Proceedings Paper:
Ye, C., Wilson, R. C. orcid.org/0000-0001-7265-3033 and Hancock, E. R. orcid.org/0000-0003-4496-2028 (2016) Analyzing graph time series using a generative model. In: Davis, L., Bimbo, A. Del and Lovell, B., (eds.) 2016 23rd International Conference on Pattern Recognition (ICPR). IEEE Computer Society , Los Alamitos, CA, USA , pp. 3338-3343.

https://doi.org/10.1109/ICPR.2016.7900149

Analyzing Graph Time Series Using a Generative Model

Cheng Ye, Richard C. Wilson and Edwin R. Hancock
Department of Computer Science, University of York, York, YO10 5GH, UK.

***Abstract*—In this paper, we present a novel method for constructing a generative model to analyze the structure of labeled data. Given a time-series of sample graphs, we aim to learn a so-called "supergraph" that best describes the underlying average connectivity structure present in the data. In this time-series the vertex set is fixed and labeled and the set of possible connections between vertices change with time. The supergraph represents these changes with a Gaussian probability distribution for the connection weights on each individual edge. This structure is fitted to the time-series data by minimizing a description length criterion, with the von Neumann entropy controlling the complexity of the fitted model structure and the Gaussian log-likelihood controlling the mean edge weights and variances. We further show this fitting process can be optimized by using a new fixed-point iteration scheme which locates the elements of the optimal weighted adjacency matrix of the supergraph. We show the iteration process is in fact governed by the partial derivative of the von Neumann entropy. In the experiments, the resulting generative model is shown to be an effective tool for analyzing the underlying connectivity structure of time-evolving networks in the financial domain, and in particular locating critical events and distinct time epochs in their evolution.**

I. Introduction

Time-evolving networks are important in several real-world domains including the analysis of citation networks, communications networks, neural networks and financial networks, to name just a few [1]. A time-evolving network gives rise to structural patterns that change over time [2]. In order to analyze such systems, efficient tools for understanding their time-dependent structure and function are required. In general, graph evolution can be approached from both macroscopic and microscopic directions. On the one hand, the macroscopic approach aims at studying how the global parameters of a dynamic graph evolve from one time step to another. This can be accomplished by directly employing a number of graph characterizations that are developed on static graphs to each time step, and then analyzing the time evolution of these characterizations. For instance, Ye et. al [3] present a novel method for characterizing the evolution of time-varying complex networks by adopting a thermodynamic representation of network structure computed from a polynomial (or algebraic) characterization of graph structure. This approach combines the theoretical tools developed for studying graph structure in the context of statistical mechanics of complex networks and clearly point the potentials of the current approach to study real-world time-varying networks. On the other hand, at the microscopic level, it is the birth-death dynamics of individual vertex or edge in the graph evolution that are under study. In the classical Barabási-Albert model [4], dynamic properties are ascribed to a preferential attachment mechanism for graph growth, i.e., the addition of new vertices that connect to the vertices already in the graph. In detail, the preferential attachment mechanism shows that the probability of an existing vertex that connects to the newly added one is proportional to its degree.

This paper focuses on the challenge of learning a generative model which best captures the underlying edge connectivity in a set of labeled graphs. Broadly speaking, there have been two main approaches to characterizing edge structure variations in graphs, namely a) graph spectral methods and b) probabilistic methods. Although the spectral approach is simple and effective in developing generative models based on the Laplacian eigenvectors, the method is limited by a lack of stability of the Laplacian spectrum under perturbations in graph structure [5]. The probabilistic approach, on the other hand, is potentially more robust, but requires accurate correspondence information to be inferred from the available graph structure [6]. It is important to stress that the graph under study is an ordered collection consisting of a vertex set, an edge set and a vertex label set which maps the vertices to a set of labels. The vertex labels of the graphs give important information on the vertex correspondences between data, which plays a vital role in problems such as graph matching. In our analysis, we focus on a simpler case where the vertex number and vertex label information do not change between sample graphs. In other words, we are dealing with graphs whose vertex correspondences are to hand, and particularly are concentrating on the edge patterns present in those graphs.

The outline of the paper is as follows. In Sec.II, we will give the detailed development of how to construct a generative model for a time-series of weighted and labeled graphs using an information theoretic framework. In Sec.III, we will investigate the practical utility of our proposed method in analyzing the time evolution of real-world dynamical systems. Finally, we conclude the paper with a summary of our contribution and suggestions for future work.

II. Generative Model Learning

To commence the development, we define some notations. Let $\mathbf{G} = \{\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_t, \cdots, \mathcal{G}_N\}$ represent the time-series graph dataset under study, and $\mathcal{G}_t$ is used to denote the t-th sample graph in the time-series. The generative model, or the supergraph, which we aim to learn from the sample data is denoted by $\tilde{\mathcal{G}} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$, with vertex set $\tilde{\mathcal{V}}$ and edge set $\tilde{\mathcal{E}}$.

We are dealing with labeled graphs. Each vertex in a network has a unique label. In our application involving the New York Stock Exchange data, there are stocks trading in the New York Stock Exchange market. The vertex indices are denoted by lower-case letters including u, v, a, b, α and β, and will interchange these vertex indices with the vertex labels.

Preprint submitted to 23rd International Conference on Pattern Recognition.
Received April 19, 2016.

We represent the connectivity structure of the sample graph $\mathcal{G}_t$ using a weighted adjacency matrix ${}^tW$ whose (u,v)-th entry ${}^tW_{uv}$ indicates the connectivity between vertices u and v in the graph, and clearly, we have ${}^tW_{uv} \in [0,1]$. Similarly, we use the matrix $\tilde{W}$ to represent the structure of the supergraph $\tilde{\mathcal{G}}$.

A. Probabilistic Framework

Having introduced the necessary formalism, we now proceed to develop the probabilistic framework for the generative model learning method. To commence, we require the posterior probability of the observed sample graphs given the structure of the generative model $p(\mathbf{G}|\hat{\mathcal{G}})$. Then, the problem of finding the optimal supergraph can be posed in terms of seeking the structure $\tilde{\mathcal{G}}$ that satisfies the condition

$$\tilde{\mathcal{G}} = \underset{\hat{\mathcal{G}}}{\operatorname{argmax}}\, p(\mathbf{G}|\hat{\mathcal{G}}).$$

We follow the standard approach to constructing the likelihood function, which has been previously used in [6][5]. This involves factorizing the likelihood function over the observed data graphs and making use of the assumption that each individual edge in the sample graphs is conditionally independent of the remainder, given the structure of the supergraph. As a result, we have

$$p(\mathbf{G}|\tilde{\mathcal{G}}) = \prod_t p(\mathcal{G}_t|\tilde{\mathcal{G}}) = \prod_t \prod_u \prod_v p({}^tW_{uv}|\tilde{W}_{uv}), \qquad (1)$$

where $t = 1, 2, \cdots, N$. Moreover, $p({}^tW_{uv}|\tilde{W}_{uv})$ is the probability that the connectivity between u and v in the sample graph $\mathcal{G}_t$ is equal to ${}^tW_{uv}$, given that the edge (u,v) in the supergraph $\tilde{\mathcal{G}}$ has connectivity $\tilde{W}_{uv}$.

To proceed, a model for the observation density $p({}^tW_{uv}|\tilde{W}_{uv})$ is required. Luo and Hancock [6] have shown that for unweighted graphs where the individual edge connectivity of both the supergraph $\tilde{A}_{uv}$ and the sample graph ${}^tA_{uv}$ is either 0 or 1, the probability distribution $p({}^tA_{uv}|\tilde{A}_{uv})$ can be modeled by a Bernoulli distribution. The idea behind this model is that the connectivity of a particular edge in the data graph is derived from that of the same edge in the supergraph through a Bernoulli distribution. In their work, such model has proved to be effective in inexact graph matching. Recently, Martin et al. [7] have used a similar model to develop methods for inferring structure for uncertain networks and particularly, for solving community detection problems.

Since we are dealing with graphs whose edge connectivity is weighted, i.e., takes on a value between 0 and 1. As a result the Bernoulli distribution is not appropriate to model the observation density in our analysis. To overcome this problem, we note that according to the central limit theorem, the distribution of the average of a large number of independent, identically distributed variables is approximately a Gaussian distribution, regardless of the underlying distribution. Therefore, to simplify matters, here we model the distribution $p({}^tW_{uv}|\tilde{W}_{uv})$ by adopting a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ of the connection weights whose mean is the weight for the edge (u,v) in the supergraph, i.e., $\mu = \tilde{W}_{uv}$ and whose variance is σ^2. It is worth noting that the choice of σ does not make a significant difference in our development (we will show this later). Even if we choose an inappropriate σ which makes the generated edge connectivity exceed the valid weight interval from 0 to 1, we could use data re-normalization techniques to guarantee that the connectivity is on a scale of 0 to 1.

Finally, with the observation density model to hand, we write

$$p({}^tW_{uv}|\tilde{W}_{uv}) = \frac{1}{\sqrt{2\pi}\sigma}\, e^{-({}^tW_{uv}-\tilde{W}_{uv})^2/2\sigma^2}.$$

With these ingredients, the likelihood function given in Eq. (1) then becomes

$$p(\mathbf{G}|\tilde{\mathcal{G}}) = \prod_t \prod_u \prod_v \frac{1}{\sqrt{2\pi}\sigma}\, e^{-({}^tW_{uv}-\tilde{W}_{uv})^2/2\sigma^2}. \qquad (2)$$

To optimize the supergraph $\tilde{\mathcal{G}}$, we maximize this likelihood function with respect to the elements of the weighted adjacency matrix $\tilde{W}_{uv}$. This can be accomplished by straightforwardly employing a maximum-likelihood estimation (MLE) method. However, this leads to the result that the generative structure of the graph time-series is simply the mean of the data graphs, i.e., the weighted adjacency matrix of the supergraph $\tilde{W}$ is obtained by taking the average of the data graph adjacency matrices ${}^tW_{uv}$. Clearly, such a structure does not capture sufficient structural properties of the observed data graphs and thus cannot represent the underlying connectivity structure of the time-series of graphs. Thus a more meaningful and effective method for estimating the generative model is required.

B. Minimum Description Length Coding

To locate the optimal supergraph. we adopt an information theoretic approach and use a two-part minimum description length (MDL) criterion. Underpinning MDL is the principle that the best hypothesis for a given set of data is the one that leads to the shortest code length of the observed data. To formalize this idea, we encode and transmit the data $\mathcal{G}_t$ together with the hypothesis $\tilde{\mathcal{G}}$, leading to a two-part message whose total length is given by

$$\mathcal{L}(\mathbf{G}, \tilde{\mathcal{G}}) = \mathcal{L}(\mathbf{G}|\tilde{\mathcal{G}}) + \mathcal{L}(\tilde{\mathcal{G}}),$$

where $\mathcal{L}(\mathbf{G}|\tilde{\mathcal{G}})$ is the code length of the data graphs given the supergraph and $\mathcal{L}(\tilde{\mathcal{G}})$ is the code length of the estimated supergraph. Determining the most likely supergraph structure can be viewed as seeking the one that minimizes the total code length of the likelihood function. To this end, we take into account the total code length and apply the MDL principle to the model, this allows us to construct a supergraph representation that trades off goodness-of-fit with the sample graphs against the complexity of the model.

To apply the two-part MDL principle, we commence by computing the code length of the data graphs given the supergraph. This can be achieved by simply using the average of the negative logarithm of the likelihood function given in Eq. (2), with the result that

$$\begin{aligned}\mathcal{L}(\mathbf{G}|\tilde{\mathcal{G}}) &= -\frac{1}{N}\ln p(\mathbf{G}|\tilde{\mathcal{G}}) \\ &= -\frac{1}{N}\sum_t\sum_u\sum_v\left\{\ln\frac{1}{\sqrt{2\pi}\sigma} - \frac{({}^tW_{uv}-\tilde{W}_{uv})^2}{2\sigma^2}\right\}, \qquad (3)\end{aligned}$$

where N is the length of the observed time-series data $\mathbf{G}$.

Preprint submitted to 23rd International Conference on Pattern Recognition.
Received April 19, 2016.

Next, we compute the code length of the supergraph structure. Traditionally, the complexity of a model is measured by counting the number of parameters in the model. However, this does not generalize well for graphs since the true graph complexity cannot be accurately reflected by information such as the numbers of vertices or edges in the graph. To overcome this problem, we adopt a more meaningful measure of graph complexity, namely the von Neumann entropy, to encode the complexity of the supergraph structure (see [8] and [9] for detailed information of this entropy). Then, we have the supergraph complexity code length as follows,

$$\mathcal{L}(\tilde{\mathcal{G}}) = 1 - \frac{1}{|\tilde{\mathcal{V}}|} - \frac{1}{|\tilde{\mathcal{V}}|^2} \sum_{(u,v)\in\tilde{\mathcal{E}}} \frac{\tilde{W}_{uv}}{w_u w_v}, \tag{4}$$

where $w_u = \sum_{(u,v)\in\tilde{\mathcal{E}}} \tilde{W}_{uv}$ is the weighted degree of vertex u, which is defined as the sum of the connectivity weights of the edges connected to u and w_v is similarly defined. In effect, the complexity of the supergraph depends on two factors. The first is the order of the supergraph, i.e., the number of the vertices while the second is based on the degree statistics of the vertices in the supergraph.

Finally, by adding together the two contributions to the total code length, the overall code length of the likelihood function is

$$\begin{aligned}\mathcal{L}(\mathbf{G},\tilde{\mathcal{G}}) &= -\frac{1}{N}\sum_t \sum_u \sum_v \left\{ \ln \frac{1}{\sqrt{2\pi}\sigma} - \frac{({}^tW_{uv} - \tilde{W}_{uv})^2}{2\sigma^2} \right\} \\ &\quad +1 - \frac{1}{|\tilde{\mathcal{V}}|} - \frac{1}{|\tilde{\mathcal{V}}|^2} \sum_{(u,v)\in\tilde{\mathcal{E}}} \frac{\tilde{W}_{uv}}{w_u w_v}. \end{aligned} \tag{5}$$

To recover the supergraph we must optimize the above code length criterion with respect to the weighted adjacency matrix $\tilde{W}$. This can be done in a number of ways. These include gradient descent and soft assign [10]. However here we use a simple fixed-point iteration scheme. To proceed with the development of a useful optimization scheme we must compute the partial derivative of the code length criterion $\mathcal{L}(\mathbf{G}|\tilde{\mathcal{G}})$ given in Eq. (3) with respect to the elements of the weighted adjacency matrix $\tilde{W}_{ab}$. After some analysis the required derivative is

$$\begin{aligned}\frac{\partial \mathcal{L}(\mathbf{G},\tilde{\mathcal{G}})}{\partial \tilde{W}_{ab}} &= \frac{1}{N\sigma^2}\sum_t (\tilde{W}_{ab} - {}^tW_{ab}) - \frac{1}{|\tilde{\mathcal{V}}|^2}\left\{ \frac{1}{w_a w_b} \right. \\ &\quad \left. - \frac{1}{w_a^2} \sum_{(a,\beta)\in\tilde{\mathcal{E}}} \frac{\tilde{W}_{a\beta}}{w_\beta} - \frac{1}{w_b^2} \sum_{(\alpha,b)\in\tilde{\mathcal{E}}} \frac{\tilde{W}_{\alpha b}}{w_\alpha} \right\}. \end{aligned} \tag{6}$$

where β denote the neighbour vertices of a and α are the neighbours of b.

To set up our fixed-point iteration scheme, we set the above derivative to zero, and re-organize the resulting equation to obtain an update equation of the form $\tilde{W}_{ab}^{(n+1)} = g(\tilde{W}_{ab}^{(n)})$, where $g(\cdots)$ is the iteration function and n is iteration number. There is of course no unique way of doing this, and for convergence the iteration function $g(\tilde{W}_{ab})$ must have a derivative of magnitude less then unity at the fixed point corresponding to the required solution. One such scheme is

$$\begin{aligned}\tilde{W}_{ab}^{(n+1)} &= \frac{1}{N\sigma^2}\sum_t {}^tW_{ab} + \frac{1}{|\tilde{\mathcal{V}}|^2}\left\{ \frac{1}{w_a^{(n)} w_b^{(n)}} - \frac{1}{{w_a^{(n)}}^2} \right. \\ &\quad \left. \sum_{(a,\beta)\in\tilde{\mathcal{E}}^{(n)}} \frac{\tilde{W}_{a\beta}^{(n)}}{w_\beta^{(n)}} - \frac{1}{{w_b^{(n)}}^2} \sum_{(\alpha,b)\in\tilde{\mathcal{E}}^{(n)}} \frac{\tilde{W}_{\alpha b}^{(n)}}{w_\alpha^{(n)}} \right\}. \end{aligned} \tag{7}$$

The update process is governed by two terms. The first is computed from the local windowed mean of the time-series $\frac{1}{N\sigma^2}\sum_t {}^tW_{ab}$, while the second term is a step away from the local time-series mean determined by the partial derivative of the von Neumann entropy. This latter update term depends on the local pattern of vertex degrees. The convergence properties of the above fixed-point scheme are clearly critical. In general, a fixed-point iteration process $x_{n+1} = g(x_n), \quad n = 0, 1, 2, \cdots$ will converge to the fixed point $x^\star = g(x^\star)$ provided that $|g'(x^\star)| < 1$, and $g(x)$ has exactly one fixed point in $[a, b]$ and the sequence x_n is initialized with $x_0 \in [a, b]$. In our case, from Eq. (7), we have

$$\tilde{W}_{ab}^{(n+1)} = g(\tilde{W}_{ab}^{(n)}).$$

Computing the derivative of $g(\tilde{W}_{ab})$ gives

$$\begin{aligned} g'(\tilde{W}_{ab}) &= \frac{2}{|\tilde{\mathcal{V}}|^2}\left\{ \frac{\tilde{W}_{ab} - w_a - w_b}{w_a^2 w_b^2} + \frac{1}{w_a^3} \sum_{(a,\beta)\in\tilde{\mathcal{E}}} \frac{\tilde{W}_{a\beta}}{w_\beta} \right. \\ &\quad \left. + \frac{1}{w_b^3} \sum_{(\alpha,b)\in\tilde{\mathcal{E}}} \frac{\tilde{W}_{\alpha b}}{w_\alpha} \right\}. \end{aligned}$$

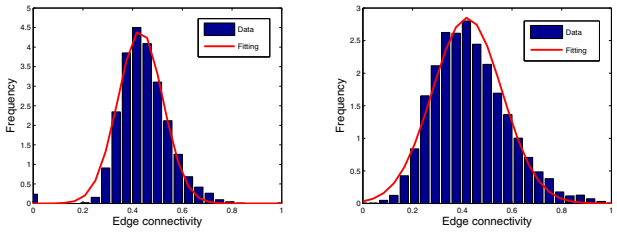
Since $\tilde{W}_{ab} \leq w_a$ and $\tilde{W}_{ab} \leq w_b$, and the vertex weighted degree is normally not small, the derivative satisfies the condition $g'(\tilde{W}_{ab}) \in [-1, 0]$ for all $\tilde{W}_{ab} \in [0, 1]$, and we are assured convergence of the fixed-point iteration scheme. Since the derivative of $g(\tilde{W}_{ab})$ is negative in sign, the convergence pattern is cobweb, i.e., from alternating sides of the fixed point.

There are a number of important points to note concerning the above analysis. First, our goal is to develop a generative model (or supergraph) that can be used to best explain the structural variations present in a time-series of graphs. To this end, we commence from a probabilistic framework which describes the likelihood of the observed data given the model structure. We then pose the problem of determining the optimal model structure as one of minimizing a code length criterion. To solve this problem numerically, we develop a simple fixed-point iteration scheme for optimizing the weighted adjacency matrix of the supergraph. The supergraph is initialized using the mean weighted adjacency matrix for the time-series, and then optimized at each time step to best fit the data in a time-window, the adjustment is determined by the von Neumann entropy.

III. Experiments

To evaluate the properties of the generative model and explore its practical utility on real-world data, in this section we report experimental results on time-evolving financial networks representing stock trading in the *NYSE Stock Market Network Dataset*. We first examine the validity of the proposed model learning method by exploring its convergence properties. Then, we compare the data graph structure with the supergraph learned from a time window of fixed length for a number of financial crisis time-series, and the result shows the supergraph is able to smooth the time-series data and more importantly, is more effective in locating critical events and distinct time epochs in financial crises. Also shown in the experiments is that by comparing the generative structure learned from different time-series data, we can better visualize and understand the structural difference of the stock market network in different time periods.

Preprint submitted to 23rd International Conference on Pattern Recognition.
Received April 19, 2016.



(a) $u = 147, v = 318$ (b) $u = 48, v = 289$

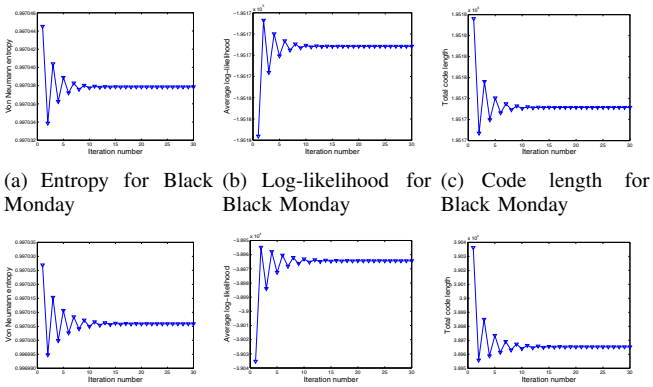
Fig. 1. Normalized histogram of the connectivity of two randomly selected edges (u, v) in the time-evolving stock correlation network and the Gaussian distribution fitting.

NYSE Stock Market Network Dataset. Is extracted from a database consisting of the daily prices of 3799 stocks traded on the New York Stock Exchange (NYSE). This data has been well analyzed in [11], which has provided an empirical investigation studying the role of communities in the structure of the inferred NYSE stock market. The authors have also defined a community-based model to represent the topological variations of the market during financial crises. Here we make use of a similar representation of the financial database, and employ the correlation-based network to represent the structure of the stock market since many meaningful economic insights can be extracted from the stock correlation matrices [12]. Particularly, to construct the dynamic network, 347 stocks that have historical data from January 1986 to February 2011 are selected [13][11]. Then, we use a time window of 28 days and move this window along time to obtain a sequence (from day 29 to day 6004) in which each temporal window contains a time-series of the daily return stock values over a 28-day period. We represent trades between different stocks as a network. For each time window, we compute the cross-correlation coefficients between the time-series for each pair of stocks, and create connections between them if the maximum absolute value of the correlation coefficient is among the highest 5% of the total cross correlation coefficients. This yields a time-varying stock market network with a fixed number of 347 vertices and varying edge structure for each of 5976 trading days.

A. *Convergence*

The first part of our experimental investigation aims to explore the convergence properties of the fixed-point iteration scheme. To this end, we first investigate whether the choice of Gaussian distribution adopted for modeling the observation density $p({}^tW_{uv}|\tilde{W}_{uv})$ is valid. Figure 1 gives the normalized histogram of the edge connectivity of two randomly selected edges in the time-evolving financial network and the probability density function curve of a Gaussian distribution whose mean is equal to the mean value of the corresponding edge connectivity and whose standard deviation is set to be a small value (0.1 here). Clearly, both plots show the normal "bell curve" or Gaussian distribution curve fits the data well, which means that using Gaussian distribution to model the probability of the observed data graph edge connectivity given the corresponding supergraph edge connectivity is effective.

To better visualize how the fixed-point iteration converges, we test the iteration process on a number of time-series graph



(a) Entropy for Black Monday (b) Log-likelihood for Black Monday (c) Code length for Black Monday

(d) Entropy for mini-crash (e) Log-likelihood for mini-crash (f) Code length for mini-crash

Fig. 2. Convergence properties of fixed-point iteration process as a function of iteration number for data graphs of different financial crises.

data of different financial crises, including the Black Monday, Friday the 13th mini-crash. For each financial crisis time-series, we initialize the supergraph $\tilde{W}_0$ with the mean data graph, which is obtained by taking the average of the structure of the sample graphs in the time-series. Then we process the iteration scheme $\tilde{W}_{ab}^{(n+1)} = g(\tilde{W}_{ab}^{(n)}),\ n = 0, 1, 2, \cdots$, which is given in Eq. (7) and observe how the complexity of the supergraph, the average log-likelihood of the sample graphs and the total code length of the likelihood function vary with the iteration number. Here, the standard deviation of the Gaussian distribution is set to be $\sigma = 0.1$, and at each iteration step, we perform a feature scaling normalization method to guarantee that the elements of the weighted adjacency matrix of the updated supergraph $\tilde{W}_{ab}^{(n)}$ is in the interval $[0, 1]$.

Figure 2 shows how various properties of the fixed-point iteration scheme for a number of financial crisis time-series data change during updating. From the plots in the first column, as the iteration processes, the supergraph entropy of both time-series fluctuates and gradually converges to a value that is lower than the initial entropy. This observation indicates that the structure of the generative model becomes less complex as the iteration scheme processes. The reason for this is that the supergraph is being optimized to best summarize the structural variations present in the time-series data during updating. Then, some negligible structural information contained in the data is discarded from the generative structure, which makes the supergraph less complex. Another interesting feature to note in the plots is that the entropy convergence of two time-series differs, which is a reasonable result as the supergraph is a structure representation that best explains the data graphs, then different financial crisis time-series yield different generative structure. In the second and third columns, the plots show the average log-likelihood function gradually increases as the increase of the number of iteration, while the total code length reduces. This is an expected observation since the goal of the proposed method is to maximize the probability of the observed data graphs given the model structure and also to minimize the total code length of the likelihood function.

Another interesting feature to note in the figure is that in both cases, the convergence of the fixed-point iteration process is oscillates from side-to-side of the fixed point. This cobweb

Preprint submitted to 23rd International Conference on Pattern Recognition.
Received April 19, 2016.

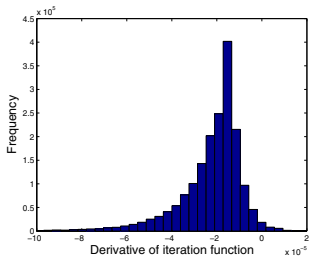
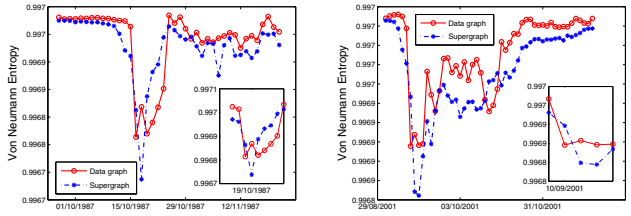


Fig. 3. Histogram of the derivative of the iteration function for both iteration processes.



(a) Black Monday (b) September 11 attacks

Fig. 4. Von Neumann entropy of supergraph and data graph as a function of time for different financial crises.

pattern is a consequence of the fact that $g'(\tilde{W}_{ab}) \in [-1, 0]$. To verify this, in Fig. 3 we show a histogram of the values of the derivative function $g'(\tilde{W}_{ab})$ for all the elements $\tilde{W}_{ab}$ obtained during the iteration processes. The result clearly conforms to our expectation as most derivative function values lie in the interval $[-1, 0]$. In particular, the values for the derivative function are close to zero, which means that the iteration process converges close to quadratically to the fixed point.

B. Event Detection

The second experimental goal is to explore whether the generative model can be used to provide a more efficient representation of the network structure in terms of capturing and detecting the structural changes during network evolution. To this end, for each time epoch t in the time-series data, we view the graphs from $t-N/2$ to $t+N/2$ as the observed sample graphs, where $N+1$ is the length of the time-series data. We then learn a generative model from this set of sample graphs, which can be viewed as a new network representation at t. We compare this new structure representation with the original data graph at each time epoch in the financial crisis time-series, by using the graph approximate von Neumann entropy.

In Fig. 4 we plot the approximate von Neumann entropy of both the data graph and the supergraph for Black Monday and the September 11 attacks time-series data. It is worth pointing out that, to construct the generative structure at each t, we use the sample graphs from $t-5$ to $t+5$ as the data. This is because two weeks is assumed to be a proper time length that can be used to effectively represent how long the effects of stock correlations last in the stock market. Moreover, the supergraph structure at each t is obtained after 15 iterations since we have observed from Fig. 2 that generally, the structure of the supergraph converges after 15 steps of updating.

From both plots, the von Neumann entropy of the supergraph generally follows the trend of that of the data graph in the time-series, implying that the supergraph is effective in capturing the graph structural properties in the data. More importantly, the generative model clearly gives a better graph structure representation than the sample graph since the supergraph entropy curve eliminates some of the random fluctuations observed in that of the sample graph. For example, from the inset plot in Fig. 4(a), at the day when Black Monday takes place, i.e., 19th October, 1987, the von Neumann entropy of the data graph clearly shows some unexpected fluctuations. However, the supergraph entropy reaches its minimum with a significant decrease, precisely representing that the stock correlation network experiences significant structural changes at that day. Overall, Fig. 4 shows that the supergraph we have learned using the generative model can be used as an efficient tool for summarizing a time-series of sample graphs and more importantly, for identifying significant structural changes during the network evolution.

C. Time Series Structure

From our development of the generative model, the supergraph represents a generative structure over a time-series of sample graphs. So it is expected that the graph whose structure significantly differs from that of the supergraph, can be viewed as a sample generated from the generative model with a relatively low probability. In contrast, graphs that have similar structure with the supergraph are the samples generated from the model with higher probabilities. In other words, the graphs corresponding to critical events and periods in the financial time-series should be highly dissimilar to the generative structure computed from the time-series data. To verify this expectation, in Fig. 5 we plot the Jensen-Shannon divergence kernel [14] between the sample graph at each time epoch and the supergraph for the entire time period in the financial data. Unlike the experiments shown above, here the supergraph is constructed over the whole collection of the financial data graphs.

The Jensen-Shannon divergence kernel is a non-extensive information theoretic kernel, which can be defined using the von Neumann entropy and mutual information computed from the structure being compared. In the work [14], the Jensen-Shannon divergence between graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ is expressed as

$$D_{JS}(\mathcal{G}_1, \mathcal{G}_2) = H_{VN}(\mathcal{G}_U) - \frac{H_{VN}(\mathcal{G}_1) + H_{VN}(\mathcal{G}_2)}{2},$$

where $\mathcal{G}_U$ denotes the disjoint union graph of $\mathcal{G}_1$ and $\mathcal{G}_2$, and H_{VN} is the von Neumann entropy. Then, the Jensen-Shannon divergence kernel is

$$k_{JSD}(\mathcal{G}_1, \mathcal{G}_2) = \exp\{-D_{JS}(\mathcal{G}_1, \mathcal{G}_2)\}.$$

It is interesting to note that the the Jensen-Shannon divergence kernel is dependent on the individual von Neumann entropies of graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ as the composite entropy $H_{VN}(\mathcal{G}_U)$ can be computed from $H_{VN}(\mathcal{G}_1)$ and $H_{VN}(\mathcal{G}_2)$.

The most important feature in the figure is that most of the significant troughs can be used to identify significant real-world financial crises. For instance, the Black Monday is the deepest one in the plot and the September 11 attacks also gives a sharp drop in the kernel curve. The reason for this

Preprint submitted to 23rd International Conference on Pattern Recognition.
Received April 19, 2016.

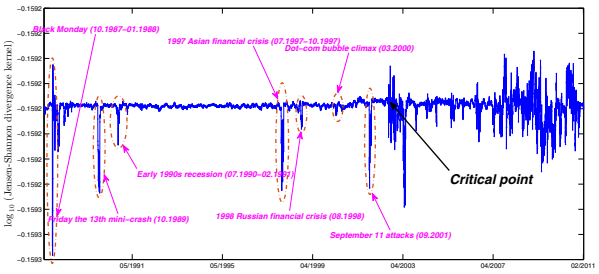


Fig. 5. Jensen-Shannon divergence kernel between sample graph and supergraph as a function of time for the time-evolving financial network.

is that during financial crisis, the stock correlation network undergoes significant structural changes, making the graph structure different from that of the generative model, which is a summary structure over the entire financial time-series. Such difference in graph topology can be efficiently captured by the shortest path kernel and the information theoretic kernel. Then, the similarity between the financial-crisis graph and the generative model becomes extremely low.

Another interesting observation to note in Fig. 5 is that the kernel measure exhibits very different patterns before and after a so-called "critical point" time epoch, which is around July and August, 2002. In particular, before the critical point, the kernel generally remains stable with time, except for a small number of fluctuations indicating the financial crises. This implies that the stock correlation network is able to return to its normal state in a short time after each financial crisis from 1987 to 2002. In contrast, after the critical point, the kernel becomes extremely unstable, which means that the network structure fluctuates significantly and becomes difficult to recover from the crises over the last decade. These observations can be mathematically verified by computing the modularity of the two generative structure computed from the time-series data before and after the critical point respectively, which are 0.3311 and 0.1860. In effect, the modularity is bounded between -1 and 1, and a network with a high modularity has modules in which the vertices are densely connected between each other but sparse connections between vertices in different modules. This shows that during 1987 to 2002, most stocks in the market have the tendency to form larger-sized groups in which the stocks are densely correlated internally. After 2002, stocks in the market are more likely to remove the connections with the stock groups, which may be related to the fact that such structure could lower the risk of stock price falling caused by the crash of the important stocks, i.e., core vertices in the network.

IV. Conclusion

To conclude, this paper is motivated by the need to develop efficient tools for analyzing time-evolving network data. To this end, we have suggested a novel method for learning a generative model from graph time-series in which the weights of the connectivity between vertices in the graph change with time while the vertex number and label do not. We concentrate on the edge patterns present in those graphs and represent the connectivity changes on each individual edge with a Gaussian probability distribution, which is characterized by the structure of the generative model (or supergraph). The paper then explores how this structure can be fitted to the graph time-series data using an information theoretic approach with an MDL criterion, whose model complexity is encoded by the graph von Neumann entropy. To solve the data-fitting problem, we present a new fixed-point iteration scheme which optimizes the structure of the generative model. In the experiments, we show our proposed method provides a number of new directions to the time-evolving network analysis.

There are a number of ways in which the work reported here can be extended. For example, it would be interesting to explore whether the method proposed for analyzing time-series data can contribute to the construction of dynamic network models. To do this, we could apply a Markov chain model to our generative model and investigate whether we could seek the evolutionary rules that govern the network dynamics.

References

[1] R. Albert and A. L. Barabási, "Topology of evolving networks: Local events and universality." *Physical Review Letters*, vol. 85, no. 24, pp. 5234–5237, 2000.

[2] P. Grindrod and D. J. Higham, "Evolving graphs: dynamical models, inverse problems and propagation." *Proceedings of the Royal Society A*, vol. 466, pp. 753–770, 2009.

[3] C. Ye, C. H. Comin, T. K. D. Peron, F. N. Silva, F. A. Rodrigues, L. d. F. Costa, A. Torsello, and E. R. Hancock, "Thermodynamic characterization of networks using graph polynomials." *Physical Review E*, vol. 92, no. 032810, 2015.

[4] A. Barabási and R. Albert, "Emergence of scaling in random networks." *Science*, vol. 286, pp. 509–512, 1999.

[5] L. Han, R. Wilson, and E. Hancock, "Generative graph prototypes from information theory." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 10, 2015.

[6] B. Luo and E. R. Hancock, "Structural graph matching using the em algorithm and singular value decomposition." *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, pp. 1120 – 1136, 2001.

[7] T. Martin, B. Ball, and M. Newman, "Structural inference for uncertain networks." *Physical Review E*, vol. 93, no. 012306, 2016.

[8] L. Han, F. Escolano, E. Hancock, and R. Wilson, "Graph characterizations from von neumann entropy." *Pattern Recognition Letters*, vol. 33, pp. 1958–1967, 2012.

[9] C. Ye, R. C. Wilson, C. H. Comin, L. d. F. Costa, and E. R. Hancock, "Approximate von neumann entropy for directed graphs." *Physical Review E*, vol. 89, no. 052804, 2014.

[10] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, p. 377388, 1996.

[11] F. N. Silva, C. H. Comin, T. K. D. Peron, F. A. Rodrigues, C. Ye, R. C. Wilson, E. R. Hancock, and L. d. F. Costa, "On the modular dynamics of financial market networks." *ArXiv e-prints*, no. 1501.05040, 2015.

[12] G. Bonanno, G. Caldarelli, F. Lillo, S. Micciché, N. Vandewalle, and R. N. Mantegna, "Networks of equities in financial markets." *European Physical Journal B*, vol. 38, pp. 363–372, 2004.

[13] T. K. D. Peron and F. A. Rodrigues, "Collective behavior in financial markets." *Europhysics Letters*, vol. 96, no. 48004, 2011.

[14] L. Bai and E. Hancock, "Graph kernels from the jensen-shannon divergence." *Journal of Mathematical Imaging and Vision*, vol. 47, pp. 60–69, 2013.

Preprint submitted to 23rd International Conference on Pattern Recognition.
Received April 19, 2016.