Switching Poisson Gamma Dynamical Systems

Wenchao Chen[1] , **Bo Chen**[1*] , **Yicheng Liu**[1] , **Qianru Zhao**[1] , **Mingyuan Zhou**[2]
[1] National Laboratory of Radar Signal Processing, Xidian University, Xian, China
[2] McCombs School of Business, The University of Texas at Austin, Austin, TX 78712, USA
wcchen_xidian@163.com, bchen@mail.xidian.edu.cn, mooooore66@gmail.com, zqr951122@163.com
mingyuan.zhou@mccombs.utexas.esu

Abstract

We propose switching Poisson-gamma dynamical systems (SPGDS) to model sequentially observed multivariate count data. Different from previous models, SPGDS assigns its latent variables into mixture of gamma distributed parameters to model complex sequences and describe the nonlinear dynamics, meanwhile, capture various temporal dependencies. For efficient inference, we develop a scalable hybrid stochastic gradient-MCMC and switching recurrent autoencoding variational inference, which is scalable to large scale sequences and fast in out-of-sample prediction. Experiments on both unsupervised and supervised tasks demonstrate that the proposed model not only has excellent fitting and prediction performance on complex dynamic sequences, but also separates different dynamical patterns within them.

1 Introduction

Temporal sequences are abundant in real world and analyzing them is always an utmost important task in machine learning. Among them, time-series count data has attracted wide attention to deal with a variety of real-world applications, such as text analysis, social network modeling and natural language processing. The widely-used dynamic models, such as hidden Markov models (HMMs) [Rabiner and Juang, 1986] and linear dynamic systems (LDSs) [Ghahramani and Roweis, 1999], have difficulty in modeling such data, which are often high dimensional, sparse, and overdispersed. To address this issue, several dynamic methods have been proposed, especially based on the Poisson-gamma structure [Zhou *et al.*, 2012; Zhou *et al.*, 2016]. Specifically, Poisson-gamma dynamical systems (PGDS) [Schein *et al.*, 2016] is a typical dynamic model for count sequence analysis and performs well in capturing cross-factor temporal dependence, which has been wildly used in text analysis, international relation study and so on. Generally, current developments of dynamic models are always focusing on modeling more complex sequences.

In this paper, we propose switching Poisson-gamma dynamical systems (SPGDS), which is a powerful dynamical model that can capture different dynamics among time-steps, so as to model complex sequential relationships efficiently. Specifically, we build a dynamic system with the assumption that the latent variables of each time-step are drawn from the gamma mixture distributions, whose shape parameter for each mixture component is factorized into a linear transformation of the latent unit of previous time-step. Combining the temporal structure and mixture model, SPGDS can not only benefit the transmitting of nonlinear and diverse temporal variation for ample representational capacity by gamma mixture distributions, but also enable our model to cluster diverse dynamics among time-steps into different patterns. We introduce a discrete indicator variable $\boldsymbol{z}_t$, called switching variable, to guide how the latent state $\boldsymbol{\theta}_t$ varies from time $t-1$ to time t, as illustrated in Fig. 1 (a). The switching mechanism in our model serves two benefits: (1) it results in better fitting and prediction performance on complex count sequences; (2) it provides an insight into which dynamical patterns contains within sequences, which can be of value in application and analysis [Fraccaro *et al.*, 2017; Becker Ehmck *et al.*, 2019]. With the assumption that the real-world data changes in dynamics at time t are causality, we assign $\boldsymbol{z}_t$ to non-linearly depend on history input $x_{1:t-1}$ via a Gumble-softmax based recurrent variational inference network. Based on this, we further develop a Weibull-distribution-based switching recurrent variational inference network for structured inference [Krishnan *et al.*, 2017] of latent variable $\boldsymbol{\theta}_t$. This structured inference network enables SPGDS to learn rich latent representations and fast in out-of-sample prediction. The detail structure of our inference network is shown in Fig. 1 (b). Moreover, we develop a mini-batch based stochastic inference algorithm that combines stochastic gradient MCMC (SG-MCMC) [Patterson and W, 2013; Ma *et al.*, 2015] and autoencoding variational inference, which accelerates our model in both training and testing phase for large scale sequences. Furthermore, to prove the flexibility and compatibility of our model with prevalent deep learning structures, we extend SPGDS into *supervised* SPGDS (sSPGDS), which incorporates the label information into the model and extract discriminative features to achieve enhanced performance on both data representation and classification.

[*]: Corresponding author

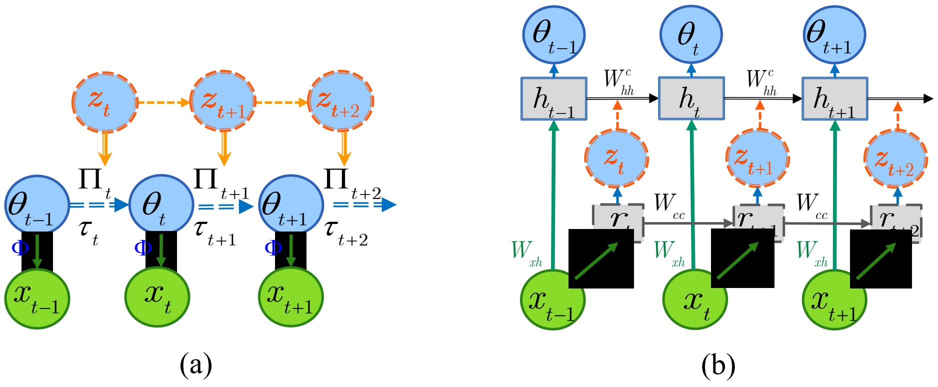

Figure 1: Graphical representation of (a): Switching PGDS, (b): Switching Recurrent Variational Inference Model.

2 Related Work

To model count sequence data, several dynamical model based on the Poisson-gamma construction has been proposed. The gamma process dynamic Poisson factor analysis (GP-DPFA) [Acharya *et al.*,] assumes that the data comes from the Poisson distribution and models the count vector at each time step under Poisson factor analysis (PFA) [Zhou *et al.*, 2012] as $\boldsymbol{x}_t \sim \text{Pois}\left(\boldsymbol{\Phi}\boldsymbol{\theta}_t\right)$. It further smoothes the translation through time by assigning $\boldsymbol{\theta}_t \sim \text{Gam}\left(\boldsymbol{\theta}_{t-1}, \beta_t\right)$. Poisson–gamma dynamical systems (PGDS) [Schein *et al.*, 2016] further improves the ability to capture cross-factor temporal dependence by introducing a translation matrix as $\boldsymbol{\theta}_t \sim \text{Gam}\left(\boldsymbol{\Pi}\boldsymbol{\theta}_{t-1}, \beta\right)$. Moreover, to capture long-range temporal dependencies and model long sequences, several multilayer probabilistic dynamic models are proposed. For instance, deep dynamic Poisson factor analysis (DDPFA) [Gong and Huang, 2017] combines recurrent neural network (RNN) [Martens and Sutskever, 2011] with PFA, to capture long-range temporal dependencies of the latent factor via RNN. Deep temporal sigmoid belief network (DTSBN) [Zhe *et al.*, 2015] is an extension of deep sigmoid belief network (DSB-N) [Gan *et al.*, 2015] with sequential feedback loops on each layer. However, DTSBN restricts its hidden units to be binary, which limits its representational power. [Guo *et al.*, 2018] extends PGDS into deep PGDS (DPGDS), a model with deep hierarchical latent structure and captures the correlations between the features across layers and over times using the gamma belief network [Zhou *et al.*, 2016].

Although these methods exhibit attractive performance in describing temporal dependencies, due to the weights sharing across different time steps, there are still some difficulties for them in modeling sequences with very complex and highly nonlinear dynamics. However, switching linear dynamical systems (SLDS) [Linderman *et al.*, 2016; Fraccaro *et al.*, 2017; Becker Ehmck *et al.*, 2019] is a widely used method to model those complex sequences by breaking down them into multiple simpler units and modeling them separately. But, it is difficult for SLDS to model count sequences for it is under the Gaussian assumption. In this paper, the proposed SPGDS could be seen as a novel switching extension of Poisson-gamma structure models, which not only fit the count sequence, but also inherits various virtues of switching dynamic models. Moreover, our model is also fast in out-of-sample prediction with the help of the structure variational inference network.

3 Switching Dynamical Systems for Count Sequences Modeling

In this section, we first propose a switching PGDS model for analysing count sequences. Then, we propose switching recurrent inference network to map the observation directly to the switching variables and latent representation of SPGDS, by which our model has benefited fast inference for out-of-sampling prediction.

3.1 Switching Poisson Gamma Dynamical Systems

We demonstrate the graphical representation of SPGDS in Fig. 1 (a). Assuming that dataset of V-dimensional sequentially observed multivariate count data $\boldsymbol{x}_1, ..., \boldsymbol{x}_t$ are represented as a $V \times T$ count matrix $\boldsymbol{X}$. The generation process of SPGDS can be expressed as

$$\begin{aligned} \boldsymbol{z}_t &\sim \text{Categorical}\left(\boldsymbol{r}_t\right), \\ \boldsymbol{\theta}_t &\sim \prod_{c=1}^{C}\left(\text{Gam}\left(\tau_0\boldsymbol{\Pi}_c\boldsymbol{\theta}_{t-1}, \tau_c\right)\right)^{\boldsymbol{z}_{tc}}, \\ \boldsymbol{x}_t &\sim \text{Pois}\left(\delta\boldsymbol{\Phi}\boldsymbol{\theta}_t\right), \end{aligned} \tag{1}$$

where the latent factors $\boldsymbol{\Phi}$, $\boldsymbol{\theta}$, $\{\boldsymbol{\Pi}_c\}_{c=1}^C$, $\{\tau_c\}_{c=1}^C$ and $\boldsymbol{z}$ are all positive. The input count data $\boldsymbol{x}_t$ of time t are factored into the loading matrix $\boldsymbol{\Phi} \in \mathbb{R}_+^{V \times K}$ and the corresponding hidden states $\boldsymbol{\theta}_t \in \mathbb{R}_+^K$. And $\delta \in \mathbb{R}_+$ is the scaling parameter. We characterize the relationship between the hidden units of adjacent time-steps as multiple gamma distribution, so as to characterize the nonlinear sequential relationship between time-steps. $\boldsymbol{z}_t = (z_{t1}, ..., z_{tC})$ is a C-dimensional categorical variable which choosing parameters $\boldsymbol{\Pi}_c$ and τ_c at time t from different gamma distribution, we call it switching variable. Marginalizing $\boldsymbol{z}_t$ in (1), we get

$$p\left(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}, \{\boldsymbol{\Pi}_c, \tau_c\}_{c=1}^C\right) = \sum_{c=1}^{C} r_t^c \text{Gam}\left(\boldsymbol{\theta}_t | \tau_0\boldsymbol{\Pi}_c\boldsymbol{\theta}_{t-1}, \tau_c\right),$$

which is a mixture of gamma distribution to characterize the complex and diverse sequential relationship between time-steps more accurately than other Poisson-gamma structure models. $\left\{\mathbf{\Pi}_c \in \mathbb{R}_+^{K\times K}\right\}_{c=1}^C$ are the latent transition matrices that capture the various temporal dependencies between components, and C represents the number of components in mixture distribution. We denote $\{\tau_c \in \mathbb{R}_+\}_{c=1}^C$ as the scaling parameters that control the variousness of temporal amplitude variation of the hidden states. $\boldsymbol{r}_t = \left(r_t^1, ..., r_t^C\right)^T$ is the parameter for categorical distribution. In this way, we can also label sequences into segmentations that exhibit different dynamics with $\boldsymbol{z}_t$. The vector $\boldsymbol{\theta}_t$ has an expected value of $E\left[\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \{\mathbf{\Pi}_c\}_{c=1}^C\right] = \sum\limits_{c=1}^{C} r_t^c \mathbf{\Pi}_c \boldsymbol{\theta}_{t-1}$, which suggests that $\{\mathbf{\Pi}_c\}_{c=1}^C$ play the role of transiting the latent presentation across time, and controlled by $\boldsymbol{r}_t$. Hence, our model is capable to capture the primitive patterns of temporal sequences concisely and accurately, when the data contains multiple dynamics.

To complete the diverse dynamic model, we introduce K factor weights in each components of mixture of gamma distribution:

$$\boldsymbol{\pi}_k^c \sim \mathrm{Dir}\left(\nu_1^c\nu_k^c, \ldots, \xi^c\nu_k^c, \ldots, \nu_{K_l}^c\nu_k^c\right), \nu_k^c \sim \mathrm{Gam}\left(\frac{\gamma_0}{K}, \beta^c\right),$$

where $\boldsymbol{\pi}_k^c = (\pi_{1k}^c, \ldots, \pi_{Kk}^c)$ is the kth column of $\mathbf{\Pi}_c$ and $\left\{\pi_{k_1 k}^c\right\}_{c=1}^C$ can be interpreted as the C different probabilities for transitioning from component k to component k_1 for different dynamics. For the latent state at the first time-step, we define its prior as $\boldsymbol{\theta}^1 \sim \mathrm{Gam}\left(\tau_0 \boldsymbol{v}, \frac{1}{c_0}\right)$. Moreover, we place Dirichlet priors over the feature factors and draw the other parameters from a noninformative gamma prior: $\boldsymbol{\phi}_k = (\phi_{1k}, \ldots, \phi_{Kk}) \sim \mathrm{Dir}\left(\eta, \ldots, \eta\right)$ and $\xi^{(c)}, \beta^{(c)} \sim \mathrm{Gam}\left(\varepsilon_0, \varepsilon_0\right)$, $\tau^{(c)} \sim \mathrm{Gam}\left(\alpha_0, 1/\beta_0\right)$.

In particular, when $C = 1$, SPGDS reduces to PGDS. For non-count observations, we use Bernoulli-Poisson distribution [Zhou, 2015] and Poisson randomized gamma distribution [Aaron *et al.*, 2019] to link binary observation and Nonnegative-real-valued observation to latent poisson count, respectively. More details can be found in [Schein *et al.*, 2016].

3.2 Switching Recurrent Inference Network for SPGDS

For efficient out-of-sample prediction, we develop a switching recurrent inference network, which will be used in hybrid SG-MCMC and variational inference method described in section 4, to map the observation directly to the latent variables. Specifically, we use Concrete distribution [Maddison *et al.*, 2016] to approximate the categorical distributed switching variables, Weibull distribution [Zhang *et al.*, 2018] to approximate gamma distributed conditioned latent representations.

Gumble-softmax based recurrent variational inference network for $\boldsymbol{z}_{tn}$: Assuming there are N count sequential data $\{\boldsymbol{x}_{1n}, ..., \boldsymbol{x}_{Tn}\}_{n=1}^N$. It is clear that categorical variable $\boldsymbol{z}_{tn}$ in control of the changes of latent states $\boldsymbol{\theta}_{tn}$ from time $t-1$ to t. In more general setting, the changes in dynamics at time t depend on the history of system and determined by the input ranging from 0 to $t-1$ [Fraccaro *et al.*, 2017; Becker Ehmck *et al.*, 2019]. Here, we let $\boldsymbol{z}_{tn}$ determined by a learnable function of $\boldsymbol{x}_{1:t-1,n}$ and modulate it by a recurrent variational parameter inference network. We construct the autoencoding variational distribution as $q(\boldsymbol{z}_{tn}) = \mathrm{Categorical}(\boldsymbol{r}_{\mathrm{tn}})$ and transform the observation to $\boldsymbol{r}_{tn}$ using recurrent structure as:

$$\boldsymbol{r}_{tn} = soft\max\left(\boldsymbol{W}_{xc}\boldsymbol{x}_{t-1,n} + \boldsymbol{W}_{cc}\boldsymbol{r}_{t-1,n} + b_{x\pi}\right) \quad (2)$$

To obtain samples from categorical distribution, and to back-propagate through the categorical latent variables, we use $q(\tilde{\boldsymbol{z}}_{tn}) = \mathrm{Gumble} - \mathrm{softmax}(\boldsymbol{r}_{\mathrm{tn}})$ to approximate $q(\boldsymbol{z}_{tn})$ [Maddison *et al.*, 2016; Jang *et al.*, 2016]. It draws samples via

$$\tilde{z}_{tn}^c = \frac{\exp\left(\left(\log r_{tn}^c + g_{tn}^c\right)/\lambda\right)}{\sum\limits_{c=1}^{C}\exp\left(\left(\log r_t^c + g_{tn}^c\right)/\lambda\right)} \quad \text{for } c = 1, ..., C, \quad (3)$$
$$g_{tn}^c \sim \mathrm{Gumbel}\left(0, 1\right) = -\log\left(-\log\left(\boldsymbol{\epsilon}_{tn}^c\right)\right),$$

where λ denotes the softmax temperature and $\boldsymbol{\epsilon}_{tn}^c$ is the standard uniform variable. As λ approaches 0, samples from the Gumbel-softmax distribution become one-hot and the Gumbel-softmax distribution $q(\boldsymbol{z}_{tn})$ becomes identical to the categorical distribution $q(\boldsymbol{z}_{tn})$.

Weibull distribution based switching recurrent variational inference network for $\boldsymbol{\theta}_{tn}$: Following [Zhang *et al.*, 2018], we approximate the gamma distributed conditional posterior of $\boldsymbol{\theta}_{tn}$ with Weibull distribution by assigning $q\left(\boldsymbol{\theta}_{tn}|z_{tn}\right) \sim \mathrm{Weibull}\left(\boldsymbol{k}_{\mathrm{tn}}, \boldsymbol{\lambda}_{\mathrm{tn}}\right)$, the random sample $\boldsymbol{\epsilon}_{tn}$ can be obtained by transforming standard uniform variables $\boldsymbol{\epsilon}_{tn}$ as: $\boldsymbol{\theta}_{tn} = \boldsymbol{\lambda}_{tn}(-\ln\left(1-\boldsymbol{\epsilon}_{tn}\right))^{1/\boldsymbol{k}_{tn}}$, where $\boldsymbol{k}_{tn}$ and $\boldsymbol{\lambda}_{tn}$ are the parameters of $\boldsymbol{\theta}_{tn}$ and they are nonlinearly transformed from the hidden units $\boldsymbol{h}_t$ as

$$\begin{aligned}\mathbf{k}_{tn} &= \ln\left[1 + \exp\left(\boldsymbol{W}_{hk}\boldsymbol{h}_{tn} + \boldsymbol{b}_1\right)\right], \\ \boldsymbol{\lambda}_{tn} &= \ln\left[1 + \exp\left(\boldsymbol{W}_{h\lambda}\boldsymbol{h}_{tn} + \boldsymbol{b}_2\right)\right],\end{aligned} \quad (4)$$

where $\boldsymbol{W}_{hk} \in \mathbb{R}^{K\times K}, \boldsymbol{W}_{h\lambda} \in \mathbb{R}^{K\times K}, \boldsymbol{b}_1 \in \mathbb{R}^K, \boldsymbol{b}_2 \in \mathbb{R}^K$. A nonlinear transformation deterministically covert $\boldsymbol{h}_{tn}$ from $\boldsymbol{x}_{tn}$ and $\boldsymbol{h}_{t-1,n}$. To exploit the various temporal information, we propose a switching recurrent inference network considering diverse temporal dependence across time-steps, as illustrated in Fig. 1 (b). Therefore, $\boldsymbol{h}_{tn}$ can be expressed as

$$\boldsymbol{h}_{tn} = \prod\nolimits_{c=1}^{C} f\left(\boldsymbol{W}_{xh}\boldsymbol{x}_{tn} + \boldsymbol{W}_{hh}^c\boldsymbol{h}_{t-1,n} + \boldsymbol{b}_3^c\right)^{z_{tc}},$$

where f is a nonlinear transformation function. $\boldsymbol{W}_{xh} \in \mathbb{R}^{V\times K}$ is a input-hidden weight matrix, and $\left\{\boldsymbol{W}_{hh}^c \in \mathbb{R}^{K\times K}\right\}_{c=1}^C$ denote hidden states connected matrices. The detail structure of our inference network is shown in Fig. 1 (b).

4 Hybrid SG-MCMC and Variational Inference

In this section, we provide a hybrid stochastic gradient MCMC and autoencoding variational inference for SPGDS, which is scalable in training phase and fast in testing phase. Aforementioned in Section 3, we develop a switching recurrent autoencoding variational inference network for switching variable $\boldsymbol{z}_t$ and latent representation $\boldsymbol{\theta}_t$, which enables our model be fast in testing phase. For inferencing global parameters in SPGDS, including Φ, $\{\Pi_c\}_{c=1}^C$ and $\{\tau_c\}_{c=1}^C$, the topic-layer-adaptive stochastic gradient Riemannian (TLASGR) MCMC algorithm [Cong *et al.*, 2017; Zhang *et al.*, 2018], which is proposed to sample simplex-constrained global parameters in a mini-batch learning setting and improve its sampling efficiency by using the Fisher information matrix (FIM), can be extended to our model. More specifically, after sampling auxiliary latent counts using augmentable techniques as in [Guo *et al.*, 2018], $\pi_k^{(c)}$, kth column of the transition matrix Π^c of component c, can be sampled as

$$\begin{aligned}(\boldsymbol{\pi}_k^c)_{n+1} =&[(\boldsymbol{\pi}_k^c)_n + \frac{\varepsilon_n}{M_k^c}[(\rho \sum_t z_{tc}\tilde{\boldsymbol{Z}}_{:kt} + \boldsymbol{\eta}_{:k}^c) \\ &- (\rho \sum_t z_{tc}\tilde{\boldsymbol{Z}}_{:kt} + \eta_{.k}^c)(\boldsymbol{\pi}_k^c)_n] \\ &+ \boldsymbol{N}(0, \frac{2\varepsilon_n}{M_k^c}[\mathrm{diag}(\boldsymbol{\pi}_k^c)_n - (\boldsymbol{\pi}_k^c)_n(\boldsymbol{\pi}_k^c)_n^T])]_\angle,\end{aligned} \tag{5}$$

where $\tilde{\boldsymbol{Z}}$ comes from the augmented latent counts and the definition of ρ, ε_n, $[\cdot]_\angle$ and M_k^c are analogous to [Cong *et al.*, 2017]'s setting. The update of Φ is the same as [Guo *et al.*, 2018]. For $\{\tau_c\}_{c=1}^C$, which are one-dimensional non-negative data, it is efficient to sample them with stochastic gradient Langevin dynamics [Welling and Teh, 2011] as

$$\begin{aligned}(\tau_c)_{n+1} = (\tau_c)_n + \frac{\varepsilon_n}{2}[(\alpha_0 - 1)\ln(\tau_c)_n - \beta_0 + \\ \rho \sum_{i=1}^{N_n}(\frac{\Pi_{\cdot:}^c \theta_{:c(t-1)}^i}{(\tau_c)_n} - \theta_{\cdot c(t)}^i)] + \boldsymbol{N}(0, \varepsilon_n I).\end{aligned} \tag{6}$$

Given the global parameters $\boldsymbol{\Phi}, \{\boldsymbol{\Pi}_c\}_{c=1}^C, \{\tau_c\}_{c=1}^C$, the task is to optimize the parameters of the switching recurrent inference network. As the usual strategy of autoencoding variational inference, this optimization can be achieved by minimizing the negative Evidence Lower Bound (ELBO). We can express the ELBO of our inference network as:

$$\begin{aligned}L = \sum_{n=1}^N \sum_{t=1}^T E_{q(\boldsymbol{z}_{tn})}\left[E_{q(\boldsymbol{\theta}_{tn})}\left[\ln p\left(\boldsymbol{x}_{tn}|\boldsymbol{\Phi}, \boldsymbol{\theta}_{tn}\right)\right]\right. \\ \left. -KL\left(q\left(\boldsymbol{\theta}_{tn}\right)||p\left(\boldsymbol{\theta}_{tn}|\boldsymbol{\Pi}, \boldsymbol{\theta}_{t-1,n}, \boldsymbol{z}_{tn}\right)\right)\right] \\ -\sum_{n=1}^N \sum_{t=1}^T KL\left(q\left(\boldsymbol{z}_{tn}\right)||p\left(\boldsymbol{z}_{tn}|\boldsymbol{r}_{t-1,n}\right)\right).\end{aligned} \tag{7}$$

Denoting $\boldsymbol{\Omega}$ as the parameters of the inference network: $\boldsymbol{\Omega} = \left\{\boldsymbol{W}_{xc}, \boldsymbol{W}_{cc}, \boldsymbol{W}_{xh}, \{\boldsymbol{W}_{hh}^c\}_{c=1}^C, \boldsymbol{W}_{hk}, \boldsymbol{W}_{h\lambda}\right\}$, the corresponding hybrid SG-MCMC and switching recurrent autoencoding variational inference method for SPGDS is described in Algorithm.1.

Furthermore, our model can be extended to a supervised version called *supervised* SPGDS (sSPGDS) to handle the categorization task of sequential data. We achieve this by concatenate the latent states across all time-steps to construct a $T \times K$-dimensional latent feature and add the softmax classifier on it [Wang *et al.*, 2019]. Then, the loss function of the entire framework should be modified as

$$L = -L_g + \xi L_s, \tag{8}$$

where L_g refers to ELBO of generative model shown in (7), L_s denotes the classification criterion, and ξ is a tradeoff parameter to balance aspects of generation and classification.

Algorithm 1 Hybrid stochastic-gradient MCMC and autoencoding variational inference for SPGDS

```
Set mini-batch size as $\boldsymbol{M}$, the width of layer $\boldsymbol{K}$ and hyperparameters.
Initialize inference model parameters $\boldsymbol{\Omega}$ and generative model parameters $\boldsymbol{D} = \left\{\boldsymbol{\Phi}, \{\boldsymbol{\Pi}_c, \tau_c\}_{c=1}^C\right\}$.
for $iter = 1, 2, ...$ do
    Randomly select a mini-batch of time sequential data to form a subset $\{\boldsymbol{x}_{t,m}\}_{t=1,m=1}^{T,M}$;
    Draw random noise $\left\{\boldsymbol{\epsilon}_{t,m}, \tilde{\boldsymbol{\epsilon}}_{t,m}\right\} \begin{matrix} T, M \\ t = 1, m = 1 \end{matrix}$ from uniform distribution for sampling latent states $\boldsymbol{\theta}_{t,m}$ and $\boldsymbol{z}_{t,m}$;
    Calculate subgradient $\nabla_{\boldsymbol{\Omega}} L\left(\boldsymbol{\Omega}, D; \boldsymbol{x}_{t,m}, \boldsymbol{\epsilon}_{t,m}, \tilde{\boldsymbol{\epsilon}}_{t,m}\right)$ according to (7), and update $\boldsymbol{\Omega}$ using the subgradient;
    for $c = 1, 2, ..., C$ and $k = 1, 2, ..., K$ do
        Update $M_k^c$ according to eqn. (18) and eqn. (19) in [Cong et al., 2017]; then $\boldsymbol{\pi}_k^c$ with (5);
        Update $\tau_c$ with (6);
        Update $\boldsymbol{\phi}_k$ according to eqn. (15) in [Cong et al., 2017];
    end for
end for
```

5 Experiments

In this section, We examine the performance of the proposed model on both unsupervised and supervised tasks.

5.1 Unsupervised Models

We first examine the performance on fitting and prediction tasks of our proposed model, both synthetic datasets and real-world datasets are exploited here. Besides, we compare our model with some existing dynamic methods introduced in section 2, including HMM [Rabiner and Juang, 1986], LDS [Ghahramani and Roweis, 1999], GP-DPFA [Acharya *et al.*,], TSBN [Zhe *et al.*, 2015] and PGDS [Schein *et al.*, 2016]. We set the hyperparameters of GP-PFA, TSBN and PGDS the same as their original settings.

Synthetic Dataset

Inspired by [Gong and Huang, 2017], we consider three different multi-dimensional synthetic datasets:

Toydata 1: $f_1(t) = t$, $f_2(x) = 2\exp(-t/15) + \exp\left(-((t-25)/10)^2\right)$ and $f_3(t) = 5\sin\left(t^2\right) + 3$ for $t = 1, \ldots, 100$.

Data	Measure	SPGDS	PGDS	HMM	LDS
Toy1	MSE	**1.17**	2.11	27.59	1.21
	PMSE	**1.92**	2.47	85.72	7.08
Toy2	MSE	**35.12**	48.24	83.98	53
	PMSE	**42.56**	65.18	250.69	104
Toy3	MSE	**102.31**	180.25	400.18	210.35
	PMSE	**3.15**	4.47	15.83	9.65

Table 1: Results on Synthetic Data

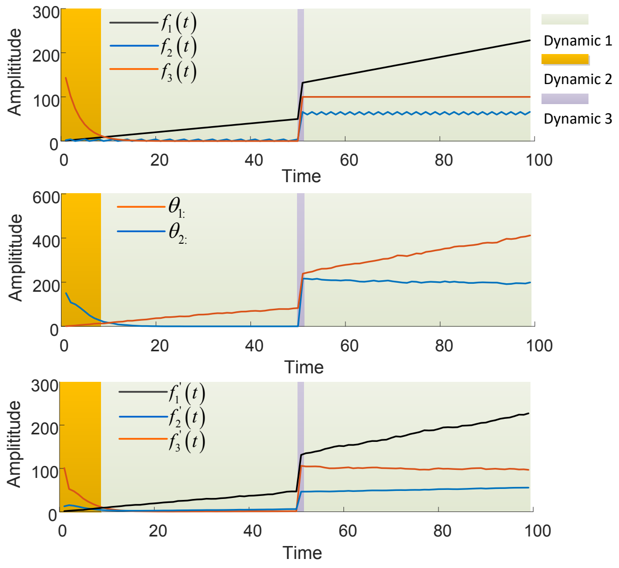

Figure 2: Visualization of data (Top), latent factors (Middle), reconstruction data (Bottom) inferred by SPGDS with three mixture components from Toydata 2. $f_1(t)$, $f_2(t)$ and $f_3(t)$ are the three row vectors of Toydata 2 matrix. $\boldsymbol{\theta}_{1:}$, $\boldsymbol{\theta}_{2:}$ represent the two row vector of latent factor $\boldsymbol{\theta}$. $f_1^{'}(t)$, $f_1^{'}(t)$, $f_1^{'}(t)$ are the reconstruction of $f_1(t)$, $f_2(t)$ and $f_3(t)$. Temporal regions with different dynamic patterns are indicated through different colors.

Toydata 2: $f_1(t) = t$, $f_2(t) = 2 \bmod (t, 3)$, $f_3(t) = 200 \exp(-t/3)$ for $t = 1, \ldots, 50$ and $f_1(t) = 2t + 30$, $f_2(t) = 3 \bmod (t, 2) + 50$ and $f_3(t) = 30t \exp(-t) + 100$ for $t = 51, \ldots, 100$, where $\bmod(t, n)$ denotes the modulo operation which returns the remainder after division of t by n.

Toydata 3: $f_1 = 5t$, $f_2 = 10t$, $f_3 = 10t + 2$ for $t = 1, \ldots, 50$ and $f_1(t) = f_1(50) + \bmod(t, 2)$, $f_2(t) = f_2(50) + 2 \bmod (t, 2) + 2$, $f_3(t) = f_3(50) + \bmod(t, 10)$ for $t = 51, \ldots, 100$ and $f_1(t) = \bmod(t, 3)$, $f_2(t) = 2 \bmod (t, 2) + 2$, $f_3(t) = \bmod(t, 5)$ for $t = 101, \ldots, 150$.

We set the number of latent states as $K = 2$, and compare the proposed SPGDS with PGDS, HMM and LDS on Mean Square Error (MSE) between the ground truth and the estimated value and Prediction Mean Square Error (PMSE), which is the MSE between the ground truth and the prediction in the next time-step. The best performance of different methods are listed in Tab. 1. Clearly, SPGDS has the best performance in fitting and prediction tasks on all datasets. We attribute these to our model's ability of capturing diverse temporal dependencies and modeling them separately in the latent space. Taking Toydata 2 as an example, SPGDS captures three dynamic patterns at different time-steps, which are $t = 1 : 7$, $t = 50 : 51$ and $t = 7 : 50,\ 51 : 99$, as shown in Fig. 2. Corresponding transition matrices and weights to these three dynamic patterns are shown in Tab. 2. For stable sequential variation as dynamic 1, the transition matrix more closely approaches a diagonal matrix and transition weight is approximately equal to one. Relatively, as dynamic 2 and 3, transition matrix and weight changes with various temporal dependencies.

Parameters	translation matrices	translation weights
Dynamic 1	$\Pi_1 = \begin{bmatrix} 0.9996 & 6 * 10^{-5} \\ 0.0004 & 0.9999 \end{bmatrix}$	$\tau_1 = 0.9879$
Dynamic 2	$\Pi_2 = \begin{bmatrix} 0.9267 & 0.1057 \\ 0.0733 & 0.8943 \end{bmatrix}$	$\tau_2 = 1.1281$
Dynamic 3	$\Pi_3 = \begin{bmatrix} 0.6731 & 0.6025 \\ 0.3273 & 0.3987 \end{bmatrix}$	$\tau_3 = 0.2743$

Table 2: Corresponding translation matrices and weights to these three dynamic patterns in Fig. 2

Real-world Datasets

Following [Gong and Huang, 2017; Schein *et al.*, 2016], five real-world datasets are used:

- **Global Database of Events, Language, and Tone (GDELT):** GDELT is an international relationship dataset, which is extracted from news corpora.
- **Integrated Crisis Early Warning System (ICEWS):** ICEWS is another international relationship dataset extracted from news corpora.
- **State-of-the-Union transcript (SOTU):** The SOTU dataset contains the text of the annual SOTU speech transcripts from 1790 to 2014.
- **DBLP conference abstract (DBLP):** DBLP corpus is a database of computer research papers.
- **NIPS corpus (NIPS):** The NIPS contains the text of every NIPS conference paper from 1987 to 2003.

For each of these datasets, we summarize it as a $V \times T$ matrix, as shown in Tab.3. Specifically, we set $V = 1000$ for all data by choosing the top 1000 most frequently used features. Similar as previous methods [Zhe *et al.*, 2015], we evaluate the prediction performance of our model by calculating the precision and recall at top-M as in [Han *et al.*, 2014], which is given by the fraction of the predicted top-M words, that matches the true ranking of the words. M is set as 50 here. We use three criterion MP, MR and PP. MP and MR are mean precision and mean recall over all years that appears in the training set, PP is the prediction precision for the final year. Moreover, we employ the setup in [Zhe *et al.*, 2015] that the entire data of the last year is held-out, while the words of the each document for the documents in the previous years are randomly partitioned into $80\%/20\%$ split. The 80% portion is used to train the model, and the prediction at the next year is tested on the rest of 20% held-out words. We compare the proposed model with several related works, including G-PDPFA, TSBN and PGDS, and the results are summarized in

		Data				
Model	Top@M	GDELT	ICEWS	SOTU	DBLP	NIPS
		$T = 365, V \approx 1000$	$T = 365, V \approx 1000$	$T = 225, V = 1000$	$T = 14, V = 1000$	$T = 17, V = 1000$
GPDPFA	MP	0.611 ±0.001	0.607 ±0.002	0.379 ±0.002	**0.435** ±0.009	0.843 ±0.005
	MR	0.145 ±0.002	0.235 ±0.005	0.369 ±0.002	0.254 ±0.005	**0.050** ±0.001
	PP	0.447 ±0.014	0.465 ±0.008	0.617 ±0.013	0.581 ±0.011	0.807 ±0.006
PGDS	MP	0.679 ±0.001	0.658 ±0.001	0.375 ±0.002	0.419 ±0.004	0.864 ±0.004
	MR	**0.150** ±0.001	0.245 ±0.005	0.373 ±0.002	0.252 ±0.004	0.050 ±0.001
	PP	0.420 ±0.017	0.455 ±0.008	0.612 ±0.018	0.566 ±0.008	0.802 ±0.020
TSBN	MP	0.594 ±0.007	0.471 ±0.001	0.360 ±0.001	0.403 ±0.012	0.788 ±0.005
	MR	0.124 ±0.001	0.158 ±0.001	0.275 ±0.001	0.194 ±0.001	**0.050** ±0.001
	PP	0.418 ±0.019	0.445 ±0.031	0.611 ±0.001	0.527 ±0.003	0.692 ±0.017
DTSBN-3	MP	0.411 ±0.001	0.431 ±0.001	0.370 ±0.008	0.390 ±0.002	0.774 ±0.002
	MR	0.141 ±0.001	0.189 ±0.001	0.274 ±0.001	0.252 ±0.004	**0.050** ±0.001
	PP	0.367 ±0.011	0.451 ±0.026	0.548 ±0.013	0.510 ±0.006	0.715 ±0.009
DPGDS-3	MP	0.689 ±0.002	0.660 ±0.001	0.380 ±0.001	0.431 ±0.012	0.887 ±0.002
	MR	**0.150** ±0.001	0.244 ±0.003	0.374 ±0.002	0.255 ±0.004	**0.050** ±0.001
	PP	**0.456** ±0.015	**0.478** ±0.024	0.628 ±0.021	0.600 ±0.001	0.839 ±0.007
SPGDS	MP	**0.705** ±0.003	**0.675** ±0.003	**0.380** ±0.002	0.428 ±0.004	**0.890** ±0.004
	MR	**0.150** ±0.001	**0.253** ±0.004	**0.377** ±0.002	**0.257** ±0.004	**0.050** ±0.001
	PP	0.440 ±0.015	0.450 ±0.008	**0.634** ±0.028	**0.605** ±0.018	**0.840** ±0.010

Table 3: Top-M Results on Real-world Text Data

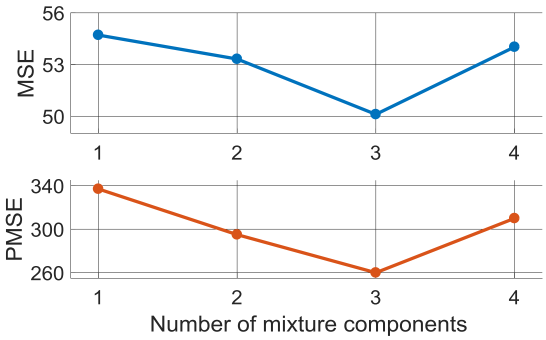

Figure 3: MSE and PMSE on NIPS for different number of mixture components

Tab. 3. Fig. 3 presents the different number of mixture components influence to the results in SPGDS on NIPS dataset. They have similar trends with different number of mixture components. Thus, cross validation where we set MSE as metric is used here to determine the number of mixture components in SPGDS for each dataset. We select $C_g = 5$, $C_i = 5$, $C_s = 3$, $C_d = 2$, $C_n = 3$ for datasets from left to right in Tab. 3 Clearly, SPGDS outperforms other methods on most of the evaluation criteria which we contribute to the superiority of SPGDS in modeling nonlinear sequential data. In Fig. 4, we present the visualization of top four latent factors inferred by SPGDS with three mixture components from the NIPS. The three dynamical patterns captured by our model, including the decline of research on neuron from 1987 to 1991, the relatively stable phase from 1991 to 1994, the decline of research on neural network and the raised of research on bayesian learning from 1994 to 2002. The sustainable growth of topics on ”learning, network” indicates the increase in the number of papers accepted by NIPS.

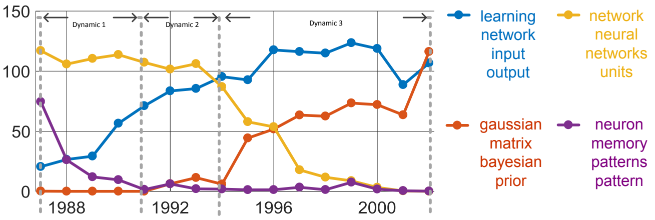

Figure 4: Visualization of top four latent factors inferred by SPGDS with three mixture components from the NIPS matrix. Temporal regions with different dynamic patterns are separated with a dotted line.

5.2 Supervised Models

To evaluate how well sSPGDS leverages the label information for feature learning, we compare its classification performance with a variety of algorithms on sequential MNIST dataset and permuted sequential MNIST dataset. For sequential MNIST, the pixels of MNIST digits [LeCun *et al.*, 1998] are presented sequentially to the network and classification is performed at the end. By permuting the image sequences by a fixed random order, we can get permuted sequential MNIST dataset. Since the MNIST image are 28×28 pixels, they are reshaped into 784×1 sequences in sequential MNIST.

We compare our model with (a)RNN(relu), that is an RNN with RELU activations, (b) IRNN [Le *et al.*, 2015], that is an RNN with RELU activations and with the recurrent weight matrix initialized to the identity, (c) Unitary evolution RNN (uRNN) [Arjovsky *et al.*, 2016], that uses orthogonal and unitary matrices in RNN, (d) Full-Capacity Unitary RNN [Wisdom *et al.*, 2016], which is a modified version of uRNN, (e) Skip RNN [Campos *et al.*, 2018], an extending of existing RNN models, that learns to skip state updates and shortens the effective size of the computational graph, (f) long short-term

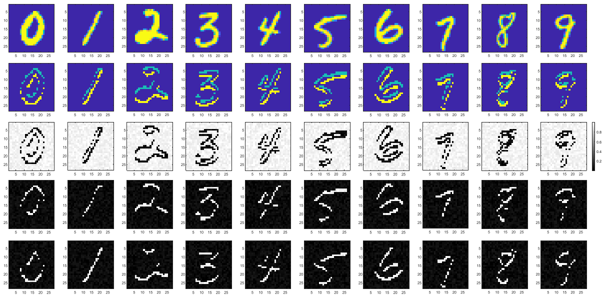

Figure 5: Top row: ten example MNIST data; second row: corresponding dynamics captured by sSPGDS. Pixels in different colors represent different dynamics; the third to fifth rows: the probability of $p(I(z_t) = 1)$, $p(I(z_t) = 2)$ and $p(I(z_t) = 3)$.

Model	sMNIST	pMNIST
RNN(RELU)	94.5	80.1
iRNN [Le *et al.*, 2015]	97.0	≈ 82.0
uRNN [Arjovsky *et al.*, 2016]	95.1	91.4
Full uRNN [Wisdom *et al.*, 2016]	97.5	**94.0**
Skip RNN [Campos *et al.*, 2018]	97.3	-
GRU [Barone, 2016]	97.6	92.5
LSTM [Zhang *et al.*, 2016]	**98.2**	88.0
sPGDS	97.3	85.4
sSPGDS	**98.4**	**92.7**

Table 4: Supervised Results on MNIST

memory (LSTM) [Greff *et al.*, 2016] and (g) the gated recurrent unit (GRU) [Cho *et al.*, 2014], which are the variants of RNN to address the gradient problem, (h) supervised Poisson gamma dynamical system, a supervised extension of PGDS. The classification accuracy of different methods are shown in Tab. 4, where the results of compared methods are provided in their corresponding papers. The latent dimension of models are 100 and Tensor Recurrent autoencoding network is developed based on RNN (relu). Clearly, our proposed sSPGDS can achieve a comparable performance among these methods and we contribute to the two characteristics of our model: (1) uncertainty is included in our hidden states [Chung *et al.*, 2015] and (2) mixture distribution is assigned to our latent variables. These two characters enable sSPGDS to be robustness when track with complex and nonlinear sequential data. In addition, we show various dynamics captured by sSPGDS in Fig. 5. The number of mixture components is set as three. We reshape $I(z_t)$, which is the index for z_t in (1), from $t = 2$ to $t = 785$ into 28×28 pixel and each pixel represents the index for dynamical patterns from the corresponding pixel in data to its next pixel. We visualize it by assign different colors to $I(z_t)$ with different values: $I(\boldsymbol{z}_t) = 1 \rightarrow$ blue, $I(\boldsymbol{z}_t) = 2 \rightarrow$ green, $I(\boldsymbol{z}_t) = 3 \rightarrow$ yellow, and show it in the second row of Fig. 5. As we can see, our model captures three different dynamical patterns including: changing within noise or target, changing from noise to target and changing from target to noise. Moreover, we also visualize the probability of $p(I(z_t) = 1)$, $p(I(z_t) = 2)$ and $p(I(z_t) = 3)$ in third to fifth row of Fig. 5.

6 Conclusion

In this paper, we propose switching Poisson gamma dynamical systems (SPGDS) that takes advantage of gamma mixture distributions to model complex and nonlinea temporal dependencies, while capturing various dynamic patterns. To model the dependency of dynamic patterns among different timesteps and achieve fast out-of-sample prediction, a switching recurrent variational inference network is developed to infer the switching variable and latent representation. A minibatch based stochastic inference method that combines both stochastic-gradient MCMC and variational inference algorithm is developed to accelerate both training and testing for large scale sequences. In addition, we provide a supervised extension of SPGDS. Experiment results show that our model not only has an excellent fitting and prediction performance on unsupervised feature extraction tasks, but also achieves comparable classification performance on supervised tasks.

Acknowledgements

B. Chen acknowledges the support of the Program for Young Thousand Talent by Chinese Central Government, the 111 Project (No. B18039), NSFC (61771361), NSFC for Distinguished Young Scholars (61525105), Shaanxi Innovation Team Project, and the Innovation Fund of Xidian University. M. Zhou acknowledges the support of the U.S. National Science Foundation under Grant IIS-1812699.

References

[Aaron *et al.*, 2019] Schein Aaron, W. Linderman Scott, Zhou Mingyuan, M. Blei David, and M. Wallach Hanna. Poisson-randomized gamma dynamical systems. *In NeurIPS*, pages 781–792, 2019.

[Acharya *et al.*,] Ayan Acharya, Joydeep Ghosh, and Mingyuan Zhou. Nonparametric Bayesian factor analysis for dynamic count matrices. *In AISTATS*, pages 1462–1471.

[Arjovsky *et al.*, 2016] Martin Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. *In ICML*, pages 1120–1128, 2016.

[Barone, 2016] Antonio Valerio Miceli Barone. Low-rank passthrough neural networks. pages 77–86, 2016.

[Becker Ehmck *et al.*, 2019] Philip Becker Ehmck, Jan Peters, and van der Smagt Patrick. Switching linear dynamics for variational bayes filtering. *In ICML*, pages 553–562, 2019.

[Campos *et al.*, 2018] Victor Campos, Brendan Jou, Xavier Giro-I-Nieto, Jordi Torres, and Shih Fu Chang. Skip rnn: Learning to skip state updates in recurrent neural networks. *In ICLR*, 2018.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merrienboer, Caglar Gulcehre, Fethi Bougares, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *Computer Science*, pages 1724–1734, 2014.

[Chung *et al.*, 2015] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. *In NeurIPS*, pages 2980–2988, 2015.

[Cong *et al.*, 2017] Yulai Cong, Chen Bo, Hongwei Liu, and Mingyuan Zhou. Deep latent dirichlet allocation with topic-layer-adaptive stochastic gradient riemannian mcmc. *In ICML*, pages 864–873, 2017.

[Fraccaro *et al.*, 2017] Marco Fraccaro, Simon Kamronn, Ulrich Paquet, and Ole Winther. A disentangled recognition and nonlinear dynamics model for unsupervised learning. *In NeurIPS*, pages 3601–3610, 2017.

[Gan *et al.*, 2015] Z Gan, R Henao, D Carlson, and L Carin. Learning deep sigmoid belief networks with data augmentation. *In AISTATS*, 2015.

[Ghahramani and Roweis, 1999] Zoubin Ghahramani and Sam T. Roweis. Learning nonlinear dynamical systems using an em algorithm. *In NeurIPS*, 11:431–437, 1999.

[Gong and Huang, 2017] C. Y Gong and W Huang. Deep dynamic poisson factorization model. *In neurIPS*, pages 1666–1674, 2017.

[Greff *et al.*, 2016] K Greff, R. K. Srivastava, J Koutnik, B. R. Steunebrink, and J Schmidhuber. Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232, 2016.

[Guo *et al.*, 2018] Dandan Guo, Bo Chen, Hao Zhang, and Mingyuan Zhou. Deep poisson gamma dynamical systems. *In NeurIPS*, pages 8451–8461, 2018.

[Han *et al.*, 2014] S Han, L Du, E Salazar, and L Carin. Dynamic rank factor model for text streams. *In NeurIPS*, pages 2663–2671, 2014.

[Jang *et al.*, 2016] Eric Jang, Gu Shixiang, and Poole Ben. Categorical reparameterization with gumbel-softmax. *arXiv preprint: 1611.01144*, 2016.

[Krishnan *et al.*, 2017] Rahul G. Krishnan, Uri Shalit, and David A. Sontag. Structured inference networks for nonlinear state space models. *In AAAI*, pages 2101–2109, 2017.

[Le *et al.*, 2015] Quoc V. Le, Navdeep Jaitly, and Geoffrey E. Hinton. A simple way to initialize recurrent networks of rectified linear units. *Computer Science*, 2015.

[LeCun *et al.*, 1998] Y LeCun, L Bottou, Y Bengio, and P Haffner. Gradient based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998.

[Linderman *et al.*, 2016] Scott W. Linderman, Andrew C. Miller, and Ryan P. Adams. Recurrent switching linear dynamical systems. *In AISTATS*, 2016.

[Ma *et al.*, 2015] Yi An Ma, Tianqi Chen, and Emily B. Fox. A complete recipe for stochastic gradient mcmc. *In NeurIPS*, pages 2917–2925, 2015.

[Maddison *et al.*, 2016] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *CoRR*, 2016.

[Martens and Sutskever, 2011] James Martens and Ilya Sutskever. Learning recurrent neural networks with Hessian-Free optimization. *In ICML*, pages 1033–1040, 2011.

[Patterson and W, 2013] S Patterson and Teh Y. W. Stochastic gradient riemannian langevin dynamics on the probability simplex. *In NeurIPS*, pages 3102–3110, 2013.

[Rabiner and Juang, 1986] L Rabiner and B Juang. An introduction to Hidden Markov Models. *IEEE ASSP magazine*, 1986.

[Schein *et al.*, 2016] Aaron Schein, Mingyuan Zhou, and Hanna Wallach. Poisson-gamma dynamical systems. *In NeurIPS*, pages 5006–5014, 2016.

[Wang *et al.*, 2019] Chaojie Wang, Bo Chen, Shucheng Xiao, and Mingyuan Zhou. Convolutional poisson gamma belief network. *In ICML*, pages 6515–6525, 2019.

[Welling and Teh, 2011] Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient langevin dynamics. *In ICML*, pages 681–688, 2011.

[Wisdom *et al.*, 2016] Scott Wisdom, Thomas Powers, John R. Hershey, Jonathan Le Roux, and Les Atlas. Full-capacity unitary recurrent neural networks. *In NeurIPS*, pages 4880–4888, 2016.

[Zhang *et al.*, 2016] Saizheng Zhang, Yuhuai Wu, Che Tong, Zhouhan Lin, and Yoshua Bengio. Architectural complexity measures of recurrent neural networks. *In NeurIPS*, pages 1822–1830, 2016.

[Zhang *et al.*, 2018] Hao Zhang, Bo Chen, Dandan Guo, and Mingyuan Zhou. Whai: Weibull hybrid autoencoding inference for deep topic modeling. *In ICLR*, 2018.

[Zhe *et al.*, 2015] Gan Zhe, Chunyuan Li, Ricardo Henao, David Carlson, and Lawrence Carin. Deep temporal sigmoid belief networks for sequence modeling. *In NeurIPS*, pages 2467–2475, 2015.

[Zhou *et al.*, 2012] Mingyuan Zhou, Lauren Hannah, David Dunson, and Lawrence Carin. Beta-negative binomial process and poisson factor analysis. *In AISTATS*, pages 1462–1471, 2012.

[Zhou *et al.*, 2016] M. Zhou, Y. Cong, and B Chen. Augmentable gamma belief networks. *Journal of Machine Learning Research*, 17(163):1–44, 2016.

[Zhou, 2015] Mingyuan Zhou. Infinite edge partition models for overlapping community detection and link prediction. *In AISTATS*, pages 1135–1143, 2015.