A memory-based method to select the number of relevant components in Principal Component Analysis

Anshul Verma[1], **Pierpaolo Vivo**[1] **and Tiziana Di Matteo**[1,2,3]

[1] Department of Mathematics, King's College London, Strand, London, WC2R 2LS, United Kingdom

[2] Department of Computer Science, University College London, Gower Street, London, WC1E 6BT, United Kingdom

[3] Complexity Science Hub Vienna, Josefstädter Strasse 39, A 1080 Vienna, Austria

E-mail: anshul.verma@kcl.ac.uk, pierpaolo.vivo@kcl.ac.uk, tiziana.di_matteo@kcl.ac.uk

Abstract. We propose a new data-driven method to select the optimal number of relevant components in Principal Component Analysis (PCA). This new method applies to correlation matrices whose time autocorrelation function decays more slowly than an exponential, giving rise to long memory effects. In comparison with other available methods present in the literature, our procedure does not rely on subjective evaluations and is computationally inexpensive. The underlying basic idea is to use a suitable factor model to analyse the residual memory after sequentially removing more and more components, and stopping the process when the maximum amount of memory has been accounted for by the retained components. We validate our methodology on both synthetic and real financial data, and find in all cases a clear and computationally superior answer entirely compatible with available heuristic criteria, such as cumulative variance and cross-validation.

Keywords: quantitative finance, financial networks, data mining

Contents

1. Introduction

With the arrival of sophisticated new technologies and the advent of the Big Data era, the amount of digital information that can be produced, processed and stored has increased at an unprecedented pace in recent years. The need of sophisticated post-processing tools – able to identify and discern the essential driving features of a given high-dimensional system – has thus become of paramount importance. Principal Component Analysis (PCA), which aims to reduce the dimensionality of the correlation matrix between data [1, 2], is continuing to prove a highly valuable method in this respect. PCA has been shown to have applications spanning from neuroscience to finance. In image processing, for instance, this technique has proven useful to identify key mixtures of colours of an image for use in compression [3]. In molecular dynamics, the increasing computational power available to researchers makes it possible to simulate more complex systems, with PCA helping to detect important chemical drivers [4]. The brain's neurons produce different responses to a variety of stimuli, hence PCA can be used in neuroscience to find common binding features that determine such responses [5]. In finance, the amount of digital storage and the length of available historical time series have dramatically increased. It has therefore become possible to probe the multivariate structure of changes in prices, but with the large universe of stocks that usually make up markets, PCA has become a valuable technique in identifying essential factors governing price evolution [6–8].

Within the class of *dimensionality reduction* methods, whose goal is to produce a faithful but smaller representation of the original correlation matrix [9], PCA plays a very important role. Other known methods include information filtering techniques [10–15], autoencoders [16, 17] and Independent Component Analysis (ICA) [18, 19]. PCA accomplishes this task using a subset of the orthogonal basis of the correlation matrix of the system. Successive *principal components* – namely the eigenvectors corresponding to the largest eigenvalues – provide the orthogonal directions along which data are maximally spread out. Since the dimension of empirical correlation matrices can be as large as $\sim 10^2 - 10^3$, a highly important parameter is the number $m^\star$ of principal components one should retain, which should strike the optimal balance between providing a faithful representation of the original data and avoiding the inclusion of irrelevant details.

Unfortunately, there is no natural prescription on how to select the optimal value $m^\star$, and many heuristic procedures and so-called *stopping criteria* have been proposed in the literature [1,2]. The most popular methods – about which more details are given in Section 7) – are i) scree plots [20], ii) cumulative explained variance [21, 22], iii) distribution-based methods [23, 24], and iv) cross-validation [25, 26]. However, they all suffer from different, but serious drawbacks: i) and ii) are essentially rules of thumb with little data-driven justification, iii) do not allow the user to control the overall significance level of the final result and are thus impractical for large data sets, and finally iv), whilst being more objective and relying on fewer assumptions, is often computationally cumbersome [1]. Efforts to improve each subclass – for instance the more "subjective" methods [20–22] – have been undertaken, but they usually resulted in adding more assumptions or were anyway unable to fully solve the issues [1].

Unlike most other methods available in the literature, in this paper we propose to take advantage of long memory effects that are present in many empirical time series [27] to select the optimal number $m^\star$ of principal components to retain in PCA. We shall leverage on the natural factor model implied by PCA (see Section 5.2 below) to assess the statistical contribution of each principal component to the overall "total memory" of the time series, using a recently introduced proxy for memory strength [15]. We test the validity of our proposal on synthetic data, namely two fractional Gaussian noise processes with different Hurst exponents (see Section 6.1), and also on an empirical dataset whose details are reported in Appendix A. Comparing our memory-based method with other heuristic criteria in the literature, we find that our procedure does not include any subjective evaluation, makes a very minimal and justifiable set of initial assumptions, and is computationally far less intensive than cross-validation.

Our methodology is generally applicable to any (however large) correlation matrix of a long-memory dataset. A typical example is provided by financial time-series, which are well-known to display long-memory effects [28]. The volatility of such time-series indeed constitutes an important input for risk estimation and dynamical models of price changes [29–31]. However, the multivariate extensions of common volatility models, such as multivariate Generalised Autogressive Conditional Heteroskedastic (GARCH) [32], stochastic covariance [33] and realised covariance [34], suffer from the curse of dimensionality, hindering their application in practice. A popular solution to this issue is to first apply PCA to the correlation matrix between volatilities, and then use the reduced form of the correlation matrix to fit a univariate volatility model for each component, as in [6]. In climate studies, PCA has been used to create 'climate indices' to identify patterns in climate data from a wide range of measurements including precipitations and temperature [35]. Here, factors such as the surface temperature are known to exhibit long range memory [36]. In neuroscience, PCA can be used to discover amongst the vast number of possible neurons those which correspond to particular responses, for example how an insect brain responds to different odorants [5]. In this case as well, long memory effects are well-known to play an important role [37]. Our framework is therefore highly suited to a wide array of problems.

The paper is organised as follows: in Section 2, we introduce and define the PCA procedure and how one selects the most relevant number of principal components. Section 3 describes the relevant quantities and results that are specific to financial data. We detail our proposed method to select the principal components based on memory in Section 5, testing the method on synthetic and empirical data in Section 6. We explore the advantages that our method offers over existing approaches in literature in Section 7, before finally drawing some conclusions in Section 8. The appendices are devoted to the description of the empirical dataset and technical details.

2. PCA and the optimal number of principal components to retain

In this Section, we give a brief introduction to PCA to make the paper self-contained. Call $\boldsymbol{X}$ the data matrix, which contains N columns – standardised to have zero mean and unit variance – of individual defining features, and T rows recording particular realisations in time of such features. PCA searches for the orthogonal linear basis with unit length $\boldsymbol{w}_{\{i=1,\ldots,N\}}$ that transforms the system to one where the highest variance is captured by the first component, the second highest by the second component and so on [1]. The first component is therefore given by

$$\boldsymbol{w}_1 = \arg\max_{||\boldsymbol{w}||=1} \left\{ ||\boldsymbol{X}\boldsymbol{w}||^2 \right\} = \arg\max_{||\boldsymbol{w}||=1} \left\{ \boldsymbol{w}^\dagger \boldsymbol{E} \boldsymbol{w} \right\} \ , \tag{1}$$

where † represents the transpose, and $\boldsymbol{E}$ is the sample correlation matrix of $\boldsymbol{X}$, defined as

$$E_{ij} = \frac{1}{T} \sum_{t=1}^{T} X_{ti} X_{tj} \ . \tag{2}$$

The search for $\boldsymbol{w}_1$ can be formulated as a constrained optimisation problem, i.e. we must maximise

$$\boldsymbol{w}^\dagger \boldsymbol{E} \boldsymbol{w} - \lambda(\boldsymbol{w}^\dagger \boldsymbol{w} - 1) \ , \tag{3}$$

where λ is the Lagrange multiplier enforcing normalisation of the eigenvectors. Differentiating Eq. (3) w.r.t. to $\boldsymbol{w}$ we get

$$\boldsymbol{E}\boldsymbol{w} - \lambda \boldsymbol{w} = 0 \ . \tag{4}$$

This means that the Lagrange multiplier must be an eigenvalue of $\boldsymbol{E}$. Also note that the variance of data along the direction $\boldsymbol{w}$ is given by

$$\boldsymbol{w}^\dagger \boldsymbol{E} \boldsymbol{w} = \lambda \boldsymbol{w}^\dagger \boldsymbol{w} = \lambda \ , \tag{5}$$

and hence the largest variance is realised by the top eigenvalue. It follows that the first principal component – i.e. the direction along which the data are maximally spread out – is nothing but the top eigenvector $\boldsymbol{w}_1$ corresponding to the top eigenvalue λ_1. A similar argument holds for the subsequent principal components.

The aim of PCA is to reduce $\boldsymbol{E}$ to a $m \times m$ matrix, where $m \ll N$ is the number of principal components that we choose to retain. Is there an optimal value $m^\star$ that one should select? Clearly, this is an important question that must be addressed, since it determines the "best" size of the reduced correlation matrix that is just enough to describe the main features of the data without including irrelevant details. In this paper, we address this question and we provide a new method to select the optimal value $m^\star$ of the number of principal components that we should retain for long-memory data.

3. Financial Data

3.1. Data Structure

In this Section, we describe the general structure of the data matrix that we use in the context of financial data. We consider a system of N stocks and T records of their daily closing prices. We calculate the time series of log-returns for a given stock i, $r_i(t)$, defined as:

$$r_i(t) = \ln p_i(t+1) - \ln p_i(t) \ , \tag{6}$$

where $p_i(t)$ is the price of stock i at time t. After standardising $r_i(t)$ so that it has zero mean and unit variance, we define the proxy we shall use for the volatility, i.e. the variability in asset returns (either increasing or decreasing), as $\ln|r_i(t)|$ [38]. Most stochastic volatility models – where the volatility is assumed to be random and not constant – assume that the return for the stock i evolve according to [39]

$$r_i(t) = \delta(t)\exp^{\omega_i(t)} \ , \tag{7}$$

where $\delta(t)$ is a white noise with finite variance and $\omega_i(t)$ are the *log volatility* terms. The exponential term encodes the structure of the volatility and how it contributes to the overall size of the return. We note that for our purposes, we are able to set the white noise term to be the same for all stocks since it contains no memory by definition [40] (we have checked that changing this assumption to include a stock dependent white noise term does not change our results). Taking the absolute value of Eq. 7 and the log of both sides, Eq. 7 becomes

$$\ln|r_i(t)| = \ln|\delta(t)| + \omega_i(t) \ . \tag{8}$$

We see that working with $\ln|r_i(t)|$ has the added benefit of making $\omega_i(t)$ – the proxy for volatility – additive, which in turn makes the volatility more suitable for factor models. Since $\delta(t)$ is a random scale factor that is applied to all stocks, we can set it to 1, so that $\omega_i(t) = \ln|r_i(t)|$. We also standardise $\omega_i(t)$ to a mean of 0 and standard deviation 1 as performed in [41]. Finally, call $\mathbf{X}$ the data matrix, which contains N columns for each individual defining stock, and T rows recording particular realisations in time of such stocks so that the i,t entry of $\mathbf{X}$ is $X_{it} = \omega_i(t)$.

3.2. Market Mode and Marčenko-Pastur

For the case of log volatilities in finance [7, 42] (see further details in Appendix B), it has been known for some time that the smallest eigenvalues of the empirical correlation matrix $\boldsymbol{E}$ may be heavily contaminated by noise due to the finiteness of the data samples. In our search for the most relevant $m^\star$ components, it is therefore important to confine ourselves to the sector of the spectrum that is less affected by noise at the outset.

To facilitate this identification, we will resort to a null distribution of eigenvalues, which are produced from a Gaussian white noise process. This is given by the celebrated Marčenko-Pastur (MP) distribution [7, 43, 44]

$$p(\lambda) = \frac{1}{2\pi q\sigma^2}\frac{\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)}}{\lambda} \ , \tag{9}$$

where $p(\lambda)$ is the probability density of eigenvalues having support in $\lambda_- < \lambda < \lambda_+$. The edge points $\lambda_\pm = \sigma\left(1 \pm \sqrt{q}\right)^2$, $q = T/N$ and σ is the standard deviation over all stocks. By comparing the empirical eigenvalue distribution of $\boldsymbol{E}$ to the MP law (9), we can therefore see how many eigenvalues, and thus principal components, are likely to be corrupted by noise and should therefore be discarded from the very beginning. More recently, this procedure has received some criticisms [45–47]: it has been argued that eigenvalues carrying genuine information about weakly correlated clusters of stocks could still be buried under the MP sea, and more refined filtering strategies may be needed to bring such correlations to the surface. Other generalisations of the MP law for non-normally distributed random data and applications to financial data can be found for instance in [48] and [49].

In practical terms, we first create the empirical correlation matrix $\boldsymbol{E} = (1/T)\boldsymbol{X}^{\dagger}\boldsymbol{X}$ from the matrix $\boldsymbol{X}$ (constructed from either synthetic or empirical data), and then we fit the MP law to the empirical distribution of its eigenvalues. This is done by considering q and σ in Eq. (9) as free parameters to take into account finite sample biases [46]. In Fig. 1(a), we plot the histogram of bulk eigenvalues for the empirical dataset described in Appendix A, and in the inset a number of outliers $\lambda > \lambda_+$ in semilog scale. It is indeed well-known that some of the eigenvalues of $\boldsymbol{E}$ extend well beyond the upper edge of the MP law, and that the largest eigenvalue lies even further away (see Fig. 1(a)). This means that the first principal component accounts for a large proportion of the variability of data, and is in fact a well-known effect of the *market mode* [41,50,51]. We plot the entries of the right eigenvector $\boldsymbol{w}_1$ of $\mathbf{E}$ (corresponding to the market mode) and $\boldsymbol{w}_2$ in Fig. 2, with the blue lines giving the length from the origin of the corresponding 2D vector. We see from Fig. 2 that the entries for $\boldsymbol{w}_1$ are all positive, which confirms that indeed the first eigenvector affects all stocks.

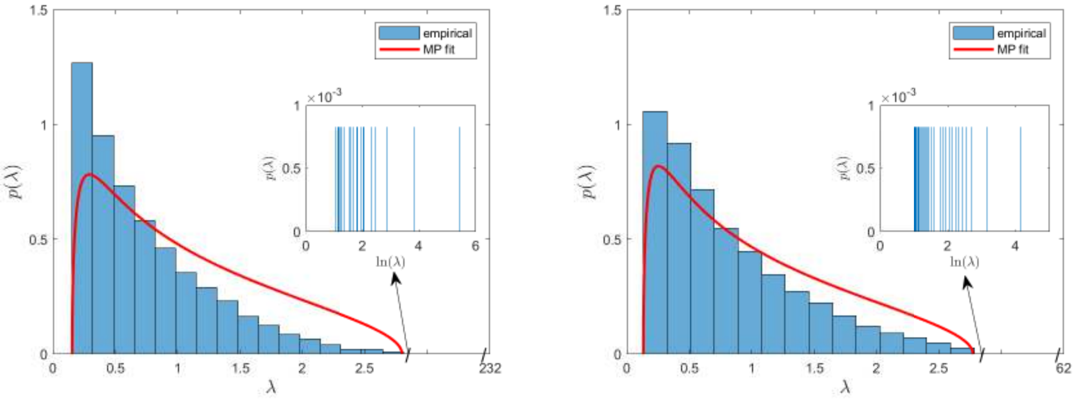
4. Long Memory

We now consider the 'long memory' features of a time series, specialising the discussion to the log volatility in a financial context.

The autocorrelation function (ACF), $\kappa(L)$, of any time series $x(t)$ is defined as

$$\kappa(L) = \mathrm{corr}(x(t+L), x(t)) = \frac{\langle [x(t+L)x(t)] \rangle}{\sigma^2} \ , \tag{10}$$

where $\langle ... \rangle$ denotes the time expectation over $x(t)$, adjusted to have zero mean. L is the lag and σ^2 is the variance of the process $x(t)$. If $\kappa(L)$ decays faster than or as fast as an exponential with L, then the time series is said to have short memory [27]. However, in many real world systems ranging from outflows in hydrology to tree ring measurements [27], $\kappa(L)$ has been found to decay much more slowly than an exponential, giving rise to an important effect known as long memory [27]. This means that the process at time t remains heavily influenced by what happened in a rather distant past. In particular for financial data (where $x(t) = |\ln r(t)|$), it is an accepted stylised fact (called *volatility clustering*) that large changes in volatilities are usually followed by other large changes in volatilities, or that the volatilities retain a long memory of



(a) $\boldsymbol{E}$ (b) $\boldsymbol{G}$

Figure 1: (a) Histogram of the eigenvalue distribution of $\boldsymbol{E}$ constructed from the empirical dataset (see Appendix A), compared to the best fit Marčenko-Pastur distribution in red. The λ axis has been split by the forward-slashes to only show the bulk eigenvalues below $\lambda_+ = 2.80$. The inset shows the 22 isolated eigenvalues for $\lambda > \lambda_+$ in semilog scale. The Marčenko-Pastur distribution is fitted with parameters $q = 0.38 \pm 0.02$ and $\sigma = 1.03 \pm 0.01$. (b) Same histogram, but applied to the correlation matrix $\boldsymbol{G}$ (see Section 5.1), where the market mode has been de-trended. Here $\lambda_+ = 2.77$, $q = 0.41 \pm 0.02$, $\sigma = 1.01 \pm 0.01$, with 35 eigenvalues above λ_+.

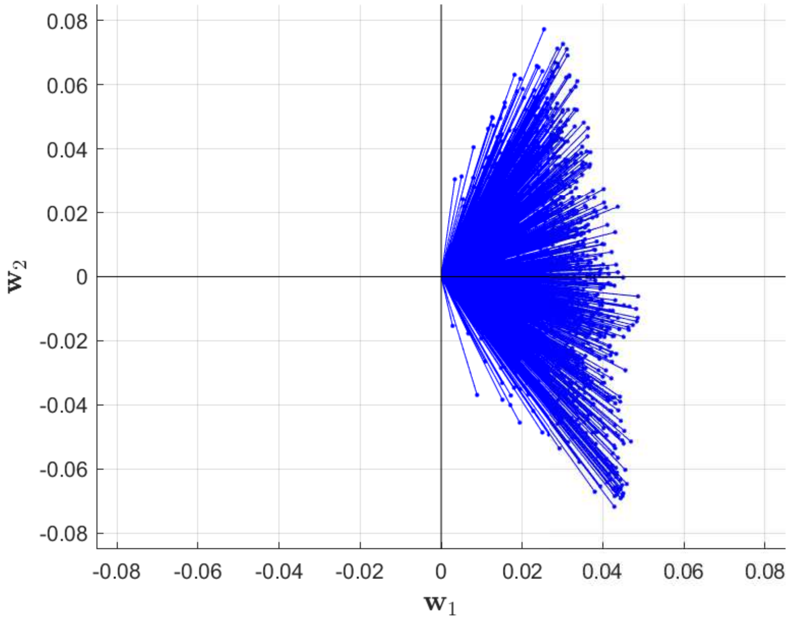


Figure 2: Each point i in this graph has coordinates (w_{i1}, w_{i2}), where w_{i1} is the i-th entry of the top eigenvector $\boldsymbol{w}_1$, and w_{i2} is the i-th entry of the second eigenvector $\boldsymbol{w}_2$ of the correlation matrix $\boldsymbol{E}$ (defined in Eq. (2)) for the dataset in Appendix A. The length of the corresponding 2D coordinate vector from the origin is given by each blue line. The plot shows that all values of w_{i1} are in fact positive.

previous values [52]. $\kappa(L)$ has also been empirically found to follow a power law decay

$$\kappa(L) \sim L^{-\beta^{\text{vol}}} \;, \tag{11}$$

where β^{vol} describes the strength of the memory effect – a lower value indicates that a longer memory of past values is retained. However, as shown in [15], to better distinguish between short and long memory it is convenient to consider the non-parametric integrated proxy η, defined as

$$\eta = \int_{L=1}^{L_{cut}} \kappa(L) \mathrm{d}L \;, \tag{12}$$

where L_{cut} is the standard Bartlett Cut at the 5% level [40]. The proxy η is less affected by the noise-dressing of $\kappa(L)$ than β^{vol} [15], and the larger the value of η the greater the degree of the memory effect. This observable will constitute an essential ingredient of our method.

5. Methods

In this Section, we describe in detail our procedure.

5.1. De-trending the market mode

The first step of our method consists in removing the influence of the market mode, the global factor affecting the data, as we hinted at in Section 3.2. To do this, we impose that the standardised log volatility $\omega_i(t) = \ln|r_i(t)|$ in Eq. (8) follows a factor model (using the Capital Asset Pricing Model, CAPM [53, 54]) with the *market mode*

$$I_0(t) = \sum_{i=1}^{N} w_{i1} \ln|r_i(t)| \tag{13}$$

as a factor. This quantity – essentially a weighted average of $\ln|r_i(t)|$ with weights $\boldsymbol{w}_1$, the top eigenvector's components – represents the effect of the market as a whole on all stocks i.e. the common direction taken by all stocks at once.

Hence we define

$$\omega_i(t) = \beta_{i0} I_0(t) + \alpha_{i0} + c_i(t) \;. \tag{14}$$

Here, β_{i0} is the responsiveness of stock i to changes in the market mode $I_0(t)$, α_{i0} is the excess volatility compared to the market and $c_i(t)$ are the residual log volatilities.

A standard linear regression of $\omega_i(t)$ against $I_0(t)$ brings to the surface the residual volatilities $c_i(t)$ that the market as a whole cannot explain. The matrix of standardised $c_i(t)$ for all stocks is labeled $\boldsymbol{X}^{(\text{market})}$. We call $\boldsymbol{G}$ the correlation matrix of $\boldsymbol{X}^{(\text{market})}$, with entries

$$G_{ij} = \frac{1}{T} \sum_{t=1}^{T} c_i(t) c_j(t) \;. \tag{15}$$

By definition, the matrix $\boldsymbol{G}$ will have the influence of the market mode removed through Eq. (14). This cleaning procedure also makes the correlation structure more stable [55], and therefore we will be working with the matrix $\boldsymbol{G}$ from now on.

As we did with the matrix $\boldsymbol{E}$, we again fit the Marčenko-Pastur (MP) distribution – this time to the empirical eigenvalue distribution of $\boldsymbol{G}$. This is justified even in the presence of autocorrelations since in the bulk the amount of memory is quite low. We can see this empirically by computing the median L_{cut} over principal components that have eigenvalues below λ_+ for the fitted MP distribution, which is 2. The values are quite close to 1, which is the value we would find for white noise. In the presence of weak autocorrelations, [56] showed that the distribution of eigenvalues in the bulk differs slightly to the MP distribution. We clearly see this distortion in our Fig. 1(b), which bears some similarity in shape with the pdf calculated and plotted in [56] (see Fig. 1 there). However, the MP distribution is a simpler and very good approximation, especially for the edge points in Fig. 1(b). We expect that the number of eigenvalues beyond the bulk should increase, since the removal of the market mode makes the true correlation structure more evident and lowly intra-correlated clusters more visible [55]. This is confirmed by the results that are detailed in Fig. 1(b), where we see that the number of eigenvalues beyond the bulk (shown in the inset plot in Fig.1(b)) has indeed increased from 22 to 35. Note that we also see from Fig. 1 that the best fit q and σ for $\boldsymbol{E}$ and $\boldsymbol{G}$ are quite similar, which matches the theoretical result of [57].

With this finding in hand, we can safely disregard all principal components corresponding to eigenvalues within the MP sea. This observation already drastically reduces the maximum value of eligible components – which we call $m_{\max}$ – from 1202 to 35 for the empirical data in Appendix A. We also recall that the eigenvectors of $\boldsymbol{G}$ have an economic interpretation according to the Industrial Classification Benchmark (ICB) supersectors – for more details, see Appendix B.

5.2. Regression of principal components

Considering the matrix $\boldsymbol{G}$ in Eq. (15), where the influence of the market mode has been removed, we must now assess how each stock i's log-volatility is related to the log-volatility of each principal component. We achieve this result by regressing $c_i(t)$ in Eq. (14) against the average behaviour of the log-volatility for the principal components. The average behaviour $I_p(t)$ is defined as

$$I_p(t) = \sum_{i=1}^{N} w_{ip} c_i(t) \ , \qquad p = 1, ..., m_{\max} \ , \tag{16}$$

i.e. it is the weighted average log-volatility of the p-th principal component, where w_{ip} is the i-th entry of the p-th eigenvector of $\boldsymbol{G}$. Eq. (16) is therefore the projection of the residue $c_i(t)$ onto the $m_{\max}$ principal components.

The principal components are an orthogonal basis for the correlation matrix $\boldsymbol{G}$, and represent important features that determine fluctuations in the c_i's. Therefore, it makes sense to define a factor model – which we call the "$m_{\max}$-based PCA factor model" – where the explanatory variables are the $m_{\max}$ principal components [1, 58]

$$c_i(t) = \sum_{p=1}^{m_{\max}} \beta_{ip} I_p(t) + \epsilon_i(t) \ , \tag{17}$$

where $\epsilon_i(t)$ is a white noise term with zero mean and finite variance, and $c_i(t)$ are the residual volatilities defined in Eq. (14). Here, β_{ip} is the responsiveness of $c_i(t)$ to changes in $I_p(t)$, indicating whether the log-volatility of stock i is higher ($\beta_{ip} > 1$) or lower than $I_p(t)$ ($\beta_{ip} < 1$).

We can now find β_{ip} by regressing the previously obtained input $c_i(t)$ against all the I_p's. This will separate the signal explained by the principal components from the residual noise present in the system. The regression will be performed using a lasso method (see Appendix C for details).

5.3. Assessing memory contribution

The next step of our methodology consists in estimating the memory contribution of the $m = 1, 2, ..., m_{\max}$ components.

Fixing m, we compute for each stock the quantity

$$d_i^{(m)}(t) = c_i(t) - \sum_{p=1}^{m} \beta_{ip} I_p(t) \ , \quad i = 1, ..., N \ . \tag{18}$$

Here, the β_{ip} are the coefficients obtained with the regression in Eq. (17). The $d_i^{(m)}(t)$ are the residues after the removal of the first m components.

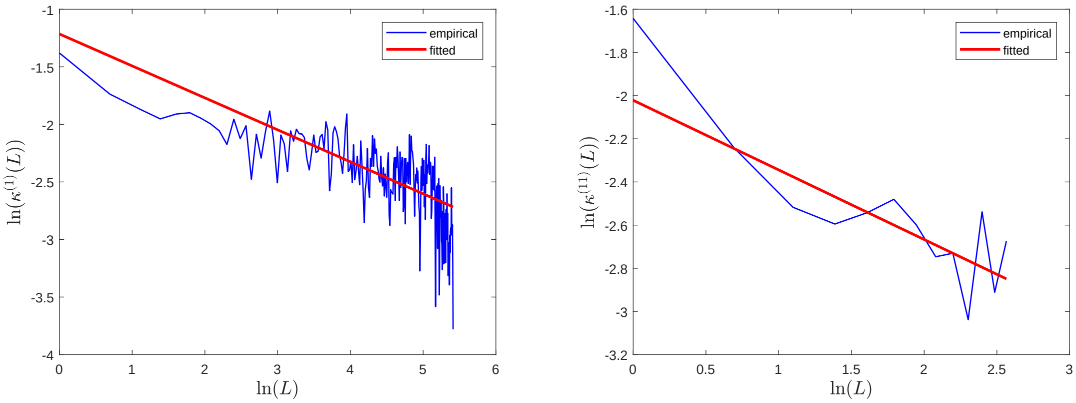
Using the $d_i^{(m)}(t)$, we can first compute their temporal autocorrelation $\kappa_i^{(m)}(L)$ in Eq. (10) for different values of the lag L between $L = 1$ and $L = T - 1$. We generically find that the $\kappa_i^{(m)}(L)$ follow a power law decay as a function of L – see examples in Fig. 3 depicting the $\kappa_i^{(m)}(L)$ for $m = 1, 11$ for ALJ Regional Holdings (ALJJ), a stock included in our empirical dataset in Appendix A. As more components are removed (i.e. as m is increased), the exponent β^{vol} defined in Eq. (11) for ALJJ and plotted in Fig. 3 increases from 0.277 to 0.322. This result is what one would expect since the amount of memory accounted for will decrease as more components are removed.

Numerically integrating the $\kappa_i^{(m)}(L)$, we obtain a set of integrated memory proxies $\eta_i^{(m)}$ (see Eq. (12)), one for each asset i and for each number m of removed components. In general, the $\eta_i^{(m)}$ are non-increasing functions of m, since the further removal of subsequent components is bound to decrease the residual memory level present in the system.

We now define

$$\zeta(m) = \mathrm{median}\left(\frac{\eta_i^{(m)}}{\eta_i^{(0)}}\right) \ , \tag{19}$$

where $\eta_i^{(m)}$ are the integrated proxies, and $\eta_i^{(0)}$ are just the integrated proxies of the residual volatilities $c_i(t)$ defined in Eq. (14). $\zeta(m)$ thus represents the "average"



(a) $\kappa^{(1)}(L)$

(b) $\kappa^{(11)}(L)$

Figure 3: Plots in log-log scale of $\kappa^{(m)}(L)$ in blue, which is given in Eq. (10) with $x(t) = d_i^{(m)}(t)$, for the stock ALJ Regional Holdings (ALJJ). Here, $m = 1$ on the left and $m = 11$ on the right. In red the lines of best fit (using the Theil Sen estimator [59]), which gives the power law decay exponent β^{vol} as 0.277 and 0.322 for the left and right plot respectively.

behaviour over all stocks of how much each of the principal components contributes to the memory. It is again a non-increasing function of m, and by definition $\zeta(m) < 1$ for all m.

In Fig. 4(c), we plot $\zeta(m)$ in log-log scale for both the empirical and synthetic datasets in Appendix A and Section 6.1 respectively. We observe a striking change in concavity at some value θ, which we interpret as follows: to the right of θ, the amount of memory left unexplained in the system changes very slowly when more and more components are progressively included. This clearly signals that we have reached the "optimal stopping" point $m^\star$ beyond which the inclusion of further components would not add more information.

Beyond θ, the behaviour of $\zeta(m)$ is power-law

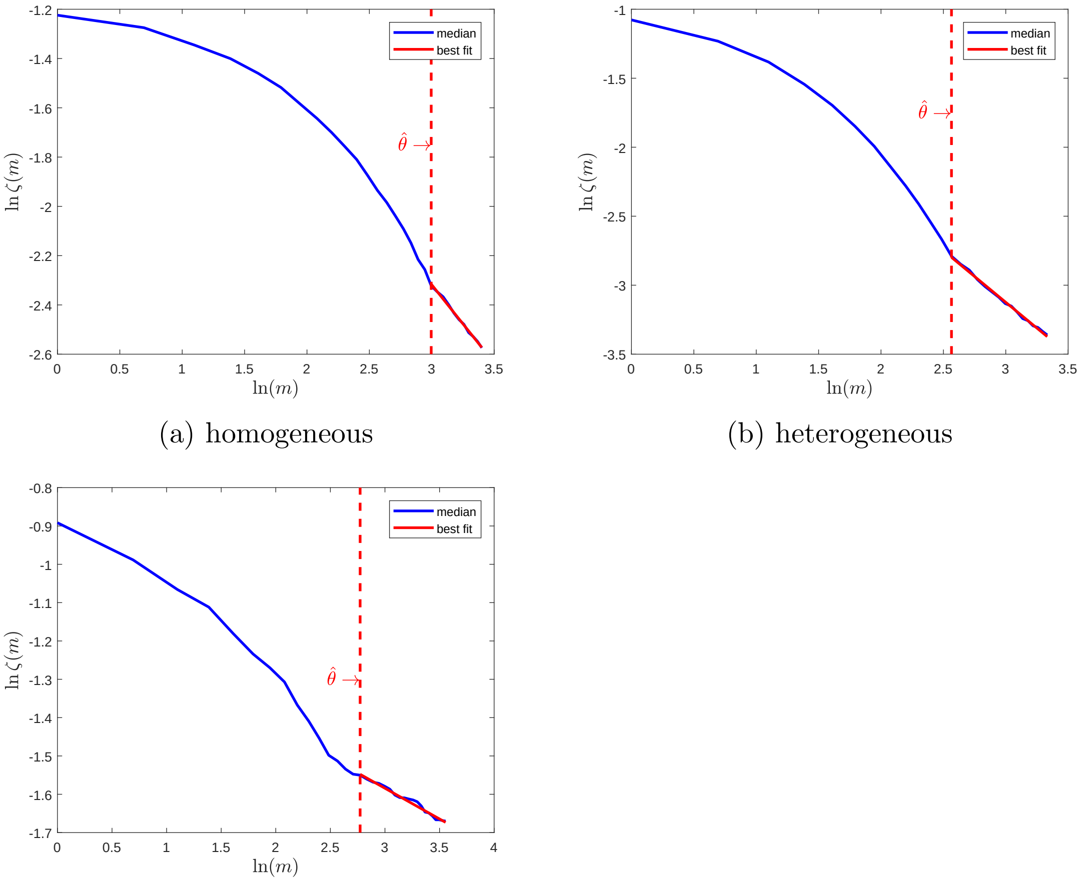
$$\zeta(m) \sim m^{-\gamma} \qquad m \geq \theta \ , \tag{20}$$

where γ is the exponent. Using the fitting procedure for θ described in Appendix D produces the optimal integer estimator $\hat{\theta}$. Since the value of $\hat{\theta}$ indicates that for $m < \hat{\theta} - 1$, $\zeta(m)$ decreases more rapidly than a power law, we can safely set

$$m^\star = \hat{\theta} - 1 \ . \tag{21}$$

5.4. Summary of the procedure

The procedure to select the optimal number $m^\star$ of principal components to retain is summarised here for a general, standardised data-matrix $\boldsymbol{X}$ containing long memory



(a) homogeneous

(b) heterogeneous

(c) empirical

Figure 4: (Top left) Plot of $\ln(\zeta(m))$ vs $\ln(m)$ for the homogeneous synthetic system defined in Section 6.1. The blue line is the value of $\zeta(m)$ across all assets, with the dashed red line indicating $\hat{\theta} = 20$, the point at which the concavity changes. (Top right) Same plot but for the heterogeneous simulated system described in Section 6.1, where $\hat{\theta} = 13$. (Bottom) Same plot but for the empirical dataset described in Appendix A, yielding $\hat{\theta} = 16$. These values of $\hat{\theta}$ imply that the number $m^\star$ of principal components to retain should be $m^\star = 19, 12, 15$ respectively.

effects (justifications for the steps can be found in the Sections labelled in brackets after each step):

(i) Remove any global effect from $\boldsymbol{X}$ to form $\boldsymbol{X}^{(\text{market})}$, whose entries are the residues $c_i(t)$ defined in Eq. (14) [Section 5.1].

(ii) Compute the correlation matrix $\boldsymbol{G}$ of $\boldsymbol{X}^{(\text{market})}$ and find the empirical probability density of its eigenvalues. Find the number of eigenvalues $m_{\max}$ exceeding λ_+, the upper edge of the Marčenko-Pastur distribution in Eq. (9) [Section 3.2].

(iii) Forming the $m_{\max}$-based PCA factor model from Eq. (17), use lasso regression (see Appendix C) to find the set of parameters β_{ip}. This is achieved by regressing the residues $c_i(t)$ against the average behaviour of principal components $I_p(t)$

$p = 1, ..., m_{\max}$ [Section 5.2].

(iv) Using these β_{ip}'s, determine for each $m = 1, ..., m_{\max}$ and stock i the residue $d_i^{(m)}(t)$ given in Eq. (18) [Section 5.3].

(v) From the $d_i^{(m)}(t)$, compute the temporal autocorrelations $\kappa_i^{(m)}(L)$ for different values of L, and by integration determine the proxies $\eta_i^{(m)}$. Construct $\zeta(m)$ from Eq. (19) [Section 5.3].

(vi) Use the fitting procedure in Appendix D to find $\hat{\theta}$, the best estimator of θ – the point at which the concavity of $\zeta(m)$ changes – defined in Eq. (20). Finally, the optimal number of principal components to retain is $m^\star = \hat{\theta} - 1$ [Section 5.3].

6. Applying our method to synthetic and empirical data

In this Section, we test our method on synthetically generated data and on an empirical data set defined in Appendix A.

6.1. Synthetic System Setup

A paradigmatic example of stochastic process with long memory is the Fractional Gaussian Noise (FGN). The FGN with Hurst exponent H is the process $Y(t)$ with an autocorrelation function [27] given by

$$\kappa_{FGN}(L) = \frac{1}{2}\left(|L-1|^{2H} - 2|L|^{2H} + |L+1|^{2H}\right) \sim H(2H-1)L^{2H-2} \ . \qquad (22)$$

Eq. (22) indeed shows that the FGN has long memory since its autocorrelation function follows a power law decay as described in Section 4. In particular, for $1/2 < H < 1$ ($H = 1/2$ corresponds to the standard white noise) we have a process with positive autocorrelation, a feature that is shared by financial data [28]. Increasing H will enhance the strength of the memory present in the FGN since $\kappa_{FGN}(L)$ will decay more slowly in this case. We shall use the method detailed in [60] to generate realisations of FGN.

For our synthetic setting, we consider a fictitious market made of N stocks, and simulate the log-volatility $\omega_i(t)$ of each stock over a time-window T. To this end, we make use of the widely recognised fact that empirical data from finance are often organised into clusters [61–64]. We therefore impose that the stocks are organised into K disjoint clusters, each containing N_k stocks.

Next, we generate a fictitious market mode $I_0(t)$ that affects all stocks [41, 50, 51]. This is simply a FGN process with Hurst exponent H_0, which we will set to 0.9 in our simulations. We also fix the variance of $I_0(t)$ to be 1.

Our simulated log-volatility processes will thus read

$$\omega_i(t) = \beta_0 I_0(t) + \beta_{k(i)} I_{k(i)}(t) + \epsilon_i(t) \ , \qquad (23)$$

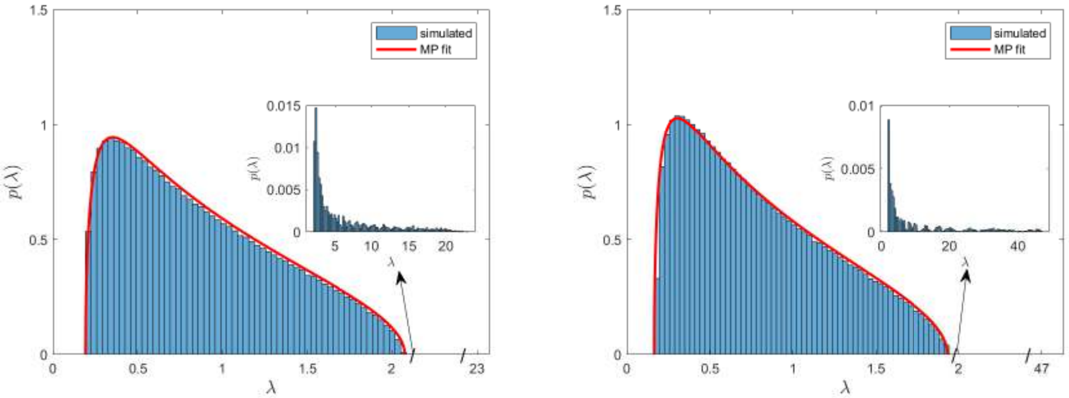
where $k(i)$ denotes the index of the cluster the asset i belongs to, and the $I_k(t)$'s are FGN processes with Hurst exponents H_k and fixed variance of 1. The ϵ_i's are white noise terms with zero mean and variance ϕ.

Typical values we use in our simulations are $N = 1200$, $T = 4000$ and $K = 30$. We simulate two markets with different internal arrangements of the clusters: the first one is homogeneous, where the size of each cluster is exactly 40, and the second is heterogeneous, meaning that each cluster has a different number of stocks. The latter case is particularly significant since the cluster sizes present in financial data as well as in many other systems are known to be heterogeneous [61, 62]. To generate the heterogeneous system, we use the procedure described in [65], which yields power-law distributed cluster sizes, a key property of real world data [66]. The particular realisation of this method that we use to generate cluster sizes for $N = 1200$ has a mean number of stocks in each cluster of 40 and a standard deviation of 26.2. We also set $\beta_0 = 1.3$, while β_k are values between 0.14 and 1, and H_k is an equally spaced sequence between 0.7 and 0.9. This choice ensures that clusters with a higher β_k will also have a higher H_k, to make contact with the empirical result of [67] that stocks with higher volatility cross-correlation have a longer memory. Finally, ϕ is fixed to be 1, the same as the variance of the time series $I_0(t)$ and $I_k(t)$. Note that we also simulate the same system using instead an Autoregressive process of 1 lag (AR(1)) [40] in Appendix E, where we show that our method can be applied in this case too, but is less accurate. This supports our reasoning that that slow decrease to the right of $\hat{\theta}$ in Fig. 4(c) is more applicable to long memory processes versus short ones.

Arranging the log-volatilities $\omega_i(t)$ in a rectangular data-matrix $\boldsymbol{X}$, we can then feed $\boldsymbol{X}$ into our algorithmic procedure and check how many significant components $m^\star$ it retrieves. A desirable feature of our synthetic model is that it is rather easy to estimate *a priori* how many eigenvalues of $\boldsymbol{E} = (1/T)\boldsymbol{X}^\dagger\boldsymbol{X}$ (or rather of its de-trended counterpart $\boldsymbol{G}$) contain information that can be separated from pure noise. This occurs because each cluster corresponds to a principal component and hence the number of eigenvalues beyond the bulk is just K. This makes the *a posteriori* comparison all the more interesting.

6.2. Results for synthetic and empirical data

We simulate 100 independent samples of our synthetic market, after checking that the statistics was sufficient to be confident on the stability of our results, and we follow the procedure set out at the end of Section 5.3 to select $m^\star$. First, we checked the eigenvalue distributions of the correlation matrix obtained from the simulated $\boldsymbol{X}$. We see from Fig. 5, which are histograms of the bulk eigenvalues of $\boldsymbol{G}$ for all samples for the homogeneous (left) and heterogeneous (right) systems respectively, that the bulk of the eigenvalues is well fitted by the Marčenko-Pastur distribution in red. There are $m_{\max} = 30$ (homogeneous) and $m_{\max} = 28$ (heterogeneous) eigenvalues beyond the bulk (depicted in the insets) that carry genuine information. This again shows that in the synthetic case the autocorrelations are also weak. We see this again by calculating the median L_{cut} of Eq. (12) for the synthetic systems to be 2 – again close to 1, which is what we would expect for white noise, hence we can still use the MP distribution as an



(a) homogeneous

(b) heterogeneous

Figure 5: Histograms of eigenvalues of the matrix $\boldsymbol{G}$ for 100 samples of the synthetic market with $N = 1200$, $T = 4000$, and $K = 30$ clusters. The values of the β coefficients and of the Hurst exponents are as in the main text. (Left) Homogeneous system with 40 stocks in each cluster. In red the best fit Marčenko-Pastur distribution of Eq. (9) with parameters $q = 0.284 \pm 0.002$ and $\sigma = 0.939 \pm 0.001$ with upper edge $\lambda_+ = 2.0756$. The inset includes the $m_{\max} = 30$ eigenvalues beyond λ_+. (Right) Same plot but for a heterogeneous system with the same parameters, but a different cluster structure defined in Section 6.1. Here $\lambda_+ = 1.9322$, $q = 0.299 \pm 0.004$, and $\sigma = 0.898 \pm 0.002$. Finally, in this case there are $m_{\max} = 28$ eigenvalues beyond λ_+.

approximation. We also remark that the MP fits in Figs. 5(a) and 5(b) are better than that of Figs. 1(a) and 1(b) because we can tune the white noise in our synthetic data so that the bulk in this region behaves more similar to white noise. This is achieved by changing the value of ϕ.

For each sample we find the median $\zeta(m)$, plotting it in log-log scale in Fig. 4(a) for the homogenous system and in Fig. 4(b) for the heterogeneous one.

As already described, the optimal value $m^\star$ turns out to be 19 and 12 for the homogenous and heterogeneous systems respectively. The fact that $m^\star$ is lower for the heterogeneous system makes sense since its broad, power law distributed values of N_k mean that more of the memory of the system is contained in earlier principal components, whose N_k are larger. Since more of the memory is concentrated in fewer principal components, it is natural that the corresponding values of $m^\star$ will be lower for the heterogenous system. On the other hand for the homogenous system, we have that the N_k are equal for all k, so we can expect that the memory is more evenly distributed across the principal components i.e. that $m^\star$ will be larger. We also apply the method in Section 5 to the data matrix $\boldsymbol{X}$ corresponding to the empirical dataset described in Appendix A, for which $m_{\max} = 35$ (see caption of Fig. 4(c) for details).

7. Comparison with other heuristic methods to select $m^\star$

In this Section, we shall compare our new method with available "stopping rules" in the literature. Many heuristic methods have been proposed in order to determine $m^\star$, generally falling into three categories: subjective methods, distribution-based methods and computational procedures [1, 2]. We describe here the most common ones in each category.

In the class of subjective methods, we find two similar procedures, the *cumulative percentage of variation* [21, 22] and *scree plots* [20]. The former is based on selecting the minimum value of m such that the cumulative percentage of variation explained by the m principal components exceeds some threshold α:

$$m^\star = \min_m \{\Lambda(m) > \alpha\} \ , \tag{24}$$

$$\Lambda(m) = 100 \frac{\sum_{p=1}^{m} \lambda_p}{N} \ , \tag{25}$$

where $\Lambda(m)$ is the % cutoff, α is the percentage cutoff threshold and $\{\lambda_p\}_{p=1}^m$ are the first m eigenvalues of $\boldsymbol{G}$. Common cutoff ranges lie somewhere between 70% to 90%, with a preference towards larger values when it is known or obvious that the first few principal components will explain most of the variability in the data [1]. An obvious disadvantage of this method is that it relies on the choice of some arbitrary value for the tolerance α.

Scree plots involve plotting a 'score' representing the amount of variability in the data explained by individual principal components, and then choosing the point at which the plot develops an 'elbow', beyond which picking further principal components does not significantly enhance the level of memory already accounted for. This procedure again has the obvious drawback of relying on graphical inspection and therefore being even more subjective than the cumulative percentage of variation.

Among the class of distribution-based methods, the most commonly used procedure is the Bartlett Test [23]. This involves testing the null hypothesis [1]

$$H_{0,m} = \lambda_{m+1} = \lambda_{m+2} = ... = \lambda_N \ , \tag{26}$$

that is whether the last $N - m$ eigenvalues are identical, against the alternative that at least two of the last $N - m$ eigenvalues are not identical, and repeating this test for various values of m. One then selects the maximum value of m for which the outcome of the hypothesis test is significant. Intuitively, this procedure tests whether the last $N-m$ eigenvalues explain roughly the same amount of variability in the data so that they can be regarded as noise, and then takes $m^\star$ to be the maximum number of "significant" eigenvalues. According to this procedure, one first tests $H_{0,N-2}$ i.e. whether $\lambda_{N-1} = \lambda_N$. If this hypothesis is not rejected, then one tests $H_{0,N-3}$, and if this is not rejected the exact same test is performed for $H_{0,N-4}$ and so on. The procedure carries on testing each individual $H_{0,m}$ until the first time ($m = m^\star - 1$) the hypothesis gets rejected at the required confidence level. Since several tests need to be conducted sequentially, the overall significance of the procedure will not be the same as the one imposed for each individual test, with no way of correcting for this bias as the number of tests to be performed is *a priori* unknown. This drawback makes distribution-based methods very impractical with real data [1].

The last category (computational procedures) involves the use of cross-validation. Cross-validation requires that some chunks of the original dataset $\boldsymbol{X}$ be initially removed. The remaining data matrix entries are used in conjunction with Eq. (17) to cast predictions on the removed entries using m principal components. We focus on so called 10-fold contiguous block cross-validation, which has been argued to be optimal in the sense that it most accurately captures the true structure of the correlation matrix (either $\boldsymbol{E}$ or $\boldsymbol{G}$) [68]. According to this procedure, we divide the data matrix $\boldsymbol{X}$ into 10 rectangular blocks row-wise, which we call $\boldsymbol{X}^{(g)}$ for $g = 1, ..., 10$. For each group g, we calculate the correlation matrix $\boldsymbol{G}^{(g)}$associated with the matrix $\boldsymbol{X}$ but with the block $\boldsymbol{X}^{(g)}$ removed. Next, we take m principal components of $\boldsymbol{G}^{(g)}$ and use them in a factor model like in Eq. (17) but with m as the upper limit for the sum to predict the values of $\boldsymbol{X}^{(g)}$, which we call $\hat{\boldsymbol{X}}^{(g,m)}$. We then repeat this procedure for every m and g.

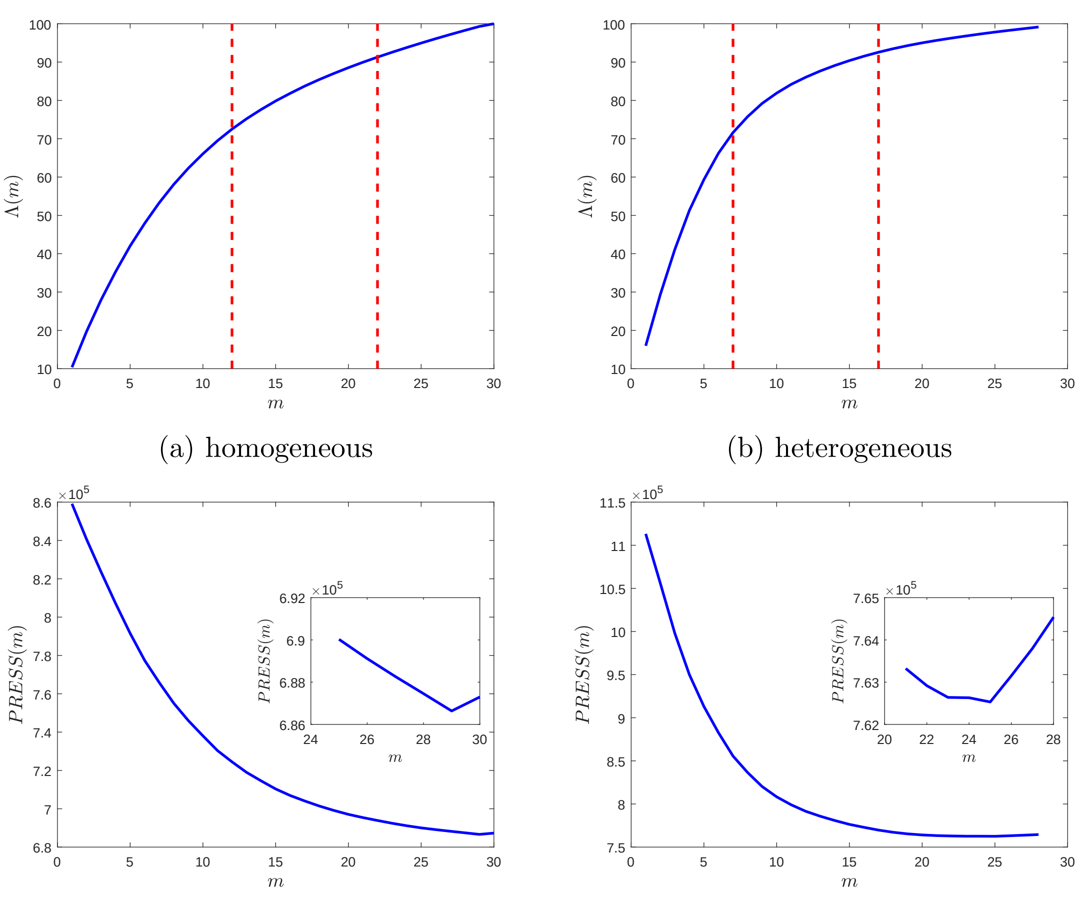
After doing so, we can calculate the Prediction Residual Error Square Sum, or PRESS(m), as a function of m. This is the total (un-normalised) squared prediction error for each value and over all blocks

$$\mathrm{PRESS}(m) = \sum_{i=1}^{N} \sum_{g=1}^{10} \sum_{t \in \mathcal{G}_g} \left(\hat{\boldsymbol{X}}_{ti}^{(g,m)} - \boldsymbol{X}_{ti}^{(g)} \right)^2 \quad , \tag{27}$$

with $\hat{\boldsymbol{X}}^{(g,m)}$ being the matrix of predicted values for block g using m principal components, and $\mathcal{G}_g$ indicating the row indices belonging to block g. Eq. (27) represents the out-of-sample error in predicting the entries of $\boldsymbol{X}$, which implies that PRESS(m) should initially decrease as m increases. However, beyond a certain threshold, PRESS(m) might start to increase instead, indicating that we are beginning to overfit the data. The optimal $m^\star$ should therefore be chosen to be the value which minimises PRESS(m), thus striking the optimal balance between increasing the model complexity and overfitting the data. This procedure has an obvious advantage over the previous two categories as it is parameter-free and not subjective. However, one significant drawback for practical purposes is that the procedure becomes computationally very expensive for large datasets due to the typically $\sim \mathcal{O}(N m_{\max})$ regressions that need to be performed from the dataset.

We compare our memory-based method, the cumulative variance method with 70% and 90% cutoffs and the 10-fold cross-validation method for 100 samples of the synthetic system described in Section 6.1 and for the empirical dataset described in Appendix A, where the numerical outputs of $m^\star$ for these methods is detailed in the columns of Table 1.

In Fig. 6 (top panel), we plot for the homogeneous and heterogeneous synthetic data the median of $\Lambda(m)$ (see Eq. (25)) over all samples, indicating the 70% and 90% cutoffs in dashed red lines. The 70% and 90% cutoffs yield an optimal number of 12



(a) homogeneous

(b) heterogeneous

(c) homogeneous

(d) heterogeneous

Figure 6: (Top) The median value of $\Lambda(m)$ (see Eq. (25)) for the cumulative variance method for the synthetic homogeneous and heterogeneous systems respectively. The 70% and 90% cutoff levels are indicated in dashed red lines, and occur at $m = 12, 22$ for the homogeneous system and $m = 7, 17$ for the heterogeneous one. (Bottom) For the homogenous and heterogeneous systems again, we plot the median of PRESS(m) (see Eq. (27)) using 10 fold cross-validation for each sample. We see from the zoomed inset figures that the minimum PRESS(m) occurs at $m = 29$ for the homogenous system and $m = 25$ for the heterogeneous one.

and 22 components for the homogeneous case and 7 and 17 for the heterogeneous case, respectively. It makes sense that fewer components are needed in the heterogeneous case as more of the total variance is accounted for by the first principal components, which correspond by construction to the larger clusters. We recall that our memory-based method predicts $m^\star = 19$ and $m^\star = 12$ for the homogeneous and heterogeneous cases respectively, and these values fall squarely between the prescribed 70% and 90% cutoffs [1]. However, our method is superior in that it gives a unique value for $m^\star$ and not a range of values, and does not use subjective criteria or rules of thumb.

Fig. 6 (bottom panel) depicts the median of PRESS(m) across all samples, from

	synthetic		empirical
	homogeneous	heterogeneous	
memory-based	**19**	**12**	**15**
cumulative variance	12–22	7–17	13–27
cross-validation	29	25	28
m_{max}	30	28	35

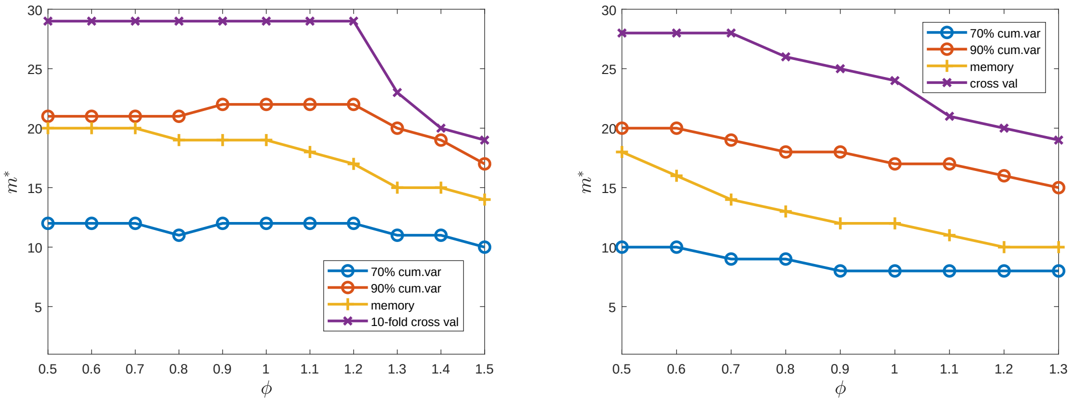
Table 1: This table summarises the $m^\star$ values obtained for the synthetic data described in Section 6.1 and empirical dataset described in Appendix A. Results from our memory-based method from Section 6.2 are included in the first row. In the second row, we have the cumulative variance rule for the cutoffs 70% and 90%. The final row includes the PRESS(m) (see Eq. (27)), using 10-fold cross-validation.

	synthetic		empirical
	homogeneous	heterogeneous	
memory-based	**138.6**	**137.6**	**209.7**
cross-validation	1136.8	1146.3	1462.3

Table 2: Computational times in seconds for our proposed memory-based method (first row) and cross-validation using 10 contiguous blocks (second row). The first two columns refer to the homogeneous and heterogeneous synthetic systems in Section 6.1. The final column is for the empirical dataset described in Appendix A. These performance times were calculated on a Windows 10, CPU Intel i7-6700 3.4 GHz, RAM 16GB PC using MATLAB 2017b.

which we see that the minimum occurs at $m^\star = 29$ for the homogenous system and $m^\star = 25$ for the heterogenous one. Hence, the cross-validation method would induce us to keep the majority of components in both systems. This is to be expected since cross-validation is based on minimising the out-of-sample prediction error (see Eq. (17)), hence performing the linear regression many times necessarily leads to a higher likelihood of including a larger number of principal components. This comes of course at the price of computational speed. Another interesting observation is that the minima occurring in both systems are not sharply defined, which indicates that the out-of-sample error made by including a larger number of components than the optimum $m^\star$ does not actually increase by a significant amount.

Compared to cross-validation, our methodology leads to keeping fewer components. Our procedure, however, is less computationally expensive since it performs far fewer regressions to find $m^\star$ (see Table 2). Another advantage of our method over cross-validation can be spotted in the top panel of Fig. 4, which highlights that only 9% and 6% of the total memory for the homogenous and heterogeneous systems after removing the market mode is unaccounted for to the right of $\hat{\theta}$. From the perspective of explaining



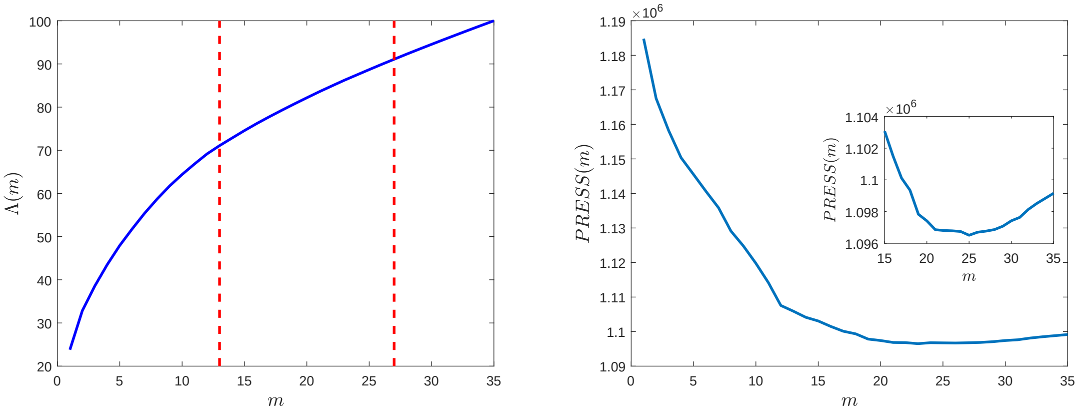
(a) (b)

Figure 7: A comparison of the different methods for selecting $m^\star$ by varying ϕ, the noise level in the simulation of synthetic data (see Eq. (23)). For each value of ϕ, 100 samples of the process are generated, with the results for the homogeneous system plotted on the left, and for the heterogeneous system on the right. The blue and red lines represent the results for the 70% and 90% cumulative variance procedure. The orange line corresponds to our method. Finally the purple line represents results from 10-fold cross-validation.

the memory in the time series, therefore, our method does on average a very good job while requiring very limited computational resources.

Now that we have compared the methods for a fixed ϕ, the variance of the noise term for our synthetic data (see Eq. (23)), we can check how robust each of the methods is to changes in ϕ. We note that fixing $\phi = 1$ constitutes already a hard regime to analyse since it implies that the fluctuations due to $I_k(t)$ are of the same magnitude as the white noise, so we can see already that our method stands well compared to others with this high value of ϕ. In Fig. 7, we compare – using 100 samples of the synthetic systems for the homogeneous and the heterogeneous cases – the optimal value $m^\star$ predicted by the cumulative variance method with 70% and 90% cutoffs, the 10-folds cross-validation method and our own memory-based procedure as we vary ϕ.

The 70% and 90% cutoffs for the cumulative variance rule remain relatively stable for most values of ϕ, before slowly decreasing for higher values of ϕ. This decrease occurs because the increased level of noise lowers the contribution to the variance from higher components, with the consequence that the cutoff is reached sooner for higher values of ϕ. Within our memory-based method, the value of $m^\star$ decreases for increasing ϕ. This decrease occurs because a higher amount of white noise increasingly masks the long-memory properties of the underlying signal, and will affect the deeper principal components more since they have a lower memory strength (lower H_k) anyway. This is a desirable property since it means that lowering the noise level will lead us to retain more principal components. Whilst the decrease in the number of components occurs



(a) (b)

Figure 8: Comparison between the cumulative variance rule, cross-validation and our memory-based method of determining $m^\star$ applied to the empirical dataset defined in Appendix A. (Left) Plot of $\Lambda(m)$ defined in Eq. (25) with the red dashed lines at $m = 13$ and $m = 27$ indicating the region where between 70% and 90% of the total variance is explained by the principal components. (Right) Plot of PRESS(m) given in Eq. (27) using 10-fold cross-validation, with a zoomed in inset version showing that the minimum occurs at $m = 28$.

earlier than for the cumulative variance method, it still remains between the 70% and 90% cutoffs, and even closer to the 90% cutoff for lower values of ϕ.

For the empirical dataset, described in Appendix A, we plot in Fig. 8 (left) the plot of $\Lambda(m)$, the cumulative percentage of variation explained by the m principal components. We see that if we set our target between 70% and 90% of the cumulative variance as prescribed in [1], this will correspond to retaining between 13 and 27 components, but again it is not clear a priori what exact value within this range we should pick. In Fig. 8 (right), we plot PRESS(m) obtained via 10-fold cross-validation, in which the minimum occurs at $m^\star = 28$, close to the 90% cutoff for the cumulative variance. Again – compared to cross-validation – our method picks out fewer principal components, but we obtain our result in far less computational time (see Table 2), and with $m^\star = 15$ we can already account for 80% of the memory.

8. Conclusion

In this paper, we have proposed a novel, data-driven method to select the optimal number $m^\star$ of principal components to retain in the Principal Component Analysis of data with long memory. The main steps are detailed in Section 5. We used the crucial fact that subsequent components contribute a decreasing amount to the total memory of the system. This allows us to identify a unique, non-subjective and computationally inexpensive stopping criterion, which compares very well with other available heuristic procedures such as cumulative variance and cross-validation (see Tables 1 and 2).

We tested our method on two synthetic systems: a homogeneous and heterogeneous version 6.1, and also on an empirical dataset of financial log-volatilities, described in Appendix A. Our results could be applied to any large dataset endowed with long-memory properties, for example in climate science [35, 36] and neuroscience [5, 37]. A potential direction for future work could be using a null hypothesis for the bulk eigenvalues which takes into account the presence of autocorrelations rather than the MP distribution used here. A comparison with the cluster driven method presented in [15] or extending the method for example to nonlinear PCA [69] could also be explored.

Appendix A.
Empirical Dataset

The empirical dataset we shall use consists of the daily closing prices of 1270 stocks in the New York Stock Exchange (NYSE), National Association of Securities Dealers Automated Quotations (NASDAQ) and American Stock Exchange (AMEX) from 1st January 2000 to 12th May 2017, which amounts to 4635 entries for each price time series. We make sure that the stocks are "aligned" through the data cleaning procedure described here. A typical source of misalignment is the fact that some stocks have not been traded on certain days. To ensure we keep as many entries as possible, we fill the gaps dragging the last available price ahead and assuming that a gap in the price time-series corresponds to a zero log-return. At the same time, we do not wish to drag ahead too many prices as doing so would compromise the statistical significance of the time-series. The detailed procedure goes as follows:

(i) Remove from the dataset the price time-series with length smaller than p times the longest one;

(ii) Find the common earliest day among the remaining time-series;

(iii) Create a reference time-series of dates when at least one of the stocks has been traded starting from the earliest common date found in the previous step;

(iv) Compare the reference time-series of dates with the time-series of dates of each stock and fill the gaps dragging ahead the last available price.

In this paper, we chose $p = 0.90$ to ensure that we keep the time-series as unmodified as possible. For example, the common earliest day for our dataset is 3rd of January 2000. In this period, the stock Ameris Bancorp (ABCB), was not traded on 35 days in the time period and therefore the last available price was used to fill these particular days. Another example is the stock Allied Healthcare Products (AHPI), which was not traded for 508 days in the time period we study, and is removed since its length is less than p times the longest time series. However, the results do not change if we pick a higher value of p. Applying this procedure leaves our dataset with $N = 1202$ stocks. Hence $\boldsymbol{X}$ and $\boldsymbol{X}^{(\text{market})}$ are 4364×1202 matrices.

Appendix B.

Financial interpretation of the eigenvectors and portfolio optimisation

Another motivation for the application of PCA to financial correlation matrices is the financial interpretation of the first principal components, which we explain here. First, we recall that the empirical correlation matrix $\boldsymbol{E}$ between the standardised log volatilities is defined as

$$E_{ij} = \frac{1}{T}\sum_{t=1}^{T} \ln|r_i(t)| \ln|r_j(t)| \ . \tag{B.1}$$

We call $\boldsymbol{w}_m$ the eigenvectors of $\boldsymbol{E}$ with λ_m its associated eigenvalue. We interpret the entries of $\boldsymbol{w}_m$ as the weights of a portfolio, with $w_{im} > 0$ indicating a long position where we buy the stock in the expectation that its value will rise, and $w_{im} < 0$ denoting a short position where we expect the stock's value to fall and hence sell it [70].

The covariance between the log volatilities of the portfolios m and m' is:

$$\frac{1}{T}\sum_{t=1}^{T}\left(\sum_{i=1}^{N} w_{im}\ln|r_i(t)|\right)\left(\sum_{j=1}^{N} w_{jm'}\ln|r_j(t)|\right) = \sum_{m'} \lambda_m \delta_{m,m'} \ , \tag{B.2}$$

where w_{im} and $w_{jm'}$ are the entries of the eigenvector $\boldsymbol{w}_m$ and $\boldsymbol{w}_{m'}$ respectively. Hence the returns defined by the portfolio $\boldsymbol{w}_m$ and another eigenvector $\boldsymbol{w}_{m'}$ are uncorrelated. Another consequence of Eq. B.2 is that the variance of the returns, which is used to measure the risk of a portfolio, is the eigenvalue λ_m. Hence larger eigenvalues of the portfolio defined by $\boldsymbol{w}_m$ have a higher risk. Knowing this information about the eigenvalues and their corresponding eigenvectors can therefore inform an investment manager in deciding how to pick portfolios both individually and to reduce a set of portfolios' overall risk by using orthogonal portfolios defined by $\boldsymbol{w}_m$.

For a given level Δ of tolerable risk, we can also find the optimal investment weights $\boldsymbol{w}_{\text{opt}}$ by solving the minimisation problem

$$\min_{\boldsymbol{w}} \boldsymbol{w}^T \boldsymbol{E} \boldsymbol{w} \tag{B.3}$$

$$\text{such that } \boldsymbol{X}\boldsymbol{w} = \Delta \ . \tag{B.4}$$

This is known as Markowitz portfolio optimisation theory [71], and can be solved via Lagrange multipliers to give

$$\boldsymbol{w}_{\text{opt}} = \Delta \frac{\boldsymbol{E}^{-1}\boldsymbol{R}}{\boldsymbol{R}^{\dagger}\boldsymbol{E}^{-1}\boldsymbol{R}} \ , \tag{B.5}$$

with $\boldsymbol{w}_{\text{opt}}$ indicating the optimal portfolio weight. We see that the distribution of the eigenvalues enters the portfolio optimisation through the inverse matrix $\boldsymbol{E}^{-1}$ in Eq. (B.5). Normally, Eq. (B.5) is applied directly by simply using the sample estimator $\boldsymbol{E}$. However, since $\boldsymbol{E}$ is empirical, it is subject to noise inherent in the data which means it is vulnerable to the noisy distribution of the eigenvalues, in turn causing the $\boldsymbol{w}_{opt}$ found to underestimate risk [7].

We also note that in line with [51], the eigenvectors corresponding to these eigenvalues beyond the MP bulk for $\boldsymbol{G}$ for the empirical data in Appendix A can be

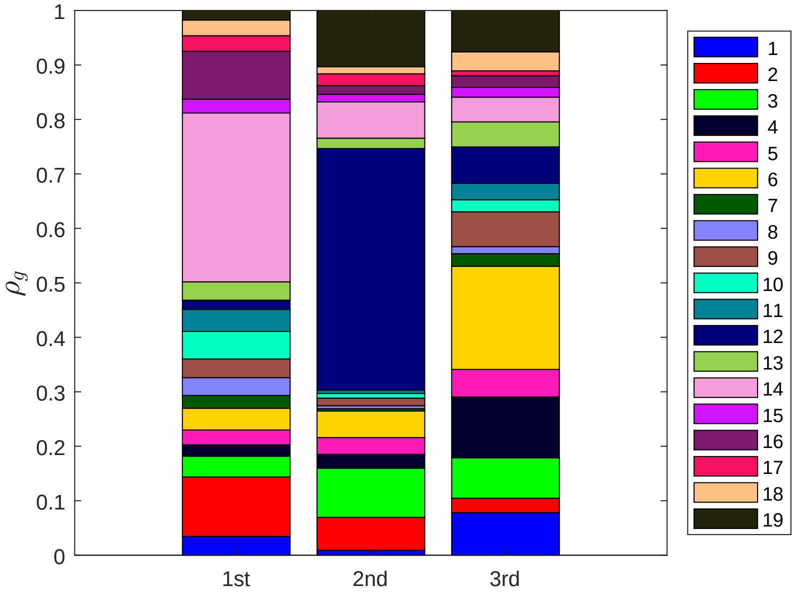


Figure B1: Plots the ϱ_g defined in Appendix B, which is the projection of the eigenvector onto the ICB supersector groups, for the eigenvectors of the first three principal components of $\boldsymbol{G}$ for the data detailed in Appendix A. The legend corresponds to each of the ICB supersector groups.

identified as belonging to particular or a mixture of 19 economic Industrial Classification Benchmark (ICB) supersectors [72]. We can quantify this for the eigenvectors of $\boldsymbol{G}$ given in Eq. (15), $\boldsymbol{v}_i$, by defining a 19-dimensional vector ϱ_i, with entries $\varrho_{g,i}$, $g = 1, ..., 19$. Specifically, we define a projection matrix $\boldsymbol{P}$ with entries

$$P_{ig} = \begin{cases} 1/N_g & \text{if } i \text{ is in supersector } g \\ 0 & \text{else} \end{cases} ,$$

where N_g is the number of stocks that are part of supersector g. From this we can define ϱ_i as

$$\varrho_i = \gamma_i \boldsymbol{P} \boldsymbol{v}_i \ , \qquad \text{(B.6)}$$

where γ_i is the normalisation constant $\sum_{g=1}^{19} \varrho_{g,i}$. Each $\varrho_{g,i}$ gives the contribution of the g-th ICB supersector to the *ith* eigenvector. We plot ϱ_g for the first three eigenvectors in Fig. B1. We can see that each eigenvector is dominated by the Real Estate (colour 14), Oil and Gas (colour 1) and Financial Services (colour 6) respectively for the first, second and third principal components.

Appendix C.

Lasso regression

Lasso regression is used to find the values of the coefficients β_{ip} using Eq. (17). Further details of the use of this method is provided in this appendix. Lasso regression [73] provides a way of dealing with overfitting explanatory variables (in our case $I_k(t)$) and also of performing feature selection, which takes into account a stock i's log-volatility not being affected by changes of $I_k(t)$. Lasso regression solves the constrained minimisation problem

$$\min_{\boldsymbol{\beta}_i} \frac{1}{T} \sum_{t=1}^{T} \left(c_i(t) - \boldsymbol{I}(t)^{\dagger} \boldsymbol{\beta}_i \right)^2 + \Upsilon P_a(\boldsymbol{\beta}_i) \ , \tag{C.1}$$

where $\boldsymbol{\beta}_i$ is the vector of loadings given by $(\beta_{i1}, \beta_{i2}, \ldots, \beta_{im_{\max}})^{\dagger}$, $\boldsymbol{I}(t)$ is the matrix whose columns are $(I_1(t), I_2(t), \ldots, I_{m_{\max}}(t))$ and Υ is a hyperparameter. $P_a(\boldsymbol{\beta}_i)$ is defined as

$$P_a(\boldsymbol{\beta}_i) = \sum_{j=1}^{m_{\max}} |\beta_{ij}| \ . \tag{C.2}$$

The sum in Eq. (C.2) is the $\mathcal{L}_1$ penalty for the lasso regression. The Υ controls the amount of regularisation: the higher it is, the more loadings are zero. To find Υ, we set its scale according to [74] and use 10 cross-validated fits [73], picking the Υ that gives the minimum prediction error. We have also investigated the stability of the results with respect to changes in Υ, and altering the penalty in (C.2) to a L2 penalty. In either case there is little difference to the calculated $m^{\star}$ values.

Appendix D.
Fitting Procedure for θ

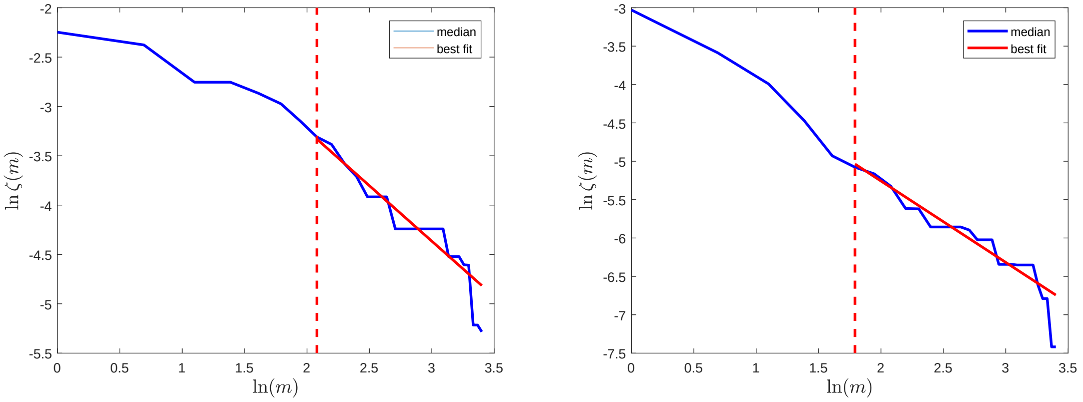
We can estimate θ by assessing what region of $\zeta(m)$ is most linear in log-log scale, which is done by assessing on each interval $m = \tilde{\theta}, ..., m_{\max}$, where $\tilde{\theta} = 2, ..., m_{\max}$, the quality of a linear fit in log-log scale of $\zeta(m)$ in this interval. The estimate of θ, $\hat{\theta}$ is then the value of $\tilde{\theta}$ that gives the best-quality linear fit. To assess the quality of the fit, we use the adjusted R^2_{adj} value [59]:

$$R^2_{\mathrm{adj}} = 1 - (1 - R^2) \frac{n-1}{n-2} \ , \tag{D.1}$$

where R^2 is the normal coefficient of determination [75], and n is the size of the interval. Note we have written the formula for our specific case where the number of explanatory variables is 1. If R^2_{adj} is higher, then the interval $m = \tilde{\theta}, ..., m_{\max}$ is better described by a linear trend. The difference between R^2_{adj} and R^2 is that the former can take into account the different sample sizes induced by the differently sized intervals by reducing the value obtained through R^2 for smaller values of n. $\hat{\theta}$ is then given by

$$\hat{\theta} = \max_{\tilde{\theta}} R^2_{\mathrm{adj}}(\tilde{\theta}) \ , \tag{D.2}$$

which is the value of $\tilde{\theta}$ which maximises R^2_{adj} and gives the region of best-quality linear fit.



(a) homogeneous

(b) heterogeneous

Figure E1: (Top left) Plot of $\ln(\zeta(m))$ vs $\ln(m)$ for the same homogeneous synthetic system described in the original manuscript but using AR(1) as the generating process for $I_{k(i)}(t)$. The blue line is the value of $\zeta(m)$ across all assets, with the dashed red line indicating $\hat{\theta} = 8$, the point at which the concavity changes. (Top right) Same plot but for the same heterogeneous simulated system described in the original manuscript, where $\hat{\theta} = 6$.

Appendix E. Exponentially decaying autocorrelations

The Autoregressive process of order 1 (AR(1)) is given by [40]

$$X(t) = \epsilon(t) + \psi X(t-1) \ , \tag{E.1}$$

where $\epsilon(t), \epsilon(t-1), ...$ are all white noise terms, ψ is the autoregressive parameter. To enforce stationarity and positive autocorrelations note that we must have that $0 < \psi < 1$ [40]. The presence of the second term in Eq. (E.1) introduces memory into the process. The autocorrelation function of $X(t)$ is known to be exponential [40], with increasing ψ increasing the strength of the memory, in contrast to the FBM we used in section 6.1. By using AR(1) to generate I_0 and the set of $I_{k(i)}(t)$ with parameters ψ_0 and ψ_k respectively, we can investigate whether the method proposed here is still valid when the autocorrelation decays exponentially. For $I_0(t)$, we fix $\psi_0 = 0.95$. Each $I_{k(i)}(t)$ is generated using an equally spaced vector from 0.65 to 0.95 for ψ_k set in a similar way described in section 6.1 to reflect the empirical result of [67]. We then repeat the steps given in section 5.4 for the same homogenous and heterogenous synthetic systems described in section 6.1. The log-log plots of $\zeta(m)$ vs m are detailed in Fig. E1. We see that for both systems whilst we do see a decrease, it is not accurately described by a straight line in log-log scales in this case, as compared to Figs. 4(a) and 4(b). Therefore we can conclude that whilst our method can be applied also in the case of faster, exponentially decaying autocorrelation, it is less precise.

Acknowledgements

We thank Bloomberg for providing the data used in this paper. We also wish to thank the ESRC Network Plus project 'Rebuilding macroeconomics'. We acknowledge support from Economic and Political Science Research Council (EPSRC) grant EP/P031730/1. We are grateful to the NVIDIA corporation for supporting our research in this area with the donation of a GPU.

[1] Ian Jolliffe. *Principal component analysis.* Wiley Online Library, 2002.

[2] Donald A Jackson. Stopping rules in principal components analysis: a comparison of heuristical and statistical approaches. *Ecology*, 74(8):2204–2214, 1993.

[3] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image processing, analysis, and machine vision.* Cengage Learning, 2014.

[4] Sarah A Mueller Stein, Anne E Loccisano, Steven M Firestine, and Jeffrey D Evanseck. Principal components analysis: a review of its application on molecular dynamics data. *Annual Reports in Computational Chemistry*, 2:233–261, 2006.

[5] Rich Pang, Benjamin J Lansdell, and Adrienne L Fairhall. Dimensionality reduction in neuroscience. *Current Biology*, 26(14):R656–R660, 2016.

[6] Carol Alexander. Principal component models for generating large garch covariance matrices. *Economic Notes*, 31(2):337–359, 2002.

[7] Joël Bun, Jean-Philippe Bouchaud, and Marc Potters. Cleaning large correlation matrices: tools from random matrix theory. *Physics Reports*, 666:1–109, 2017.

[8] J.H.M. Darbyshire. *The Pricing and Trading of Interest Rate Derivatives: A Practical Guide to Swaps.* Aitch and Dee Limited, 2016.

[9] Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10:66–71, 2009.

[10] Tomaso Aste, Tiziana Di Matteo, and ST Hyde. Complex networks on hyperbolic surfaces. *Physica A: Statistical Mechanics and its Applications*, 346(1-2):20–26, 2005.

[11] Michele Tumminello, Tomaso Aste, Tiziana Di Matteo, and Rosario N Mantegna. A tool for filtering information in complex systems. *Proceedings of the National Academy of Sciences*, 102(30):10421–10426, 2005.

[12] Tiziana Di Matteo, Francesca Pozzi, and Tomaso Aste. The use of dynamical networks to detect the hierarchical organization of financial market sectors. *The European Physical Journal B*, 73(1):3–11, 2010.

[13] Tomaso Aste, Ruggero Gramatica, and Tiziana Di Matteo. Exploring complex networks via topological embedding on surfaces. *Physical Review E*, 86(3):036109, 2012.

[14] Wolfram Barfuss, Guido Previde Massara, Tiziana Di Matteo, and Tomaso Aste. Parsimonious modeling with information filtering networks. *Physical Review E*, 94(6):062306, 2016.

[15] Anshul Verma, Riccardo Junior Buonocore, and Tiziana Di Matteo. A cluster driven log-volatility factor model: a deepening on the source of the volatility clustering. *Quantitative Finance*, pages 1–16, 2018.

[16] Geoffrey E Hinton and Richard S Zemel. Autoencoders, minimum description length and helmholtz free energy. In *Advances in neural information processing systems*, pages 3–10, 1994.

[17] Jeffrey Regier and Jon McAuliffe. Second-order stochastic variational inference. In *Bay Area Machine Learning Symposium (BayLearn)*, 2016.

[18] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[19] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent component analysis*, volume 46. John Wiley & Sons, 2004.

[20] Raymond B Cattell. The scree test for the number of factors. *Multivariate behavioral research*, 1(2):245–276, 1966.

[21] T Sugiyama and Howell Tong. On a statistic useful in dimensionality reduction in multivariable linear stochastic system. *Communications in statistics-theory and methods*, 5(8):711–721, 1976.

[22] Deng-Yuan Huang and Sheng-Tsaing Tseng. A decision procedure for determining the number of components in principal component analysis. *Journal of statistical planning and inference*, 30(1):63–71, 1992.

[23] Maurice S Bartlett. Tests of significance in factor analysis. *British Journal of Mathematical and Statistical Psychology*, 3(2):77–85, 1950.

[24] Asghar Ali, GM Clarke, and K Trustrum. Principal component analysis applied to some data from fruit nutrition experiments. *The Statistician*, pages 365–370, 1985.

[25] Svante Wold. Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics*, 20(4):397–405, 1978.

[26] HT Eastment and WJ Krzanowski. Cross-validatory choice of the number of components from a principal component analysis. *Technometrics*, 24(1):73–77, 1982.

[27] Jan Beran. *Statistics for long-memory processes.* Routledge, 2017.

[28] Rama Cont. Empirical properties of asset returns: stylized facts and statistical issues. *Quantitative Finance*, 1:223–236, 2001.

[29] John Hull and Alan White. The pricing of options on assets with stochastic volatilities. *The journal of finance*, 42(2):281–300, 1987.

[30] John C Hull. *Options, futures, and other derivatives.* Pearson Education India, 2006.

[31] Jean-Philippe Bouchaud and Marc Potters. *Theory of financial risk and derivative pricing: from statistical physics to risk management.* Cambridge university press, 2009.

[32] Luc Bauwens, Sébastien Laurent, and Jeroen VK Rombouts. Multivariate garch models: a survey. *Journal of applied econometrics*, 21(1):79–109, 2006.

[33] Peter K Clark. A subordinated stochastic process model with finite variance for speculative prices. *Econometrica: journal of the Econometric Society*, pages 135–155, 1973.

[34] Torben G Andersen, Tim Bollerslev, Francis X Diebold, and Paul Labys. Modeling and forecasting realized volatility. *Econometrica*, 71(2):579–625, 2003.

[35] Hans Von Storch and Francis W Zwiers. *Statistical analysis in climate research.* Cambridge university press, 2001.

[36] Christian Franzke. Nonlinear trends, long-range dependence, and climate noise properties of surface temperature. *Journal of Climate*, 25(12):4172–4183, 2012.

[37] Klaus Linkenkaer-Hansen, Vadim V Nikouline, J Matias Palva, and Risto J Ilmoniemi. Long-range temporal correlations and scaling behavior in human brain oscillations. *Journal of Neuroscience*, 21(4):1370–1377, 2001.

[38] Stephen J Taylor. Modeling stochastic volatility: A review and comparative study. *Mathematical finance*, 4(2):183–204, 1994.

[39] F Jay Breidt, Nuno Crato, and Pedro De Lima. The detection and estimation of long memory in stochastic volatility. *Journal of econometrics*, 83(1-2):325–348, 1998.

[40] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*, page 33. John Wiley & Sons, 2015.

[41] Ajay Singh and Dinghai Xu. Random matrix application to correlations amongst the volatility of assets. *Quantitative Finance*, 16(1):69–83, 2016.

[42] Rudi Schäfer, Nils Fredrik Nilsson, and Thomas Guhr. Power mapping with dynamical adjustment for improved portfolio optimization. *Quantitative Finance*, 10(1):107–119, 2010.

[43] Vladimir A Marčenko and Leonid Andreevich Pastur. Distribution of eigenvalues for some sets of random matrices. *Sbornik: Mathematics*, 1(4):457–483, 1967.

[44] Giacomo Livan, Marcel Novaes, and Pierpaolo Vivo. *Introduction to Random Matrices: Theory*

and Practice, volume 26. Springer, 2018.

[45] Thomas Guhr and Bernd Kälber. A new method to estimate the noise in financial correlation matrices. *Journal of Physics A: Mathematical and General*, 36(12):3009, 2003.

[46] Giacomo Livan, Simone Alfarano, and Enrico Scalas. Fine structure of spectral properties for random correlation matrices: An application to financial markets. *Physical Review E*, 84(1):016113, 2011.

[47] Mateusz Wilinski, Yuichi Ikeda, and Hideaki Aoyama. Complex correlation approach for high frequency financial data. *Journal of Statistical Mechanics: Theory and Experiment*, 2018(2):023405, 2018.

[48] Giulio Biroli, Jean-Philippe Bouchaud, and Marc Potters. The student ensemble of correlation matrices: Eigenvalue spectrum and kullback-leibler entropy. *Acta Physica Polonica B*, 38(13), 2007.

[49] AY Abul-Magd, Gernot Akemann, and P Vivo. Superstatistical generalizations of wishart–laguerre ensembles of random matrices. *Journal of Physics A: Mathematical and Theoretical*, 42(17):175207, 2009.

[50] Laurent Laloux, Pierre Cizeau, Jean-Philippe Bouchaud, and Marc Potters. Noise dressing of financial correlation matrices. *Physical review letters*, 83(7):1467, 1999.

[51] Vasiliki Plerou, Parameswaran Gopikrishnan, Bernd Rosenow, Luis A Nunes Amaral, Thomas Guhr, and H Eugene Stanley. Random matrix approach to cross correlations in financial data. *Physical Review E*, 65(6):066126, 2002.

[52] Benoit B Mandelbrot. The variation of certain speculative prices. In *Fractals and Scaling in Finance*, pages 371–418. Springer, 1997.

[53] William F Sharpe. Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, 19(3):425–442, 1964.

[54] Robert C Merton. An intertemporal capital asset pricing model. *Econometrica: Journal of the Econometric Society*, pages 867–887, 1973.

[55] Christian Borghesi, Matteo Marsili, and Salvatore Miccichè. Emergence of time-horizon invariant correlation structure in financial returns by subtraction of the market mode. *Physical Review E*, 76(2):026104, 2007.

[56] Zdzisław Burda, Jerzy Jurkiewicz, and Bartłomiej Wacław. Spectral moments of correlated wishart matrices. *Physical Review E*, 71(2):026111, 2005.

[57] Alex Bloemendal, Antti Knowles, Horng-Tzer Yau, and Jun Yin. On the principal components of sample covariance matrices. *Probability theory and related fields*, 164(1-2):459–552, 2016.

[58] Giancarlo Diana and Chiara Tommasi. Cross-validation methods in principal component analysis: a comparison. *Statistical Methods and Applications*, 11(1):71–82, 2002.

[59] Henry Theil. *Economic forecasts and policy.* North-Holland, 1958.

[60] Claude R Dietrich and Garry N Newsam. Fast and exact simulation of stationary gaussian processes through circulant embedding of the covariance matrix. *SIAM Journal on Scientific Computing*, 18(4):1088–1107, 1997.

[61] Rosario N Mantegna. Hierarchical structure in financial markets. *The European Physical Journal B-Condensed Matter and Complex Systems*, 11(1):193–197, 1999.

[62] Nicolo Musmeci, Tomaso Aste, and Tiziana Di Matteo. Relation between financial market structure and the real economy: comparison between clustering methods. *PloS one*, 10(3):e0116201, 2015.

[63] Nicolo Musmeci, Tomaso Aste, and Tiziana Di Matteo. Risk diversification: a study of persistence with a filtered correlation-network approach. *Journal of Network Theory in Finance*, 1(1):77–98, 2015.

[64] Nicolo Musmeci, Tomaso Aste, and Tiziana Di Matteo. Interplay between past market correlation structure changes and future volatility outbursts. *Scientific Reports 6*, 6:36320, 2016.

[65] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Physical review E*, 78(4):046110, 2008.

[66] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814, 2005.

[67] Salvatore Micciché. Empirical relationship between stocks cross-correlation and stocks volatility clustering. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(05):P05015, 2013.

[68] Joël Bun, Jean-Philippe Bouchaud, and Marc Potters. On the overlaps between eigenvectors of correlated random matrices. *arXiv preprint arXiv:1603.04364*, 2016.

[69] Juha Karhunen and Jyrki Joutsensalo. Representation and separation of signals using nonlinear pca type learning. *Neural networks*, 7(1):113–127, 1994.

[70] Jean-Philippe Bouchaud and Marc Potters. Financial applications of random matrix theory: a short review. *arXiv preprint arXiv:0910.1205*, 2009.

[71] Harry Markowitz. Portfolio selection. *The journal of finance*, 7(1):77–91, 1952.

[72] ftserussel. Industry classification benchmark (icb), 2017.

[73] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[74] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

[75] Norman R Draper and Harry Smith. *Applied regression analysis*, volume 326. John Wiley & Sons, 2014.

[76] Satya N Majumdar and Pierpaolo Vivo. Number of relevant directions in principal component analysis and wishart random matrices. *Physical review letters*, 108(20):200601, 2012.