Journal of the Royal Statistical Society Series A: Statistics in Society, 2023, **186**, 508–529
https://doi.org/10.1093/jrsssa/qnad042
Advance access publication 17 April 2023
Original Article

ROYAL STATISTICAL SOCIETY DATA | EVIDENCE | DECISIONS
Journal of the Statistics Society
Series A
Statistics in Society

Dynamic latent space relational event model

I. Artico and E.C. Wit

Università della Svizzera italiana, Lugano, Switzerland

Address for correspondence: I. Artico, via Zurigo 30, Lugano, Switzerland. Email: igor.artico@usi.ch

Abstract

Dynamic relational processes, such as e-mail exchanges, bank loans, and scientific citations, are important examples of dynamic networks, in which the relational events constitute time-stamped edges. There are contexts where the network might be considered a reflection of underlying dynamics in some latent space, whereby nodes are associated with dynamic locations and their relative distances drive their interaction tendencies. As time passes, nodes can change their locations assuming new configurations, with different interaction patterns. The aim of this manuscript is to define a dynamic latent space relational event model. We then develop a computationally efficient method for inferring the locations of the nodes. We make use of the expectation maximization algorithm, which embeds an extension of the universal Kalman filter. Kalman filters are known for being effective tools in the context of tracking objects in the space, with successful applications in fields such as geolocalization. We extend its application to dynamic networks by filtering the signal from a sequence of adjacency matrices and recovering the hidden movements. Besides the latent space, our formulation includes also more traditional fixed and random effects, thereby achieving a general model that can suit a large variety of applications.

Keywords: dynamic interaction networks, EM, Kalman filter, latent space, patent citations, relational event model

1 Introduction

Networks appear in many contexts. Examples include gene regulatory networks (Signorelli et al., 2016), financial networks (Cook & Soramaki, 2014), psychopathological symptom networks (De Vos et al., 2017), political collaboration networks (Signorelli & Wit, 2018), and contagion networks (Užupytė & Wit, 2020). Studying networks is important for understanding complex relationships and interactions between the components of the system. The analysis can be difficult due to the many endogeneous and exogeneous factors that may play a role in the constitution of a network. The aim of statistical modelling in this context is to describe the underlying generative process in order to assist in identifying drivers of these complex interactions. These models can assist in learning certain features of the process, filtering noise from the data, thereby making interpretation possible.

In this manuscript, we are considering temporal random networks, whereby nodes make instantaneous time-stamped directed or undirected connections. Examples are email exchanges, bank loans, phone calls, article citations. A common approach to these networks has been flattening the time variable and studying the resulting static network. Although this method simplifies the complexity of the calculations, clearly there is a loss of information about the temporal structure of the process. Most networks are inherently dynamic. Subjects repeatedly create ties through time. Since the adjustment of ties is influenced by the existence and non-existence of other ties, the network is both the dependent and the explanatory variable in this process (Brandes et al., 2009). Thus, rather than viewing this as a static network, we consider the generative process as a network structure in which the actors interact with each other through the time. Edges are defined as instantaneous events. This quantitative framework is known as *relational event modelling*.

Received: December 23, 2021. **Revised:** March 16, 2023. **Accepted:** March 20, 2023
© (RSS) Royal Statistical Society 2023.
This is an Open Access article distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

The basic form of a relational event model (REM) as an event history model can be found in Butts (2008) with an application to the communications during the World Trade Center disaster. The model has been extended by Brandes et al. (2009) to weighted networks: nodes involved in these events are actors, such as countries, international organizations, or ethnic groups. An event is assigned a positive or negative weight depending on a cooperative or hostile type of interaction, respectively. Other examples of relational event modelling include the work by Vu et al. (2017) on interhospital patient transfers within a regional community of health care organizations or the analysis of social interaction between animals (Tranmer et al., 2015).

In a relational event model, the connectivity may depend on the past evolution of the network. Keeping track of the past is challenging for dynamic networks because of the high number of possible configurations (k-stars, k-triangles, etc.) that could be taken into account, as well as their closure time and the time they keep affecting future configurations. We thus propose to take some kind of summary of the past configurations. A solution that can both summarize the process and approximate effectively the past information is the idea of a dynamic latent space. To describe the latent structure of a network, one can think of placing the vertices in a space where the distance between two points describes the tendency or lack of tendency to connect. Among social scientists this is typically called a *social space* where actors with more interactions are close together and vice versa (Bourdieu, 1989). The locations are allowed to change in time. At each time point, new connections are formed and the subjects develop attraction/repulsion that force them to change their social space configuration. The new configuration is the one that best reflect the new connectivity behaviour. As a result, one location at a certain time reflects past information, within the limits of the latent space formulation. This evolution describes the social history of the subjects, their preferences, and the groups they might join or leave.

There are other temporal network models. The stochastic actor oriented model (Snijders & Pickup, 2017) defines relationships between social actors that can be created and destroyed. This model is very useful to model interactions that extend in time but are less suitable to model instantaneous interactions, such as communication, patent citations, or financial transactions. The temporal exponential random graph models (Hanneke et al., 2010) model sequences of networks. This approach is agnostic about the underlying generative process but typically would also focus on persistent network relations. Here we focus on instantaneous interactions, which makes the use of relational event models the method of choice.

1.1 Related work and novelty of the proposed method

The problem of tracking latent locations has been studied by many authors, specifically for the static case, i.e., tracking locations under the assumption that they are fixed over time. For static binary random graphs, Hoff et al. (2002) provide a framework for inference. Some extensions of that model have been developed to overcome the limitations of the latent space formulation (Hoff, 2005, 2008, 2009). The well-known stochastic block model describes the similarity between the actors by grouping them together, which is similar to latent space formulation. An extension of stochastic block modelling to relational event data is provided by DuBois et al. (2013).

An approach for modelling latent space dynamic binary networks was proposed by Sarkar and Moore (2005). The method is based on an initial preprocessing phase where rough location guesses are found through generalized multidimensional scaling, followed by an estimation phase in which the dynamic locations are treated as fixed parameters and optimized via a conjugate gradient method. The distances between nodes are approximated by thresholding larger ones and including an additional penalty for forcing distant nodes to be closer. In this work, we avoid making ad hoc inference assumptions.

Sewell and Chen (2015) propose a Bayesian latent space model for temporal binary networks where its radius interpretation of the linear predictor reduces to a Hoff et al. (2002) model with the addition of node specific random effects. The method employs a Metropolis-within-Gibbs approach, whose computational burden of MCMC integration increases exponentially with the latent dimension d, the number of nodes p, and the number of time points n. Although case-control sampling (Raftery et al., 2012) reduces the likelihood computation from $O(np^2)$ to $O(np)$, its accuracy depends on extensive stratification. By considering one control stratum, Sewell and Chen (2015) weigh heterogeneous distances in the same way, producing a bias. This leads to paradoxical

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

overlapping of unconnected nodes. Durante and Dunson (2016) developed a Bayesian approach using Polya-Gamma data augmentation for binary links and Gaussian processes for parameter dynamics combined with a non-Euclidean dissimilarity measure. In contrast to the previous two Bayesian approaches, we tackle the problem from a frequentist perspective, which does not require data augmentation. Our expectation-maximization (EM) algorithm combined with a Kalman filter is deterministic and does not suffer from Bayesian convergence issues. It scales linearly with the number of time periods and achieves a good latent representation after few iterations. It can scale to several hundred nodes without case-control subsampling. Moreover, whereas Durante and Dunson (2016) assume a discrete time sequence of binary adjacency matrices, we embed our discrete time observation process into a often more realistic continuous time relational event process. Furthermore, we explicitly consider the availability of covariates, which allow for further disentanglement of known drivers of the interaction dynamics from the unknown factors. Although non-Euclidean alternatives can easily be added, our implementation focuses on an easily interpretable Euclidean latent space.

1.2 The methodology presented

A dynamic latent space model is particularly useful in an exploratory stage of the analysis. It allows for an interactive investigation of the data to generate hypotheses about the drivers of the generative process by seeing which nodes are close and which nodes are far apart, as well as the way they develop through time. The most obvious example of this approach is simply by visualizing the development of the latent node locations in two dimensions. However, simple multivariate analysis tools, such as Principal Component Analysis, can also explore latent spaces with higher dimensions. If the aim of the analysis is entirely predictive, then the latent space model itself may be of interest as it can be used to generate predictions without knowing the underlying drivers of the process.

The aim of this manuscript is to develop an efficient inference scheme for a relational event process embedded in a latent Gaussian process. The framework is very general and can be extended to networks with weighted edges of any exponential family distribution. There are two dual representations of the process, either as a continuous time exponential or as discrete Poisson counts. Depending on the sparsity of the observed process, one or the other can be selected in the inference procedure. Furthermore, the theoretical burden of the expectation maximization framework in the model has been reduced to two analytical steps: for the E-step, a Kalman filter and smoother is used, whereas for the M-step, a generalized linear model framework is derived. Both are provided by modern packages. Our latent space relational event framework provides an accurate, simple, and computationally efficient way for inferring a wide general class of dynamic social network models.

Section 2 describes a motivating patent citation network example. In Section 3, several formulations of the latent space relational event model are presented. In Section 4, we propose an efficient inference method that is based on combining the state-space formulation of the model with the EM algorithm. In Section 5, we check the performance and limitations of our method via a simulation study. In Section 6, we analyze the latent structure of technological innovations, by studying over 23 million patent citations from 1967 until 2006.

2 Patent citation networks

Patents are legal documents of intellectual property that testify some technological innovation. Innovation itself is a complicated process and involves both true novelty as well as the adaptation of existing ideas in a new context. Within the patenting process, this borrowing of existing ideas are referred to as *patent citations*: each inventor that submits a patent to a patent office is required by law to include the current state-of-the-art on which the current patent is based by citing those patents in which those ideas have been deposited.

By tracing which patents cite which other patents, it is possible to establish a dynamic network in which patents accumulate over time citations from other patents. Alternatively, it is possible to group patents together into clusters and to track how these clusters cite and are cited by other clusters. Either way, the process of citation shows how certain patents at certain times are particularly important in the technological innovation process. As innovation is important for economical

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

progress and prosperity, it is little surprise that the analysis of the patent citation network has become an important field of study. It is of particular interest to find out what drives technological innovation (Lafond & Kim, 2019). Furthermore, economists are eager to find out whether or not the innovation process is changing over time.

The International Patent Classification (IPC) scheme is a hierarchical clustering scheme for patents. It assigns each patent to eight main classes, to with,

(A) : Human necessities: agriculture, foods, tobacco, personal, or domestic articles, health, life-saving, amusement.
(B) : Performing operations and transporting: separating, mixing, shaping, printing, transporting, nanotechnology.
(C) : Chemistry and metallurgy
(D) : Textiles; papers.
(E) : Fixed constructions: building, earth drilling.
(F) : Mechanical engineering; lightning; heating; weapons; blasting.
(G) : Physics: instruments, nuclear.
(H) : Electricity.

Within each main class, there are a large number of subclasses, resulting in overall roughly 500 subclasses. Each subclass has again a number of groups and subgroups, which for the purposes of the analysis here we will ignore. Also, other grouping schemes are possible (Younge & Kuhn, 2016).

The National Bureau of Economic Research in the U.S. released in 2010 patent citation data, consisting of 3.1 million patents, 23.6 million citations over the period 1967–2006, with collection intervals of 1 year length. By studying how citing behaviour and being cited tendency of the classes and the subclasses changes over time, we aim to answer some of the questions we posed above. The latent representation allows for a straightforward similarity assessment, showing which fields are becoming more heterogeneous in their citation patterns. The aim is to develop a methodological framework for inferring dynamic latent space tracking of the technology classes and to show how this changes the nature of patent citations.

3 Latent space relational event models

In this section, we introduce a general version of a latent space relational event model. We consider a set of actors, defined as a finite vertex set $V = \{1, \ldots, p\}$, that can exchange links or edges in time. In principle, we will consider the exchange of relational events, such as discrete interaction, e.g., sending an email or citing a patent, but we will also consider extensions to the quantitative exchanges, such as import and export. As drivers of the exchange process, we consider both endogeneous, such as reciprocity, and exogeneous variables, such as vertex characteristics. One particular exogeneous variable is the relative location of the vertices in some Euclidean latent space, which itself is defined as a dynamic process.

We consider a non-homogeneous multivariate Poisson counting process $N = \{N_{ij}(t) \mid i, j \in V, t \in [0, T]\}$ and a state-space process $X = \{X_i(t) \in \mathbb{R}^d \mid t \in [0, T], i = 1, \ldots, p\}$ relative to some standard filtration $\mathcal{F}$. In particular, we consider $\mathcal{F}$-measurable rate functions $\lambda_{ij}(t)$ that drive the components of the counting process. In particular, we assume that the rates $\lambda_{ij}(t)$ are functions of the underlying positions $X_i(t)$ and $X_j(t)$, besides possible other exogeneous characteristics $B_{ij}(t)$ and endogeneous features $N(t)$,

$$\lambda_{ij}(t) = g(d(X_i(t), X_j(t)), B_{ij}(t), N(t)),$$

for some measurable function g. Two common choices for the way that the rate depends on the locations is either as a function of the squared distance,

$$d(X_i(t), X_j(t)) = \|X_i(t) - X_j(t)\|^2$$

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

or the relative activity dissimilarity $d(X_i(t), X_j(t)) = \frac{\langle X_i(t), X_j(t)\rangle}{\|X_i(t)\|}$ between i and j (Hoff et al., 2002). The former induces a symmetric interpretation, where the latter allows for a more complex asymmetric interpretation of the state-space. In this manuscript, we mainly focus on the Euclidean distance, as we prioritize visual interpretation of the results. However, it is important to mention that switching to another dissimilarity measure requires very little effort. The interaction dynamics $\lambda_{ij}(t)$ can be highly structured and parametrized, i.e., $g = g_\theta$, whereas the state-space dynamics is assumed to be a random walk at equally spaced time points t_k^x in $[0, T]$,

$$X_{t_k^x} = X_{t_k^x} + \nu_k, \quad k = 1, \ldots, n_x, \tag{1}$$

with $\nu_k \sim N(0, \Sigma)$ and $t_0^x = 0$. In this manuscript, we use sometimes the more compact notation $x_k = X(t_k^x)$ or $X(k)$ when we find it more convenient. The covariance matrix Σ regulates the evolution of the latent process: a large variance allows longer jumps. Given the joint formulation (X, N) of the state-space and interaction process, we will assume that only the interaction process N is observed and the main aim of this manuscript is to infer the structure of the state-space X and the rate functions λ, or more specifically, the parameter β associated with functional form $\lambda = g_\beta$.

Next, we will consider two particular special cases of the latent space formulation of the interacting point process defined above. First we consider the general case, in which the relational events are observed in continuous time. This is the traditional setting for relational events. We will also define a relational event model where the interactions can only happen at specific times. For example, bibliometric citations or patent citations only happen at prespecified publication dates. Furthermore, this model allows a generalization to non-binary relational events, such as export between countries, that can be dealt with in the same inferential framework.

3.1 Continuous time relational event process *N*

We consider a sequence of n_e relational events, $\{(i_1, j_1, t_1^e), \ldots, (i_{n_e}, j_{n_e}, t_{n_e}^e) \mid t_i^e \in [0, T], i, j \in V\}$ observed according to the above defined relational counting process N. In a latent space relational event model, the rate is defined as

$$\log \lambda_{ij}(t, x, \beta) = -d(x_i(t), x_j(t)) + \beta_G^t B_{ij}(t) + \beta_D^t s(\{N(\tau) \mid \tau < t\}), \tag{2}$$

where the latent space effect $d(X_i(t), X_j(t))$ that captures the 'vicinity' of the actors. The drivers of the network dynamics can be of various types: *exogeneous effects*, $\beta_G^t B_{ij}(t)$, such as global covariates, node covariates, edge covariates, as well as *endogeneous effects*, $\beta_D^t s(\{N(\tau) \mid \tau < t\})$, where network statistics $s()$ capture endogeneous quantities such as popularity, reciprocity, and triadic closure. The parameter vector β determines the relative importance of the various effects.

Conditional on the process X, the distribution of the interarrival time for interaction $i \to j$ is generalized exponentials, with instantaneous rates as described in (2) and interval rates,

$$\mu_{k,ij}(x_k, \beta) = \int_{t_k^x}^{t_{k+1}^x} \lambda_{ij}(t, x, \beta)\, \mathrm{d}t = \mathrm{e}^{-d(x_i(t_k^x), x_j(t_k^x))} c_{ij}(k, \beta), \tag{3}$$

where $c_{ij}()$ is the remaining integral and latent distance $d()$ between the nodes is constant over the interval. The full log-likelihood of the complete process $\{X, N\}$, can be factorized into two components,

$$\ell(\beta, \Sigma) = \log p_\beta(n \mid x) + \log p_\Sigma(x), \tag{4}$$

where $\log p_\Sigma(x) = -\frac{n_x}{2} \log |\Sigma| - \frac{1}{2} \sum_{k=1}^{n_x} (x_k - x_{k-1})' \Sigma^{-1} (x_k - x_{k-1})$ and $\log p_\beta(n \mid x) = -\sum_{i \neq j} \sum_{k=1}^{n_x} \mu_{k,ij}(x_k, \beta) + \sum_{k=1}^{n_e} \log \lambda_{i_k j_k}(t_k^e, x_{t_k^e}, \beta)$, where the generalized exponential formulation is the one adopted by Rastelli and Corneli (2021). Although it is common in the REM literature to simplify inference by using the partial likelihood, we keep the generalized exponential component, as it can be estimated more easily in the M-step of the EM algorithm, described in Section 4.

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

3.2 Discrete time relational event process Y

Often relational events are 'published' only on prespecified discrete event times $\mathcal{T} = \{t_1^e, \ldots, t_n^e\}$. For simplicity of notation, we will assume that the relational event collection process and the jumps of the latent space are equal, i.e., $n = n_x = n_e$ and $\{t_1 = t_1^x = t_1^e, \ldots, t_n = t_n^x = t_n^e\}$. We make an additional assumption that the rate λ is constant with respect to the endogeneous and exogeneous variables inside the collection intervals $(t_k, t_{k+1}]$. In fact, with respect to the endogeneous variable N it makes sense that no further information between the publication dates affects the rates. In other words, assuming a log link for the hazard, for $t \in (t_k, t_{k+1}]$,

$$\log \lambda_{ij}(t, x, \beta) = -d(x_i(t_k), x_j(t_k)) + \beta_G^t B_{ij}(t_k) + \beta_D^t s(\{N(\tau) \mid \tau \leq t_k\}). \tag{5}$$

As the interactions $i \to j$ are collected at t_{k+1} from the observation intervals $(t_k, t_{k+1}]$, the resulting interval counts

$$y_{k,ij} = N_{ij}(t_{k+1}) - N_{ij}(t_k)$$

of the number of interactions between i and j are Poisson distributed with interval rate,

$$\mu_{k,ij}(x_k, \beta) = \int_{t_k}^{t_{k+1}} \lambda_{ij}(t, x, \beta)\, \mathrm{d}t = (t_{k+1} - t_k)\lambda_{ij}(t_k, x, \beta). \tag{6}$$

An advantage of using discrete time is the reduction of the model complexity. It is not uncommon to observe thousands, even millions of links. Such numbers are not surprising when we consider $p(p-1)$ processes having an expected number of links $\mathbb{E}[\sum_{p(p-1)} N_{ij}(t)]$ that grows rapidly. The model can be written as a discrete-time state space process,

$$\begin{cases} x_k \sim N(x_{k-1}, \Sigma), & k = 1, \ldots, n, \\ y_{k,ij} \sim \mathrm{Poi}(\mu_{k,ij}(x_k, \beta)), & 1 \leq i \neq j \leq p. \end{cases} \tag{7}$$

Given the complete observations (x, y), the complete log-likelihood for the state space model in (7) can again be factorized into two components,

$$\ell(\beta, \Sigma) = \log p_\beta(y \mid x) + \log p_\Sigma(x), \tag{8}$$

where $\log p_\beta(y \mid x) = -\sum_{kij} \mu_{k,ij}(x_k, \beta) + \sum_{kij} y_{ij}(k) \log \mu_{k,ij}(x_k, \beta)$ and $\log p_\Sigma(x)$ as above, where the factorization is according to the directed graph in Figure 1, where $y_k \perp y_{-k}, x_{-k} \mid x_k$ and $x_{k+1} \perp x_{k-1} \mid x_k$. Similar to Butts (2008) and Perry and Wolfe (2013), who focused on non-homogeneous exponential waiting times, this approach focuses on non-homogeneous Poisson counts.

One advantage of the latent space formulation is the dimensionality reduction in the latent representation. As the number of nodes p increases, the number of observed counts $p(p-1)n$ grows quadratically, while the latent space grows linearly as pdn.

Dynamic exponential family network model

Given the state space formulation in (7), it is possible to generalize the model considering connections drawn from any exponential family distribution without changing the inference procedure. In fact,

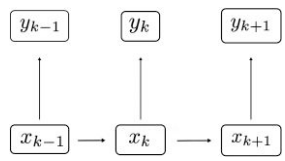

Figure 1. The observed counts y_k are a result of the dynamics in node locations x_k. Hence, y is independent conditionally to the latent locations x.

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

ignoring the connection with any underlying counting process, we could define a temporal network process on discrete time intervals k ($k \in \{1, \ldots, n\}$) between nodes i and j as $f(y_{ij}(k)) = \exp((y_{ij}(k)\theta_{ij} - b(\theta_{ij}))/a(\varphi) + c(y_{ij}(k), \varphi)$, where θ_{ij} is the edge-specific canonical parameter. Using the canonical link function, we can specify the canonical parameter in a similar fashion to (5),

$$\theta_{ij}(x_k) = -d(x_{ki}, x_{kj}),$$

where the values for x are the latent states as before. It is also possible to add additional covariates, but we do not consider this case here. In Online Supplementary Materials D, we show how to obtain the Kalman update equation for any exponential family. The inferential method presented in this manuscript remains mostly the same with a minimal change, effectively replacing the mean $\mu(x_k)$ and variance R_k of the process by

$$\mu(x_k) = b'(\theta)|_{x_k} \quad \text{and} \quad R_k = b''(\theta)a(\varphi)|_{x_k}.$$

This generalized temporal network model can be used to model import and export or other dynamic networks with weighted edges.

Marginalization

One of the main advantages of our latent space network model is that, unlike many other network models, it is coherent under sampling a subset of nodes. Given that any subset V' of V maintains the same distances among nodes, the distribution of the restricted node set $P_{V'}$ is the same as the marginalized distribution of the full model $P_{V|V'}$. This invariance means that it is unimportant to which node set the observed nodes actually belong. Therefore, for the true latent dimension d, as well as for any dimension higher than that, the model is invariant under marginalization. The only effect of subsampling is on inference, in that the conditional variance of the latent locations given the restricted nodes is larger than when given the full node set V, as they have fewer triangulation opportunities.

4 Inference

In this section, we develop all the necessary steps for making inference on the latent states x_k and the parameters Σ an β. Since the latent process x_k is unobserved, we aim to maximize $\int_x L(\beta, \Sigma; y, x)\, dx$. We use the expectation maximization (EM) algorithm (Dempster et al., 1977). The EM algorithm is widely used in problems where certain variables are missing or latent. The EM algorithm consists of an iterative maximization of the conditional expectation of the latent process $X \mid N, \beta, \Sigma$ with respect to the data.

Due to the stepwise dynamic of the latent locations (1), the expectation step is equivalent for both models presented in Sections 3.1 and 3.2. As the locations are constant within intervals $\mathcal{T}$, the continuous time non-homogeneous exponential relational event model N reduces to a discrete time Poisson model during the E-Step,

$$Q(\beta, \Sigma \mid \beta^*, \Sigma^*) = \mathbb{E}_X[\ell(\beta, \Sigma) \mid y],$$

where β^*, Σ^* denotes the parameters estimated at the previous EM iteration. In the maximization step, $Q(\beta, \Sigma \mid \beta^*, \Sigma^*)$ is maximized with respect to the parameters β, Σ. The two steps above are iterated until convergence is reached. The expectation step is typically challenging due to the high dimensional nature of the integral.

The expectation of the log-likelihood can approximately be written as a function of the first two conditioned moments $\mathbb{E}[x_k \mid y_{1:n}]$ and $\mathbb{V}[x_k \mid y_{1:n}]$. Exploiting the state space formulation of the model (7) we can estimate these two quantities with a Kalman filter and smoother (Kalman, 1960). The filter derives mean and variance of the latent process x_k conditioned to the information on y up to time k,

$$\hat{x}_{k|k} = \mathbb{E}[x_k \mid y_{1:k}] \quad V_{k|k} = \mathbb{V}[x_k \mid y_{1:k}].$$

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

The smoother refines these quantities accounting for the complete information on y up to time n,

$$\hat{x}_{k|n} = \mathbb{E}[x_k \mid y_{1:n}] \quad V_{k|n} = \mathbb{V}[x_k \mid y_{1:n}].$$

The expected log-likelihood can be then calculated using these quantities obtained from the smoother.

4.1 E-step: extended Kalman filter

The Kalman filter is one of the most popular algorithms for making inference on state space models and it provides a solution that is both computationally cheap and accurate. Kalman filter is an iterative method that calculates the conditional distribution of the latent x_k. Given the causal Directed Acyclic Graph at Figure 1, x_k depends on x_{k-1} and the observed y_k. Assuming a prior knowledge on the distribution of x_{k-1}, the conditional distribution of x_k is calculated easily. The procedure is applied sequentially from time 1 to n, where the conditional distribution achieved at time k becomes the prior knowledge for the next time point. An arbitrary distribution is specified for the initial x_0. Calculating the conditional distribution entirely could be difficult so the first moments are calculated only. The calculation of the conditional probability involves two steps that are universal in the filtering literature: predict and update. In order to be consistent with the forementioned literature, we denote $\hat{x}_{k|k} = \mathbb{E}[x_k \mid y_{1:k}]$ and $V_{k|k} = \mathbb{V}[x_k \mid y_{1:k}]$ as the expectation and variance conditioned of having observed y_k. Note that x_k and y_k are vectors of length $p_x = pd$ and $p_y = p(p-1)$ or $p(p-1)/2$ in case of an undirected network, respectively. These correspond to the vectorized coordinate and adjacency matrices at time k, respectively. Σ is a $pd \times pd$ matrix and is constant over time. R_k the observed data variance is a diagonal $p_y \times p_y$ matrix. The latent process conditional variance V_k is a $p_x \times p_x$ matrix, whereas the Jacobian matrix H_k is of size $p_x \times p_y$.

Predict

Assume that, at time $k-1$, the approximated conditional distribution of the latent locations is $x_{k-1|k-1} \sim N(\hat{x}_{k-1|k-1}, V_{k-1|k-1})$. For the initial case $k = 1$, we set arbitrarily $x_{0|0} = \upsilon_0$ and $V_{0|0} = \Sigma_0$. The predict step calculates the first moments of x_k conditioned to y_{k-1}. In fields such physics, chemistry, or engineering it is common to employ a forward function $x_k = f(x_{k-1}) + \upsilon_k$ which is related to the physical properties of the system. In our case, the random walk formulation makes no constraints on the latent process evolution. The forward function is the identity with moments

$$\begin{aligned}\hat{x}_{k|k-1} &= \mathbb{E}[x_{k-1} + \upsilon_k \mid y_{1:k-1}] = \hat{x}_{k-1|k-1},\\ V_{k|k-1} &= \mathbb{V}[x_{k-1} + \upsilon_k \mid y_{1:k-1}] = V_{k-1|k-1} + \Sigma.\end{aligned}$$

These are called the apriori mean and variance of the latent locations before observing y_k. The prior distribution is $x_{k|k-1} \sim N(\hat{x}_{k|k-1}, V_{k|k-1})$.

Update

The update step finalizes the calculation of the conditional distribution. We consider the mean vector of all the pairwise relationships $\mu(x_k, \beta) : \mathbb{R}^{p_x} \to \mathbb{R}^{p_y}$ described at (3) and (6) and covariance matrix $\mathbb{V}[y_k] = R_k$ where counts are independent with variance equal to the mean $R_k = \mu(x_k, \beta)\mathbb{I}_{p_y}$. In case a general dynamic network model using exponential family weighted edges, as described in Section 3.2, is considered then the mean $\mu(x_k)$ and variance R_k vary accordingly.

Kalman filters assume that the observed process y_k is Gaussian and the transformations involved are linear. The extended Kalman filter (EKF)(Anderson & Moore, 2012) overcomes the Kalman filter limitations. By means of a first order Taylor expansion

$$\mu(x_k, \beta) = \mu(\hat{x}_{k|k-1}, \beta) + H_k(x_k - \hat{x}_{k|k-1}), \quad H_k = \left.\frac{\partial \mu(x, \beta)}{\partial x}\right|_{\hat{x}_{k|k-1}}, \tag{9}$$

we calculate the expectation $\mathbb{E}[y_k|y_{k-1}] = \mu(\hat{x}_{k|k-1}, \beta)$, variance $\mathbb{V}[y_k|y_{k-1}] = H_k V_{k|k-1} H_k' + R_k$, and covariance $\mathbb{C}ov[x_k, y_k \mid y_{k-1}] = V_{k|k-1} H_k'$ of the conditional predictive distribution of y_k.

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

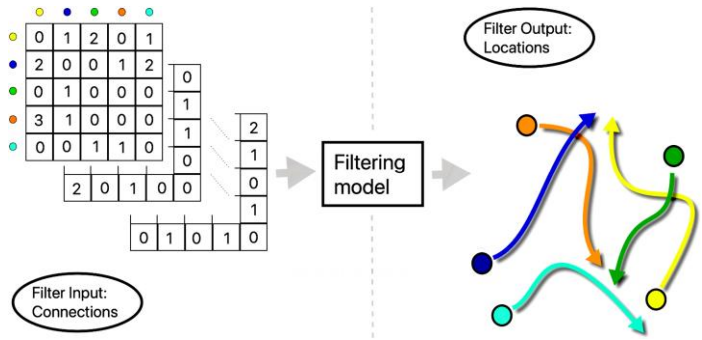

Figure 2. The filtering model takes as input a sequence of adjacency matrices and update the node locations in the latent space.

The joint multivariate distribution of the observed and latent process is

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} \Big| y_{1:k-1} \sim \mathcal{L}\left(\begin{bmatrix} \hat{x}_{k|k-1} \\ \mu(\hat{x}_{k|k-1}, \beta) \end{bmatrix}, \begin{bmatrix} V_{k|k-1} & H_k V_{k|k-1} \\ V_{k|k-1} H_{k'} & H_k V_{k|k-1} H_{k'} + R_k \end{bmatrix} \right),$$

where $\mathcal{L}$ is some probability law parametrized by the first two moments. Using the multivariate regression formulation, we have the conditional moments of x_k

$$\begin{aligned} \hat{x}_{k|k} &= \mathbb{E}[x_k \mid y_{1:k}] = \hat{x}_{k|k-1} + K_k(y_k - \mu(\hat{x}_{k|k-1}, \beta),) \\ V_{k|k} &= \mathbb{E}[(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})' \mid y_{1:k}] = (\mathbb{I} - K_k H_k) V_{k|k-1}, \\ K_k &= V_{k|k-1} H_k' (R_k + H_k V_{k|k-1} H_k')^{-1}, \end{aligned} \tag{10}$$

see at Online Supplementary Materials A for more details. We hence obtain posterior distribution $x_{k|k} \sim N(\hat{x}_{k|k}, V_{k|k})$, which is approximated to be Gaussian. This will be the starting distribution for the inference at time $k+1$. The filtering procedure is shown in Algorithm 1. In Figure 2, we show a visual representation of the algorithm: at each time point, the model takes as input an adjacency matrix and returns the locations in the latent space.

Algorithm 1 Extended Kalman Filter

Initialize $\hat{x}_{0|0} = \nu_0$ and $V_{0,0} = \Sigma_0$

for k = 1, …, n **do**

(a) *Filter prediction step*

$$\hat{x}_{k|k-1} = \hat{x}_{k-1|k-1}$$
$$V_{k|k-1} = V_{k-1|k-1} + \Sigma$$

(b) *Filter update step*

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - \mu(\hat{x}_{k|k-1}, \beta))$$
$$V_{k|k} = (I - K_k\, H_k) V_{k|k-1}$$

where

$$K_k = V_{k|k-1} H_k^{'} (H_k V_{k|k-1} H_k^{'} + R_k)^{-1}$$
$$H_k = \frac{\partial \mu(x, \beta)}{\partial x}\Big|_{\hat{x}_{k|k-1}}$$
$$R_k = \mu(\hat{x}_{k|k-1}, \beta)\mathbb{I}_{p_y}$$

In the update step, the latent locations are updated according to the magnitude of the prediction error: a larger error in the prediction corresponds to a wider change in the locations. The filtering matrix K_k, capturing the linear relationship between the latent and observed processes, weights this prediction error. K_k is the ratio between the noise R_k and the latent variance Σ. Thus, K_k filters

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

the prediction error according to the signal/noise ratio. Fahrmeir (1992) simply considers it as a single Fisher scoring step, see Online Supplementary Materials D.

The Kalman filter can be interpreted both in a frequentist and Bayesian way. From a Bayesian Perspective, the filtering procedure consists of a sequence of updates of the posterior mean and variance (Gamerman, 1991, 1992; West et al., 1985), whereas from a frequentist side, the estimation based on the posterior mode is equivalent to the maximization of a penalized likelihood (Fahrmeir, 1992; Fahrmeir & Kaufmann, 1991), see Online Supplementary Materials D. Approximating the posterior distribution with the same family of the prior, i.e., Gaussian, the posterior mean is equivalent to the posterior mode and hence the equivalence of the two approaches. This double interpretation makes Kalman filters appealing for both types of applications.

Smoother

The smoother moves backward from the last prediction to the first. It calculates the first moments of the latent process conditioned to the information of all time points. Similarly as the EKF, the backward matrix B can be calculated considering the multivariate distribution of the latent locations at two consecutive time points,

$$\begin{bmatrix} x_{k-1} \\ x_k \end{bmatrix} \Big| y_{1:k-1} \sim N\left(\begin{bmatrix} \hat{x}_{k-1|k-1} \\ \hat{x}_{k|k-1} \end{bmatrix}, \begin{bmatrix} V_{k-1|k-1} & V_{k-1|k-1} \\ V_{k-1|k-1} & V_{k|k-1} \end{bmatrix} \right).$$

Using the multivariate regression formula, we have the conditioned mean of x_{k-1} over x_k,

$$\mathbb{E}\left[x_{k-1} \mid x_k, y_{1:k-1}\right] = \hat{x}_{k-1|k-1} + B_k(x_k - \hat{x}_{k|k-1}) \quad \text{with} \quad B_k = V_{k-1|k-1} V_{k|k-1}^{-1}$$

According to the conditional independence in Figure 1, we have $(x_{k-1} \perp y_{k:n}) \mid x_k$ since x_k closes the dependency path. Using the iterated expectation rule, we have

$$\begin{aligned} \hat{x}_{k-1|n} &= \mathbb{E}\left[x_{k-1} \mid y_{1:n}\right] = \mathbb{E}\left[\mathbb{E}\left[x_{k-1} \mid x_k, y_{1:n}\right] \mid y_{1:n}\right] = \mathbb{E}\left[\mathbb{E}\left[x_{k-1} \mid x_k, y_{1:k-1}\right] \mid y_{1:n}\right] \\ &= \mathbb{E}\left[\hat{x}_{k-1|k-1} + B_k(x_k - \hat{x}_{k|k-1}) \mid y_{1:n}\right] \\ &= \hat{x}_{k-1|k-1} + B_k(\hat{x}_{k|n} - \hat{x}_{k|k-1}) \end{aligned}$$

where $\hat{x}_{k-1|k-1}$ and $\hat{x}_{k|k-1}$ are constants. In the same way using the iterated variance rule

$$\begin{aligned} \mathbb{V}\left[x_{k-1} \mid y_{1:n}\right] &= \mathbb{E}\left[\mathbb{V}\left[x_{k-1} \mid x_k, y_{1:n}\right] \mid y_{1:n}\right] + \mathbb{V}\left[\mathbb{E}\left[x_{k-1} \mid x_k, y_{1:n}\right] \mid y_{1:n}\right] \\ &= V_{k-1|k-1} - B_k V_{k|k-1} B_k' + B_k V_{k|n} B_k' \\ &= V_{k-1|k-1} + B_k(V_{k|n} - V_{k|k-1}) B_k', \end{aligned}$$

see at Online Supplementary Materials B for more details. The smoothing procedure is presented in Algorithm 2 and it is known as the RauchTung–Striebel smoother. The final iteration of the smoother updates the starting values $\hat{x}_{0|0}$ and $V_{0|0}$. These values will be used as starting points for the successive EM iteration.

Algorithm 2 Smoother

for k = n, …, 1 **do**

Backward step

$$\hat{x}_{k-1|n} = \hat{x}_{k-1|k-1} + B_k(\hat{x}_{k|n} - \hat{x}_{k|k-1})$$
$$V_{k-1|n} = V_{k-1|k-1} + B_k(V_{k|n} - V_{k|k-1}) B_k'$$

where

$$B_k = V_{k-1|k-1} V_{k|k-1}^{-1}$$

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

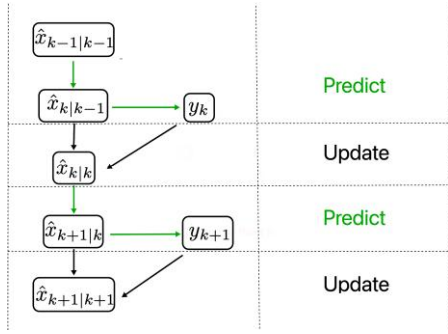

Figure 3. The filtering procedure can be summarized as a sequence of predictions and updates. At each time step, a prediction on the observed link count is made. The prediction error is then propagated back to the nodes for updating their positions.

4.2 M-step: generalized linear model

In the maximization step, we maximize the log-likelihood with respect to the parameters β, Σ and we make the first distinction between the continuous (4) and discrete (8) time models. For the continuous time process N, the expected log-likelihood is

$$Q^N(\beta, \Sigma \mid \beta^*, \Sigma^*) = \mathbb{E}_X[\log p_\beta(N \mid X) \mid y_{1:n}] + \mathbb{E}_X[\log p_\Sigma(X) \mid y_{1:n}] = Q^E(\beta) + Q^G(\Sigma).$$

For the discrete time process Y, the expected log-likelihood is

$$Q^Y(\beta, \Sigma \mid \beta^*, \Sigma^*) = \mathbb{E}_X[\log p_\beta(Y \mid X) \mid y_{1:n}] + \mathbb{E}_X[\log p_\Sigma(X) \mid y_{1:n}] = Q^P(\beta) + Q^G(\Sigma).$$

Notice that the Poisson component $Q^P(\beta)$ and exponential component $Q^E(\beta)$ do not depend on Σ, whereas the Gaussian component $Q^G(\Sigma)$ does not depend on the remaining parameters β. These quantities can, therefore, be optimized separately.

Gaussian component

We can maximize the Gaussian component

$$Q^G(\Sigma) = -\frac{1}{2}\sum_{k=1}^{n} \mathbb{E}[(x_k - x_{k-1})'\Sigma^{-1}(x_k - x_{k-1}) \mid y_{1:n}] - n \log |\Sigma| - \frac{n}{2}\log(2\pi).$$

finding the zero of the first derivative with respect to Σ. Rearranging the elements and taking the expectation as shown in Online Supplementary Materials C, we obtain

$$\begin{aligned}\hat{\Sigma} &= \mathbb{E}\left[\frac{1}{n}\sum_{k=1}^{n}(x_k - x_{k-1})(x_k - x_{k-1})' \mid y_{1:n}\right] \\ &= \frac{1}{n}\sum_{k=1}^{n} V_{k|n} + V_{k-1|n} + B_k V_{k|n} + V_{k|n}B_k' + (\hat{x}_{k|n} - \hat{x}_{k-1|n})(\hat{x}_{k|n} - \hat{x}_{k-1|n})'.\end{aligned}$$

This result corresponds to the one presented in Fahrmeir (1994). Substituting $V_{k|n}B_k' = \mathbb{C}\text{ov}(x_{k|n}, x_{k-1|n} \mid y_{1:n})$, we have the equivalence with the result of Watson and Engle (1983).

The estimate of Σ plays the major role in the bias/variance trade-off. It can find interpretation in the univariate scenario. If the latent process has a small variance, then a little portion of the prediction error is used to update the locations and therefore the latent process moves slowly and

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

delayed. When the variance is high, the estimated latent process is heavily influenced by the last observation and have a tendency to overfit the observed process. In some practical fields, the variance is tuned manually by searching for overfitting or delayed behaviours in the errors. Our EM provides a precise solution and avoids manual tuning.

Poisson component

For arbitrary exponential family distributed edges, as described in Section 3.2, the observed process component can be maximized numerically with a general optimization algorithm. However, for Poisson distribution a more elegant solution is available. Consider the conditional expected rate in the interval $t \in (t_k, t_{k+1}]$

$$\log\left(\mathbb{E}[\lambda_{ij}(t, x_k, \beta)) \mid y_{1:n}\right] = \log\left(\mathbb{E}[\mathrm{e}^{-d(x_{ki}, x_{kj})} \mid y_{1:n}]\right) + \beta_G^t B_{ij}(t_k) + \beta_D^t s(\{N(\tau) \mid \tau \leq t_k\}), \tag{11}$$

with its associated expected cumulative hazard across the entire interval $\mu^*_{k,ij}(y_{1:n}, \beta) = (t_{k+1} - t_k)\mathbb{E}[\lambda_{ij}(t, x_k, \beta)) \mid y_{1:n}]$. The expectation of the Poisson component for the discrete time process Y can then be rearranged as follows:

$$\begin{aligned} Q^P(\beta) &= \sum_{kij} \mathbb{E}[-\mu_{k,ij}(x_k, \beta) + y_{k,ij} \log\left(\mu_{k,ij}(x_k, \beta)\right) - \log\left(y_{k,ij}!\right) \mid y_{1:n}] \\ &= \sum_{kij} -\mu^*_{k,ij}(y_{1:n}, \beta) + y_{k,ij} \log\left(\mu^*_{k,ij}(y_{1:n}, \beta)\right) - \log\left(y_{k,ij}!\right) + C, \end{aligned}$$

which, up to an additive constant, is a Poisson log-likelihood parametrized by $\mu^*_{k,ij}(y_{1:n}, \beta)$. The optimization can be performed by fitting a generalized linear model (McCullagh, 2018) with the above linear predictor and the offset $\log\left(\mathbb{E}[\mathrm{e}^{-d(x_{ki}, x_{kj})} \mid y_{1:n}]\right)$. See Online Supplementary Materials C for the full derivation. The expected value in the offset cannot be further simplified. We use a second order Taylor approximation, which can be expressed as a function of the first two moments of the latent locations, $\hat{x}_{k|n} = \mathbb{E}[x_k \mid y_{1:n}]$ and $V_{k|n} = \mathbb{V}[x_k \mid y_{1:n}]$. Consider $g_{ij}(x) = \mathrm{e}^{-d(x_{ki}, x_{kj})}$, then the expectation within the off-set is approximately

$$\mathbb{E}[g_{ij}(x) \mid y_{1:n}] \approx g_{ij}(\hat{x}_{k|n}) + \frac{1}{2}\mathrm{trace}\left(\frac{\partial^2 g_{ij}(x)}{\partial^2 x}|_{\hat{x}_{k|n}} V_{k|n}\right),$$

since the expectation of the first derivative is zero. Simulation studies show that if the latent space changes smoothly, i.e., a low value on the diagonal of Σ, the approximation is almost perfect.

Above we have described the linear fixed effect case. In the case non-linear or random effects are required then generalized additive modelling (Wood, 2006) can be inserted in this part of the M-step. This formulation is very general and employs spline bases for estimating non-linear or time-varying effects.

Exponential component

The expectation of the exponential component for the continuous time process N is

$$Q^E(\beta) = \mathbb{E}\left[-\sum_{i \neq j}\sum_{k=1}^{n_x} \mu_{k,ij}(x_k, \beta) + \sum_{k=1}^{n_e} \log \lambda_{i_k j_k}(t_k, x_{t_k}, \beta)\right]$$

Note that, up to a multiplicative constant $y_{k,ij}$, the exponential log-likelihood factorizes similarly to that of the Poisson. Also in this case the expected log-likelihood can be rewritten as an exponential log-likelihood with the same offset as in Equation (11). Inference involves survival regression with exponential waiting times. In case the hazard in Equation (2) would also contain an unknown time-varying baseline hazard $\lambda_0(t)$ common to all nodes V, then the M-step could proceed using the partial likelihood as in Cox proportional hazard regression (Cox, 1972).

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

Algorithm 3 Expectation Maximization

Initialize $\hat{x}_{0|0} = \nu_0$, $V_{0|0} = \Sigma$, $\Sigma = \Sigma_0$ and $\beta = \beta_0$

While not converged **do**

(a) Expectation:
- *Extended Kalman Filter*
- *Smoother*

(b) Maximization and update of starting values:
- $\beta = GLM$
- $\Sigma = \hat{\Sigma}$
- $\hat{x}_{0|0} = \hat{x}_{0|n}$
- $V_{0|0} = V_{0|n}$

(c) Check for convergence

4.3 Computational aspects

The $p^2 \times p^2$ matrix inversion in (10) represents a computational bottleneck in many Kalman filter applications. However, there are cases where the dimension of the latent process is much smaller than the observed process dimension. The Sherman–Morrison–Woodbury identity can be employed

$$\left(R_k + H_k V_{k|k-1} H_k'\right)^{-1} = R_k^{-1} - R_k^{-1} H_k (V_{k|k-1}^{-1} + H_k' V_{k|k-1} H_k)^{-1} H_k' R_k^{-1}$$

and requires $p \times p$ matrices inversion only. As the latent space employed by our model has a cheap p-dimensional representation our scenario is particularly appealing for the application of the ShermanMorrison–Woodbury identity. The identity is closely related to the Information Filter (see the Online Supplementary Materials D). The overall computational cost of the algorithm is therefore dominated by the inversion of a $p \times p$ matrix (Mandel, 2006).

4.4 Goodness-of-fit and model selection

The conditional distribution of the latent space x conditioned to the observed process y can be used for assessing the uncertainty about the latent process. Variability bands can be drawn by using the quantiles of the distribution $x_{k|n} \sim N(\hat{x}_{k|n}, V_{k|n})$ and the user can visually check whether the dynamic locations are far from being a constant line, as shown in Figure 4.

Akaike Information Criterion

The dimension d of the latent space can be selected by using some Information Criterion such as the cAIC

$$\text{cAIC} = -2 \log f(y \mid \hat{\beta}, \hat{x}) + 2\Phi,$$

where Φ is the effective degrees of freedom of the fixed and random latent part of the model. Saefken et al. (2014) present a unifying approach for calculating the conditional Akaike information in generalized linear models that can be used in this context. This allows us to select the latent space dimension d that minimizes the conditional Akaike criterion. The cAIC can also be used for choosing between different variance structures, e.g., a diagonal matrix Σ with either the same or different diagonal elements, or for choosing between a static or a dynamic latent model. The static model, where all the locations are fixed in time, can be obtained by modifying our algorithm, as the static model can be viewed as a dynamic model with one single time interval, grouping together all time intervals. The filtering procedure is reduced to updating the locations with $\hat{\Sigma} = 0$.

4.5 Identifiability and divergence

The latent space formulation is identifiable with respect to the relative distances but unidentifiable in the locations (Hoff et al., 2002): infinite combinations of rotations and translations have the

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

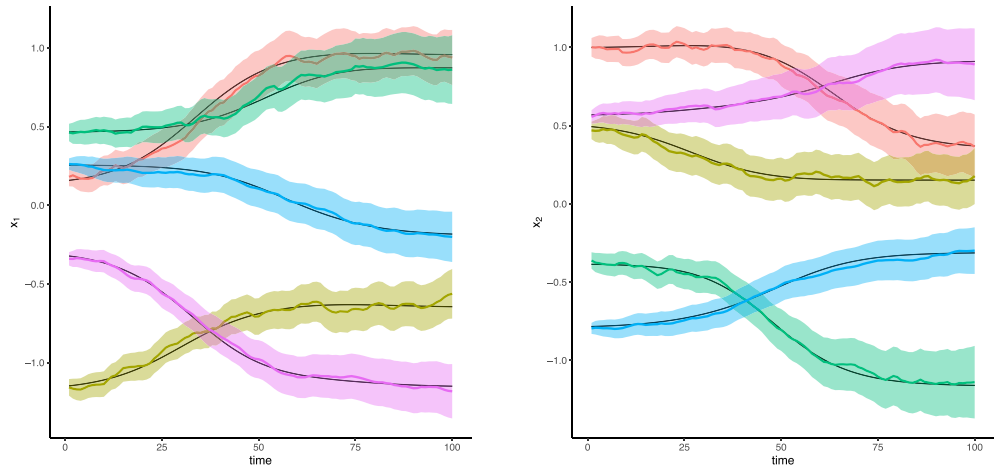

Figure 4. An example of the model fit of the latent space on simulated data with 10 nodes. The two plots represent the $d = 2$ latent space dimensions, x_1 and x_2, across time k for five nodes, by plotting $\hat{x}_{k|n}$ and their variability bands $\hat{x}_{k|n} \pm 1.96\sqrt{V_{k|n}}$. Such quantities are produced by the Kalman smoother, allowing for a straightforward assessment of the model fit. The black line represents the true locations of the simulated data. Procrustes rotation is used to find the best match between the fit and the truth.

same distances and therefore the same likelihood. This implies the non-identifiability of Σ, as the coordinate system rotates. Each update of the filter and smoother may involve a certain shift and rotation in the next location configuration. As a result, when we update the starting points $x_{0|0}$ for the next EM iteration they may be shifted and rotated, with related rotation for Σ. These movements become stable as the starting points $x_{0|0}$ converge. It is however possible to make Σ fully identifiable, by fixing $d + 1$ constraints on the node locations. Alternatively, one can specify Σ spherical or spherical within each node, to obtain an identifiable Σ. In principle, it is possible to extend the latent model to steps with time-varying Σ_t, but it would require additional assumptions. For example, assuming that the $d \times d$ diagonal submatrices of the $dp \times dp$ matrix Σ_t are identical makes it identifiable. However, this is undesirable from a practical point of view as it would make each node equally variable, which is clearly not the case in many scenarios. Instead, we prefer to interpret the time-homogeneity of Σ as a Bayesian prior on X: rather than being an assumption on the underlying generating process of X, it guarantees the 'continuity' of X as well as identifiability of a particular axis of rotation of the latent space. Clearly, this assumption affects the posterior distribution of X, but not strongly its posterior mean, which is our main quantity of interest.

A practical aspect of Kalman filter users may encounter when working on real data is divergency issues of the algorithm, defined as generating unbounded state value residuals within the procedure (Fitzgerald, 1971). Many factors can influence the divergence tendency such as a wrong variance specification in R_k, poor approximation of non-linearity, inappropriate initial choice β, abrupt changes in link rates, too large variances on the diagonal of $V_{0|0}$ and Σ or poor initial latent state values x_0. In the case of bad starting points x_0, the update of locations might have abrupt changes because in a non-convex likelihood optimization locations jump to find a more stable configuration.

Fine-tuning parameters and starting points can resolve the above problems. Artificially inflating R_k solves the overdispersion problems, although inferring the correct variance function of the data might take some extra effort. Sufficiently good $x_{0|0}$ points can be calculated via multidimensional scaling or reversing the time dimension and running the Kalman Filter backward. Furthermore, starting the EM close to the static model, by setting the diagonal values of $V_{0|0}$ and Σ low, always leads to stable Kalman update. In fact, the latent space variances can be seen as tuning parameters that can be expanded slowly to allow for more movement in the latent space. Where possible, one eventually expands them towards the maximum likelihood values. Otherwise, a profile maximum likelihood estimate will be the best alternative.

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

5 Simulation study

In order to assess the method performance, we carry out a simulation study. We specify logistic functions for the latent location trajectories x_k, rescaling and shifting these functions in different ways. The link counts are generated from a Poisson distribution with $\log(\mu_{k,ij}(x_k)) = \alpha - \|x_{ki} - x_{kj}\|_2^2$ for p nodes across n intervals with d latent dimensions. The simulation study involves varying the number of nodes, intervals, and dimension. We also propose some challenges to the model such as the misspecification of the distribution family, high clustering, or sparsity behaviour. Optimal starting points are calculated via the static model as described in Section 4.5. We use the out-of-fold Kullback–Leibler divergence as performance measure,

$$KL(\hat{x}, x_{\text{true}}) = \mathbb{E}_y\left[\log p(y \mid x_{\text{true}}) - \log p(y \mid \hat{x})\right] \approx \frac{\sum \log p(y_{\text{new}} \mid x_{\text{true}}) - \log p(y_{\text{new}} \mid \hat{x})}{np(p-1)/2},$$

where y_{new} denotes an additional sample that is generated from x_{true}. The Kulback–Leibler is a performance measure based on the distance matrix, which is invariant to rotations and translations of the locations.

Varying the number of nodes p

Figure 5a shows the results of varying the number of nodes $p = 5, 10, 25, 50$. The EKF performance improves as p increases dramatically. This is a consequence of, on the one hand, a quadratic increase in the number of possible interactions and, on the other, a quadratic increase of the number of triangulation opportunities in the latent space.

Varying the number of intervals n

Figure 5b shows the results of varying the number of observed time sub-intervals $n = 10, 50, 100, 1000$. Again, the EKF performance improves with the increase of n. The reason for the improvement is that when the same time interval is divided in a larger number of sub-intervals, it reduces the effective latent space variance and it increases the number of observations.

Varying the latent dimension d

Figure 5c shows a slight decrease in the performance when increasing the true latent dimension d. Clearly, when the latent dimension increases, the number of observations remains constant, but the dynamics becomes more complex, resulting in an increase of the KL divergence.

Effect of model misspecification: overdispersion

In Figure 5d we investigate the inference behaviour under one type of model misspecification, namely, overdispersion. We simulate data from a negative binomial with mean $\mu_{k,ij}(x_k)$ and a quadratic variance function $\mu_{k,ij}(x_k) + \mu_{k,ij}(x_k)^2$ and compare the performance of the EKF to data simulated from a Poisson distribution with the same increasing mean $\mu_{k,ij}(x_k)$. For low rates, the negative binomial variance is almost the same as that of the Poisson, and here we observe the same EKF performances over the two distributions. For high rates, the fit on negative binomial counts deteriorates and starts to become comparable to that of the static model. For the highest rate in the simulation study, the signal-to-noise ratio in the data is so low that the inference procedure diverges in all the simulations. However, it is interesting to note that for highly sparse counts of relational events, the inference procedure always converges (for more details, see Online Supplementary Materials F).

Alternative methods

In the various simulations, we compare the EKF implementation with two possible competitors. The Unscented Kalman filter uses a so-called unscented transformation as an alternative to the EKF linear approximation of non-linear equations. For details, we remand the reader to the Online Supplementary Materials E. The static model refers to the latent space implementation

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

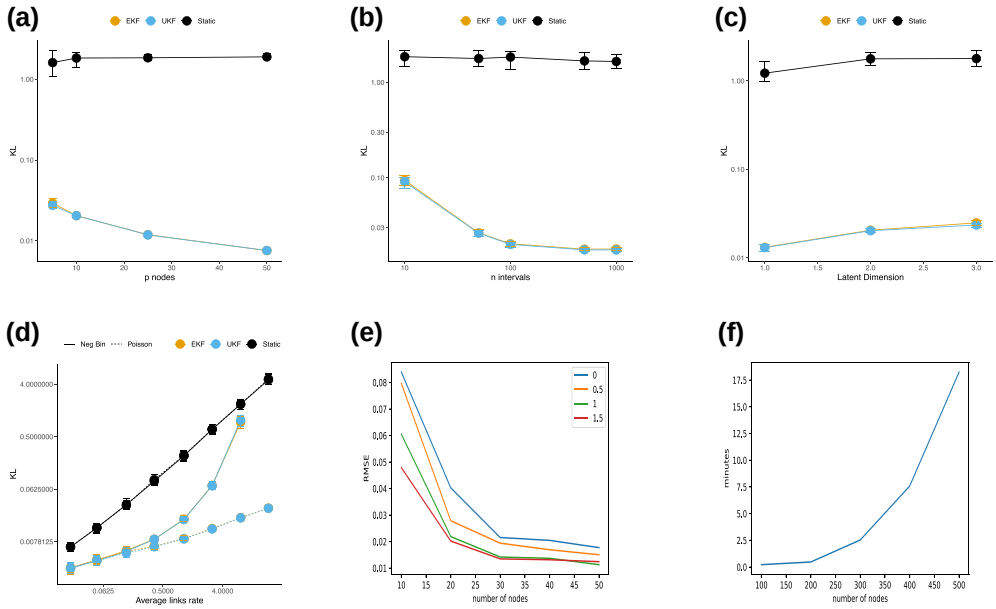

Figure 5. Kullback–Leibler measure shows that the EKF and UKF both improve performance with (a) additional number of nodes p and (b) interval n, while slightly deteriorates when (c) increasing the latent dimension d. (d) Shows the effects of model misspecification, (e) the reliability of endogeneous effect estimation in our latent space formulation. (f) Computational time grows markedly in the number of nodes p. (a) Nodes. (b) Sub-intervals. (c) Latent dimension. (d) Overdispersion. (e) Reciprocity strength. (f) Computational time.

with non-dynamic states, described in Section 4.4. Figure 5a–d show that the EKF and UKF have very similar performances in terms of KL divergence, whereas the computational costs are very similar (Online Supplementary Materials F). In general, it can be seen that ignoring state dynamics can be highly detrimental, as the KL divergence of the static model is typically much higher than that of the EKF. However, there is one exception: if the model is highly misspecified and the dispersion is much higher than that of a Poisson, then the static model becomes more robust and starts to become competitive.

Modeling endogeneous effects

On the one hand, endogeneous effects, such as reciprocity or triadic effect, are drivers of relational events that depend on the past structure of the network. Other the other hand, the latent space itself also encapsulates part of the network structure. Therefore, it is important to check whether endogeneous effects are identifiable in the presence of latent dynamics. Figure 5e shows the mean squared error (MSE) of the estimated reciprocity for four different reciprocity strength in a simulation study across increasing number of nodes p. The results show that the MSE decreases roughly as $1/p$, which is consistent with the fact that the information grows quadratic with the number of nodes.

Modeling larger networks

The simulations so far were performed on relatively small networks with $p \leq 50$, a dimension that is achievable for a custom implementation in the R language. For larger networks, we created an implementation in Tensorflow and performed the simulations on *Google Colab* using its free GPU resources. Figure 5f shows the computational time for larger networks. The 100 nodes model converges in roughly 22 s, whereas for networks with 500 nodes roughly 20 min are needed. Computational time seems to increase roughly quadratically in the number of nodes. Another common computational bottleneck in large networks is that the number of observations carried by the adjacency matrix and the related machine operations grow quadratically with the number of nodes. In that case stratified subsampling (Raftery et al., 2012) on the adjacency matrix

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

elements could reduce the computational burden. Using this idea, a pilot Kalman filter can be run to calculate the stratum contribution via the increment in the expected log-likelihood. Other ideas, such as parallel Kalman Filters (Särkkä & García-Fernández, 2020) where multiple time points can be computed in parallel, can only be implemented if the memory consumption of each individual Kalman filter iteration is small, which is not our case.

6 Dynamics of patent citation patterns

The patent citation process introduced in Section 2 presents some peculiar characteristics with respect to the underlying relational event: patents are added in tranches to the system, and citations happen only at the moment of patent creation. Furthermore, patents can cite only those patents that have previously been created and not the ones that are added to the network in the future. Therefore, rather than focusing on the individual patents, we focus on the citations between groups of patents, such as the patent classes and subclasses, described above. Our aim is to describe the relative changing importance of each of these (sub)classes over time in being cited as prior art in novel patents. We consider the latent space model for the number of citations $y_{k,ij}$ from patents of field i to patents of field j at time k,

$$\begin{aligned} y_{k,ij} &\sim \text{Poi}(\mu_{k,ij}(x_k, \beta)) \\ \log(\mu_{k,ij}(x_k, \beta)) &= \log C_i(k) + \alpha_0 - \|x_{ki} - x_{kj}\|_2^2 + \text{sender}_i + \text{receiver}_j, \end{aligned} \tag{12}$$

where α_0 is an intercept and sender_i and receiver_j are, respectively, the sender and receiver random effects. We include random effects in the linear predictor as the usual conditional formulation of the regression model. The citation rate is proportional to the number of patents $C_i(k)$ added in a field within a year. If in a certain year there are no patents added in a field, the rate would clearly be zero. We therefore specify an additional offset $\log C_i(k)$ that accounts for the number of patents added in field i at time k. The inclusion of the offset has the advantage that the interpretation of the latent space locations and other effects is with respect to a single patent in each (sub)class. As the aim is to explore the major relative movement of each of the (sub)classes, we consider here a bidimensional latent space. Optimal starting points are calculated via the static model as described in Section 4.5.

Figure 6 shows a peculiar behaviour of the latent locations of the eight main technology classes. They seem to be more or less static in the initial 10 years from 1967 until 1976. Patents can only cite back in time and therefore the first patents added in the system cannot cite patents submitted before the year 1967. The apparent stationarity may therefore be an artifact. The figure suggests

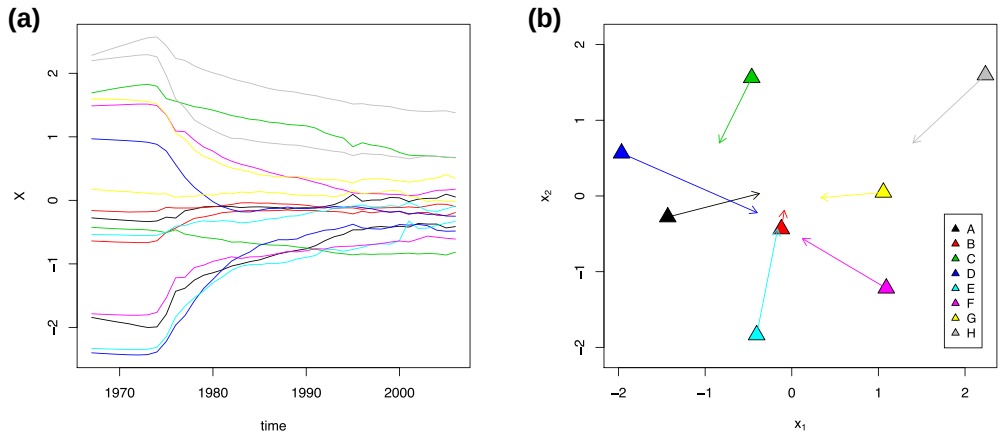

Figure 6. Changes in latent space patent locations. (a) The two coordinates for each of the eight main classes are shows in the same figure. The first ten years show a static behaviour in citations. After that point, the fields start moving closer as the citations between fields intensify; (b) the overall change in latent space locations of the eight main classes over the entire period of 1967–2006.

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

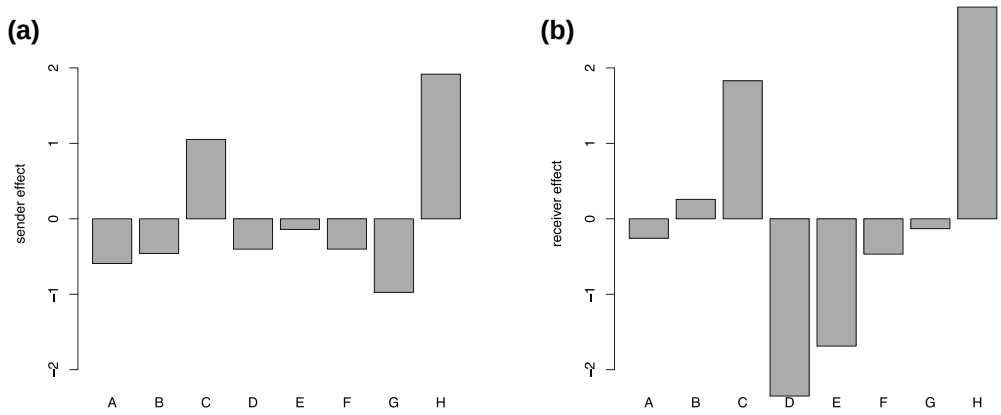

Figure 7. Model inference on dynamic locations for the relational event model with sender and receiver effects. (a) shows a summary of the movement of the patent classes in the observed time interval. (a) Sender effect. (b) Receiver effect.

that around 1976 the patent citation process start behaving more 'normally', i.e., it starts to represent more representatively the bulk of the citation process. This seems reasonable as patents cite an average of 10 years back in time, with a mode that is significantly less than 10 years.

In general, we observe that the exchange of citations between different fields increases through time, ending with a large cluster including the majority of the ICL categories. Only classes C (chemistry and metallurgy) and H (electricity) remain somewhat separate from the other main classes. The overall conclusion is that except for classes C and H, the other main technology classes lose their specific characteristics and patents tend to cite more across technology class borders. This suggests that most technology classes are becoming less dissimilar: there is an increasing heterogeneity *within* the fields, as they communicate with other technology fields, and thus a higher homogeneity *between* the fields.

The sender and receiver effects can be interpreted as the asymmetry between fields citations that the symmetric latent space representation fails to capture. Figure 7b shows how the Textile, Papers, and Fixed constructions classes are very low receiver classes, meaning that they are cited below average. Figure 7 shows that Physics patents a low tendency to cite others. The high sending and receiving tendencies of the chemistry, metallurgy, and electricity patents must be seen in the context of Figure 6: the fact that we observe such huge effects jointly together with their distant location to the other patent classes might suggest some violation of the model assumptions. The two locations should be closer to the main cluster but there does not exist a 2D latent configuration that makes a good fit. An analysis without sender and receiver effects (Online Supplementary Materials G) indeed shows that those two classes would be apparently closer, joining the other technology classes.

6.1 Extending the analysis to subclass dynamics

The eight main technology classes give a rough overview of the patent dynamics. However, given that we analyze more than 23 million citations, a finer analysis should reveal more detailed results. We therefore extend the analysis to the subclass level of the patent classification system. The eight main technology classes consist of a total of 487 more specific subclasses.

Figure 8 shows the latent dynamics for all the 487 subclasses, where the color refers to the original eight main technology classes. What is immediately clear is that the dynamics within a single technology class is quite diverse. Figure 8a shows that the subclasses are evenly spread in the latent plane. Moreover, by inspecting single subclass trajectories, it emerges that a subclass tends to move with few subclasses from within the same main technology class, but also with some subclasses from another class. This is consistent with the raw data, as approximately half of a patent's connectivity is within the same class, while the rest is towards other classes.

Figure 8 also shows that subclasses are heterogeneous in their citation behaviour from the beginning, and that not all the subclasses converge to a single cluster. Technology subclasses end

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

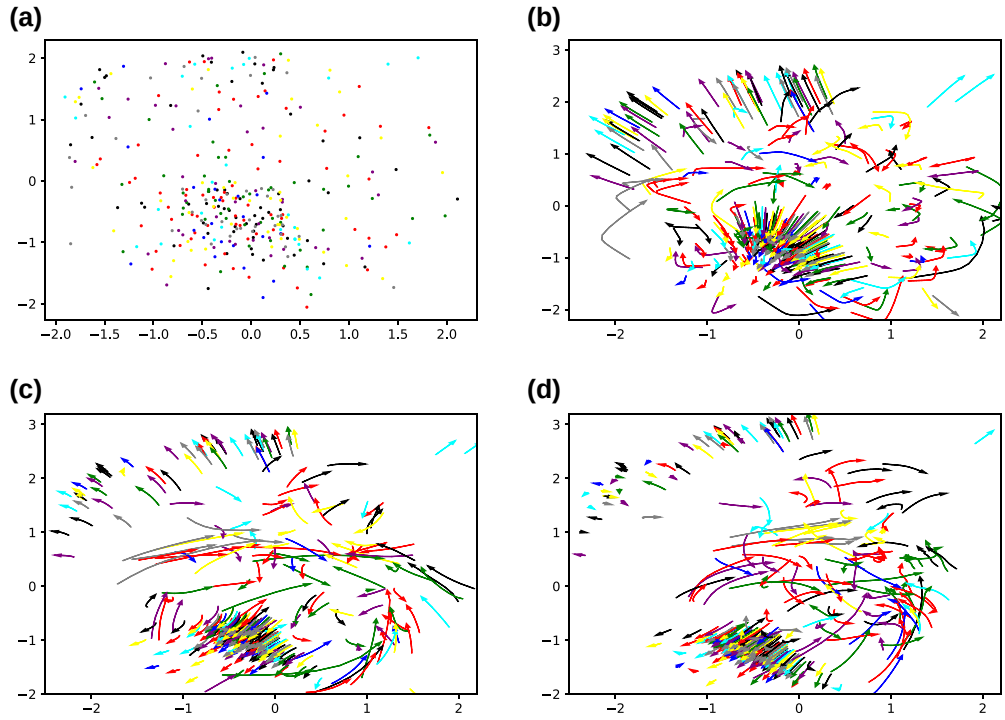

Figure 8. Dynamic latent locations for the 487 technology subclasses. The colors correspond to the original eight main technology classes. (a) Initial configuration in 1967. (b) Changes in years 1967–1980, (c) changes in years 1980–1993, and (d) changes in years 1993–2006.

up forming three heterogeneous clusters, as evidenced by Figure 8d. As time passes, the bottom left nodes separate from the centre and converge into a dense cluster, revealing an increasing heterogeneity in their citation. On the top left, something similar happen although this cluster is less dense as its nodes do not seem to shorten their distances. Nodes belonging to clusters with such a flatten shape typically present high connectivity with the immediate neighbour, but this connectivity does not extend to distant nodes, creating a chain where the two poles share little similarity. At the centre, by looking at the inward arrows, it is possible to spot a third, low density cluster which is separating from the other two. We conclude that the increment of heterogeneity in patent citations is not uniform across all subclasses. There is some coordinated movement from the three clusters of subclasses. Patents within these clusters tend to get more similar citing behaviour, whereas patents between these clusters tend to cite each other less. It is interesting to note that the apparent converging behaviour of the main technology classes in Figure 6 is simply the result of aggregating the subclasses where the diverging movements are averaged out.

7 Conclusion

In the last decade, REMs have been used for describing the drivers of dynamic network interactions. Traditional approaches focus on endogeneous and exogeneous drivers, which may not always be able to capture all heterogeneity in the data. Our aim has been to extend relational event modelling by letting their interactions depend on dynamic locations in a latent space.

The model defines the latent locations as missing states, where the observations are the time-stamped relational events or aggregates of those events within a certain interval. We use an EM algorithm, whereby a Kalman filter calculates their conditional expectation and a generalized linear model formulation performs the maximization step. Kalman filters are effective methods for estimating latent dynamic processes. Their simplicity and computational efficiency make them suitable for many problems common in engineering contexts. The filter relies on a sequence of

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

linear operations and easily calculates the expectation step, typically untractable for non-trivial cases. The Kalman filter dual interpretation in the Bayesian and frequentist literature would also make an effective Gibbs possible. Current Bayesian approaches, such as Sewell and Chen (2015), rely on a simplified stratified case-control sampling of non-events. As there are many more non-events with distant nodes, mid-distances are either never sampled or sampled and overweighted by an inappropriate case-control weight. Although this reduces computational complexity, this produces bias in the inference procedure.

It is easy to extend the linearity of the exogeneous and endogeneous effects in the model formulation (2) to smooth effects. The generalized linear model approach for the M-step can easily be replaced by a generalized additive setup for incorporating smooth and time-varying effects as well as random effects (Wood, 2006). The simulation results show that the modelling and inference set-up is accurate, computationally feasible, and insightful under different scenarios.

We applied the model to 23 million patent citations from the US patent office in order to investigate the innovation dynamics in the period 1967–2006. Focusing on the eight main technology classes suggests that there is an overall convergence in the latent space, meaning that the patents classes are becoming either more similar or more internally dissimilar. A subsequent analysis on the 487 subclasses revealed that the second hypothesis explains most of the apparent convergence: it seems that the subclasses within each main technology class have coordinated, but diverging dynamics, which suggest that the main technology classes have become more dissimilar over time. This may be because the original class denominations refer to distinctions that have become less relevant over time. For this reason, it would probably be good to avoid using the main technology classes as important descriptors of patents, and instead focus on their subclass denominations.

Conflict of interest: The authors declare that there are no conflicts of interest associated with this manuscript.

Funding

The authors acknowledges funding from the Swiss National Science Foundation (SNSF 188534).

Data availability

The patent citation data are publicly available in the NBER repository.

Supplementary material

Supplementary material are available at *Journal of the Royal Statistical Society: Series A* online.

References

Anderson B. D., & Moore J. B. (2012). *Optimal filtering*. Courier Corporation.

Bourdieu P. (1989). Social space and symbolic power. *Sociological Theory*, *7*(1), 14–25. https://doi.org/10.2307/202060

Brandes U., Lerner J., & Snijders T. A. (2009). Networks evolving step by step: statistical analysis of dyadic event data. In *2009 International Conference on Advances in Social Network Analysis and Mining* (pp. 200–205). IEEE.

Butts C. T. (2008). 4. A relational event framework for social action. *Sociological Methodology*, *38*(1), 155–200. https://doi.org/10.1111/j.1467-9531.2008.00203.x

Cook S., & Soramaki K. (2014). *The global network of payment flows*. (Technical Report). SWIFT Institute Working Paper.

Cox D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, *34*(2), 187–202. https://doi.org/10.1111/j.2517-6161.1972.tb00899.x

Dempster A. P., Laird N. M., & Rubin D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*(1), 1–22. https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

De Vos S., Wardenaar K. J., Bos E. H., Wit E. C., Bouwmans M. E., & De Jonge P. (2017). An investigation of emotion dynamics in major depressive disorder patients and healthy persons using sparse longitudinal networks. *PLoS One*, *12*(6), e0178586. https://doi.org/10.1371/journal.pone.0178586

DuBois C., Butts C., & Smyth P. (2013). Stochastic blockmodeling of relational event dynamics. In *Artificial intelligence and statistics* (pp. 238–246).International Conference on Artificial Intelligence and Statistics

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

Durante D., & Dunson D. B. (2016). Locally adaptive dynamic networks. *The Annals of Applied Statistics*, *10*(4), 2203–2232. https://doi.org/10.1214/16-AOAS971
Fahrmeir L. (1992). Posterior mode estimation by extended Kalman filtering for multivariate dynamic generalized linear models. *Journal of the American Statistical Association*, *87*(418), 501–509. https://doi.org/10.1080/01621459.1992.10475232
Fahrmeir L. (1994). Dynamic modelling and penalized likelihood estimation for discrete time survival data. *Biometrika*, *81*(2), 317–330. https://doi.org/10.1093/biomet/81.2.317
Fahrmeir L., & Kaufmann H. (1991). On Kalman filtering, posterior mode estimation and Fisher scoring in dynamic exponential family regression. *Metrika*, *38*(1), 37–60. https://doi.org/10.1007/BF02613597
Fitzgerald R. (1971). Divergence of the Kalman filter. *IEEE Transactions on Automatic Control*, *16*(6), 736–747. https://doi.org/10.1109/TAC.1971.1099836
Gamerman D. (1991). Dynamic Bayesian models for survival data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *40*(1), 63–79. https://doi.org/10.2307/2347905
Gamerman D. (1992). A dynamic approach to the statistical analysis of point processes. *Biometrika*, *79*(1), 39–50. https://doi.org/10.1093/biomet/79.1.39
Hanneke S., Fu W., & Xing E. P. (2010). Discrete temporal models of social networks. *Electronic Journal of Statistics*, *4*, 585–605. https://doi.org/10.1214/09-EJS548
Hoff P. D. (2005). Bilinear mixed-effects models for dyadic data. *Journal of the American Statistical Association*, *100*(469), 286–295. https://doi.org/10.1198/016214504000001015
Hoff P. D. (2008). Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in neural information processing systems* (pp. 657–664). NeurIPS Conference.
Hoff P. D. (2009). Multiplicative latent factor models for description and prediction of social networks. *Computational and Mathematical Organization Theory*, *15*(4), 261. https://doi.org/10.1007/s10588-008-9040-4
Hoff P. D., Raftery A. E., & Handcock M. S. (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Association*, *97*(460), 1090–1098. https://doi.org/10.1198/016214502388618906
Kalman R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, *82*(1), 35–45. https://doi.org/10.1115/1.3662552
Lafond F., & Kim D. (2019). Long-run dynamics of the US patent classification system. *Journal of Evolutionary Economics*, *29*(2), 631–664. https://doi.org/10.1007/s00191-018-0603-3
Mandel J. (2006). *Efficient implementation of the ensemble Kalman filter* (University of Colorado at Denver and Health Sciences Center). Center for Computational Mathematics Reports.
McCullagh P. (2018). *Generalized linear models*. Routledge.
Perry P. O., & Wolfe P. J. (2013). Point process modelling for directed interaction networks. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *75*(5), 821–849. https://doi.org/10.1111/rssb.12013
Raftery A. E., Niu X., Hoff P. D., & Yeung K. Y. (2012). Fast inference for the latent space network model using a case-control approximate likelihood. *Journal of Computational and Graphical Statistics*, *21*(4), 901–919. https://doi.org/10.1080/10618600.2012.679240
Rastelli R., & Corneli M. (2021). 'Continuous latent position models for instantaneous interactions', arXiv, arXiv:2103.17146, preprint: not peer reviewed.
Saefken B., Kneib T., van Waveren C.-S., & Greven S. (2014). A unifying approach to the estimation of the conditional Akaike information in generalized linear mixed models. *Electronic Journal of Statistics*, *8*(1), 201–225. https://doi.org/10.1214/14-EJS881
Sarkar P., & Moore A. W. (2005). Dynamic social network analysis using latent space models. *Acm Sigkdd Explorations Newsletter*, *7*(2), 31–40. https://doi.org/10.1145/1117454.1117459
Särkkä S., & García-Fernández Á. F. (2020). Temporal parallelization of Bayesian smoothers. *IEEE Transactions on Automatic Control*, *66*(1), 299–306. https://doi.org/10.1109/TAC.2020.2976316
Sewell D. K., & Chen Y. (2015). Latent space models for dynamic networks. *Journal of the American Statistical Association*, *110*(512), 1646–1657. https://doi.org/10.1080/01621459.2014.988214
Signorelli M., Vinciotti V., & Wit E. C. (2016). NEAT: an efficient network enrichment analysis test. *BMC Bioinformatics*, *17*(1), 1–17. https://doi.org/10.1186/s12859-016-1203-6
Signorelli M., & Wit E. C. (2018). A penalized inference approach to stochastic block modelling of community structure in the Italian parliament. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *67*(2), 355–369. https://doi.org/10.1111/rssc.12234
Snijders T. A., & Pickup M. (2017). Stochastic actor-oriented models for network dynamics. *Annual Review of Statistics and its Application*, *4*(1), 343–363. https://doi.org/10.1146/annurev-statistics-060116-054035
Tranmer M., Marcum C. S., Morton F. B., Croft D. P., & de Kort S. R. (2015). Using the relational event model (REM) to investigate the temporal dynamics of animal social networks. *Animal Behaviour*, *101*, 99–105. https://doi.org/10.1016/j.anbehav.2014.12.005
Užupytė R., & Wit E. C. (2020). Test for triadic closure and triadic protection in temporal relational event data. *Social Network Analysis and Mining*, *10*(1), 1–12. https://doi.org/10.1007/s13278-020-0632-4

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024

Vu D., Lomi A., Mascia D., & Pallotti F. (2017). Relational event models for longitudinal network data with an application to interhospital patient transfers. *Statistics in Medicine*, *36*(14), 2265–2287. https://doi.org/10.1002/sim.7247

Watson M. W., & Engle R. F. (1983). Alternative algorithms for the estimation of dynamic factor, mimic and varying coefficient regression models. *Journal of Econometrics*, *23*(3), 385–400. https://doi.org/10.1016/0304-4076(83)90066-0

West M., Harrison P. J., & Migon H. S. (1985). Dynamic generalized linear models and Bayesian forecasting. *Journal of the American Statistical Association*, *80*(389), 73–83. https://doi.org/10.1080/01621459.1985.10477131

Wood S. N. (2006). *Generalized additive models: An introduction with R*. Chapman and Hall/CRC.

Younge K. A., & Kuhn J. M. (2016). Patent-to-patent similarity: a vector space model. *Available at SSRN 2709238*.

Downloaded from https://academic.oup.com/jrsssa/article/186/3/508/7126053 by Biblioteca universitaria di Lugano user on 04 September 2024