NETWORK SCIENCE

Community detection in networks without observing edges

Till Hoffmann[1], Leto Peel[2], Renaud Lambiotte[3]*, Nick S. Jones[1,4]*

We develop a Bayesian hierarchical model to identify communities of time series. Fitting the model provides an end-to-end community detection algorithm that does not extract information as a sequence of point estimates but propagates uncertainties from the raw data to the community labels. Our approach naturally supports multiscale community detection and the selection of an optimal scale using model comparison. We study the properties of the algorithm using synthetic data and apply it to daily returns of constituents of the S&P100 index and climate data from U.S. cities.

INTRODUCTION

Detecting communities in networks provides a means of coarse-graining the complex interactions or relations (represented by network edges) between entities (represented by nodes) and offers a more interpretable summary of a complex system. However, in many complex systems, the exact relationship between entities is unknown and unobservable. Instead, we may observe interdependent signals from the nodes, such as time series, which we may use to infer these relationships. Over the past decade, a multitude of algorithms have been developed to group multivariate time series into communities with applications in finance (*1*–*4*), neuroscience (*5*, *6*), and climate research (*7*). For example, identifying communities of assets whose prices vary coherently can help investors gain a deeper understanding of the foreign exchange market (*1*, *2*) or manage their market risk by investing in assets belonging to different communities (*8*). Classifying regions of the brain into distinct communities allows us to predict the onset of psychosis (*6*) and learn about the aging of the brain (*9*). Global factors affecting our climate are reflected in the community structure derived from sea surface temperatures (*7*).

Current methods for detecting communities when network edges are unobservable typically involve a complicated process that is highly sensitive to specific design decisions and parameter choices. Most approaches consist of three steps: First, a measure is chosen to assess the similarity of any pair of time series such as Pearson correlation (*1*–*3*, *7*, *9*, *10*), partial correlation (*6*, *11*, *12*), mutual information (*13*), or wavelet correlation coefficients (*5*, *14*, *15*). Second, the similarity is converted to a dense weighted network (*1*–*3*, *15*) or a binary network. For example, some authors connect the most similar time series such as to achieve a desired network density (*13*), threshold the similarity matrix at a single value (*5*, *7*), or demand statistical significance under a null model (*6*, *11*, *12*). Others threshold the similarity matrix at multiple values to perform a sensitivity analysis (*9*, *10*, *14*). After the underlying network has been inferred, community detection is applied to uncover clusters of time series, for example, by maximizing the modularity (*1*, *2*, *5*, *10*, *14*, *15*) or using the map equation (*7*, *9*, *16*).

This type of approach faces a number of challenges: First, most community detection methods rely on the assumption that the network edges have been accurately observed (*17*). In addition, Newman-Girvan modularity (*18*), a popular measure to evaluate community structure in networks, is based on comparing the network to a null model that does not apply to networks extracted from time series data (*8*). Second, when the number of time series is large, computing pairwise similarities is computationally expensive, and the entries of the similarity matrix are highly susceptible to noise. For example, the sample covariance matrix does not have full rank when the number of observations is smaller than or equal to the number of time series (*19*). Third, at each step of the three-stage process, we generally only compute point estimates and discard any notion of uncertainty such that it is difficult to distinguish genuine community structure from noise, a generic problem in network science (*20*). Fourth, missing data can make it difficult to compute similarity measures such that data have to be imputed (*10*) or incomplete time series are dropped (*3*, *8*). Last and more broadly, determining an appropriate number of communities is difficult (*21*) and often relies on the tuning of resolution parameters without a quality measure to choose one value over another (*2*, *22*).

More broadly, this work is related to the problem of series clustering (*23*), whose purpose is to take a set of time series as input and to group them according to a measure of similarity. Most of these methods are not constructed from a network perspective, but they tend to face the same challenges outlined above. In particular, they often comprise separate steps combined in a relatively ad hoc manner, e.g., transformations based on wavelets or piecewise approximations (*24*, *25*). Accordingly, the resulting disconnected pipelines produce point estimates at each step and do not propagate uncertainty from the raw data to the final output.

Our approach is motivated by the observation that inferring the presence of edges between all pairs of nodes in a network is an unnecessary, computationally expensive step to uncover the presence of communities. Instead, we propose a Bayesian hierarchical model for multivariate time series data that provides an end-to-end community detection algorithm and propagates uncertainties directly from the raw data to the community labels. This shortcut is more than a computational trick, as it naturally allows us to address the aforementioned challenges. In particular, our approach naturally supports multiscale community detection and the selection of an optimal scale using model comparison. Furthermore, it enables us to extract communities even in the case of short observation time windows. The rest of this paper will be organized as follows. After introducing the algorithm, we

[1]Department of Mathematics, Imperial College London, London SW7 2AZ, UK. [2]Institute of Information and Communication Technologies, Electronics and Applied Mathematics (ICTEAM), Université Catholique de Louvain, Louvain-la-Neuve B-1348, Belgium. [3]Mathematical Institute, University of Oxford, Radcliffe Observatory Quarter, Woodstock Road, Oxford OX2 6GG, UK. [4]EPSRC Centre for Mathematics of Precision Healthcare, Imperial College London, London SW7 2AZ, UK.
*Corresponding author. Email: nick.jones@imperial.ac.uk (N.S.J.); renaud.lambiotte@maths.ox.ac.uk (R.L.)

validate and study its properties in a series of synthetic experiments. We then apply it to daily returns of constituents of the S&P100 index to identify salient communities of similar stocks and to climate data of U.S. cities to identify homogeneous climate zones. For the latter, we characterize the quality of the communities in terms of the predictive performance provided by the model.

MATERIALS AND METHODS

A Bayesian hierarchical model

The variability of high-dimensional time series is often the result of a small number of common, underlying factors (*26*). For example, the stock price of oil and gas companies tends to be positively affected by rising oil prices, whereas the manufacturing industry, which consumes oil and gas, is likely to suffer from rising oil prices (*27*). Motivated by this observation, we modeled the multivariate time series y using a latent factor model, i.e., the n-dimensional observations at each time step t are generated by a linear transformation A of a lower-dimensional, latent time series x and additive observation noise. More formally, the conditional distribution of y is

$$y_{ti} \mid A, x, \tau \sim \text{Normal}\left(\sum_{q=1}^{p} x_{tq} A_{iq}, \tau_i^{-1}\right) \quad (1)$$

where y_{ti} is the value of the ith time series at time t, x_{tq} is the value of the qth latent time series, and p is the number of latent time series. The precision (inverse variance) of the additive noise for each time series is τ_i, and Normal(μ, σ^2) denotes the normal distribution with mean μ and variance σ^2. The entries A_{iq} of the $n \times p$ factor loading matrix encode how the observations of time series i are affected by the latent factor q. Using our earlier example, the entry of A connecting an oil company with the (unobserved) oil price would be positive, whereas the corresponding entry for an automobile company would be negative.

Variants of this model abound. For example, the mixture model of factor analyzers (*28*, *29*) assumes that there are not one but many latent factors to account for a possibly nonlinear latent manifold (*30*, *31*). Huopaniemi *et al.* (*32*) and Zhao *et al.* (*33*) demanded that most of the entries of the factor loading matrix are zero such that each observation only depends on a subset of the latent factors. Inoue *et al.* (*34*) modeled gene expression data and assumed that the factor loadings of all genes belonging to the same community are identical.

We aim to strike a balance between the restrictive assumption that observations belonging to the same community have identical factor loadings (*34*) and the more complex mixtures of factor analyzers (*30*): We define a community of time series as having factor loadings drawn from a common latent distribution. Each time series i belongs to exactly one community $g_i \in \{1, \ldots, K\}$, i.e., $\mathbf{g}$ is the vector of community memberships and K is the number of communities. The factor loadings are drawn from a multivariate normal distribution conditional on the community membership of each time series such that

$$A_i \sim \sum_{k=1}^{K} z_{ik} \text{Normal}\left(\mu_k, \Lambda_k^{-1}\right) \quad (2)$$

$$\text{where } z_{ik} = \begin{cases} 1 & \text{if } g_i = k \\ 0 & \text{otherwise} \end{cases}$$

The parameters μ_k and Λ_k are the p-dimensional mean and precision matrix of the kth component, respectively. The intuition behind the model is captured in Fig. 1A: We can identify communities because time series that behave similarly are close in the space spanned by the factor loading matrix. This idea relates to latent space models of networks in which nodes that are positioned closer together in the latent space have a higher probability of being linked (*35*). Extending the notion of communities to such a model implies clusters of nodes within the latent space (*36*).

The priors for the mean and precision parameters of the different communities require careful consideration because they can have a

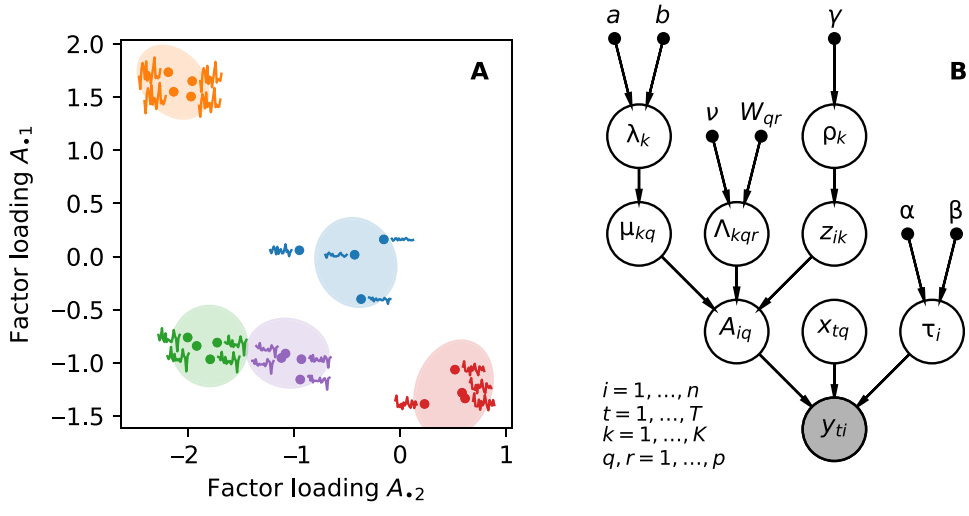

Fig. 1. A Bayesian hierarchical model for time series with community structure. Time series y are generated by a latent factor model with factor loadings A shown as dots in (**A**). The factor loadings are drawn from a Gaussian mixture model with mean μ and precision Λ. Generated time series are shown next to each factor loading for illustration. (**B**) Directed acyclic graph (DAG) representing the mixture model (A and all of its parents) and the probabilistic principal components analysis (PCA) (A, its siblings, and y). Observed nodes are shaded gray, and fixed hyperparameters are shown as black dots.

significant impact on the outcome of the inference (*37*): If the priors are too broad, then the model evidence is penalized heavily for each additional community, and all time series are assigned to a single community. If the priors are too narrow, then the inference will fail because it is dominated by our prior beliefs rather than being data driven. To minimize the sensitivity of our model to prior choices, we use an automatic relevance determination (ARD) prior, which can learn an appropriate scale for the centers of the communities μ (*38*). In particular

$$\mu_{kq} \sim \text{Normal}(0, \lambda_{kq}^{-1})$$

$$\lambda_{kq} \sim \text{Gamma}(a = 10^{-3}, b = 10^{-3})$$

Conjugate ARD priors are not available for the precision matrices of the communities, and we used Wishart priors such that

$$\Lambda_k \sim \text{Wishart}(\nu, W)$$

where $\nu > p - 1$ and $W \in \mathbb{R}^{p \times p}$ are the shape and scale parameters of the Wishart distribution, respectively. We set W to be a diagonal matrix that scales according to the number of latent factors such that $W = pwI_p$, where I_p is the p-dimensional identity matrix. To obtain a relatively broad prior (*39*), we let $\nu = p$ such that the prior precision, i.e., the expectation of the precision under the prior, is $\langle\Lambda\rangle = w^{-1}I_p$. We perform inference for a range of prior precisions because we cannot learn it automatically using an ARD prior.

Latent factor models as defined in Eq. 1 are not uniquely identifiable because we can obtain an equivalent solution by, for example, multiplying the factor loading matrix A by an arbitrary constant and dividing the latent factors x by the same value. We impose a zero-mean, unit-variance Gaussian prior on the latent factors to identify the scale of x and A (*40*). This approach does not identify the model with respect to rotations and reflections, but the lack of identifiability does not affect the detection of communities because the Gaussian mixture model defined in Eq. 2 is invariant to orthogonal transformations.

The community memberships follow a categorical distribution

$$g_i \sim \text{Categorical}(\boldsymbol{\rho})$$

where $\boldsymbol{\rho}$ represents the normalized sizes of communities such that $\sum_{k=1}^{K}\rho_k = 1$. To ensure that no community is favored a priori, we assign a symmetric Dirichlet prior

$$\boldsymbol{\rho} \sim \text{Dirichlet}(\gamma \mathbf{1}_K)$$

to the community sizes, where $\gamma = 10^{-3}$ is a uniform concentration parameter for all elements of the Dirichlet distribution and $\mathbf{1}_K$ is a K-dimensional vector with all elements equal to one. We use a broad gamma prior for the precision parameter of the idiosyncratic noise. In particular

$$\tau_i \sim \text{Gamma}(\alpha = 10^{-3}, \beta = 10^{-3})$$

Figure 1B shows a graphical representation of the model as a directed acyclic graph (DAG). Because the observations $\boldsymbol{y}$ only appear as leaf nodes of the DAG, any missing observations can be marginalized analytically.

Inference using the variational mean field approximation

Exact inference for the hierarchical model is intractable, and we use a variational mean field approximation of the posterior distribution to learn the parameters (*41*). The basic premise of variational inference is to approximate the posterior distribution $P(\Theta \mid y)$ by a simpler distribution $Q(\Theta)$, where Θ is the set of all parameters of the model. Variational inference algorithms seek the approximation $Q^*(\Theta)$ that minimizes the Kullback-Leibler divergence between the approximation and the true posterior. More formally

$$Q^*(\Theta) = \underset{Q \in \mathcal{Q}}{\text{argmin}}\, \text{KL}(Q(\Theta) \| P(\Theta \mid y))$$

where $\mathcal{Q}$ is the space of all approximations that we are willing to consider. Minimizing the Kullback-Leibler divergence is equivalent to maximizing the evidence lower bound (ELBO)

$$L(Q) = \langle \log P(y, \Theta) - \log Q(\Theta) \rangle \leq \log \int d\Theta\, P(y, \Theta) \qquad (3)$$

where $\langle\cdot\rangle$ denotes the expectation with respect to the approximate posterior Q and the right-hand side of Eq. 3 is the logarithm of the model evidence (*41*). The maximized ELBO (henceforth, just ELBO) serves as a proxy for the model evidence to perform model comparison, and we use it to determine the number of latent factors and the prior precision.

We further assume that the posterior approximation factorizes with respect to the nodes of the graphical model shown in Fig. 1A. More formally, we let $Q(\theta) = \prod_{\theta_i \in \Theta} Q_{\theta_i}(\theta_i)$, which restricts the function space $\mathcal{Q}$. Under this assumption, known as the mean field approximation, the individual factors can be optimized, in turn, until the ELBO converges to a (local) maximum. The general update equation is (up to an additive normalization constant)

$$\log Q_{\theta_i}(\theta_i) \rightarrow \langle \log P(\Theta \mid y) \rangle_{\setminus \theta_i}$$

where $\langle\cdot\rangle_{\setminus\theta_i}$ denotes the expectation with respect to all parameters except the parameter θ_i under consideration. See Blei *et al.* (*42*) for a recent review of variational Bayesian inference and appendix B for the update equations specific to our model.

RESULTS

Simulation study

Having developed an inference algorithm for the model, we would like to assess under which conditions the algorithm fails and succeeds. We start with a simple, illustrative example by drawing $K = 5$ community means μ from a two-dimensional normal distribution with zero mean and unit variance, i.e., we consider two latent time series and a two-dimensional space of factor loadings. The community precisions Λ are drawn from a Wishart distribution with shape parameter $\nu = 50$ and identity scale parameter. The communities are well separated because the within-community variability ($1/\sqrt{50} \approx 0.14$) is much smaller than the between-community variability (≈1), as shown in Fig. 2A. We assign $n = 50$ time series to the five communities using a uniform distribution of community sizes $\rho_k = 1/K$. Last, we draw $m = 100$ samples of the two-dimensional latent factors x and obtain the observations y using Eq. 1, i.e., by adding Gaussian observation noise with precision τ drawn from a Gamma(100,10) distribution to the linear transformation xA^T.

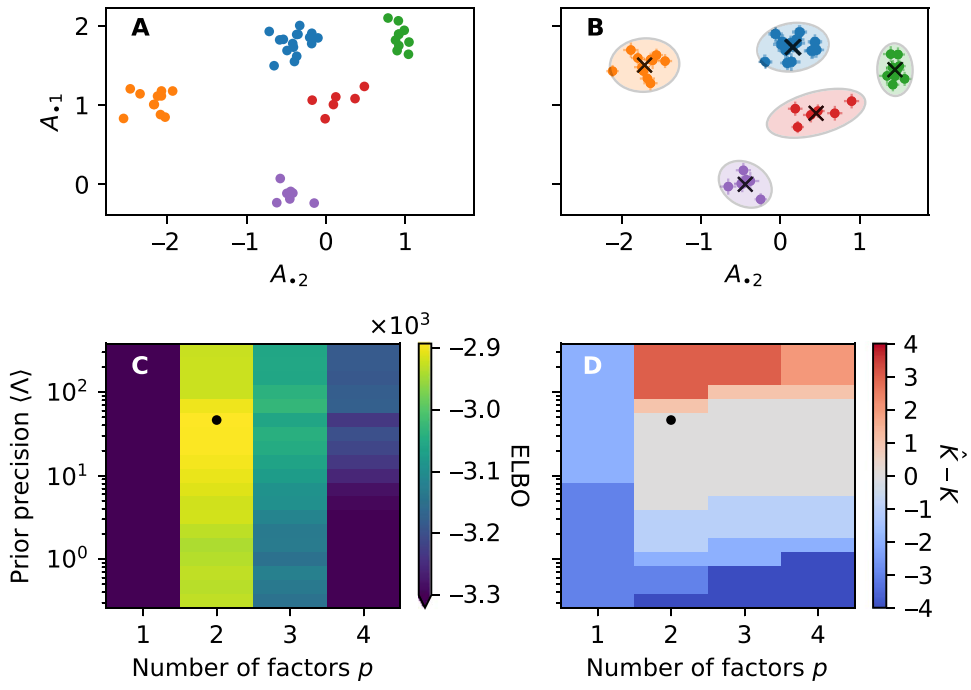

Fig. 2. The algorithm successfully identifies synthetic communities of time series. (**A**) Entries of a synthetic factor loading matrix A as a scatter plot. (**B**) Inferred factor loading matrix together with the community centers as black crosses and the community covariances as ellipses; error bars correspond to 3 SDs of the posterior. (**C**) ELBO as a function of the number of latent factors and the prior precision. (**D**) Difference between the estimated number of communities $\hat{K}$ and the true number of communities K. The model with the highest ELBO is marked with a black dot in (C) and (D); it recovers two latent factors and five communities.

Optimizing the ELBO is usually a nonconvex problem (*42*), and the results are sensitive to the initialization of the posterior factors. Choosing a good initialization is difficult in general, but the optimization can be aided to converge more quickly by initializing it using a simpler algorithm (*43*). We run the inference algorithm in three stages: First, we fit a standard probabilistic principal components analysis (PCA) (*44*) to initialize the latent factors, factor loadings, and noise precision. Second, we perform 10 independent runs of k-means clustering on the factor loading matrix (*45*) and update the community assignments z according to the result of the best run of the clustering algorithm, i.e., the clustering with the smallest sum of squared distances between the factor loadings A and the corresponding cluster centers μ. Third, we optimize the posterior factors of all parameters according to the variational update equations in appendix B until the ELBO does not increase by more than a factor of 10^{-6} in successive steps. The entire process is repeated 50 times, and we choose the model with the highest ELBO to mitigate the optimization algorithm getting stuck in local optima.

The number of communities and the prior precision are tightly coupled: Suppose that we choose a large prior precision for the Wishart distribution encoding a prior belief that each individual community occupies a small volume in the space of factor loadings. Consequently, the algorithm is incentivized to separate the time series into many small communities. In the limit $\langle\Lambda\rangle \to \infty$ (where vanishing within-community variation is permitted), the algorithm assigns each time series to its own community. In contrast, if we choose a small prior precision, then our initial belief is that each community occupies a large volume in the latent space, and time series are aggregated into few, large communities. Fortunately, the number of communities is determined automatically once the prior precision has been specified: In practice, we define the inferred cluster labels as

$$\hat{g}_i = \underset{k}{\operatorname{argmax}} \langle z_{ik} \rangle_{Q_z}$$

and determine the number of inferred communities $\hat{K}$ by counting the number of unique elements in $\hat{\mathbf{g}}$.

For the synthetic data discussed above, we set the maximum number of communities to 10 and run the inference for a varying number of latent factors and prior precisions. Increasing the maximum number of communities would not have any effect because the algorithm identifies, at most, eight communities. The ELBO of the best model for each parameter pair is shown in Fig. 2C. The model with the highest ELBO correctly identifies the number of factors and the number of communities; the inferred parameters are shown in Fig. 2B. As mentioned in the previous section, the model is not identifiable with respect to rotations and reflections, and consequently, the factor loadings in Fig. 2 (A and B) differ. However, the precise values do not affect the community assignments, and the difference is immaterial. Figure 2D shows the difference between the inferred and actual number of communities. As expected, choosing too small or large a prior precision leads to the algorithm inferring too few or too many communities, respectively.

Choosing the hyperparameters, such as the number of factors and the prior precision, to maximize the ELBO is known as empirical Bayes (*41*). In theory, it is preferable to introduce hyperpriors and treat the number of factors and the prior precision as proper model parameters similar to the ARD prior. However, dealing with the variable dimensionality of the latent space is difficult in practice, and computationally convenient conjugate priors for the scale parameter of Wishart distributions do not exist.

Multiscale community detection

Treating the dimensionality p of the latent space and the extent Λ of communities in the latent space as input parameters not only lets us avoid complicated inference but also provides us with a natural approach to multiscale community detection. We create nine communities arranged in a hierarchical fashion in the factor loading space similar to a truncated Sierpiński triangle and assign $n = 50$ time series to the communities, as shown in Fig. 3B. As in the previous section, we generate $T = 100$ observations of the time series with noise precision drawn from a Gamma(100,10) distribution.

In this example, we assume that the number of latent factors is known, set the maximum number of communities to 20, and vary the prior precision over several orders of magnitude. Figure 3A shows the ELBO as a function of the prior precision exhibiting two local maxima: The larger of the two corresponds to a large prior precision and identifies the nine communities used to generate the data, as shown in Fig. 3B. The smaller maximum occurs at a smaller prior precision, and the algorithm aggregates time series into mesoscopic communities, as shown in Fig. 3D. Decreasing the prior precision further forces the algorithm to assign all time series to a single community, and increasing the prior precision beyond its optimal value results in communities being fragmented into smaller components, as can be seen in Fig. 3C. Our algorithm not only is able to select an appropriate scale automatically but also allows the user to select a particular scale of interest if desired.

Testing the limits

In both of the examples we have considered so far, the communities were well separated from one another, which made it easier to assign time series to communities. Similarly, the number of observations T was twice as large as the number of time series n such that the algorithm could constrain the factor loading matrix well. In this section, we consider how the performance of the algorithm changes as we change the separation between communities and the number of observations. We define the community separation

$$h = \sqrt{\langle \Lambda \rangle \mathrm{var}[\mu]}$$

which measures the relative between-community and within-community scales such that communities are well separated in the factor loading space if $h \gg 1$ and are overlapping if $h \ll 1$. The expectation and variance in the definition of h are taken with respect to the generative model for the synthetic data.

For each combination of the number of observations and the separation h, we run 100 independent simulations with $K = 5$ communities, prior precision $\Lambda = 10I_2$ for each community, and $p = 2$ latent factors. For the inference, we assume that the number of latent factors is known and impose a limit of, at most, 10 communities. The prior precision is varied logarithmically from 0.625 to 20, and we retain the model with the highest ELBO. We use two criteria to measure the performance of the algorithm.

First, we measure the normalized mutual information (NMI) between the inferred community labels $\hat{\mathbf{g}}$ and the true community labels $\mathbf{g}$. The NMI is equal to one if the inferred and true community labels match exactly and is equal to zero if the community labels are independent. The NMI is defined as (*46*)

$$\mathrm{NMI}(\mathbf{g}, \hat{\mathbf{g}}) = \frac{I(\mathbf{g}, \hat{\mathbf{g}})}{\sqrt{H(\mathbf{g})H(\hat{\mathbf{g}})}}$$

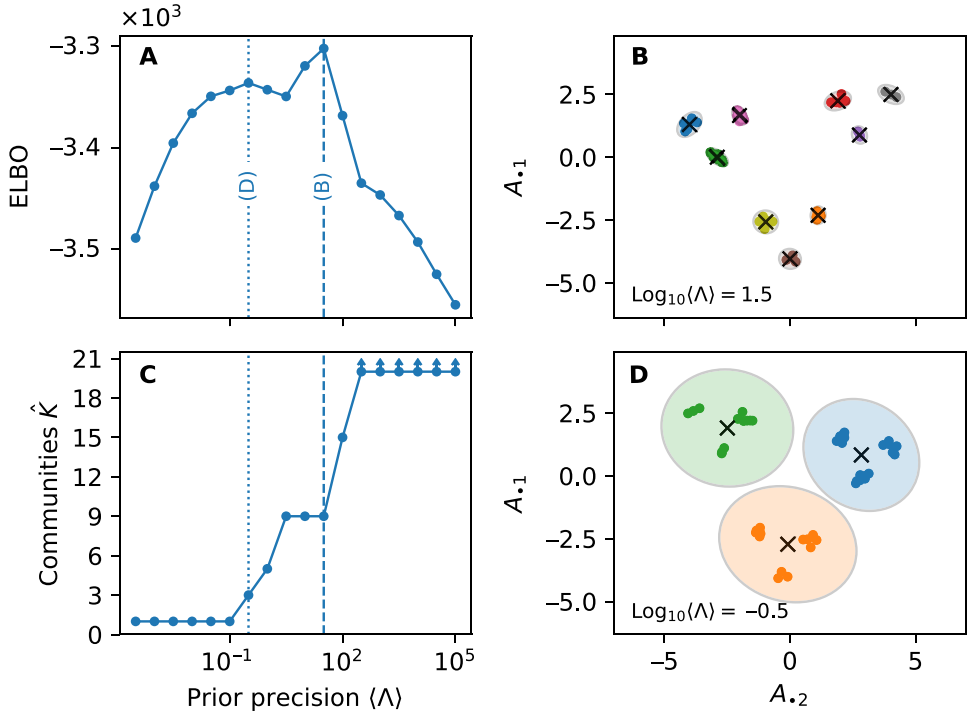

Fig. 3. The prior precision Λof the communities affects the number of detected communities. (**A**) ELBO of the model as a function of the prior expectation of the precision matrices $\langle \Lambda \rangle$. The ELBO has two distinct peaks corresponding to the community assignments shown in panels (**B**) and (**D**), respectively. (**C**) Number of identified communities as a function of the prior precision; data points with arrows represent a lower bound on the number of inferred communities.

where $I(\mathbf{g}, \hat{\mathbf{g}})$ is the mutual information between the true and inferred community assignments and $H(\mathbf{g})$ is the entropy of $\mathbf{g}$. The NMI displayed in Fig. 4A shows a clear and expected pattern: The larger the separation and the larger the number of observations, the better the inference. The separation poses a fundamental limit to how well we can infer the community labels. Even if we could estimate the factor loadings perfectly, we could not determine the community memberships if the communities are overlapping. This observation is analogous to the detectability limit for community detection on fully observed networks: The ability to recover community assignments diminishes as the difference of within-community and between-community connections decreases (*47*). However, provided that the communities are well separated, we can estimate the community labels well with a relatively small number of observations. We only require that the estimation errors of the factor loadings are small compared to the separation between communities. Of course, the community separation is not under our control, in practice, so we should ensure that we collect enough data to estimate the factor loadings well.

Second, we compare the inferred number of communities $\hat{K}$ with the true number of planted communities, as shown in Fig. 4B. When the communities are overlapping, the algorithm infers a smaller number of communities because aggregating time series into fewer communities with more constituents provides a more parsimonious explanation of the data. Similarly, when the number of observations is too small, the factor loadings are not estimated well, and the algorithm chooses fewer communities because the data do not provide sufficient evidence to split the set of time series into smaller communities.

To assess the effect of fitting a hierarchical Bayesian model compared with a simpler approach using point estimates at each stage of the process, we also infer community labels for each simulation as follows. First, we compute the correlation matrix and obtain an embedding for each time series by evaluating the two leading eigenvectors of the correlation matrix. Second, we apply k-means clustering with $K = 5$ clusters to the embeddings to recover community assignments. While the NMI, shown in Fig. 4C, displays a similar pattern to our hierarchical model, the difference between the NMIs of the two algorithms exhibits three types of behavior, as shown in Fig. 4D. When the separation between communities is small [labeled (i) in Fig. 4D], the hierarchical model has a lower NMI than the simpler model. The hierarchical model recovers fewer communities because there is not enough evidence to support multiple clusters, whereas the simpler model only performs better because it has access to additional information, the number of planted partitions. When we provide the hierarchical model with this additional information (see appendix C), the simpler model no longer outperforms the hierarchical one. For intermediate separation between clusters [labeled (ii)], the hierarchical model achieves a higher NMI because it does not discard information at each stage, especially when the number of observations is small. When the clusters are well separated [labeled (iii)], both approaches recover the communities well and there is little difference.

Application to financial time series

Having studied the behavior of the algorithm on synthetic data, we apply it to daily returns of constituents of the S&P100 index comprising 102 stocks of 100 large companies in the United States. Google and 21st Century Fox have two classes of shares, and we discard FOXA and GOOG in favor of FOX and GOOGL, respectively, because the latter have voting rights. We obtained 252 daily closing prices for all stocks from 4 January to 30 December 2016 from Yahoo! finance. Before feeding the data to our algorithm, we compute the daily logarithmic returns for each time series and standardize them by subtracting the mean and dividing by the SD.

In contrast to performing a grid search over the number of latent factors and the prior precision jointly as in the previous section for

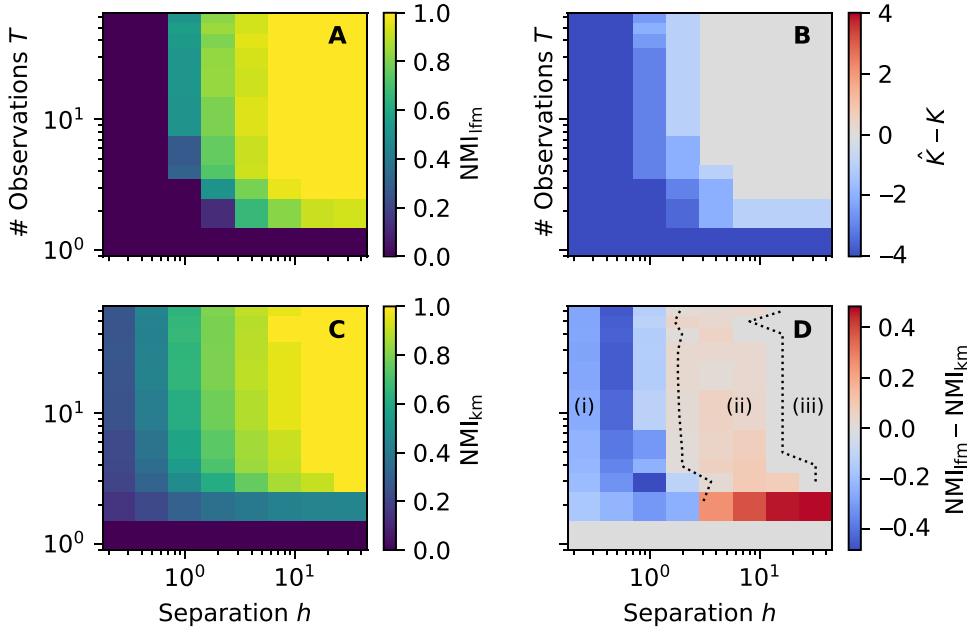

Fig. 4. Communities can be recovered even from very short time series. (**A**) Median NMI between the true and inferred community assignments obtained using our hierarchical model for $n = 100$ time series and $K = 5$ groups as a function of the number of observations T and the community separation h. (**B**) Median difference between the number of inferred communities and the true number of communities. (**C**) Median NMI obtained using PCA followed by k-means clustering. (**D**) Difference in NMI between the two algorithms [see the main text for description of regions (i) to (iii)].

the simulation study, we run the inference in two steps. First, we fit a standard probabilistic PCA model (*44*) and use the ELBO to choose the number of latent factors, as shown in Fig. 5A. Having identified the optimal number of factors as $\hat{p} = 10$, we perform a grid search over the prior precision to select an appropriate scale for the communities. The algorithm selects $\hat{K} = 11$ communities, as shown in Fig. 5 (B and C). Among an ensemble of 50 independently fitted models for each prior precision, the model with the highest ELBO tends to have the smallest number of communities: The algorithm tries to find a parsimonious description of the data, and representations with too many communities are penalized.

The factor loading matrix A has a nontrivial structure, as can be seen in Fig. 5D: The columns of the factor loading matrix are ordered descendingly according to the column-wise L_2 norm. The first column explains most of the variance of the data, and the corresponding factor is often referred to as the market mode, which captures the overall sentiment of investors (*8*, *48*). Additional factors capture ever more refined structure. Because visualizing the 10-dimensional factor loading matrix is difficult, we obtain a lower-dimensional embedding using t-distributed stochastic neighbor embedding (t-SNE) (*49*) shown in Fig. 5E. The shaded regions are the convex hulls of time series belonging to the same community.

The community assignments capture salient structure in the data. For example, the three smallest communities that have only two members consist of Mastercard (MA) and Visa (V), both credit card companies; Lockheed Martin (LMT) and Raytheon (RTN), both defense companies; and DuPont (DD) and Dow Chemical (DOW), both chemical companies. Dow Chemical and DuPont merged to form the conglomerate DowDuPont (DWDP) in August 2017. The algorithm also identifies a large community of companies from diverse industry sectors. More specialized communities consist of biotechnology and pharmaceutical companies [e.g., Merck (MRK) and Gilead Sciences (GILD)], financial services companies [e.g., Citigroup (C) and Goldman Sachs (GS)], and manufacturing and shipping companies [e.g., Boeing (BA), Caterpillar (CAT), FedEx (FDX), and United Parcel Service (UPS)].

Some of the community assignments appear to be less intuitive. For instance, the nuclear energy company Exelon (EXC) is assigned to a community of telecommunications companies rather than to a community of other energy companies as we might expect. This result does not necessarily indicate an error in community assignment, as the "true" communities in real data are not known (*50*). See Table 1 for a full list of companies and community assignments.

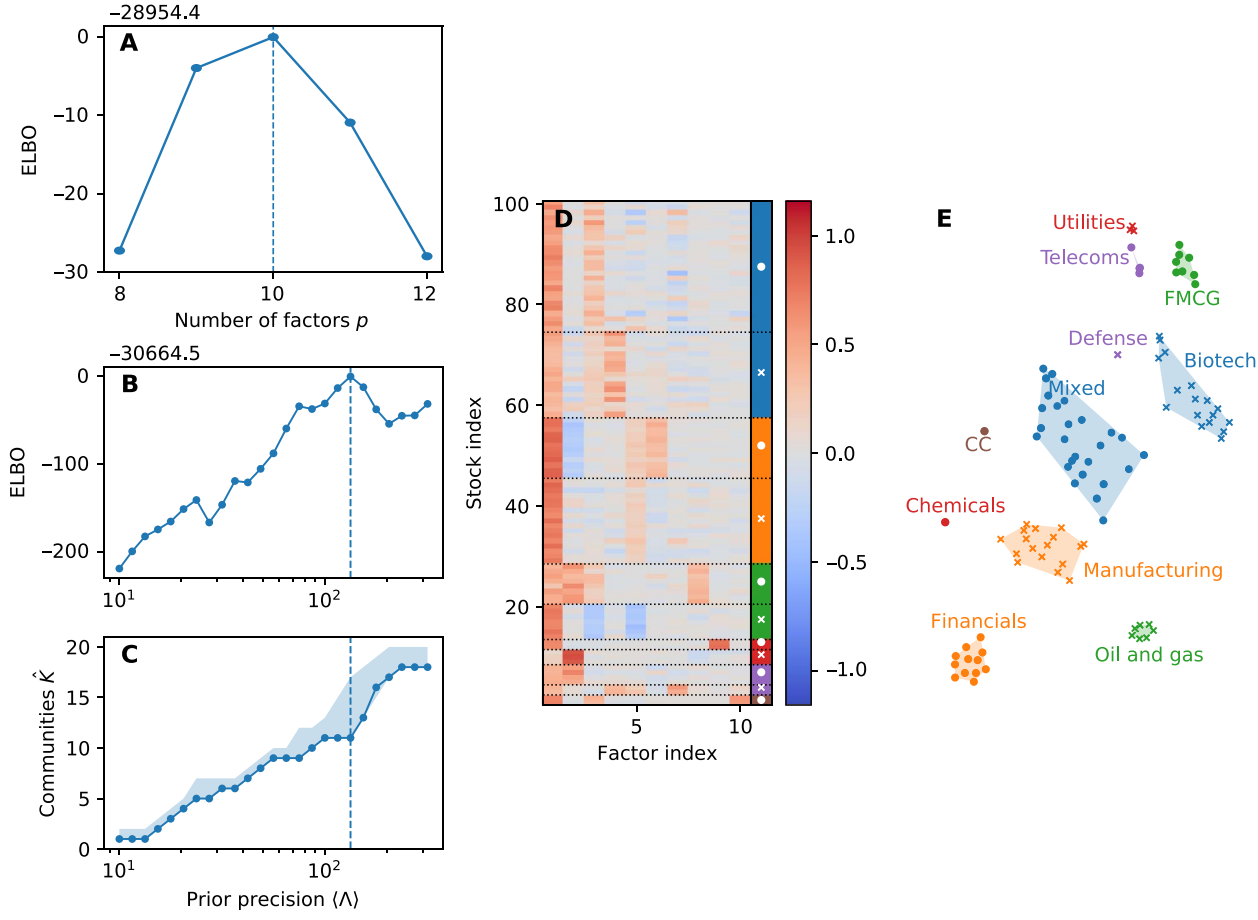

Fig. 5. The algorithm identifies 11 communities of stocks in a 10-dimensional factor loading space. (**A**) ELBO as a function of the number of latent factors of the model peaking at $p = 10$ factors. The ELBO of the best of an ensemble of 50 independently fitted models is shown in blue. (**B**) ELBO as a function of the prior precision. (**C**) Number of communities identified by the algorithm. The shaded region corresponds to the range of the number of detected communities in the model ensemble. (**D**) Factor loadings inferred from 1 year of daily log returns of constituents of the S&P100 index as a heat map. Each row corresponds to a stock, and each column corresponds to a factor. The last column of the loading matrix serves as a color key for different communities. (**E**) Two-dimensional embedding of the factor loading matrix using t-SNE together with cluster labels including credit card (CC) and fast-moving consumer goods (FMCG) companies.

Table 1. Constituents of the S&P100 grouped by inferred community assignment.

Group	Constituents
Mixed	Apple (AAPL), Abbott Laboratories (ABT), Accenture (ACN), Amazon (AMZN), American Express (AXP), Cisco (CSCO), Danaher (DHR), Walt Disney (DIS), Facebook (FB), 21st Century Fox (FOX), Google (GOOGL), Home Depot (HD), Intel (INTC), Lowe's (LOW), Medtronic (MDT), Monsanto (MON), Microsoft (MSFT), Nike (NKE), Oracle (ORCL), Priceline.com (PCLN), Paypal (PYPL), Qualcomm (QCOM), Starbucks (SBUX), Time Warner (TWX), Texas Instruments (TXN), Walgreen (WBA)
Biotech	AbbVie (ABBV), Actavis (AGN), Amgen (AMGN), Biogen (BIIB), Bristol-Myers Squibb (BMY), Celgene (CELG), Costco (COST), CVS (CVS), Gilead (GILD), Johnson & Johnson (JNJ), Eli Lilly (LLY), McDonald's (MCD), Merck (MRK), Pfizer (PFE), Target (TGT), UnitedHealth (UNH), Walmart (WMT)
Financials	American International Group (AIG), Bank of America (BAC), BNY Mellon (BK), BlackRock (BLK), Citigroup (C), Capital One (COF), Goldman Sachs (GS), JPMorgan Chase (JPM), MetLife (MET), Morgan Stanley (MS), US Bancorp (USB), Wells Fargo (WFC)
Manufacturing and shipping	Allstate (ALL), Barnes Group (B), Boeing (BA), Caterpillar (CAT), Comcast (CMCSA), Emerson Electric (EMR), Ford (F), FedEx (FDX), General Dynamics (GD), General Electric (GE), General Motors (GM), Honeywell (HON), International Business Machines (IBM), 3M (MMM), Union Pacific (UNP), United Parcel Service (UPS), United Technologies (UTX)
Fast-moving consumer goods	Colgate-Palmolive (CL), Kraft Heinz (KHC), Coca Cola (KO), Mondelez International (MDLZ), Altria (MO), PepsiCo (PEP), Procter & Gamble (PG), Philip Morris International (PM)
Oil and gas	ConocoPhillips (COP), Chevron (CVX), Halliburton (HAL), Kinder Morgan (KMI), Occidental Petroleum (OXY), Schlumberger (SLB), ExxonMobil (XOM)
Chemicals	DuPont (DD), Dow Chemical (DOW)
Utilities	Duke Energy (DUK), Nextera (NEE), Southern Company (SO)
Telecoms	Exelon (EXC), Simon Property Group (SPG), AT&T (T), Verizon (VZ)
Defense	Lockheed Martin (LMT), Raytheon (RTN)
Credit cards	MasterCard (MA), Visa (V)

Application to climate data

We now apply our method to climate data from 1429 U.S. cities. Each "node" represents a city, and the signals that we observe at each of the nodes are monthly values (averaged over 20 years) for the high and low temperatures and the amount of precipitation received, so instead of T observations of a time series, we have T attributes of the nodes, in this case $T = 36$ (three times 12 months). In this context, communities represent climate zones in which the temperature and precipitation vary similarly. In climatology, locales are classified into climate zones according to man-made climate classification schemes. One of the most popular climate classification schemes is the Köppen-Geiger climate classification system (*51*), first developed in 1884 by Wladimir Köppen (*52*) but has since received a number of modifications. The system divides climates into groups on the basis of seasonal temperature and precipitation patterns. Figure 6A shows the Köppen-Geiger classification of the U.S. cities we studied.

We infer the parameters of our model and community assignments using a similar approach to the previous section except for two notable differences. First, we found that the ELBO increased monotonically with increasing number of latent factors when fitting the standard probabilistic PCA, which is likely the result of the data having significant skewness of 0.70: The more complex the data, the more latent factors are required to fit the distribution. While the model is able to fit arbitrary data distributions by adding more latent factors, similar to a Gaussian mixture model (*53*), it may be advantageous to limit the number of factors for performance reasons. In this case, we decided to use six latent factors as the rate of increase of the ELBO drops when we increase the number of factors further. Second, instead of choosing the number of communities by maximizing the ELBO, we set the number of communities to the number of Köppen-Geiger climate zones to allow for a more direct comparison. Figure 6B shows the communities inferred by our model. Both sets of climate zones display similar qualitative features such as the division between the humid East and the arid West along the 100th meridian. However, a direct quantitative comparison of the two climate partitions is not necessarily meaningful, as we do not expect that there is only a single good way to partition the nodes. For reference, we find that the NMI between the two community assignments is ≈0.4. The low correlation between our inferred communities and the manually labeled Köppen-Geiger zones does not imply poor performance of our model (*50*), nor does it validate it.

Instead of trying to recover man-made labels, in the next section, we consider the predictive performance of our model on held-out, previously unseen data.

Imputing missing data

Often when dealing with real data, some values may be missing, e.g., because of measurement or human errors. We may also wish to artificially hold out a subset of values during the inference and attempt to impute these values to assess the goodness of fit of the model. Either way, imputing missing values consists of two steps. First, fit the model to the available data (all observed or not held-out values) using the inference procedure as described earlier. Second, use estimates of the factor loadings A and inferred latent time series x to impute the missing signal values as

$$\hat{y}_i = A_i^T x \tag{4}$$

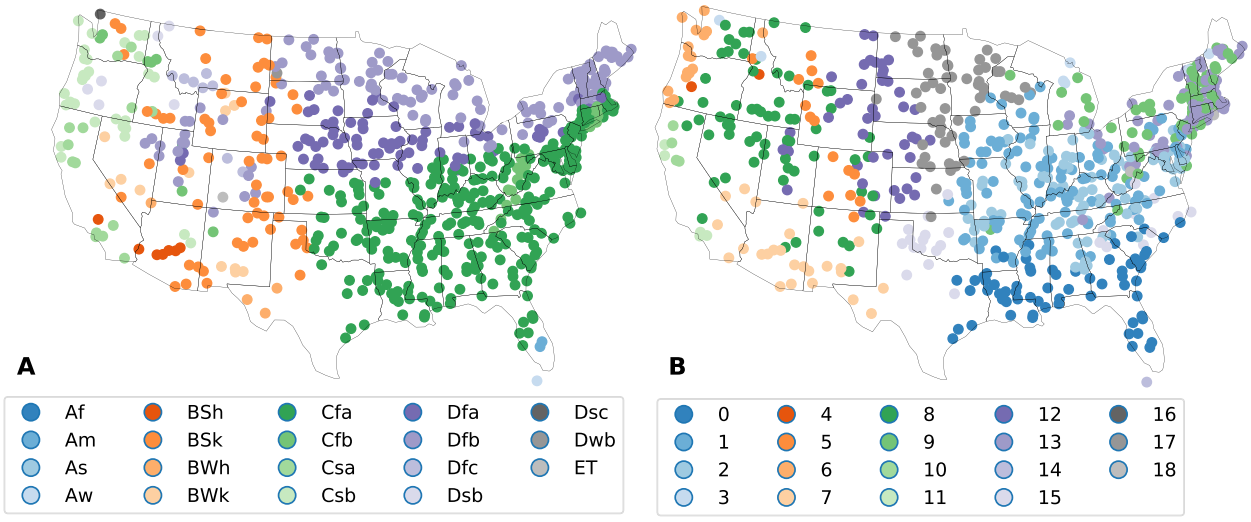

Fig. 6. Climate zones of U.S. cities. (**A**) City locations colored according to the Köppen-Geiger climate classification system (*51*). (**B**) Inferred climate zones based on the monthly average high and low temperatures and precipitation amounts. We observe qualitative similarities between the two sets of climate zones, but a quantitative comparison reveals a relatively low correlation (NMI ≈ 0.4).

We demonstrate the performance of imputing missing data on the financial and climate data in a cross-validation experiment. We first fit the model to the complete series of about half of the nodes (50 companies and 760 cities) selected uniformly at random, which acts as a training set to learn the latent factors x, community means μ, and community precisions Λ. Second, we perform a 10-fold cross validation on the remaining nodes by holding out a tenth of the data, inferring their factor loadings A and their community assignment, and predicting the missing signal values according to Eq. 4.

For comparison, we infer community assignments using a typical network-based method for clustering time series used by Fenn *et al.* (*2*). In particular, we apply the Louvain algorithm (*54*) with resolution parameter set such that we get approximately the same number of communities as the hierarchical model ($\gamma = 0.953$ for the financial time series and $\gamma = 0.95$ for the climate data) to the weighted adjacency matrix M

$$M_{ij} = \frac{\rho_{ij} + 1}{2} - \delta_{ij}$$

where ρ_{ij} is the Pearson correlation between series i and j and the Kronecker delta δ_{ij} removes self-edges (*2*). This approach does not make node-specific predictions but instead predicts the community mean. Therefore, to provide a more direct comparison with the communities found by our method, we also compare the predictions using the community means of the hierarchical model, i.e., $\hat{y}_i = \mu_{g_i}^T x$. For the climate data, we also impute the missing values using the mean value of each signal type, i.e., the mean temperature or precipitation for each month, within each Köppen-Geiger climate zone (*51*).

Table 2 shows the root mean square error for each approach on the financial and climate data, respectively. Our method outperforms the others in terms of predictive ability. While this observation provides some validation of our approach, it should not come as a surprise that our data-driven method, which is trained on the same type of data that we are trying to predict, outperforms the hand-crafted zones of Köppen and Geiger. However, the approach detecting communities using the method of Fenn *et al.* (*2*) performs worse than the Köppen-Geiger climate zones despite being trained on the same data: The method may identify spurious communities, at least with respect to those that have good predictive performance.

Note that because ground-truth communities are not available, we cannot determine which algorithm provides "better" community assignments (*50*), but we believe that the community assignments inferred by our algorithm are more intuitive than the community assignments inferred using the method of Fenn *et al.* (*2*) (e.g., for the financial data, compare the community assignments of our method shown in Table 1 with those of Fenn *et al.* shown in table S1).

DISCUSSION

We have developed a model for community detection for networks in which the edges are not observed directly. Using a series of interdependent signals observed for each of the nodes, our model detects communities using a combination of a latent factor model, which provides a lower-dimensional latent space embedding, and a Gaussian mixture model, which captures the community structure. We fit the model using a Bayesian variational mean field approximation, which allows us to determine the number of latent factors and an appropriate number of communities using the ELBO for model comparison. The method is able to recover meaningful communities from daily returns of constituents of the S&P100 index and climate data in U.S. cities. The code to run the inference is publicly available.

Our proposed method presents an important advancement over current methods for detecting communities without observing network edges. Recall that these methods typically consist of three steps: calculate pairwise similarity, threshold similarity to create a network, and apply community detection to the network. In contrast, our approach is end to end, i.e., the method propagates uncertainties from the raw data to the community labels instead of relying on a sequence of point estimates. As a result, the model is able to recover community structure even when the number of observations T is possibly much smaller than the number of n. Current methods for detecting communities when network edges are unobservable struggle in this setting because of the uncertainty in the estimate of the similarity matrix. The asymptotic

Table 2. Root mean square error predicting held-out values of the real-world datasets. The first column indicates the dataset. The second column displays the error of our method using the specific factor loadings of the time series. We include the third column to indicate the error of our method when we predict missing values according to the community means as a more direct comparison to the baseline methods of Köppen-Geiger and Fenn *et al.* (*2*). N/A, not applicable.

Dataset	Our method $A_i^T x$	Our method $\mu_{g_i}^T x$	Köppen-Geiger (*51*)	Fenn *et al.* (*2*)
S&P100	0.731	0.750	N/A	0.803
U.S. climate	0.301	0.578	0.706	0.727

complexity of algorithms that rely on pairwise similarities scales (at least) quadratically with the number of nodes, whereas each iteration of our algorithm scales linearly. We report the empirical run times of performing the inference on various synthetic networks in appendix E.

There are several avenues for future work. For example, using the same prior precision for all communities reflects our prior belief that all communities should occupy roughly similar volumes in the factor loading space. In analogy, in the case of standard community detection with modularity optimization, balanced sizes between communities are induced by the so-called diversity index in the quality function (*55*). Whether this assumption holds in practice is unclear, and we may be able to find communities of heterogeneous sizes in the factor loading space by lifting this assumption. Furthermore, Gaussian distributions are a standard choice for mixture models, but mixtures of other distributions such as Student's *t* distributions may provide better clustering results. Similarly, we modeled the community assignments as categorical variables such that each node belongs to exactly one community. Our approach could be extended to a mixed-membership model by allowing the community assignments to encode a weight of belonging to different communities (*56*).

Despite being motivated by time series, our algorithm does not model the dynamics of the data explicitly. Using a dynamical model such as a linear state space model may capture additional information in the data to help infer better community labels and allow us to predict future values of the time series.

As shown in the previous section, our algorithm can recover communities from observations of different attributes. While this use of the model violates the assumption that node observations are identically distributed, it does not prevent us from identifying meaningful communities. However, it may perform poorly in a posterior predictive check that compares statistics of the posterior distribution $P(y' \mid y)$ with the observed data. Promoting the observations y and factor loadings A to three-dimensional tensors would allow us to model different attributes in a principled fashion. In particular, the lth attribute of node i at time t would have distribution

$$y_{til} \mid A, x, \tau \sim \text{Normal}\left(\sum_{q=1}^{p} x_{tq} A_{ilq}, \tau_{il}^{-1}\right)$$

where A_{ilq} controls the effect of the qth latent factor on attribute l of node i. While increasing the number of independent observations T can only help us constrain the factor loadings A, collecting data about additional attributes provides fundamentally new information. Provided that the community assignments for the Gaussian mixture model are shared across the factor loadings of different attributes, we would be able to assign nodes to the correct community even if the components are not resolvable independently, i.e., $h \ll 1$, as discussed in the simulation study, similar to the enhanced detectability of fixed communities in temporal (*57*) and multilayer (*58*) networks.

Here, we have considered the setting in which the community structure of the network is assumed to be constant over time. Another avenue for future work may be adapting the model to investigate if and when changes occur in the underlying community structure (*59*).

Last, this work provides a new perspective on how to perform network-based measurements in empirical systems where edges are not observed. This opens the way to other end-to-end methods for, e.g., estimating centrality measures or motifs in complex dynamical systems.

SUPPLEMENTARY MATERIALS

Supplementary material for this article is available at http://advances.sciencemag.org/cgi/content/full/6/4/eaav1478/DC1

Appendix A. Exponential family distributions

Appendix B. Update rules for variational inference

Appendix C. Comparison against PCA and *k*-means with known *K*

Appendix D. Community assignments using the method of Fenn *et al.*

Appendix E. Empirical run times

Fig. S1. Recovering communities when the number of communities is known.

Fig. S2. Mean run time for inferring communities from synthetic data with varying number of latent factors *p*.

Fig. S3. Mean run time for inferring communities from synthetic data with varying number of observations *T*.

Fig. S4. Mean run time for inferring communities from synthetic data with varying number of network nodes *n*.

Table S1. Constituents of the S&P100 grouped by inferred community assignment using the Louvain algorithm applied to a correlation matrix.

REFERENCES AND NOTES

1. D. J. Fenn, M. A. Porter, M. McDonald, S. Williams, N. F. Johnson, N. S. Jones, Dynamic communities in multichannel data: An application to the foreign exchange market during the 2007–2008 credit crisis. *Chaos* **19**, 033119 (2009).
2. D. J. Fenn, M. A. Porter, P. J. Mucha, M. McDonald, S. Williams, N. F. Johnson, N. S. Jones, Dynamical clustering of exchange rates. *Quant. Finance* **12**, 1493–1520 (2012).
3. M. Bazzi, M. A. Porter, S. Williams, M. McDonald, D. J. Fenn, S. D. Howison, Community detection in temporal multilayer networks, with an application to correlation networks. *Multiscale Model. Simul.* **14**, 1–41 (2016).
4. T. Ando, J. Bai, Clustering huge number of financial time series: A panel data approach with high-dimensional predictors and factor structures. *J. Am. Stat. Assoc.* **112**, 1182–1198 (2017).
5. D. Meunier, R. Lambiotte, A. Fornito, K. D. Ersche, E. T. Bullmore, Hierarchical modularity in human brain functional networks. *Front. Neuroinform.* **3**, 37 (2009).
6. L.-D. Lord, P. Allen, P. Expert, O. Howes, M. Broome, R. Lambiotte, P. Fusar-Poli, I. Valli, P. McGuire, F. E. Turkheimer, Functional brain networks before the onset of psychosis: A prospective fMRI study with graph theoretical analysis. *Neuroimage Clin.* **1**, 91–98 (2012).
7. A. Tantet, H. A. Dijkstra, An interaction network perspective on the relation between patterns of sea surface temperature variability and global mean surface temperature. *Earth Syst. Dynam.* **5**, 1–14 (2014).

8. M. MacMahon, D. Garlaschelli, Community detection for correlation matrices. *Phys. Rev. X* **5**, 021006 (2015).
9. M. Y. Chan, D. C. Park, N. K. Savalia, S. E. Petersen, G. S. Wig, Decreased segregation of brain systems across the healthy adult lifespan. *Proc. Natl. Acad. Sci. U.S.A.* **111**, E4997–E5006 (2014).
10. S. Wu, M. Tuo, D. Xiong, Community structure detection of shanghai stock market based on complex networks, in *LISS 2014: Proceedings of 4th International Conference on Logistics, Informatics and Service Science* (Springer, 2015), pp. 1661–1666.
11. A. S. Pandit, P. Expert, R. Lambiotte, V. Bonnelle, R. Leech, F. E. Turkheimer, D. J. Sharp, Traumatic brain injury impairs small-world topology. *Neurology* **80**, 1826–1833 (2013).
12. T. Yu, Y. Bai, Network-based modular latent structure analysis. *BMC Bioinformatics* **15**, S6 (2014).
13. J. F. Donges, Y. Zou, N. Marwan, J. Kurths, The backbone of the climate network. *Europhys. Lett.* **87**, 48007 (2009).
14. A. Alexander-Bloch, R. Lambiotte, B. Roberts, J. Giedd, N. Gogtay, E. Bullmore, The discovery of population differences in network community structure: New methods and applications to brain functional networks in schizophrenia. *Neuroimage* **59**, 3889–3900 (2012).
15. R. F. Betzel, T. D. Satterthwaite, J. I. Gold, D. S. Bassett, A positive mood, a flexible brain. arXiv:1601.07881 [q-bio.NC] (2016).
16. M. Rosvall, C. T. Bergstrom, Maps of random walks on complex networks reveal community structure. *Proc. Natl. Acad. Sci. U.S.A.* **105**, 1118–1123 (2008).
17. S. Fortunato, Community detection in graphs. *Phys. Rep.* **486**, 75–174 (2010).
18. M. E. J. Newman, M. Girvan, Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113 (2004).
19. T. Cai, W. Liu, X. Luo, A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *J. Am. Stat. Assoc.* **106**, 594–607 (2011).
20. M. E. J. Newman, Network structure from rich but noisy data. arXiv:1703.07376 [cs.SI] (2017).
21. P. Latouche, E. Birmelé, C. Ambroise, Variational Bayesian inference and complexity control for stochastic block models. *Stat. Modelling* **12**, 93–115 (2012).
22. J. Reichardt, S. Bornholdt, Statistical mechanics of community detection. *Phys. Rev. E* **74**, 016110 (2006).
23. S. Aghabozorgi, A. S. Shirkhorshidi, T. Y. Wah, Time-series clustering—A decade review. *Inf. Syst.* **53**, 16–38 (2015).
24. J. Lin, M. Vlachos, E. Keogh, D. Gunopulos, Iterative incremental clustering of time series, in *International Conference on Extending Database Technology* (Springer, 2004), pp. 106–122.
25. E. J. Keogh, M. J. Pazzani, An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback, in *KDD'98 Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining* (AAAI, 1998), pp. 239–243.
26. E. F. Fama, K. R. French, Common risk factors in the returns on stocks and bonds. *J. Financ. Econ.* **33**, 3–56 (1993).
27. M. Nandha, R. Faff, Does oil move equity prices? A global view. *Energy Econ.* **30**, 986–997 (2008).
28. Z. Ghahramani, G. E. Hinton, "The em algorithm for mixtures of factor analyzers" (Technical Report CRG-TR-96-1, University of Toronto, 1996).
29. Z. Ghahramani, M. J. Beal, Variational inference for Bayesian mixtures of factor analysers, in *Advances in Neural Information Processing Systems* (MIT Press, 2000), vol. 12, pp. 449–455.
30. M. E. Tipping, C. M. Bishop, Mixtures of probabilistic principal component analyzers. *Neural Comput.* **11**, 443–482 (1999).
31. J. Taghia, S. Ryali, T. Chen, K. Supekar, W. Cai, V. Menon, Bayesian switching factor analysis for estimating time-varying functional connectivity in fMRI. *Neuroimage* **155**, 271–290 (2017).
32. I. Huopaniemi, T. Suvitaival, J. Nikkilä, M. Orešič, S. Kaski, Two-way analysis of high-dimensional collinear data. *Data Min. Knowl. Discov.* **19**, 261–276 (2009).
33. S. Zhao, C. Gao, S. Mukherjee, B. E. Engelhardt, Bayesian group factor analysis with structured sparsity. *J. Mach. Learn. Res.* **17**, 1–47 (2016).
34. L. Y. T. Inoue, M. Neira, C. Nelson, M. Gleave, R. Etzioni, Cluster-based network model for time-course gene expression data. *Biostatistics* **8**, 507–525 (2007).
35. P. D. Hoff, A. E. Raftery, M. S. Handcock, Latent space approaches to social network analysis. *J. Am. Stat. Assoc.* **97**, 1090–1098 (2002).
36. M. S. Handcock, A. E. Raftery, J. M. Tantrum, Model-based clustering for social networks. *J. R. Stat. Soc. A* **170**, 301–354 (2007).
37. R. E. Kass, A. E. Raftery, Bayes factors. *J. Am. Stat. Assoc.* **90**, 773–795 (1995).
38. J. Drugowitsch, Variational bayesian inference for linear and logistic regression. arXiv:1310.5438 [stat.ML] (2013).
39. I. Alvarez, J. Niemi, M. Simpson, Bayesian inference for a covariance matrix, in *Annual Conference on Applied Statistics in Agriculture* (New Prairie Press, 2014), vol. 26, pp. 71–82.
40. J. Luttinen, Fast variational Bayesian linear state-space model, in *European Conference on Machine Learning and Knowledge Discovery in Databases* (Springer, 2013), vol. 8188, pp. 305–320.
41. C. M. Bishop, *Pattern Recognition and Machine Learning* (Springer, 2007).
42. D. M. Blei, A. Kucukelbir, J. D. McAuliffe, Variational inference: A review for statisticians. *J. Am. Stat. Assoc.* **112**, 859–877 (2017).
43. M. Salter-Townshend, T. B. Murphy, Variational Bayesian inference for the latent position cluster model for network data. *Comput. Stat. Data Anal.* **57**, 661–671 (2013).
44. M. E. Tipping, C. M. Bishop, Probabilistic principal component analysis. *J. R. Stat. Soc. B* **61**, 611–622 (1999).
45. D. Arthur, S. Vassilvitskii, K-means++: The advantages of careful seeding, in *SODA '07 Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (Society for Industrial and Applied Mathematics, 2007), pp. 1027–1035.
46. A. Strehl, J. Ghosh, Cluster ensembles: A knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.* **3**, 583–617 (2003).
47. A. Decelle, F. Krzakala, C. Moore, L. Zdeborová, Inference and phase transitions in the detection of modules in sparse networks. *Phys. Rev. Lett.* **107**, 065701 (2011).
48. D. J. Fenn, M. A. Porter, S. Williams, M. McDonald, N. F. Johnson, N. S. Jones, Temporal evolution of financial-market correlations. *Phys. Rev. E* **84**, 026109 (2011).
49. L. van der Maaten, G. Hinton, Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
50. L. Peel, D. B. Larremore, A. Clauset, The ground truth about metadata and community detection in networks. *Sci. Adv.* **3**, e1602548 (2017).
51. M. Kottek, J. Grieser, C. Beck, B. Rudolf, F. Rubel, World map of the Köppen-Geiger climate classification updated. *Meteorol. Z.* **15**, 259–263 (2006).
52. W. Köppen, Die wärmezonen der erde, nach der dauer der heissen, gemässigten und kalten zeit und nach der wirkung der wärme auf die organische welt betrachtet. *Meteorol. Z.* **1**, 5 (1884).
53. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, 2016).
54. V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks. *J. Stat. Mech.* **2008**, P10008 (2008).
55. J.-C. Delvenne, S. N. Yaliraki, M. Barahona, Stability of graph communities across time scales. *Proc. Natl. Acad. Sci. U.S.A.* **107**, 12755–12760 (2010).
56. E. M. Airoldi, D. M. Blei, S. E. Fienberg, E. P. Xing, Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.* **9**, 1981–2014 (2008).
57. A. Ghasemian, P. Zhang, A. Clauset, C. Moore, L. Peel, Detectability thresholds and optimal algorithms for community structure in dynamic networks. *Phys. Rev. X* **6**, 031005 (2016).
58. D. Taylor, S. Shai, N. Stanley, P. J. Mucha, Enhanced detectability of community structure in multilayer networks through layer aggregation. *Phys. Rev. Lett.* **116**, 228301 (2016).
59. L. Peel, A. Clauset, *Detecting Change Points in the Large-Scale Structure of Evolving Networks* (AAAI, 2015), vol. 15, pp. 1–11.

Acknowledgments

Funding: This work was supported, in part, by EPSRC (UK) grant no. EP/I005986/1 (T.H. and N.S.J.), EPSRC (UK) grant no. EP/N014529/1 (T.H. and N.S.J.), F.R.S-FNRS (BE) grant no. 1. B.336.18F (L.P.), and Concerted Research Action (ARC) program of the Federation Wallonia-Brussels (BE) grant no. ARC 14/19-060 (L.P.) **Author contributions:** All authors designed the study and wrote the manuscript. T.H. and L.P. analyzed the data. **Competing interests:** The authors declare that they have no competing interests. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are available for download via www.usclimatedata.com, https://finance.yahoo.com/, https://github.com/tillahoffmann/time_series/, and/or the Supplementary Materials. Additional data related to this paper may be requested from the authors. Code is also provided.

Submitted 18 August 2018
Accepted 20 November 2019
Published 24 January 2020
10.1126/sciadv.aav1478

Citation: T. Hoffmann, L. Peel, R. Lambiotte, N. S. Jones, Community detection in networks without observing edges. *Sci. Adv.* **6**, eaav1478 (2020).

Community detection in networks without observing edges

Till Hoffmann, Leto Peel, Renaud Lambiotte and Nick S. Jones

Sci Adv **6** (4), eaav1478.
DOI: 10.1126/sciadv.aav1478