$ are the means $\mathbf{\Sigma}_F$ is the covariance of P pervasive factors, while $\boldsymbol{\mu}_\Phi$ are the means and $\mathbf{\Sigma}_\Phi$ is the covariance of Q cluster-specific factors. In our simulations, the means are all zero, and the covariances are both diagonal matrices with equal variances σ_F^2 and σ_Φ^2 on the diagonal. The full set of simulation parameters is given in Table 1.

TABLE 1. Simulation parameters.

Parameter	Symbol	Value
Number of assets	N	1000
Number of pervasive factors	P	5
Number of clusters	K	5
Number of cluster-specific factors	C	$[1, 2, 3, 4, 5]$
Pervasive factor variance	σ_F^2	0.1
Cluster-specific factor variance	σ_Φ^2	0.1
Idiosyncratic variance	σ_e^2	0.5

To simulate asset returns, we use the asset mean and covariance (11) to simulate T returns, drawing from the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_X, \mathbf{\Sigma}_\mathbf{X})$ and the Student's t-distribution, with 6 degrees of freedom $t_5(\boldsymbol{\mu}_X, \mathbf{\Sigma}_\mathbf{X})$ and 4 degrees of freedom $t_3(\boldsymbol{\mu}_X, \mathbf{\Sigma}_\mathbf{X})$. Although such models and estimation methods are often tested using simulations of normally distributed data [18], [22], we additionally use the heavy-tailed Student's distribution, since they replicate the properties of financial returns, as seen in Figure 5.

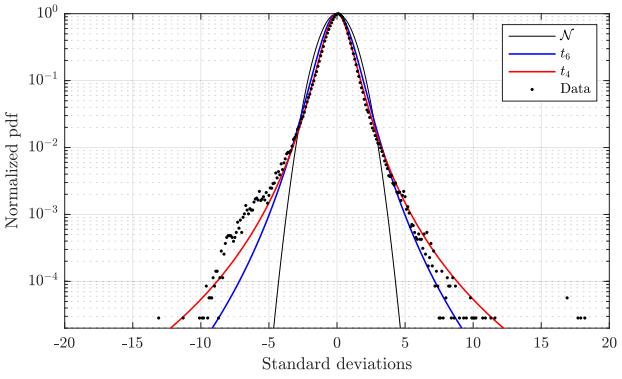


FIGURE 5. The normalized pdfs of the three considered theoretical distributions, together with the empirical histogram of the weekly returns of NASDAQ global equity indices between 2005 and 2020.

In addition, we also use two dataset containing weekly financial return time series. Firstly we use a dataset of NASDAQ global equity indices between 2005 and 2020 [56]. The original dataset contains a large number of redundant time series, from which we select only the total return NASDAQ indices available in the considered period, leaving us with $N = 797$ assets. We also consider a dataset of $N = 621$ ETFs[4] available between 2010 and 2020. The data for this dataset is obtained by downloading historical return data using Yahoo Finance for the ETF tickers available on etf.com, and then again selecting only those time series which have price data for the entire considered period. Both of these datasets cover a wide range of exchanges, countries and specific sectors, and can be used to represent and study the latent risk factors in global financial markets.

V. RESULTS

A. SIMULATION RESULTS

We verify the proposed algorithm and measure the accuracy of the clustering and model selection methods using the proposed simulation scenario. Using the parameters defined in Table 1), we randomly generate models and for each random model we simulate time series of length T.

We apply the proposed method given the correct P and K, and measure the quality of clustering and the accuracy of the detected number of cluster-specific factors C. The estimated clustering $\hat{\mathbf{g}}$ and ground truth clustering $\mathbf{g}$ are compared using the Rand statistic and Jaccard coefficient (see Appendix C). Moreover, for both of these cluster validation measures and any pair of clustering methods we define a paired statistical test[5] in order to test the hypothesis H_0: *There is no difference between two clustering methods*, with a one-sided alternative H_1: *Method 2 outperforms Method 1*. For each randomly generated model $m = 1, \ldots, m_{\max}$, the considered clustering methods are applied and the cluster validation measure is calculated for both results (for instance $\text{Rand}_1(m)$ and $\text{Rand}_2(m)$), then the p-value is calculated as the fraction of pairs for which Method 2 outperforms Method 1 (in this example, the fraction of samples for which $\text{Rand}_2(m) > \text{Rand}_1(m)$). We repeat this procedure for the both cluster validation measures, pairing our proposed model-based method with several commonly used clustering approaches (k-means algorithm, spectral clustering [55], and the Ando-Bai estimation procedure [23]). The k-means method uses $1 - |\rho_{ij}|$ as a distance measure, and the spectral clustering method employs the proposed asset graph estimated directly from $\mathbf{X}$. The Ando-Bai procedure iteratively estimates clusters and latent factors, but using a procedure which does not account for the bias in clusters with different numbers of cluster-specific factors.

We generate a number of $m_{\max} = 1000$ models and for each we simulate time series realizations of length $T = 1000, 500, 250$, and apply the considered clustering methods and tests. The average Rand and Jaccard statistics, as well as the p-values of the paired resampling tests (comparing the proposed model-based method with each of the other considered clustering methods) are shown in Table 2. These results demonstrate the advantage of the proposed model-based approach, as well as the fact that the existence of pervasive factors may severely hinder clustering accuracy when they are not taken into account. Moreover, in the paired tests, the proposed method outperformed the considered methods for virtually all of the 1000 resampled model realizations (the p-values of < 0.001 mean that in the $m_{\max} = 1000$ simulated models, none were found for which the considered method outperformed our algorithm). To better visualize the paired comparison for these two methods across the simulations, we show the Rand statistic for the Ando-Bai and the proposed model-based method across all 1000 simulations in Figure 6.

In order to look into the bias in clustering when the number of cluster-specific factors differ between clusters, in Figure 7 we also look into the average number of assets in clusters with different numbers of cluster-specific factors, given by two different approaches. The first uses the real C_k as the number of cluster-specific factors in each cluster (corresponding to the Ando-Bai method [23]), while the second uses the fixed C_0 in each cluster during the clustering phase. The bias towards the clusters with a larger C_k is evident and might be a large source of inaccuracy in the clustering procedure, while our method with C_0 seems to provide accurate clustering without any evident bias in the cluster sizes. These results are obtained for the $T = 500$ window and the t_4 distribution, but hold for all of the considered combinations.

We also evaluate the performance of the model selection method, using the same simulation environment and the simulated time series lengths. In addition to measuring the percentage of correctly estimated number of pervasive factors, clusters and cluster-specific factors, we also measure the mean absolute deviation for each of these. The results are shown in Table 3. The accuracy of the proposed model selection method is remarkably high, even when presented with heavy tailed data and short time window length. Only the number of pervasive factors seems to suffer a bit in case of the t_4 distribution and $T = 250$ – nevertheless, the accuracy of 90% achieved for this case is high.

B. FINANCIAL MARKET DATA

The simulation results confirm the ability of the proposed method to provide accurate estimates, even in the presence of correlated residuals, heavy tails, and high-dimensional sample data. However, in real financial market data, such as the NASDAQ global equity indices and ETF data, the latent factors are unknown, as well as the clustering and the number of clusters and latent factors. The proposed method allows us

[4]ETF stands for *exchange-traded fund* – these are relatively novel assets which mostly follow certain known index methodologies and offer exposure to certain asset classes such as equities, commodities, bonds, while reducing costs for investors and increasing transparency.

[5]Since the models are randomly generated, each model realization presents different conditions for the considered clustering methods, which need to be taken into account in a paired fashion.

TABLE 2. Clustering performance on simulation data for the proposed method and other considered clustering techniques, using different simulation time window lengths and data distributions. The brackets below each value contain the *p*-value of the paired resampling test of the considered method compared to the proposed model-based algorithm. All of the values are obtained using simulation parameters given in Table 1.

	Rand			Jaccard		
$T = 1000$	$\mathcal{N}$	t_6	t_4	$\mathcal{N}$	t_6	t_4
k-means	68.03% (< 0.001)	68.03% (< 0.001)	67.98% (< 0.001)	11.09% (< 0.001)	11.10% (< 0.001)	11.12% (< 0.001)
Spectral clust.	77.02% (< 0.001)	74.10% (< 0.001)	74.10% (< 0.001)	29.34% (< 0.001)	23.78% (< 0.001)	23.77% (< 0.001)
Ando-Bai	89.99% (< 0.001)	88.90% (< 0.001)	88.34% (< 0.001)	64.38% (< 0.001)	61.16% (< 0.001)	59.40% (< 0.001)
Model-based	**99.03%**	**98.83%**	**98.53%**	**95.27%**	**94.35%**	**92.95%**
$T = 500$	Rand			Jaccard		
	$\mathcal{N}$	t_6	t_4	$\mathcal{N}$	t_6	t_4
k-means	68.03% (< 0.001)	68.01% (< 0.001)	67.94% (< 0.001)	11.10% (< 0.001)	11.11% (< 0.001)	11.15% (< 0.001)
Spectral clust.	75.94% (< 0.001)	71.91% (< 0.001)	71.90% (< 0.001)	27.07% (< 0.001)	20.00% (< 0.001)	19.99% (< 0.001)
Ando-Bai	89.31% (0.003)	88.40% (0.003)	87.73% (< 0.001)	62.36% (0.003)	59.57% (0.003)	57.80% (< 0.001)
Model-based	**98.75%**	**98.39%**	**98.01%**	**93.92%**	**92.31%**	**90.59%**
$T = 250$	Rand			Jaccard		
	$\mathcal{N}$	t_6	t_4	$\mathcal{N}$	t_6	t_4
k-means	68.02% (< 0.001)	67.98% (< 0.001)	67.90% (< 0.001)	11.11% (< 0.001)	11.13% (< 0.001)	11.17% (< 0.001)
Spectral clust.	73.68% (< 0.001)	69.75% (< 0.001)	69.74% (< 0.001)	22.69% (< 0.001)	16.26% (< 0.001)	16.26% (< 0.001)
Ando-Bai	88.09% (0.001)	87.26% (0.001)	86.44% (< 0.001)	58.86% (0.001)	56.43% (0.001)	54.47% (< 0.001)
Model-based	**98.13%**	**97.58%**	**97.15%**	**91.10%**	**88.62%**	**86.75%**

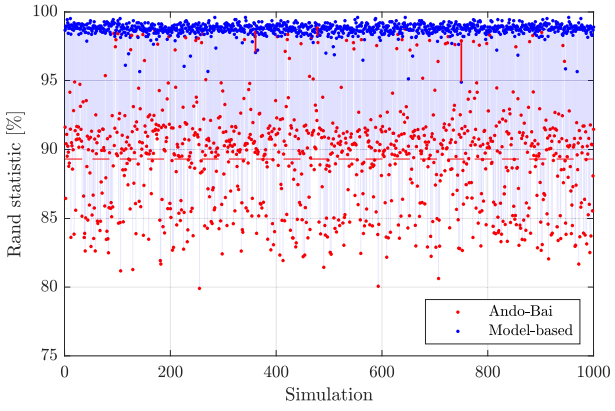


FIGURE 6. The Rand statistic for all the 1000 simulations and $T = 500$, given for the Ando-Bai and our proposed method. The two statistics for each simulation are connected with a transparent blue line if our method outperforms the Ando-Bai method, and a red line otherwise (only 3 samples in this case). The dashed lines represent the average values of the statistics, corresponding to the values in Table 2.

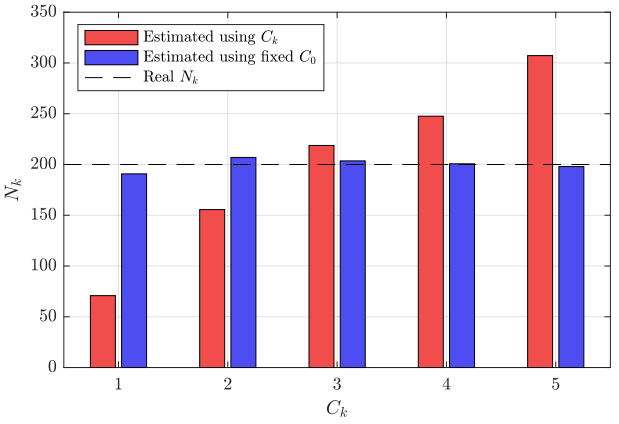


FIGURE 7. The sizes of clusters (number of assets N_k for different numbers of cluster-specific factors C_k, given by two estimation methods. The real number of assets in each cluster is known in the simulation and is equal to $N_k = 200$ for each k.

to study and estimate these from the data directly. We first focus on two distinct periods in the NASDAQ dataset: Figure 8 shows the asset graph for the period 2007-2009 around the global recession, and Figure 9 show the graph for the subsequent period 2010-2020 which corresponds to one of the strongest and longest bull markets in the

TABLE 3. Model selection accuracy on simulation data over different simulation time window lengths.

	$T = 1000$					
	Acc.			MAD		
	$\mathcal{N}$	t_6	t_4	$\mathcal{N}$	t_6	t_4
P	100%	100%	95%	0.00	0.00	0.05
K	100%	100%	100%	0.00	0.00	0.00
C_k	95.2%	100%	99.6%	0.10	0.00	0.01
	$T = 500$					
	Acc.			MAD		
	$\mathcal{N}$	t_6	t_4	$\mathcal{N}$	t_6	t_4
P	100%	100%	93%	0.00	0.00	0.08
K	99%	100%	100%	0.01	0.00	0.00
C_k	100%	99.4%	97.4%	0.00	0.01	0.03
	$T = 250$					
	Acc.			MAD		
	$\mathcal{N}$	t_6	t_4	$\mathcal{N}$	t_6	t_4
P	100%	98%	90%	0.00	0.02	0.14
K	100%	98%	98%	0.00	0.02	0.04
C_k	99.8%	99.8%	98.4%	0.01	0.01	0.02

FIGURE 8. The asset graph for NASDAQ indices between 2007 and 2009.

history of financial markets. In both graphs, some common clusters emerge (shown in same colors on both graphs): European markets (pink), Brazil and Latin America (purple), North America and global developed market indices (blue), Asian emerging markets (teal), Middle East and Africa (darker green), Asian developed markets (light green). The 2007-2009 graph contains another cluster for India and New Zealand (yellow), and European emerging markets (red) - both of which are encapsulated within other clusters in the 2010-2020 graph. The clusters found in the ETF dataset reflect different asset classes, rather than geographic origin, mainly because of the assets within – they mostly follow broad indices from various countries, but differ between stocks, bonds, commodities and others (to remain concise in

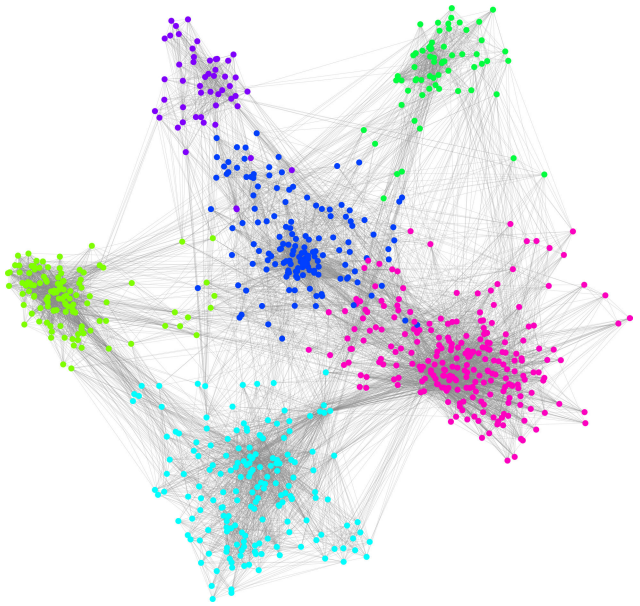


FIGURE 9. The asset graph for NASDAQ indices between 2010 and 2020.

this section, we do not display them additionally). In addition to serving as a sanity check for the meaning behind the estimated clusters, these results suggest that the clusters and latent factor structures in the data may change through time. This is why, in the rest of our analysis, we focus on rolling time window estimates of the latent factors and clusters, and use out-of-sample data from subsequent future windows to measure the quality of our estimates.

To validate our approach on the available financial market data, we propose a backtesting framework in which the model is estimated on return time series $\mathbf{X}$ on look-back windows of fixed length T. Using the estimated model (mainly, the factor loadings $\hat{\mathbf{A}} = [\hat{\mathbf{B}}, \hat{\mathbf{\Lambda}}]$), a reconstruction of any time series $\mathbf{X}'$ can be obtained using the $N \times N$ filtering matrix of rank $P + Q$: $\hat{\mathbf{M}} = \hat{\mathbf{A}}(\hat{\mathbf{A}}^\intercal\hat{\mathbf{A}})^{-1}\hat{\mathbf{A}}^\intercal$. This enables us to obtain a reconstruction of any out-of-sample time series $\mathbf{X}'$ using the in-sample loadings estimates from which $\hat{\mathbf{M}}$ is calculated:

$$\hat{\mathbf{X}}' = \mathbf{X}'\hat{\mathbf{M}}. \tag{12}$$

Using the reconstructed time series $\hat{\mathbf{X}}$, we can calculate the unexplained variance in each asset (either for the in-sample or out-of-sample data):

$$V_i = \frac{\sum_t^T (X_{ti} - \hat{X}_{ti})^2}{\sum_t^T (X_{ti} - \overline{X}_i)^2}, \tag{13}$$

where X_{ti} is the realization of time series i at time t, $\hat{X}_{ti}$ is the model reconstruction given by (12), and $\overline{X}_i$ is the sample mean of time series i. This framework enables us to apply cross-validation principles for estimating the out-of-sample model performance. Specifically, the model estimates from a look-back window of length T are used to reconstruct the future out-of-sample returns on a look-ahead window of

TABLE 4. Unexplained variances of the model estimates compared to the PC estimator given different lengths of the look-back windows, on both considered datasets.

$T = 4$ years, $T' = 1$ year						
	NASDAQ data			ETF data		
	V	V'	d	V	V'	d
Model	27.40%	**34.88%**	**28.05%**	25.05%	**31.92%**	**29.14%**
PC	**26.69%**	36.95%	39.52%	**24.22%**	32.00%	34.71%
$T = 2$ years, $T' = 1$ year						
	NASDAQ data			ETF data		
	V	V'	d	V	V'	d
Model	30.43%	**38.00%**	**28.66%**	24.34%	**29.79%**	**23.79%**
PC	**29.34%**	39.66%	41.15%	**23.25%**	31.34%	35.68%
$T = 1$ year, $T' = 1$ year						
	NASDAQ data			ETF data		
	V	V'	d	V	V'	d
Model	31.29%	**39.62%**	**33.03%**	25.44%	**32.44%**	**32.39%**
PC	**29.62%**	41.33%	47.01%	**24.47%**	34.10%	44.83%

length T' – using these we can measure the average unexplained variance $V = \frac{1}{N}\sum_i^N V_i$ both in-sample and out-of-sample (denoted V_i and V_i', respectively).

In addition, to measure the decline in out-of-sample model performance, we calculate the average percentage deterioration in out-of-sample vs. in-sample unexplained variance: $d = 1/N \sum_i^N V_i'/V_i$. We compare the proposed estimation method with the PC estimator, where the number of components for the PC estimator is selected so that it explains at least the amount of variance explained by the proposed model.

The results in Table 4 demonstrate that the proposed approach finds relevant estimates of latent factors in the data which outperform the PC estimates in out-of-sample data, for both considered datasets. Even though the PC estimator yields the lowest in-sample unexplained variance V, the PC estimates deteriorate much more than the proposed model, as seen in the out-of-sample unexplained variance V' and the average deterioration d. Moreover, all of these results are in line with other econometric and unsupervised learning studies which find that approximately 30-50% of variance in financial data corresponds to idiosyncratic components [37], [51]. In addition, we find that the model performance in terms of unexplained variance deteriorates fairly less than the PC estimates, suggesting that the proposed estimation method finds more persistent and relevant latent factors in high-dimensional financial time series. In other words, the proposed model-based estimation method generalizes very well to out-of-sample data. This result is expected since the proposed method utilizes the assumed clustering structures within the markets to reduce the number of parameters, thus providing a type of structural regularization of the estimates.

To demonstrate the applicability of the proposed method to risk modelling and portfolio optimization, we consider global minimum variance (GMV) portfolios [51], obtained by solving the following problem:

$$\begin{aligned} &\min_{\mathbf{w}} \mathbf{w}^\intercal \mathbf{\Sigma} \mathbf{w} \\ &\text{s. t. } \mathbf{w}^\intercal \mathbf{1} = 1, \end{aligned} \tag{14}$$

where $\mathbf{\Sigma}$ is the asset covariance matrix. Since the estimation procedure only depends on the covariance, it is a suitable way to demonstrate the ability of the proposed method to provide reliable estimates of asset risk, and is often used to benchmark covariance estimation methods [21]. The covariance matrix is obtained using the expression (11), based on latent factors estimated on a look-back window of length T. The idiosyncratic covariance $\mathbf{\Sigma}_e$ is estimated using a thresholding approach, as described in the Appendix VI. At each time step, the estimated GMV portfolios $\mathbf{w}$ are held on a look-ahead window of length T' and their risks are measured as the realized volatility: $\sqrt{\mathbf{w}^\intercal \mathbf{\Sigma}' \mathbf{w}}$. We compare the volatilities GMV portfolios estimated using the empirical covariance and the covariances calculated using the latent factors estimated by PC and the proposed method. We also use all of the out-of-sample returns of the portfolios built using the PC estimator based covariance and the model based covariance, and apply a test for the equality of their variances. Since these data are paired (measured at same timesteps and thus dependent on some common market factors), and may exhibit heavy tails, we apply a non-parametric version of the Morgan-Pitman test for the equality of variances with paired data, proposed by McCulloch [57]. This test for $H_0 : \sigma_x^2 = \sigma_y^2$ uses variables $u = x + y$ and $v = x - y$, and H_0 is rejected if the Spearman correlation coefficient between u and v is significant. The out-of-sample GMV portfolio volatilities and the p-value for the equality of variance tests (comparing our method with the PC estimator) are given in Table 5.

The results demonstrate that not only the out-of-sample portfolio risk is reduced by implementing latent factor models

TABLE 5. Out-of-sample GMV portfolio volatilities for the portfolios estimated using the covariance matrices obtained by the considered estimators. The *p*-values of the McCulloch non-parametric version of the Morgan-Pitman test are given in brackets.

		Emp.	PC	Model
NASDAQ data	$T = 4$ years	1.79	0.53	**0.45** (0.014)
	$T = 2$ years	2.87	0.57	**0.38** $(1.6 \cdot 10^{-18})$
	$T = 1$ year	4.31	0.32	**0.26** $(3.9 \cdot 10^{-8})$
ETF data	$T = 4$ years	0.48	0.33	**0.27** (0.370)
	$T = 2$ years	0.92	0.57	**0.37** $(1.1 \cdot 10^{-5})$
	$T = 1$ year	1.19	0.57	**0.27** $(1.3 \cdot 10^{-5})$

as opposed to empirical estimates, but also that the proposed method works best at reducing out-of-sample portfolio risk. All of the out-of-sample portfolio variances are significantly reduced in comparison to the PC estimator, except for the $T = 4$ years case in the ETF dataset, where the variance is still reduced but the statistical evidence is not strong enough to reject the equality of variances. Moreover, these results suggest that the empirical covariance based portfolios deteriorate with reducing the look-back time window length, but the model-based covariance estimates yield portfolios which perform very well in short time windows. Evidently, accurate estimates of latent factors allow for the usage of shorter time windows for estimation, which in turn may improve the properties of optimal portfolios since they adapt more quickly to new market conditions.

The results presented in this section demonstrate that the proposed method yields relevant and robust estimates of latent pervasive and cluster-specific factors, which can be applied to global equity market data with the goal of modelling and managing financial risk.

VI. CONCLUSION

In this paper we consider latent factor models of high-dimensional financial time series with pervasive and cluster-specific factors. We propose an estimation method which performs time series clustering and estimates latent pervasive and cluster-specific factors iteratively. In order to estimate the unknown number of clusters and latent pervasive and cluster-specific factors, we also propose a model selection method based on the asset correlation matrices and asset graphs.

We test the method using several data generating processes under the approximate factor model assumptions, featuring heavy tailed returns with some off-diagonal correlations of residuals. The simulation study shows that the proposed method yields very accurate clustering results, even for the most severe high-dimensional setting and heavy-tailed distributions. Moreover, we show that the proposed two-phase model-based method estimates clusters which are not biased towards those clusters with a larger number of cluster-specific factors, as is the case with other clustering methods using cluster-specific factors. In addition, the simulation study results suggest that the proposed model selection method provides stable and accurate estimates of the number of clusters, latent pervasive, and latent cluster-specific factors.

We also apply our methods to datasets of return time series of NASDAQ indices and ETFs in a backtesting approach which allows us to use in-sample model estimates to reconstruct out-of-sample return data. By doing so we cross-validate the unexplained variance, and find that the proposed model-based estimation method, although explaining less variance in-sample than the PC estimator, explains more variance out-of-sample, meaning that it generalizes better and provides more robust estimates. In addition, we apply the method for estimating the asset covariance matrix, based on which we build optimized minimum variance portfolios. The results demonstrate the ability of the proposed method to reduce risk in the minimum variance portfolios, which outperform the portfolios built on empirical and PC estimates of the covariance matrix. Moreover, we find that, whereas the empirical covariances deteriorate with the shorter look-back windows, the model-based estimates thrive in these high-dimensional situations, allowing one to use short look-back windows and thus being more adaptive to changing market conditions.

The results presented in this paper suggest that the clustering assumption in high-dimensional financial time series data holds, and that the model-based estimation method indeed extracts useful information about the latent factor structure. These findings affirm and refine asset pricing theories based on multi-factor models, providing evidence on the clustering structures of latent risk factors. This approach may help shed more light on the intricate latent factor structures in global financial markets, as is demonstrated in our results. Ultimately, the robust estimates of pervasive and cluster-specific factors may be used to improve risk assessment and enhance the out-of-sample performance of portfolios built on the estimated models.

APPENDIX A
SPARSE COVARIANCE ESTIMATION

To estimate a sparse covariance matrix $\mathbf{\Sigma}_e^{(\text{sp.})}$ we start from a full sample covariance estimate $\mathbf{\Sigma}_e^{(\text{est.})}$ and apply an adaptive thresholding technique [58], [59]. A specific threshold is set for each element of the matrix $\mathbf{\Sigma}_e^{(\text{est.})}$, so that the scale (the variance of each time series) is taken into account. The simplest way to do this is to consider the sample correlation matrix $\mathbf{R}_e^{(\text{est.})}$, and apply a fixed threshold ε_r to all elements:

$$\mathbf{R}_e^{(\text{sp.})} = (r_{ij}^{(\text{sp.})})_{N\times N}, \quad r_{ij}^{(\text{sp.})} = \begin{cases} 0, & \text{if } |r_{ij}^{(\text{est.})}| < \varepsilon_r \\ r_{ij}^{(\text{est.})}, & \text{if } |r_{ij}^{(\text{est.})}| \geq \varepsilon_r. \end{cases} \quad (15)$$

The sparse correlation matrix $\mathbf{R}_e^{(\text{sp.})}$ thus contains only elements larger than ε_r or smaller than $-\varepsilon_r$. However, this simple hard thresholding rule does not always produce

positive-definite matrices $\mathbf{R}_e^{(\text{sp.})}$, since certain elements $r_{ik}^{(\text{sp.})}$ and $r_{jk}^{(\text{sp.})}$ may be non-zero (pass above the threshold ε_r, but the element $r_{ij}^{(\text{sp.})}$ may be zero (fall under ε_r). This case may be generalized in the term of graphs - the sparse correlation matrix defines a graph where the edges are only those pairwise correlations which pass the threshold value ε_r. This graph is actually a very sparse graph with a relatively large number of connected components - however each component may not necessarily be fully connected, and as long as they are not, the resulting correlation matrix will not necessarily be positive-definite. Thus, in order to correct this, we iterate over all connected components defined by matrix $\mathbf{R}_e^{(\text{sp.})}$, and assure that all links in those components are non-zero - thus adding additional non-zero elements $r_{ij}^{(\text{sp.})}$ (if $r_{ik}^{(\text{sp.})}$ and $r_{jk}^{(\text{sp.})}$ exist). The resulting new matrix $\mathbf{R}_e^{(\text{sp.})}$ is still sparse, but will be positive-definite. Finally, the sparse covariance matrix is reconstructed from the sparse correlation matrix:

$$\mathbf{\Sigma}_e^{(\text{sp.})} = \sqrt{\text{diag}(\mathbf{\Sigma}_e^{(\text{est.})})}\mathbf{R}_e^{(\text{sp.})}\sqrt{\text{diag}(\mathbf{\Sigma}_e^{(\text{est.})})}. \tag{16}$$

In our simulations, to generate random sparse correlation matrices, we first generate $N \times N$ random matrices $\mathbf{U}$ where each element is drawn from a uniform distribution $\mathcal{U}(0, 1)$, and then define the simulated full correlation matrix $\mathbf{R}_e^{(\text{est.})} = \sqrt{\mathbf{U}\mathbf{U}^\intercal}\mathbf{U}\mathbf{U}^\intercal\sqrt{\mathbf{U}\mathbf{U}^\intercal}$. Based on this correlation matrix and the procedure proposed above, we obtain the sparse correlation matrix $\mathbf{R}_e^{(\text{est.})}$ - the threshold in simulations is set so that approximately 10% of off-diagonal elements are kept non-zero. Finally, the sparse covariance $\mathbf{\Sigma}_e^{(\text{est.})}$ is calculated similarly as in (16) so that the diagonals are the idiosyncratic component variance σ_e^2. In the estimation procedure on the NASDAQ equity indices dataset, we set the threshold equal to the ϵ-neighborhood graph threshold used in the model selection procedure.

APPENDIX B
HYPERPARAMETER SELECTION

The proposed estimation method depends on a handful of hyperparameters: the fixed number of cluster-specific factors C_0 in the clustering phase, number of neighbors k in the kNN graph, and the neighborhood threshold ϵ in the ϵ-N graph. Although the algorithm is not too sensitive to small changes in these hyperparameters, here we provide some quick guidelines on how to select them. Firstly, the algorithm in its clustering phase will not depend too much on the selection of C_0 since the cluster-specific factors in clusters where $C_k < C_0$ will model the C_k latent factors and the rest will be noise, while for clusters where $C_k > C_0$, all C_0 latent factors will be relevant. Nevertheless, we find that a balanced C_0 which is not too large but encapsulates most of the cluster-specific factors will be best, thus we use $C_0 = 4$ in all of our simulations and results. Furthermore, the number of neighbors k in the kNN graph should primarily reflect the size of the clusters we want to detect in the data. These are naturally dependent on the number of time series N - we find that as a rule of thumb, a good choice will be somewhere between $\log N$ and $\sqrt{N}$. In our simulations and results we use: $k = \lceil(\log N + \sqrt{N})/2\rceil$. Finally, for the selection of the neighborhood threshold in the ϵ in the ϵ-N graph, we suggest that both the length of the time series T and their number N are taken into account. Since longer time series will provide smaller estimation error and more accurate correlations between asserts ρ_{ij}, the standard error in the estimates will be reduced and the threshold may be lower - however, the threshold still needs to be above a certain level ρ_0 above which we wish the pairs of assets to be connected in the graph. To account for the statistical uncertainty in the estimation, we propose to set the threshold to the critical value of the approximate Pearson correlation test for the hypothesis $H_0 : \rho_{ij} = \rho_0$:

$$\epsilon = \frac{1+\rho_0}{1-\rho_0}\exp\left(\frac{2z}{\sqrt{T}-3}\right), \tag{17}$$

where we set $\rho_0 = 0.4$, T is the time window length, and z is the $1-\alpha$ quantile of the standard normal distribution $\mathcal{N}(0, 1)$. To account for the fact that the test is applied to all pairwise coefficients $\rho_i j$, we use to Bonferroni correction and set $\alpha = 0.05/\binom{N}{2}$. These values are used in our simulations and results for all different lengths of time windows.

APPENDIX C
CLUSTER VALIDATION

To measure the clustering performance of the proposed method, we calculate the Rand statistic and Jaccard coefficient, both of which are commonly used techniques to measure the agreement between different partitions of the same set and can be used even when there are no class labels available [60]. Given the estimated clustering $\hat{\mathbf{g}}$ and the ground truth clustering $\mathbf{g}$, define the following variables:

$$\begin{aligned}
SS &= \sum_i^N \sum_{j=i+1}^N 1[(\hat{g}_i = \hat{g}_j) \wedge (g_i = g_j)],\\
SD &= \sum_i^N \sum_{j=i+1}^N 1[(\hat{g}_i = \hat{g}_j) \wedge (g_i \neq g_j)],\\
DS &= \sum_i^N \sum_{j=i+1}^N 1[(\hat{g}_i \neq \hat{g}_j) \wedge (g_i = g_j)],\\
DD &= \sum_i^N \sum_{j=i+1}^N 1[(\hat{g}_i \neq \hat{g}_j) \wedge (g_i \neq g_j)],
\end{aligned} \tag{18}$$

where $1[c]$ is an indicator function with value 1 if the condition c in the brackets holds, and 0 otherwise. The variable SS simply counts the number of pairs of assets which belong to the same cluster in both clusterings $\hat{\mathbf{g}}$ and $\mathbf{g}$; SD counts the number of pairs belonging to the same cluster in $\hat{\mathbf{g}}$ and different clusters in $\mathbf{g}$; DS counts the number of pairs belonging to different clusters in $\hat{\mathbf{g}}$ and the same cluster in $\mathbf{g}$; DD counts the number of pairs belonging to different clusters in both clusterings $\hat{\mathbf{g}}$ and $\mathbf{g}$. Given these variables, the Rand statistic

and the Jaccard coefficient can be calculated:

$$\text{Rand} = \frac{SS + DD}{SS + SD + DS + DD},$$
$$\text{Jaccard} = \frac{SS}{SS + SD + DS}. \quad (19)$$

Following the above expression, in our case the Rand statistic simply measures the proportion of pairs which are correctly clustered together or apart, and the Jaccard coefficient measures the intersection of the correctly clustered pairs in proportion to the union of all the pairs of assets. Both of these can be interpreted as focusing on the sets of pairs, rather than the original set of assets, and look into whether the pairwise clustering properties match in the two given clusterings.

ACKNOWLEDGMENT

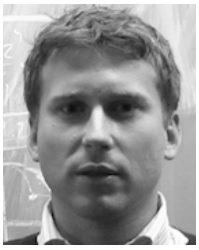
The authors would like to thank the participants and discussants at the International Conference on Quantitative Finance - Forecasting Financial Markets, Venice, Italy 2019, and the IEEE International Symposium on Image and Signal Processing and Analysis, Special Session on Signal Processing and Machine Learning for Finance, Dubrovnik, Croatia, 2019. They would also like to thank the anonymous reviewers for their suggestions which ultimately helped to improve the quality of the article.

REFERENCES

[1] D. Johnston and P. Djurić, "The science behind risk management," *IEEE Signal Process. Mag.*, vol. 28, no. 5, pp. 26–36, Sep. 2011.
[2] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi, "Machine learning for predictive maintenance: A multiple classifier approach," *IEEE Trans. Ind. Informat.*, vol. 11, no. 3, pp. 812–820, Jun. 2015.
[3] F. Barboza, H. Kimura, and E. Altman, "Machine learning models and bankruptcy prediction," *Expert Syst. Appl.*, vol. 83, pp. 405–417, Oct. 2017.
[4] E. F. Fama and K. R. French, "Common risk factors in the returns on stocks and bonds," *J. Financial Econ.*, vol. 33, no. 1, pp. 3–56, Feb. 1993.
[5] J. Bai and S. Ng, "Evaluating latent and observed factors in macroeconomics and finance," *J. Econometrics*, vol. 131, nos. 1–2, pp. 507–537, Mar. 2006.
[6] E. F. Fama and K. R. French, "International tests of a five-factor asset pricing model," *J. Financial Econ.*, vol. 123, no. 3, pp. 441–463, Mar. 2017.
[7] C. Asness, A. Frazzini, R. Israel, T. J. Moskowitz, and L. H. Pedersen, "Size matters, if you control your junk," *J. Financial Econ.*, vol. 129, no. 3, pp. 479–509, Sep. 2018.
[8] M. Lettau and M. Pelger, "Factors that fit the time series and cross-section of stock returns," *Rev. Financial Stud.*, vol. 33, no. 5, pp. 2274–2325, May 2020.
[9] M. Agrawal, D. Mohapatra, and I. Pollak, "Empirical evidence against CAPM: Relating alphas and returns to betas," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 4, pp. 298–310, Aug. 2012.
[10] X.-P.-S. Zhang and F. Wang, "Signal processing for finance, economics, and marketing: Concepts, framework, and big data applications," *IEEE Signal Process. Mag.*, vol. 34, no. 3, pp. 14–35, May 2017.
[11] M. M. L. de Prado, *Advances in Financial Machine Learning*. Hoboken, NJ, USA: Wiley, 2018.
[12] D. Wang, B. Podobnik, D. Horvatić, and H. E. Stanley, "Quantifying and modeling long-range cross correlations in multiple time series with applications to world stock indices," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 83, no. 4, Apr. 2011, Art. no. 046121.
[13] Z. Zheng, B. Podobnik, L. Feng, and B. Li, "Changes in cross-correlations as an indicator for systemic risk," *Sci. Rep.*, vol. 2, no. 1, pp. 1–8, Nov. 2012.
[14] Z. Kostanjčar, S. Begušić, H. E. Stanley, and B. Podobnik, "Estimating tipping points in feedback-driven financial networks," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 6, pp. 1040–1052, Sep. 2016.
[15] J. Bun, J.-P. Bouchaud, and M. Potters, "Cleaning large correlation matrices: Tools from random matrix theory," *Phys. Rep.*, vol. 666, pp. 1–109, Jan. 2017.
[16] F. Rubio, X. Mestre, and D. P. Palomar, "Performance analysis and optimal selection of large minimum variance portfolios under estimation risk," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 4, pp. 337–350, Aug. 2012.
[17] E. Chong, C. Han, and F. C. Park, "Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies," *Expert Syst. Appl.*, vol. 83, pp. 187–205, Oct. 2017.
[18] J. Bai and S. Ng, "Rank regularized estimation of approximate factor models," *J. Econometrics*, vol. 212, no. 1, pp. 78–96, Sep. 2019.
[19] R. Cont, "Empirical properties of asset returns: Stylized facts and statistical issues," *Quant. Finance*, vol. 1, no. 2, pp. 223–236, Feb. 2001.
[20] A. Verma, R. J. Buonocore, and T. Di Matteo, "A cluster driven log-volatility factor model: A deepening on the source of the volatility clustering," *Quant. Finance*, vol. 19, no. 6, pp. 981–996, Nov. 2018.
[21] Y. Aït-Sahalia and D. Xiu, "Using principal component analysis to estimate a high dimensional factor model with high-frequency data," *J. Econometrics*, vol. 201, no. 2, pp. 384–399, Dec. 2017.
[22] T. Ando and J. Bai, "Panel data models with grouped factor structure under unknown group membership," *J. Appl. Econometrics*, vol. 31, no. 1, pp. 163–191, Jan. 2016.
[23] T. Ando and J. Bai, "Clustering huge number of financial time series: A panel data approach with high-dimensional predictors and factor structures," *J. Amer. Stat. Assoc.*, vol. 112, no. 519, pp. 1182–1198, Jul. 2017.
[24] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.
[25] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.
[26] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2009.
[27] I. Iguyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003, doi: 10.1162/153244303322753616.
[28] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2010, pp. 333–342.
[29] N. Kambhatla and T. K. Leen, "Dimension reduction by local principal component analysis," *Neural Comput.*, vol. 9, no. 7, pp. 1493–1516, Oct. 1997.
[30] P. Mitra, C. A. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 301–312, Mar. 2002.
[31] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 507–514.
[32] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *Proc. ACM Int. Conf. Ser.*, vol. 227, New York, NY, USA, 2007, pp. 1151–1157.
[33] D. Huang, X. Cai, and C.-D. Wang, "Unsupervised feature selection with multi-subspace randomization and collaboration," *Knowl.-Based Syst.*, vol. 182, Oct. 2019, Art. no. 104856.
[34] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, Oct. 2007.
[35] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *J. Mach. Learn. Res.*, vol. 3, pp. 1289–1305, 2003, doi: 10.5555/944919.944974.
[36] M. Gong, M. Zhang, and Y. Yuan, "Unsupervised band selection based on evolutionary multiobjective optimization for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 544–557, Jan. 2016.
[37] G. Connor, "The three types of factor models: A comparison of their explanatory power," *Financial Analysts J.*, vol. 51, no. 3, pp. 42–46, May 1995.
[38] C. Lam, Q. Yao, and N. Bathia, "Estimation of latent factors for high-dimensional time series," *Biometrika*, vol. 98, no. 4, pp. 901–918, Dec. 2011.
[39] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," in *Proc. World Wide Web Conf. WWW*, 2018, pp. 689–698.
[40] S. Gu, B. Kelly, and D. Xiu, "Autoencoder asset pricing models," *J. Econometrics*, Jul. 2020, doi: 10.1016/j.jeconom.2020.07.009.

[41] S. Basu and G. Michailidis, "Regularized estimation in sparse high-dimensional time series models," *Ann. Statist.*, vol. 43, no. 4, pp. 1535–1567, Aug. 2015.
[42] M. Macmahon and D. Garlaschelli, "Community detection for correlation matrices," *Phys. Rev. X*, vol. 5, no. 2, Apr. 2015, Art. no. 021006.
[43] S. Begušić, Z. Kostanjčar, D. Kovač, H. E. Stanley, and B. Podobnik, "Information feedback in temporal networks as a predictor of market crashes," *Complexity*, vol. 2018, pp. 1–13, Sep. 2018.
[44] M. Tumminello, F. Lillo, and R. N. Mantegna, "Correlation, hierarchies, and networks in financial markets," *J. Econ. Behav. Org.*, vol. 75, no. 1, pp. 40–58, Jul. 2010.
[45] Z. Kakushadze and W. Yu, "Statistical industry classification," *J. Risk Control*, vol. 3, no. 1, pp. 17–65, Jun. 2017.
[46] V. Tola, F. Lillo, M. Gallegati, and R. N. Mantegna, "Cluster analysis for portfolio optimization," *J. Econ. Dyn. Control*, vol. 32, no. 1, pp. 235–258, Jan. 2008.
[47] M. L. de Prado, "Building diversified portfolios that outperform out of sample," *J. Portfolio Manage.*, vol. 42, no. 4, pp. 59–69, Jul. 2016.
[48] S. Begušić and Z. Kostanjčar, "Cluster-based shrinkage of correlation matrices for portfolio optimization," in *Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA)*, Sep. 2019, pp. 301–305.
[49] F. G. Duarte and L. N. De Castro, "A framework to perform asset allocation based on partitional clustering," *IEEE Access*, vol. 8, pp. 110775–110788, Jun. 2020.
[50] O. Ledoit and M. Wolf, "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection," *J. Empirical Finance*, vol. 10, no. 5, pp. 603–621, Dec. 2003.
[51] R. Clarke, H. De Silva, and S. Thorley, "Minimum-variance portfolio composition," *J. Portfolio Manage.*, vol. 37, no. 2, pp. 31–45, 2011, doi: 10.3905/jpm.2011.37.2.031.
[52] A. Meucci, "Quant nugget 2: Linear vs. compounded returns-common pitfalls in portfolio management," *GARP Risk Prof.*, pp. 49–51, May 2010.
[53] I. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer-Verlag, 2002.
[54] S. C. Ahn and A. R. Horenstein, "Eigenvalue ratio test for the number of factors," *Econometrica*, vol. 81, no. 3, pp. 1203–1227, May 2013.
[55] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.
[56] Quandl. (2020). *NASDAQ OMX Global Index Data*. [Online]. Available: https://www.quandl.com/data/NASDAQOMX-NASDAQ-OMX-Global-Index-Data
[57] C. E. McCulloch, "Tests for equality of variances with paired data," *Commun. Statist.-Theory Methods*, vol. 16, no. 5, pp. 1377–1391, Jan. 1987.
[58] T. Cai and W. Liu, "Adaptive thresholding for sparse covariance matrix estimation," *J. Amer. Stat. Assoc.*, vol. 106, no. 494, pp. 672–684, Jun. 2011.
[59] J. Fan, Y. Liao, and H. Liu, "An overview of the estimation of large covariance and precision matrices," *Econometrics J.*, vol. 19, no. 1, pp. C1–C32, Feb. 2016.
[60] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *J. Intell. Inf. Syst.*, vol. 17, nos. 2–3, pp. 107–145, Dec. 2001.

STJEPAN BEGUŠIĆ (Member, IEEE) received the M.Sc. degree in information and communication technology from the Faculty of Electrical Engineering and Computing, University of Zagreb, in 2014. He is currently pursuing the Ph.D. degree in computer science with the Faculty of Electrical Engineering and Computing, University of Zagreb. He is also a Research Associate with the Laboratory for Financial and Risk Analytics, Faculty of Electrical Engineering and Computing, University of Zagreb. His main research interests include statistical and machine learning methods for high-dimensional financial data, risk modeling, and portfolio optimization.



ZVONKO KOSTANJČAR (Member, IEEE) received the Dipl.-Ing. degree from the Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia, in 2002, the degree in financial mathematics from the Faculty of Science, in 2008, and the Ph.D. degree from the University of Zagreb, in 2010. He is currently an Associate Professor with the Faculty of Electrical Engineering and Computing, University of Zagreb, and the Head and the Founder of the Laboratory for Financial and Risk Analytics. He has served as the President for the Signal Processing Chapter, IEEE Croatia Section. His main research interests include statistical and machine learning methods for finance.