Bayesian tensor factorisations for time series of counts

Zhongzhen Wang[1] · **Petros Dellaportas**[1,2] · **Ioannis Kosmidis**[3]

Abstract

We propose a flexible nonparametric Bayesian modelling framework for multivariate time series of count data based on tensor factorisations. Our models can be viewed as infinite state space Markov chains of known maximal order with non-linear serial dependence through the introduction of appropriate latent variables. Alternatively, our models can be viewed as Bayesian hierarchical models with conditionally independent Poisson distributed observations. Inference about the important lags and their complex interactions is achieved via MCMC. When the observed counts are large, we deal with the resulting computational complexity of Bayesian inference via a two-step inferential strategy based on an initial analysis of a training set of the data. Our methodology is illustrated using simulation experiments and analysis of real-world data.

Keywords Dirichlet process · MCMC · Poisson distribution · Tensor factorisation

1 Introduction

We consider a time-index sequence of multivariate random variables of size T, $\{y_t\}_{t=1}^{T}$, taking values in $\{0, 1, \ldots\}$. We build a non-parametric model by (i) assuming that the transition probability law of the sequence $\{y_t\}$ conditional on the filtration up to time $t-1$, $\mathbf{F}_{t-1}$, is that of a Markov chain of maximal order q, (ii) allowing non-linear dependence of the values at the previous q time points and (iii) incorporating complex interactions between lags.

We propose a Bayesian model for multivariate time series of counts based on tensor factorisations. Our development is inspired by Yang and Dunson (2016) and (Sarkar & Dunson, 2016). Yang and Dunson (2016) introduced conditional tensor factorisation models that lead to parsimonious representations of transition probability vectors together with a simple, powerful

✉ Petros Dellaportas
  p.dellaportas@ucl.ac.uk

[1] University College London, London, UK

[2] Athens University of Economics and Business, Athens, Greece

[3] University of Warwick, Coventry, UK

Bayesian hierarchical formulation based on latent allocation variables. This framework has been exploited in Sarkar and Dunson (2016) to build a nonparametric Bayesian model for categorical data together with an efficient MCMC inferential framework. We adopt the ideas and methods of these papers to build flexible models for time series of counts. The major difference that distinguishes our work to Sarkar and Dunson (2016) is that, unlike categorical data, we deal with time series that are infinite, rather than finite, state space Markov chains. The resulting computational complexity of our proposed model is grown as the observed counts become larger, so we propose a two-step inferential strategy in which an initial, training part of the time series data, is utilized to facilitate the inference and prediction of the rest of the data.

A common way to analyse univariate time series of counts is by assuming that the conditional probability distribution of $y_t \mid y_{t-1}, \dots, y_{t-q}$ can be expressed as a Poisson density with rate λ_t that depends either on previous counts $y_{t-1}, \dots, y_{t-q}$ or previous intensities $\lambda_{t-1}, \dots, \lambda_{t-q}$. For example, one such popular model is the Poisson autoregressive model (without covariates) of order q, PAR(q):

$$y_t \sim \text{Poisson}(\lambda_t),$$
$$\log(\lambda_t) = \beta_0 + \sum_{i=1}^{q} \beta_i \log(y_{t-i} + 1) \tag{1}$$

where $\beta_0, \beta_1, \dots, \beta_q$ are unknown parameters; see (Cameron & Trivedi, 2001). Grunwald et al. (2000), Grunwald et al. (1997) and (Fokianos, 2011) discuss the modelling and properties of a PAR(1) process. Brandt and Williams (2001) generalise PAR(1) to a PAR(q) process and apply it to the modelling of presidential vetoes in the United States. Kuhn et al. (1994) adopt such processes to model the counts of child injury in Washington Heights. When we deal with M distinct time series of counts, the PAR(q) model is written, for $m = 1, \dots, M$, as

$$y_{m,t} \sim \text{Poisson}(\lambda_{m,t}),$$
$$\log(\lambda_{m,t}) = \beta_0 + \sum_{m=1}^{M} \sum_{i=1}^{q} \beta_{i,m} \log(y_{m,t-i} + 1); \tag{2}$$

see, for example, Liboschik et al. (2015). In the above equation, q is fixed for each $m = 1, \dots, M$. We will use this model formulation as a benchmark for comparison against our proposed methodology. Other approaches to modelling time series of counts include the integer-valued generalised autoregression conditional heteroscedastic models (Heinen, 2003; Weiß 2014) and the integer-valued autoregression processes (Al-Osh & Alzaid, 1987). We have not dealt with these models here because a proper Bayesian evaluation of their predictive performance requires a challenging Bayesian inference task which is beyond the scope of our work.

The rest of the paper is organised as follows. We specify our model in Section 2, followed by estimation and inference details in Section 3. Simulation experiments and applications are provided in Section 4 and 5, respectively.

2 Model specification

2.1 The Bayesian tensor factorisation model

2.1.1 Univariate time series

We build a probabilistic model by assuming that the transition probability law of y_t conditional on $\mathbf{F}_{t-1}$ is that of a Markov chain of maximal order q:

$$p(y_t \mid \mathbf{F}_{t-1}) = p(y_t \mid \{y_{t-j}\}_{j\in[1,q]}), \tag{3}$$

for $t \in [q + 1, T]$ where the set containing all integers from i to j is denoted as $[i, j]$. This formulation includes the possibility that only a subset of the previous q values affects y_t. We follow (Sarkar & Dunson, 2016) and introduce a series of latent variables as follows. First, let k_j denote the maximal number of clusters that the values of y_{t-j} can be separated into for predicting y_t. To demonstrate the use of k_j we present a simple example. Assume that y_t depends only on y_{t-1} and the relationship in which the observed values of y_{t-1} affect the density of y_t is based on the following stochastic rule: if $y_{t-1} > 1$ then $y_t \sim \text{Poisson}(1)$ and if $y_{t-1} \leq 1$ then $y_t \sim \text{Poisson}(2)$. Then $k_1 = 2$ since the values of y_{t-1} are separated into two clusters that determine the distribution of y_t. Note that if $k_j = 1$ the value of y_{t-j} does not affect the density of y_t. The collection of all these latent variables $K := \{k_j\}_{j\in[1,q]}$ determines how past values of the time series affect the distribution of y_t.

We also define a collection of time-dependent latent allocation random variables $Z_t := \{z_{j,t}\}_{j\in[1,q]}$ where $z_{j,t}$ specifies which of the k_j clusters of y_{t-j} affects y_t. We will write $Z_t = H$ meaning that all latent variables in Z_t equal to another collection of latent variables $H := \{h_j\}_{j\in[1,q]}$ that do not depend on t. Finally, denote the collection $\mathbf{H} := \{h_j \in [1, k_j], j \in [1, q]\}$ that depends on K. The connection among Z_t, H and $\mathbf{H}$ is that for any $t \in [q + 1, T]$, Z_t is sampled with value $H \in \mathbf{H}$.

We are now in a position to define our model. Let λ_{Z_t} be the Poisson rate for generating y_t given the random variable Z_t. The conditional transition probability law (3) can be written as a Bayesian hierarchical model, for $j \in [1, q]$, $H \in \mathbf{H}$ and $t \in [q + 1, T]$, as

$$y_t \mid Z_t = H \sim \text{Poisson}(\lambda_H), \tag{4}$$

$$z_{j,t} \mid y_{t-j} \sim \text{Multinomial}\left([1, k_j], \left(\pi_1^{(j)}(y_{t-j}), \dots, \pi_{k_j}^{(j)}(y_{t-j})\right)\right). \tag{5}$$

Expressions (4) and (5) imply that

$$p(y_t \mid \{y_{t-j}\}_{j\in[1,q]}) = \sum_{H\in\mathbf{H}} \text{PD}(y_t; \lambda_H) \prod_{j\in[1,q]} \pi_{h_j}^{(j)}(y_{t-j}). \tag{6}$$

with constraints $\lambda_H \geq 0$ for any $H \in \mathbf{H}$ and $\sum_{h_j=1}^{k_j} \pi_{h_j}^{(j)}(y_{t-j}) = 1$ for each combination of (j, y_{t-j}). Multinomial$([1, k], \pi)$ is a multinomial distribution selecting a value from $[1, k]$ with a probability vector π. The formulation (6) is referred to as a conditional tensor factorisation with the Poisson density $\text{PD}(y_t; \lambda_H)$ being the core tensor; see (Harshman, 1970; Harshman & Lundy, 1994; Tucker, 1966; De Lathauwer et al., 2000) for a description of tensor factorisations. It can also be interpreted as a Poisson mixture model with $\prod_{j\in[1,q]} \pi_{h_j}^{(j)}(y_{t-j})$ being the mixture weights that depend on previous values of y_t.

A more parsimonious representation for our tensor factorisation model is obtained by adopting a Dirichlet process for Poisson rates λ_H. Independently, for each $H \in \mathbf{H}$, we use the stick-breaking construction introduced by Sethuraman ([1994]) in which

$$\lambda_H \sim \sum_{l=1}^{\infty} \pi_l^* \delta(\lambda_l^*), \tag{7}$$

where $\delta(.)$ is a Dirac delta function and independently, for $l \in [1, \infty)$,

$$\pi_l^* = V_l \prod_{s=1}^{l-1} (1 - V_s), \quad V_l \sim \text{Beta}(1, \alpha_0), \quad \lambda_l^* \sim \text{Gamma}(a, b)$$

where λ_l^* represents a label-clustered Poisson rate. By letting $\mathbf{Z}_{Z_t}^*$ denote the label of the cluster that Z_t belongs to at time $t \in [q+1, T]$, we complete the model formulation as

$$\begin{aligned}
p(\mathbf{Z}_H^* = l) &= \pi_l^*, \quad \text{independently for each } H \in \mathbf{H}, \\
(\lambda_H \mid \mathbf{Z}_H^* = l) &= \lambda_l^*, \\
(\mathbf{Z}_{Z_t}^* \mid Z_t = H) &= \mathbf{Z}_H^*, \\
(y_t \mid \mathbf{Z}_{Z_t}^* = l) &\sim \text{Poisson}(\lambda_l^*).
\end{aligned} \tag{8}$$

2.1.2 Multivariate time series

The model of the previous section can be readily extended to deal with multivariate responses $\{Y_t\}_{t=1}^T$, where $Y_t = (y_{1,t}, \ldots, y_{M,t})^\top$ taking values in $\mathbb{N}_0$. The idea is similar to the way the univariate PAR model (1) is generalised to its multivariate counterpart (2). We assume that the transition probability for any $\tau \in [1, M]$ and $t \in [q+1, T]$ is

$$p(y_{\tau,t} \mid \mathbf{F}_{t-1}) = p(y_{\tau,t} \mid \{y_{m,t-j}\}_{m \in [1,M], j \in [1,q]}). \tag{9}$$

The idea is that each univariate time series may depend on all or some of the q previous values of all, or some, univariate time series. Model (9) assumes that conditional on past q values of all time series before time t, the M univariate random variables at time t are independent. The formulation requires M different latent variables for each dimension but, other than that, its specific details have no essential difference from those in the univariate case.

2.1.3 Two-step inference for large counts

Imagine that based on observed data $\{y_t\}_{t=1}^T$, one has to recursively forecast future observations $y_{T+1}, y_{T+2}, \ldots$. Clearly, the observed values of $\{y_t\}_{t=1}^T$ determine the form of our models in Sections 2.1.1 and 2.1.2 and as a result of this construction we may face the unfortunate situation in which a count that has been unobserved up to time T appears in the future observations. This problem can be solved by re-estimating the model but in cases where this is not desirable, we propose the following solution. We separate $\{y_t\}_{t=1}^T$ into two segments of size T_1 and T_2, representing the size of *pre-training* dataset and *training* dataset, respectively, so $\{y_t\}_{t \in [1, T_1]}$ and $\{y_t\}_{t \in [T_1+1, T_1+T_2]}$ are the corresponding observations in these sets. We aim to use the pre-training dataset to cluster all the counts in time series and the training dataset to model the time series with labelled counts.

We first define a collection of latent variables $\{w_{1:c-1}, \mu_{1:c}, c\}$ that models the pre-training data $\{y_t\}_{t \in [1, T_1]}$ as

$$p(y_t \mid w_{1:c-1}, \mu_{1:c}, c) = \sum_{i=1}^{c} w_i \text{PD}(y_t; \mu_i) \tag{10}$$

for any $t \in [1, T_1]$, $0 < w_i < 1$, $\sum_{i=1}^{c} w_i = 1$, $\mu_i \geq 0$. Thus, (10) assumes that any y_t in the pre-training dataset is distributed as a finite mixture of Poisson distributions with c components, weights w_i and intensities μ_i. The usual latent structure for such mixture models assumes indicator variables d_t representing the estimated label of the mixture component that y_t belongs to, so $p(d_t = i) = w_i$ for all $i \in [1, c]$.

We exploit this finite mixture clustering of the pre-training dataset to build our model for the training dataset. We define another collection of latent variables as $D_t = \{d_{j,t}\}_{j \in [1,q]}$ and by setting $d_{j,t} = d_{t-j}$ for all $j \in [1, q]$ and $t \in [T_1 + 1 + q, T_1 + T_2]$. We then build a probabilistic model for the training dataset by assuming that the transition probability law of the sequence $\{y_t\}_{t \in [T_1 + 1 + q, T_1 + T_2]}$ conditional on $\mathbf{F}_{t-1}$ is that of a probabilistic model of this target sequence conditional on D_t. That is, we have

$$p(y_t \mid \mathbf{F}_{1:t-1}) = p(y_t \mid D_t). \tag{11}$$

The conditional transition probability law (11) can then be written as a Bayesian hierarchical model, for $j \in [1, q]$ and $t \in [T_1 + 1 + q, T_1 + T_2]$, as

$$y_t \mid Z_t = H \sim \text{Poisson}(\lambda_H), \tag{12}$$

$$z_{j,t} \mid d_{j,t} \sim \text{Multinomial}\left([1, k_j], \{\pi_1^{(j)}(d_{j,t}), \dots, \pi_{k_j}^{(j)}(d_{j,t})\}\right). \tag{13}$$

(12) and (13) immediately imply that

$$p(y_t \mid D_t) = \sum_{H \in \mathbf{H}} \text{PD}(y_t; \lambda_H) \prod_{j \in [1,q]} \pi_{h_j}^{(j)}(d_{j,t}). \tag{14}$$

with constraints $\lambda_H \geq 0$ for any $H \in \mathbf{H}$ and $\sum_{h_j=1}^{k_j} \pi_{h_j}^{(j)}(d_{j,t}) = 1$ for each combination of $(j, d_{j,t})$. It is clear that (14) is equivalent to (12) and (13). From (14) the expectation of y_t conditional on D_t is

$$\mathbb{E}(y_t \mid D_t) = \sum_{H \in \mathbf{H}} \lambda_H \prod_{j \in [1,q]} \pi_{h_j}^{(j)}(d_{j,t}). \tag{15}$$

The rest of the model which utilises the stick-breaking process for λ_H is similar to the one used in Sect. 2.1.1.

2.1.4 Priors

We assign independent priors on $\pi^{(j)}(d_{j,t})$ as

$$\pi^{(j)}(d_{j,t}) = \{\pi_1^{(j)}(d_{j,t}), \dots, \pi_{k_j}^{(j)}(d_{j,t})\} \sim \text{Dirichlet}(\gamma_j, \dots, \gamma_j),$$

with $\gamma_j = 0.1$. Also, we follow (Sarkar & Dunson, 2016) and set priors

$$p(k_j = \kappa) \propto \exp(-\varphi j \kappa),$$

where $j \in [1, q]$, $\kappa \in [1, c]$. Notice that φ controls $p(k_j = \kappa)$ and the number of important lags for the proposed conditional tensor factorisation; for all our experiments throughout this paper, we set $\varphi = 0.5$. Following (Viallefont et al., 2002), we place for the Gamma density of λ_l^* parameters a as the mid-range of y_t in the training dataset $a = \frac{1}{2}[\max(\{y_t\}_{t \in [T_1+1+q, T_1+T_2]}) - \min(\{y_t\}_{t \in [T_1+1+q, T_1+T_2]})]$ and $b = 1$. We set $\alpha_0 = 1$ for the Beta prior to V_l. Finally, we truncate the series (7), by assuming

$$\lambda_H \sim \sum_{l=1}^{L} \pi_l^* \delta(\lambda_l^*),$$

and we set $L = 100$.

3 Estimation and inference

The joint density of the general model of Section 2.2.3 can be expressed as $p(y, Z, \mathbf{Z}^*, D, \lambda^*, \pi^*, \pi_K)$, where $D = \{D_t\}_{t \in [T_1+1, T_1+T_2]}$ and $K = \{k_j\}_{j \in [1,q]}$. The Poisson mixture model in the pre-training set is estimated with the MCMC algorithm of Marin et al. (2005). For any $t > T_1$ we then estimate $d_t = \arg_i \max \mathrm{PD}(y_t, \mu_i)$, $i \in [1, c]$. Our BTF model has a finite number of mixture components with an unknown number of components due to the randomness of the random variable matrix K. We follow (Yang & Dunson, 2016) and estimate K separately through a stochastic search variable selection (George & McCulloch, 1997) based on approximated marginal likelihood. As (Yang & Dunson, 2016) point out, such an approach is helpful since it fixes the numbers of inclusions of the tensor and the sampling process of K can indicate whether a predictor is important. The rest of the inference proceeds by sampling all other random variables conditional on K and D through MCMC.

3.1 MCMC for finite Poisson mixtures

We follow the procedure in Marin et al. (2005). $\{y_t\}_{t \in \mathbb{Z}_{[1,T_1]}}$ is a mixture of c univariate Poisson distributions with density $\sum_{i=1}^{c} w_i \mathrm{PD}(y_t; \mu_i)$, $\{w_i\}_{i \in \mathbb{Z}_{[1,c]}}$ are weights with $\sum_{i=1}^{c} w_i = 1$ and $\{\mu_i\}_{i \in \mathbb{Z}_{[1,c]}}$ are the corresponding Poisson rates. By setting the priors as $\mu_i \sim \mathrm{Gamma}(1, 1)$, $\{w_i\}_{i \in \mathbb{Z}_{[1,c]}} \sim \mathrm{Dirichlet}(1, \dots, 1)$, the corresponding Gibbs sampler is as follows: (i) Generate the label of y_t, ι_t, for $t \in \mathbb{Z}_{[1,T_1]}$, $i \in \mathbb{Z}_{[1,c]}$ as $p(\iota_t = i) \propto w_i (\mu_i)^{y_t} \exp(-\mu_i)$ and set $n_i = \sum_{t \in \mathbb{Z}_{[1,T_1]}} \mathbb{1}_{\iota_t = i}$ and $\mathbf{I}_i = \sum_{t \in \mathbb{Z}_{[1,T_1]}} \mathbb{1}_{\iota_t = i} y_t$ (ii) Generate $\{w_i\}_{i \in \mathbb{Z}_{[1,c]}} \sim \mathrm{Dirichlet}(1 + n_1, \dots, 1 + n_c)$ and (iii) for $i \in \mathbb{Z}_{[1,c]}$, generate $\mu_i \sim \mathrm{Gamma}(1 + \mathbf{I}_i, 1 + n_i)$.

3.2 Important lags selection

Important lags are inferred by the variable $K = \{k_j\}_{j \in \mathbb{Z}_{[1,q]}}$. The basic calculations are as follows. Following (Sarkar & Dunson, 2016), the posterior of $K = \{k_j\}_{j \in \mathbb{Z}_{[1,q]}}$ can be sampled as

$$p(k_j| \dots) \propto \exp(-\varphi j k_j) \prod_{\omega=1}^{c} \frac{\Gamma(k_j \gamma_j)}{\Gamma(k_j \gamma_j + n_{j,\omega})}$$

with $k_j = \max\left(\{z_{j,t}\}_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}}\right), \dots, c$ and $n_{j,\omega} = \sum_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} \mathbb{1}\{d_{j,t} = \omega.\}$ The levels of $d_{j,t}$ are partitioned into k_j clusters $\{C_{j,r} : r = 1, \dots, k_j\}$ with each cluster $C_{j,r}$ assumed to correspond to its own latent class $h_j = r$. With independent Dirichlet priors on the mixture kernels $\lambda_H \sim \text{Gamma}(a, b)$ marginalised out, the likelihood of our targeted response $\{y_t\}_{t \in \mathbf{T}_2^*}$ conditional on the cluster configuration $C = \{C_{j,r} : j \in \mathbb{Z}_{[1,q]}, r \in \mathbb{Z}_{[1,k_j]}\}$ is given by

$$p(\{y_t\}_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} \mid C) = \prod_{H \in \mathbf{H}} \int_0^\infty f(\{y_t\}_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} \mid \lambda_H) p(\lambda_H \mid C) d\lambda_H$$

$$= \prod_{H \in \mathbf{H}} \int_0^\infty \left(\prod_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} (y_t \xi)! \right)^{-1} \exp\left(-(\sum_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} \xi)\lambda_H \right) \cdot$$

$$\lambda_H^{\sum_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} y_t \xi} \frac{1}{(1/b)^a \Gamma(a)} \lambda_H^{a-1} \exp(-\lambda_H b) d\lambda_H$$

$$= \prod_{H \in \mathbf{H}} \frac{1}{(1/b)^a \Gamma(a)} \left(\prod_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} (y_t \xi)! \right)^{-1} \Gamma\left(a + \sum_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} y_t \xi \right) \cdot$$

$$\left(\sum_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} \xi + b \right)^{-(a + \sum_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}} y_t \xi)},$$

where $\xi = \mathbb{1}\{d_{1,t} \in C_{1,h_1}, \dots, d_{q,t} \in C_{q,h_q}\}$. Then the MCMC steps for $j \in \mathbb{Z}_{[1,q]}$ are: (i) If $1 \le k_j \le c$, we propose to either increase k_j to $(k_j + 1)$ or decrease k_j to $(k_j - 1)$. (ii) If an increasing move is proposed, we randomly split a cluster of $d_{j,t}$ into two clusters. We accept this move with an acceptance rate based on the approximated marginal likelihood. (iii) If a decrease move is proposed, we randomly merge two clusters of $d_{j,t}$ into a single cluster. We accept this move with an acceptance rate based on the approximated marginal likelihood. If K^* and C^* are the updated model index and cluster, $\alpha(\cdot;\cdot)$ is the Metropolis-Hastings acceptance rate, $L(\cdot)$ is the likelihood function and $q(\cdot \to \cdot)$ is the proposal function, we obtain

$$\alpha(K, C; K^*, C^*) = \frac{L(\{y_t\}_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}}, K^*, C^*) q(K^*, C^* \to K, C)}{L(\{y_t\}_{t \in \mathbb{Z}_{[T_1+1+q,T_2]}}, K, C) q(K, C \to K^*, C^*)}.$$

3.3 Full conditional densities

For given D and K, denote by ζ a generic variable that collects the variables that are not explicitly mentioned, including y. Then the corresponding Gibbs sampling steps are

- Sample $\mathbf{Z}_H^*$ for each $H \in \mathbf{H}$ from $p(\mathbf{Z}_H^* = l \mid \zeta) \propto \pi_l^*(\lambda_l^*)^{n_H^*} \exp\left(-n_H \lambda_l^*\right)$ where $n_H^* = \sum_{t \in \mathbb{Z}_{[T_1+1+q,T_1+T_2]}} \mathbb{1}\{Z_t = H\} y_t$ and $n_H = \sum_{t \in \mathbb{Z}_{[T_1+1+q,T_1+T_2]}} \mathbb{1}\{Z_t = H\}$.
- Sample V_l for $l \in \mathbb{Z}_{[1,L]}$ from $V_l \mid \zeta \sim \text{Beta}\left(1 + \mathbf{N}_l^*, \alpha_0 + \sum_{l'>l} \mathbf{N}_{l'}^*\right)$ where $\mathbf{N}_l^* = \sum_{H \in \mathbf{H}} \mathbb{1}\{\mathbf{Z}_H^* = l\}$, and update π_l^* accordingly.
- Sample each λ_l^* with $l \in \mathbf{L}$ from $\lambda_l^* \mid \zeta \sim \text{Gamma}\left(a + N_H^*(l), b + N_H(l)\right)$, where $N_H^*(l) = \sum_{H \in \mathbf{H}} \mathbb{1}\{\mathbf{Z}_H^* = l\} n_H^*$ and $N_H(l) = \sum_{H \in \mathbf{H}} \mathbb{1}\{\mathbf{Z}_H^* = l\} n_H$.
- For $j \in \mathbb{Z}_{[1,q]}$ and $\omega \in \mathbb{Z}_{[1,c]}$, sample

$$\left\{\pi_1^{(j)}(\omega), \ldots, \pi_{k_j}^{(j)}(\omega)\right\} \mid \zeta \sim \text{Dirichlet}\{\gamma_j + n_{j,\omega}(1), \ldots, \gamma_j + n_{j,\omega}(k_j)\}$$

  where $n_{j,\omega}(h_j) = \sum_{t \in \mathbb{Z}_{[T_1+1+q,T_1+T_2]}} \mathbb{1}\{z_{j,t} = h_j, d_{j,t} = \omega\}$.
- Sample $z_{j,t}$ for $j \in \mathbb{Z}_{[1,q]}$ and $t \in \mathbb{Z}_{[T_1+1+q,T_1+T_2]}$ from

$$p(z_{j,t} = h | z_{j',t} = h_{j'}, j' \neq j, \zeta) \propto \pi_h^{(j)}(d_{j,t}) \left(\lambda_{\mathbf{Z}_{H_{\ldots/j=h}}^*}^*\right)^{y_t} \exp\left(-\lambda_{\mathbf{Z}_{H_{\ldots/j=h}}^*}^*\right),$$

  where $H_{\ldots/j=h}$ is equal to H at all position except the j-th position taking the value h.

4 Simulation experiments

We tested our methodology with simulated data from designed experiments against the Poisson autoregressive model (1) through the log predictive score calculated in an out-of-sample (test) dataset $\mathfrak{T}$ of size $\tilde{T}$. For each model the log predictive score is estimated by

$$\frac{-\sum_{t \in \mathfrak{T}} \sum_{i=1}^N \log \hat{p}^{(i)}(y_t)}{\tilde{T} N}$$

where $\hat{p}^{(i)}(y_t)$ denotes the one-step ahead estimated transition probability of observing $y_{t \in \mathfrak{T}}$ calculated using the parameter values at the i-th iteration of MCMC with total N iterations. It measures the predictive accuracy of the model by assessing the quality of the uncertainty quantification. A model predicts better when the log predictive score is smaller; see, for example, Czado et al. (2009). For each designed scenario, we generated 10 datasets with 5, 000 data points and out-of-sample predictive performance for all models was tested by using either the first 4, 000 or 4, 500 data points as training datasets and calculating the log predictive scores approximated via the MCMC output at the rest 1, 000 or 500 test data points respectively. The resulting mean log predictive score that is reported in Tables 1, 2, 3 is the average log predictive score across the 10 generated datasets. The pre-training dataset for the BTF model has been chosen to be the first 3, 000 points. All MCMC runs were based on the following burn-in and posterior samples respectively: 2, 000 and 5, 000 for fitting the Poisson mixtures on the pre-training dataset; 1, 000 and 2, 000 for selecting the important lags and their corresponding number of inclusions; and 2, 000 and 5, 000 for sampling the rest of the parameters. Bayesian inference for Poisson autoregressive model

Table 1 Mean log predictive scores (with standard deviations in brackets) for Bayesian Poisson autoregressive models and our Bayesian tensor factorisations model (BTF) based on 10 Poisson autoregression generated data sets for each one of 6 Scenarios

Scenarios	Data sizes	Bayesian Poisson autoregression		BTF
		AIC	BIC	
$(A): \beta_0 = 1, \beta_1 = 0.5$	4000 : 1000	**2.436(0.024)**	**2.436(0.024)**	2.443(0.022)
	4500 : 500	**2.441(0.031)**	**2.441(0.031)**	2.450(0.030)
$(B): \beta_0 = 1, \beta_7 = 0.5$	4000 : 1000	2.450(0.019)	**2.449(0.019)**	2.458(0.022)
	4500 : 500	2.454(0.028)	**2.452(0.031)**	2.463(0.030)
$(C): \beta_0 = 1, \beta_{29} = 0.7$	4000 : 1000	3.126(0.018)	3.126(0.018)	**3.108(0.014)**
	4500 : 500	3.123(0.024)	3.123(0.024)	**3.106(0.021)**
$(D): \beta_0 = 1, \beta_1 = -0.5, \beta_7 = 0.5$	4000 : 1000	**1.870(0.016)**	**1.870(0.016)**	1.882(0.024)
	4500 : 500	**1.876(0.020)**	**1.876(0.020)**	1.885(0.017)
$(E): \beta_0 = 1, \beta_{19} = -0.5, \beta_{29} = 0.5$	4000 : 1000	1.873(0.015)	1.873(0.015)	**1.857(0.017)**
	4500 : 500	1.869(0.018)	1.869(0.018)	**1.852(0.020)**
$(F): \beta_0 = 1, \beta_1 = -0.5, \beta_7 = -0.5, \beta_{19} = 0.5$	4000 : 1000	1.683(0.013)	1.683(0.013)	**1.631(0.009)**
	4500 : 500	1.689(0.017)	1.689(0.017)	**1.635(0.012)**

AIC and BIC columns indicate that the best model has been chosen with the corresponding criterion. Models with the best performance are highlighted in bold

Table 2 Mean log predictive scores (with standard deviations in brackets) for Bayesian Poisson autoregressive models and our Bayesian tensor factorisations model (BTF) based on 10 nonlinear generated data sets for each one of 6 Scenarios

Scenarios	Data sizes	Bayesian Poisson autoregression		BTF
		AIC	BIC	
$(A): \nu_+ = 30, \nu_- = 50$	4000 : 1000	**3.860(0.032)**	**3.860(0.032)**	3.956(0.251)
Important lag: y_{t-1}	4500 : 500	**3.869(0.029)**	**3.869(0.029)**	3.982(0.181)
$(B): \nu_+ = 30, \nu_- = 50$	4000 : 1000	3.892(0.056)	3.890(0.055)	**3.691(0.155)**
Important lag: y_{t-7}	4500 : 500	3.897(0.064)	3.897(0.064)	**3.724(0.173)**
$(C): \nu_+ = 20, \nu_- = 100$	4000 : 1000	3.615(0.207)	3.615(0.207)	**3.437(0.078)**
Important lags: y_{t-3}, y_{t-7}	4500 : 500	3.665(0.225)	3.668(0.222)	**3.448(0.113)**
$(D): \nu_+ = 20, \nu_- = 100$	4000 : 1000	3.857(0.172)	3.858(0.172)	**3.489(0.088)**
Important lags: y_{t-7}, y_{t-9}	4500 : 500	3.822(0.192)	3.820(0.187)	**3.470(0.102)**
$(E): \nu_+ = 20, \nu_- = 100$	4000 : 1000	3.426(0.030)	3.426(0.030)	**3.380(0.057)**
Important lags: $y_{t-3}, y_{t-7}, y_{t-9}$	4500 : 500	3.440(0.023)	3.441(0.024)	**3.396(0.089)**
$(F): \nu_+ = 20, \nu_- = 100$	4000 : 1000	5.338(0.092)	5.338(0.092)	**3.772(0.130)**
Important lags: $y_{t-7}, y_{t-8}, y_{t-9}$	4500 : 500	5.270(0.120)	5.270(0.120)	**3.692(0.164)**

AIC and BIC columns indicate that the best model has been chosen with the corresponding criterion. Models with the best performance are highlighted in bold

Table 3 Mean log predictive scores (with standard deviations in brackets) for Bayesian Poisson autoregressive models and our Bayesian tensor factorisations model (BTF) based on 10 nonlinear generated data sets for each one of 6 Scenarios

Scenarios and non-zero coefficients	Data sizes	Bayesian Poisson autoregression		BTF
		AIC	BIC	
(A) : $M = 2$; $v_- = 20, v_+ = 10$;	4000 : 1000	3.225(0.029)	3.225(0.029)	**3.013(0.037)**
Non-zero coefficients for $y_{1,t}$: $y_{1,t-1}, y_{2,t-1}$;				
No non-zero coefficient for $y_{2,t}$	4500 : 500	3.243(0.057)	3.243(0.057)	**3.110(0.033)**
(B) : $M = 2$; $v_- = 20, v_+ = 10$;	4000 : 1000	2.993(0.030)	2.993(0.030)	**2.705(0.033)**
Non-zero coefficients for $y_{1,t}$: $y_{1,t-3}, y_{2,t-5}$;				
No non-zero coefficient for $y_{2,t}$	4500 : 500	3.002(0.058)	3.003(0.058)	**2.711(0.040)**
(C) : $M = 2$; $v_- = 20, v_+ = 10$;	4000 : 1000	3.488(0.047)	3.487(0.47)	**2.877(0.021)**
Non-zero coefficient for $y_{1,t}$: $y_{2,t-1}$;				
Non-zero coefficient for $y_{2,t}$: $y_{1,t-2}$	4500 : 500	3.452(0.059)	3.452(0.059)	**2.843(0.024)**
(D) : $M = 2$; $v_- = 20, v_+ = 10$;	4000 : 1000	3.207(0.039)	3.206(0.039)	**2.855(0.026)**
Non-zero coefficients for $y_{1,t}$: $y_{1,t-3}, y_{2,t-4}$;				
Non-zero coefficients for $y_{2,t}$: $y_{1,t-1}, y_{2,t-3}, y_{2,t-5}$	4500 : 500	3.159(0.044)	3.159(0.044)	**2.797(0.029)**
(E) : $M = 3$; $v_- = 20, v_+ = 10$;	4000 : 1000	3.632(0.052)	3.632(0.052)	**2.903(0.033)**
Non-zero coefficient for $y_{1,t}$: $y_{2,t-1}$;				
Non-zero coefficient for $y_{2,t}$: $y_{3,t-2}$;	4500 : 500	3.622(0.044)	3.622(0.044)	**2.772(0.030)**
Non-zero coefficient for $y_{3,t}$: $y_{1,t-3}$				
(F) : $M = 3$; $v_- = 60, v_+ = 20$;	4000 : 1000	6.117(0.149)	6.117(0.149)	**3.508(0.227)**
Non-zero coefficients for $y_{1,t}$: $y_{1,t-3}, y_{2,t-4}, y_{3,t-1}$;				
Non-zero coefficients for $y_{2,t}$: $y_{1,t-1}, y_{2,t-2}, y_{3,t-5}$;	4500 : 500	6.306(0.202)	6.306(0.202)	**3.574(0.173)**
Non-zero coefficients for $y_{3,t}$: $y_{1,t-3}, y_{2,t-2}, y_{3,t-5}$				

AIC and BIC columns indicate that the best model has been chosen with the corresponding criterion. Models with the best performance are highlighted in bold

was obtained by 'rjags' (Plummer et al., 2016) package based on 5,000 burn-in and 10,000 MCMC samples respectively. We first chose the order q of the model by choosing among all models with maximum order up to $q + 2$ using the AIC and BIC criteria. We set the priors for parameters as $\beta_0 \sim N(0, 10^{-6})$ and $\beta_i \sim N(0, 10^{-4})$ for any $i \in [1, q]$.

Table 1 presents the results of out-of-sample comparative predictive ability based on six generated Poisson autoregressive models based on (1). Notice that when the order q is high and there are only a few true coefficients, as in cases C, E and F, the maximal order Markov structure of the BTF model achieves a comparative, satisfactory predictive performance. Given that the data generating process is based on Poisson autoregressive models these results are very promising.

Next, we generated data in which past values affect current random variables in a nonlinear fashion as follows. There are $\mathbf{K}$ important lag(s) $\{y_{t-i_1}, \ldots, y_{t-i_{\mathbf{K}}}\}$ and, for given v_+, v_-, if $\sum_{j=1}^{\mathbf{K}} y_{t-i_j} \geq \mathbf{K} v_+$, then $y_t \sim \text{Poisson}(v_+)$; else $y_t \sim \text{Poisson}(v_-)$. We designed 6 scenarios and the results are shown in Table 2. Our proposed modelling formulation outperforms the Bayesian Poisson autoregressive model in all but one scenario.

Finally, we replicated the last exercise by testing the models in a more challenging data generation mechanism in which the response is multivariate. We designed 6 different

scenarios by generating an M-dimensional time series $\{y_{m,t}\}_{m\in[1,M]}$ and assuming that we are interested in predicting $y_{1,t}$. For $t \leq 10$, we generated $y_{m,t}$ from $\text{Pois}(v_-)$ for each m; for $t > 10$, if $\sum_{i=1}^{K} y_{m_i,t-j_i} \geq v_-$ we generate $y_{1,t} \sim \text{Poisson}(v_+)$, else $y_{1,t} \sim \text{Poisson}(v_-)$. We fitted an M-dimensional multivariate Poisson autoregressive model of order q that predicts $y_{\ell,t}$ with covariates $\{y_{m,t-1}\}_{m\in M, m\neq\ell}$ as

$$y_{\ell,t} \sim \text{Poisson}(\lambda_{\ell,t}),$$

$$\log(\lambda_{\ell,t}) = \beta_{\ell,0} + \sum_{i=1}^{q} \beta_{\ell,i} \log(y_{\ell,t-i} + 1) + \sum_{m\neq\ell} \zeta_{\ell,m} y_{m,t-1} \tag{16}$$

where $\beta_{\ell,0}$, $\beta_{\ell,i}$ and $\zeta_{\ell,m}$ are unknown parameters. Table 3 shows that for all 6 Scenarios, the Bayesian tensor factorisation model achieves impressively better predictive performance than the Bayesian Poisson autoregressive model.
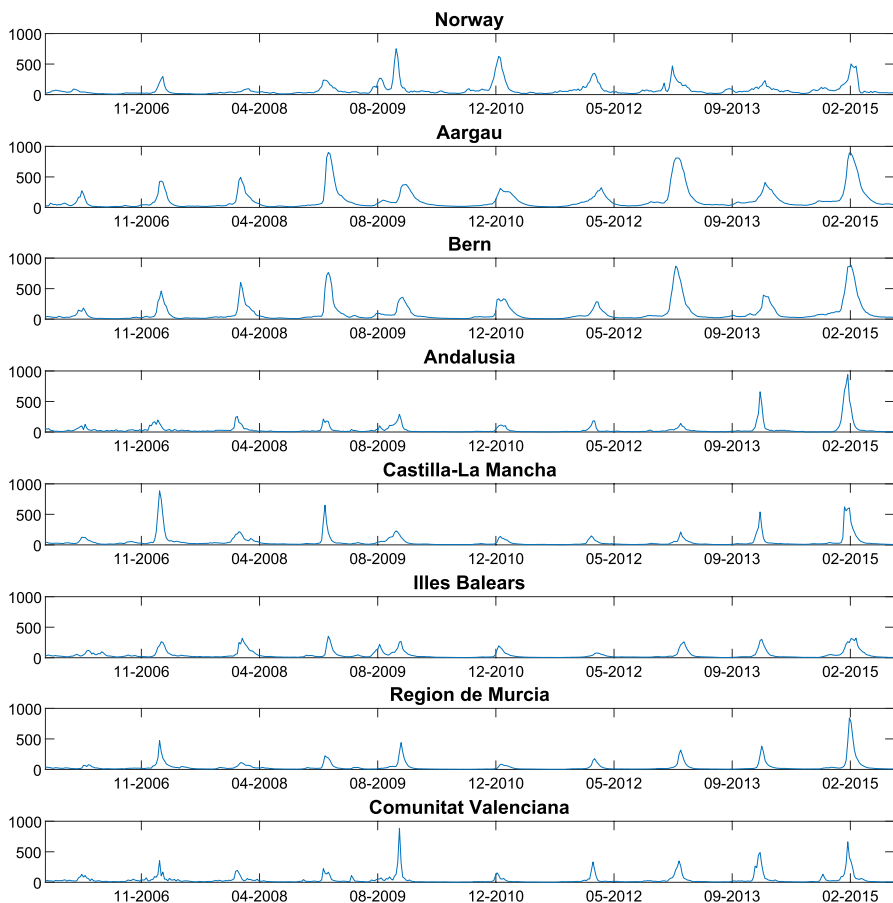
Fig. 1 Trace plot of 514 time-series data points counting flu cases in Norway, Aargau as well as Bern in Switzerland and five regions in eastern Spain including Andalusia, Castilla-La Mancha, Illes Balears, Region de Murcia and Valencian Community counted by each week from 09-Oct-2005 to 09-Aug-2015

The times needed to run the MCMC algorithms for Bayesian Poisson autoregressive and BTF models are comparable. For 1000 iterations we needed, on average, 20 s for the BTF model implemented with our matlab code and 25 s for the Bayesian Poisson autoregressive models implemented with rjags.

5 Applications

5.1 Univariate flu data

We compared our Bayesian tensor factorisation model to Bayesian Poisson autoregressive model with two datasets from Google Flu Trends that refer to 514 Norway, Switzerland and Castilla-La Mancha weekly flu counts in Spain, see Fig. 1. We chose the maximum lag q to be 10 for all models we applied to the data. We examined the sensitivity to the size of the pre-training data by considering three scenarios. We used 103(20%), 154(30%) and 206(40%) pre-training sizes and compared their predictive ability against the best models for Bayesian Poisson autoregression formulations based on AIC and BIC criteria. The last 103 and 52 data points were chosen for out-of-sample test comparison for each dataset. To demonstrate how our methodology works, we will present MCMC results for the Norway dataset based on 154 training points; results for both datasets and for all training sizes are given at the end of the Section.

The pre-training results are illustrated in Fig. 2. There are barely significant differences among 6 of the 10 clusters in the left panel so we fix the number of clusters to be 5, see

(a) Total number of clusters to be 10

(b) Total number of clusters to be 5
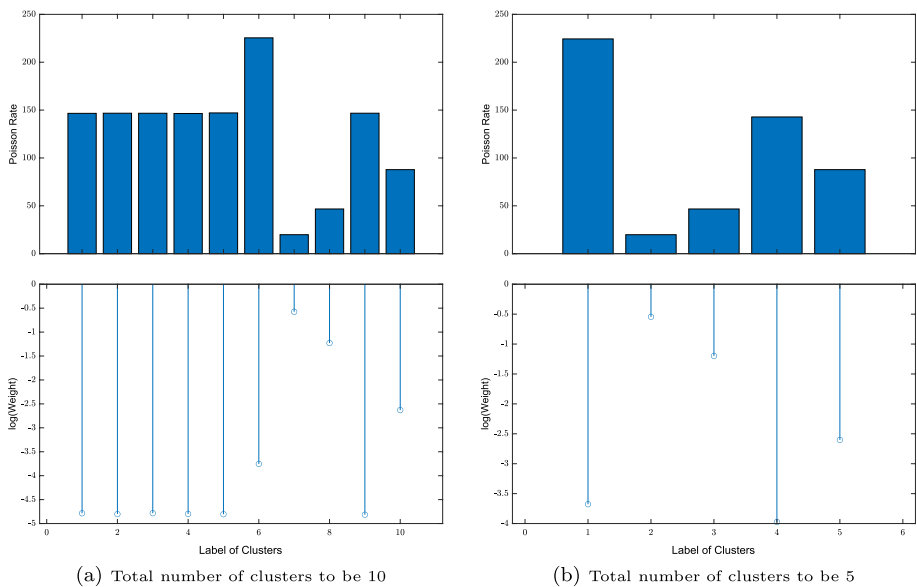
Fig. 2 Fitting of a mixture of Poisson distributions. The dataset used is the pre-training data from flu cases in Norway counted by each week from 09-Oct-2005 to 09-Aug-2015. Panels **a** and **b** indicate that the outcome for a total number of clusters c are 10 and 5 respectively. The top panels illustrate the Poisson rates of their corresponding label of clusters, whilst the bottom panels show their corresponding log weights
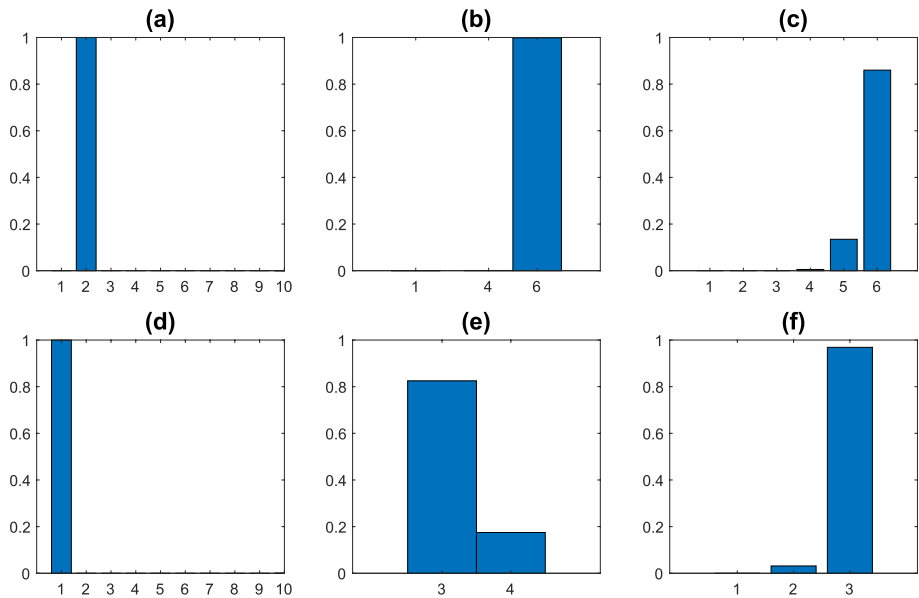
Fig. 3 MCMC frequency results. In all panels, the *x*-axis represents the number and the *y*-axis does the relative frequency. Top three panels: 411 training data points; bottom three panels: 462 training data points. **a**, **d**: The relative frequency distributions for the number of important predictor(s). **b**, **e**: The relative frequency distributions of $\prod_{j=1}^{q} k_j$, or the total number of possible combinations of $(h_1, \ldots, h_q)$. **c**, **f**: The relative frequency distributions of the number of unique $\lambda_{h_1, \ldots, h_q}$

Fig. 2. Figure 3 shows some MCMC results for the rest of our Bayesian tensor factorisation model. With 411 training data points, Panels (a),(b) and (c) provide strong evidence that there are two important predictors, 6 possible combinations of $(h_1, ..., h_q)$ and 6 unique $\lambda_{h_1, \ldots, h_q}$. Similarly, when the length of the training dataset is 462 panels (d),(e) and (f) indicate that there is evidence for only one important predictor, the total number of possible combinations of $(h_1, ..., h_q)$ is either 3 or 4, and that there are 3 unique $\lambda_{h_1, \ldots, h_q}$.

Model selection results for the Poisson autoregression models are illustrated in Fig. 4. MCMC was based on 5,000 burn-in and 10,000 runs by using 'rjags', see (Plummer et al., 2016). For the resulting parameter estimates see Table 4.

Table 5 indicates that in all pre-training size scenarios BTF outperform, in terms of predictive ability expressed with log predictive scores, Bayesian Poisson autoregression models. There is clearly a trade-off between good mixture estimation and adequate training size that is expressed in small and large pre-training sizes respectively. In our small empirical study it seems that there is evidence for some robustness in the inference procedure when the pre-training size is small, since 103 points outperform 206 points with the 154 points being the best performing pre-training size. The predictive means and 95% credible intervals of BTF and of the PAR(5) model that had one of the best predictive performances based on 103 test data are depicted in Fig. 5.

The average run times for the MCMC algorithms for BTF and the Bayesian Poisson autoregression models are comparable. For the former, 1000 iterations take approximately
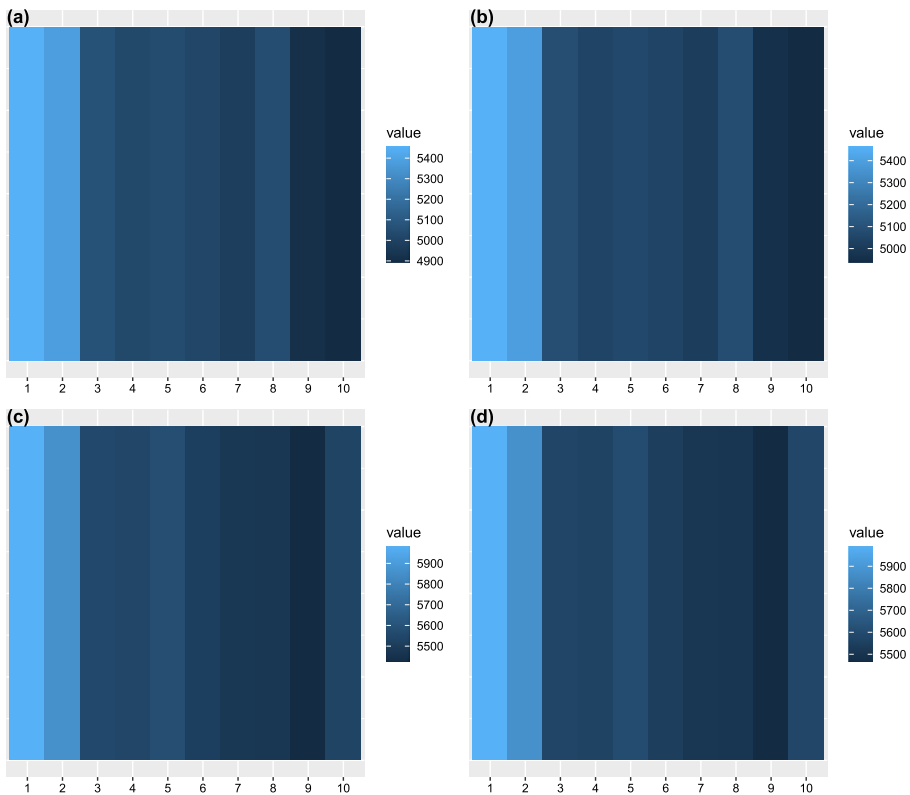
Fig. 4 AIC and BIC scores given by PAR(q) models with q labelled in the x-axis for flu cases in Norway counted by each week from 09-Oct-2005 to 09-Aug-2015. **a**: The AIC scores for the scenario with 411 training data points and 103 testing data points; **b**: The BIC scores for the scenario with 411 training data points and 103 testing data points; **c**: The AIC scores for the scenario with 462 training data points and 52 testing data points; **d**: The BIC scores for the scenario with 462 training data points and 52 testing data points

20 s with our code written in matlab, whereas the latter takes approximately 25 s for 1000 iterations in the R package 'rjags'.

5.2 Multivariate flu data

We revisit the flu data of the previous subsection by jointly modelling flu cases in (i) the adjacent Swiss cantons of Bern and Aargau and (ii) in five neighbouring regions in south-eastern Spain, namelyAndalusia, Castilla-La Mancha, Illes Balears, Region de Murcia and Valencian Community. The data are depicted in Fig. 1 and consist of 514 weekly counts from 09-Oct-2005 to 09-Aug-2015.

We chose the maximum lag q to be ten for all multivariate BTF models we applied to the data. The sizes of training against the testing dataset are 411 : 103 and 462 : 52 respectively. Our BTF considered the first 154 data points as the pre-training dataset.

Table 4 Means of coefficients (with standard deviations in brackets) based on 5,000 burn-in and 10,000 MCMC runs

Sizes	411 : 103	462 : 52
Model selected	PAR(10)	PAR(9)
Coefficient		
β_0	0.252(0.035)	0.225(0.033)
β_1	1.472(0.016)	1.547(0.023)
β_2	−0.395(0.019)	−0.634(0.038)
β_3	−0.172(0.027)	−0.048(0.029)
β_4	−0.005(0.022)	0.071(0.021)
β_5	−0.006(0.028)	0.103(0.021)
β_6	0.050(0.023)	−0.204(0.017)
β_7	−0.111(0.028)	0.018(0.021)
β_8	0.168(0.029)	0.196(0.018)
β_9	0.111(0.029)	−0.105(0.014)
β_{10}	−0.179(0.025)	

Two scenarios with different sizes of training against testing data are shown in each columns with their corresponding selected models indicated

Table 5 Log predictive scores for Bayesian Poisson autoregression models and our Bayesian tensor factorisations model (BTF) for flu counts datasets in Norway and Castilla-La Mancha, Spain

Country/region	Data sizes	Poisson autoregression		BTF		
		AIC	BIC	103 PTDPs	154 PTDPss	206 PTDPs
Norway	411 : 103	7.560(10)	7.560(10)	6.054	**5.846**	6.440
	462 : 52	7.805(9)	7.805(9)	6.110	**6.079**	6.289
Castilla-La Mancha, Spain	411 : 103	12.295(10)	12.607(5)	5.667	**5.416**	6.760
	462 : 52	15.073(9)	15.073(9)	5.858	**5.664**	6.268

The BTF model has performed with 103, 154 and 206 pre-training data points (PTDPs). AIC and BIC columns indicate that the best model has been chosen (in brackets) with the corresponding criterion. Training and testing data sizes appear in the second column. Models with the best performance are highlighted in bold

Figures 6, 7, 8, 9 illustrate how lags were selected in each real data application. Note that a lag is considered to be important, and thus is selected, when its corresponding relative frequency distribution is higher than 0.5.

The predictive ability of the models compared to the Bayesian Poisson autoregression models are given in Tables 6 and 7. In the Swiss cantons it seems that the BTF model underperforms when Aargau flu cases are predicted from past flu cases of Aargu and Bern, whereas it outperforms when we predict Bern cases based on past data from Aargau and Bern. An informal justification of this behaviour is that from the data it seems that the two series have very high positive contempoaneous and lag-one correlations so naturally the model (16) that captures very well these linear dependencies outperforms our model. Such situations are expected when a general non-parametric model is compared to a linear model with the corresponding data generating mechanism to be primarily linear-based.
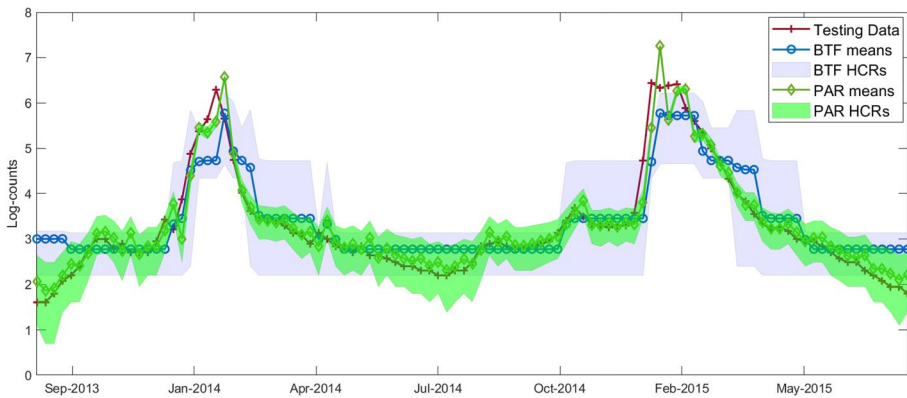
Fig. 5 Out of sample predictive means and 95% highest credible regions (HCRs) of Bayesian tensor factorisations (BTF) and Poisson autoregressive models (PAR) compared against the Castilla-La Mancha data. The sizes of training and testing data are 411 and 103 respectively
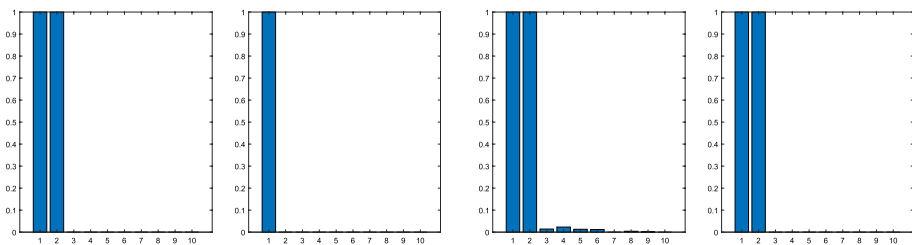

Fig. 6 Lag selection for the Norway (left pair) and the Castilla-La Mancha (right pair) flu datasets. Each pair of figures represents (i) the inclusion proportions (y-axis) of different lags (x-axis) for the scenario with 411 training and 103 testing data points and (ii) the inclusion proportions (y-axis) of different lags (x-axis) for the scenario with 462 training data points and 52 testing data points
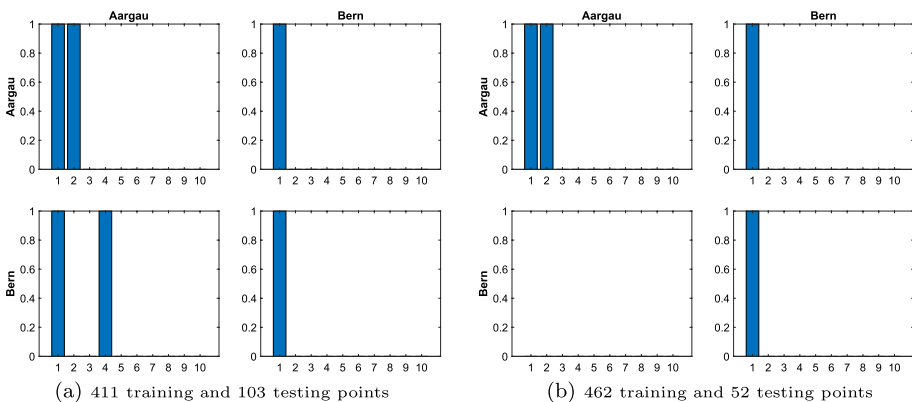
Fig. 7 Important lag selection for the Swiss flu dataset. Y-axis represents the inclusion proportions of different lags in x-axis
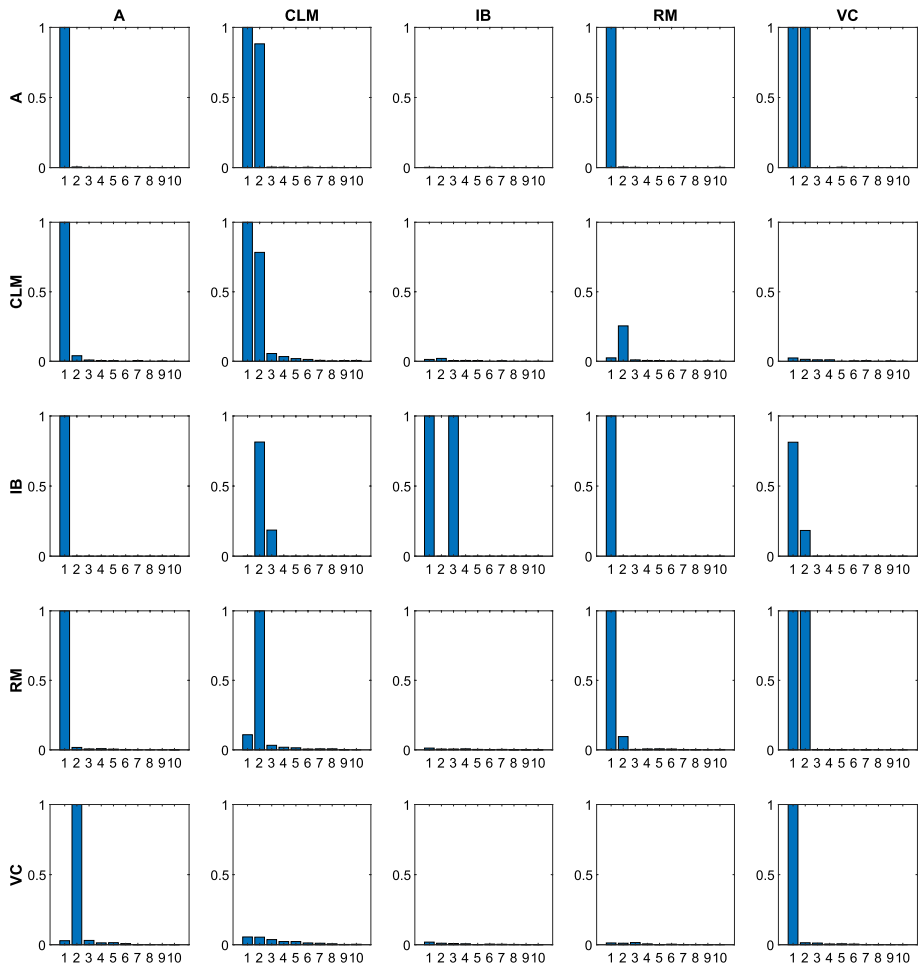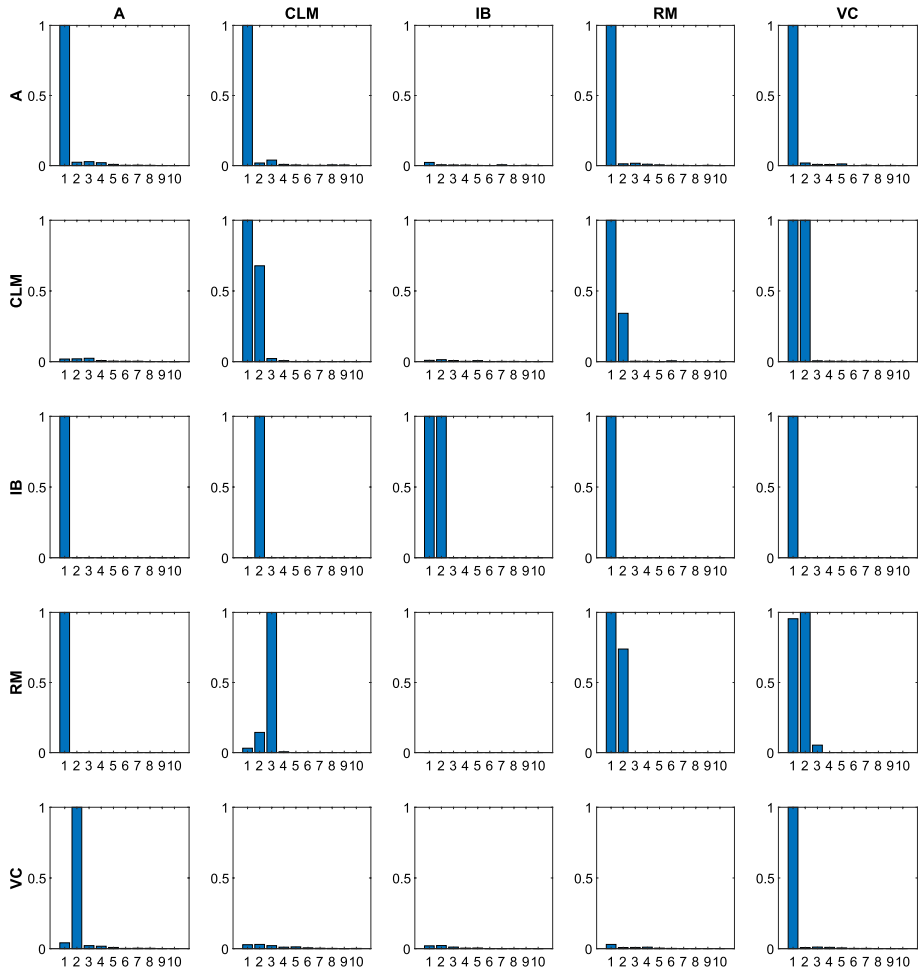
Fig. 8 Important lag selection for the south-eastern Spain flu dataset. Y-axis represents the inclusion proportions of different lags in x-axis for the scenario with 411 training data points and 103 testing data points. A: Andalusia; CLM: Castilla-La Mancha; IB: Illes Balears; RM: Region de Murcia; VC: Valencian Community

Table 7 presents the five-dimensional example of Spanish regions in which the counts of each region are predicted from past counts of all five regions. Here, in eight out of ten cases BTF outperforms the Poisson autoregression model and in particular the log-predictive scores are dramatically lower in all cases with smaller training (462) and higher test (103) sizes. This is not surprising since our model is capable of capturing the complicated five-dimensional dependencies created in these Spanish regions.

Fig. 9 Important lag selection for the south-eastern Spain flu dataset. Y-axis represents the inclusion proportions of different lags in x-axis for the scenario with 462 training data points and 52 testing data points. A: Andalusia; CLM: Castilla-La Mancha; IB: Illes Balears; RM: Region de Murcia; VC: Valencian Community

Table 6 Log predictive score between Bayesian Poisson autoregressive model and Bayesian tensor factorisations model (BTF) for multivariate flu counts datasets

Region	Data Sizes	Bayesian Poisson autoregression		BTF
		AIC	BIC	
Aargau	411 : 103	**4.574(10)**	**4.574(10)**	5.719
	462 : 52	**5.001(10)**	5.041(4)	5.836
Bern	411 : 103	6.155(10)	6.155(10)	**5.328**
	462 : 52	6.632(10)	6.632(10)	**6.103**

Multiple datasets include flu counts in two cantons in Switzerland, Aargau and Bern. AIC and BIC columns indicate that the best model has been chosen (in brackets) with the corresponding criterion. Models with the best performance are highlighted in bold

Table 7 Log predictive score between Bayesian Poisson autoregressive model and Bayesian tensor factorisations model (BTF) for multivariate flu counts datasets

Region	Data Sizes	Bayesian Poisson autoregression		BTF
		AIC	BIC	
Andalucia	411 : 103	**9.097(10)**	9.296(4)	17.205
	462 : 52	17.550(4)	17.550(4)	**14.006**
Castilla-La Mancha	411 : 103	14.589(7)	14.467(5)	**5.760**
	462 : 52	23.765(10)	23.708(9)	**5.992**
Illes Balears	411 : 103	14.334(10)	14.297(3)	**4.335**
	462 : 52	6.788(10)	7.019(5)	**5.178**
Region de Murcia	411 : 103	25.593(10)	25.379(8)	**10.665**
	462 : 52	**5.771(10)**	**5.771(10)**	15.078
Valencian Community	411 : 103	13.760(10)	14.601(8)	**5.997**
	462 : 52	21.532(10)	21.532(10)	**6.450**

Multiple datasets include flu counts in Andalusia, Castilla-La Mancha, Illes Balears, Region de Murcia and Valencian Community. AIC and BIC columns indicate that the best model has been chosen (in brackets) with the corresponding criterion. Models with the best performance are highlighted in bold

6 Discussion

We have introduced a new flexible modelling framework for that extends Bayesian tensor factorisations to multivariate time series of count data. Extensive simulation studies and analysis of real data provide evidence that the flexibility of these models offers an important alternative to other multivariate time series models for counts.

An important aspect of our proposed models is that direct MCMC inference cannot avoid an increased computational complexity as observed counts grow. We have dealt with this issue with a two-stage inferential procedure that successfully deals with large observed counts.

Declarations

Conflict of interest Not applicable.

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

References

Al-Osh, M. A., Alzaid, & Aus A. (1987). First-order integer-valued autoregressive (inar (1)) process. *Journal of Time Series Analysis, 8*(3), 261–275.

Brandt, P. T., & Williams, J. T. (2001). A linear poisson autoregressive model: The poisson ar (p) model. *Political Analysis*, pp. 164–184.

Cameron, A. C., & Trivedi, P. K. (2001). Essentials of count data regression. *A companion to theoretical econometrics*, 331.

Czado, C., Gneiting, T., & Held, L. (2009). Predictive model assessment for count data. *Biometrics, 65*(4), 1254–1261.

De Lathauwer, L., De Moor, B., & Vandewalle, J. (2000). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications, 21*(4), 1253–1278.

Fokianos, K. (2011). Some recent progress in count time series. *Statistics, 45*(1), 49–58.

George, E. I., & McCulloch, R. E. (1997). Approaches for bayesian variable selection. *Statistica sinica*, pp. 339–373.

Grunwald, G. K., Hamza, K., & Hyndman, R. J. (1997). Some properties and generalizations of non-negative bayesian time series models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 59*(3), 615–626.

Grunwald, G. K., Hyndman, R. J., Tedesco, L., & Tweedie, R. L. (2000). Theory & Methods: Non-Gaussian Conditional Linear AR (1) Models. *Australian & New Zealand Journal of Statistics, 42*(4), 479-495.

Harshman, R. (1970). Foundations of the parafac procedure: Model and conditions for an explanatory factor analysis. *Technical Report UCLA Working Papers in Phonetics 16, University of California, Los Angeles, Los Angeles, CA*.

Harshman, R. A., & Lundy, M. E. (1994). Parafac: Parallel factor analysis. *Computational Statistics and Data Analysis, 18*(1), 39–72.

Heinen, A. (2003). Modelling time series count data: an autoregressive conditional poisson model. *Available at SSRN 1117187*.

Kuhn, L., Davidson, L. L., & Durkin, M. S. (1994). Use of poisson regression and time series analysis for detecting changes over time in rates of child injury following a prevention program. *American Journal of Epidemiology, 140*(10), 943–955.

Liboschik, T., Fokianos, K., & Fried, R. (2015). *tscount: An R package for analysis of count time series following generalized linear models*. Germany: Universitätsbibliothek Dortmund Dortmund.

Marin, J.-M.., Mengersen, K., & Robert, C. P. (2005). Bayesian modelling and inference on mixtures of distributions. *Handbook of statistics, 25*, 459–507.

Plummer, M. et al. (2016). rjags: Bayesian graphical models using mcmc. *R package version, 4*(6)

Sarkar, A., & Dunson, D. B. (2016). Bayesian nonparametric modeling of higher order markov chains. *Journal of the American Statistical Association, 111*(516), 1791–1803.

Sethuraman, J. (1994). A constructive definition of dirichlet priors. *Statistica sinica*, pp. 639–650.

Tucker, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika, 31*(3), 279–311.

Viallefont, V., Richardson, S., & Green, P. J. (2002). Bayesian analysis of poisson mixtures. *Journal of nonparametric statistics, 14* (1–2), 181–202.

Weiß, C. H. (2014). Ingarch and regression models for count time series. *Wiley StatsRef: Statistics Reference Online*, pp. 1–6.

Yang, Y., & Dunson, D. B. (2016). Bayesian conditional tensor factorizations for high-dimensional classification. *Journal of the American Statistical Association, 111*(514), 656–669.