LAMORE: A Stable, Scalable Approach to Latent Vector Autoregressive Modeling of Categorical Time Series

Yubin Park[1] **Carlos M. Carvalho**[2] **Joydeep Ghosh**[1]

[1]Department of Electrical Computer Engineering
The University of Texas at Austin

[2]McCombs School of Business
The University of Texas at Austin

Abstract

Latent vector autoregressive models for categorical time series have a wide range of potential applications from marketing research to healthcare analytics. However, a brute-force particle filter implementation of the Expectation-Maximization (EM) algorithm often fails to estimate the maximum likelihood parameters due to the Monte Carlo approximation of the E-step and multiple local optima of the log-likelihood function. This paper proposes two auxiliary techniques that help stabilize and calibrate the estimated parameters. These two techniques, namely *asymptotic mean regularization* and *low-resolution augmentation*, do not require any additional parameter tuning, and can be implemented by modifying the brute-force EM algorithm. Experiments with simulated data show that the proposed techniques effectively stabilize the parameter estimation process. Also, experimental results using Medicare and MIMIC-II datasets illustrate various potential applications of the proposed model and methods.

1 Introduction

Categorical time series, i.e. temporal sequences of alphabets, are pervasive across multiple domains such as healthcare, bio-medicine, econometrics, and marketing research. For example, in the healthcare domain, a patient's diagnosis history has been an informative source of information to score the risk of mortality and potential illness (Gadzhanova et al., 2007). In the fraud detection community, purchase records and click streams have been key factors for detecting fraudulent activities in electronic commerce (Kou et al., 2004). Although categorical time series are relatively less studied than continuous time series, they are valuable assets for forecasting events, analyzing temporal patterns, and extracting meaningful information (Fokianos and Kedem, 2003).

Several statistical models have been developed for categorical time series data. These models fall into two classes depending on the use of latent variables: fully observation-based and latent variable models. The fully observation-based models include Mixture Transition Distribution Model (MTDM) (Raftery, 1985), Markovian regression model (Kaufmann, 1987; Zeger et al., 1988), and Discrete Autoregressive Moving Average model (DARMA) (Jacobs and Lewis, 1983). On the other hand, latent variable models have been successfully demonstrated in various applications such as decoding algorithms (Viterbi, 1967) and speech recognition (Juang and Rabiner, 1991). Such latent variable models can be further grouped into two sub-categories based on the representation of latent variables; the class of Hidden Markov Models (HMM) (Zucchini and MacDonald, 2009) uses discrete latent variables, whereas the class of State-Space Models (SSM) (Zhen and Basawa, 2009) adopts continuous latent variables.

This paper focuses on a latent vector autoregressive (VAR) model for categorical time series. The model of our interest is a state-space model for a categorical time series that has been less popular than HMM and SSM. This is partly because such continuous latent variables are notoriously difficult to reconstruct from categorical observations. However, the use of the latent VAR process provides two substantial advantages: *interpretability* and *extensibility*. Indeed, the VAR process has a rich history with parsimonious theoretical results (Canova and Cicarelli, 2013; Litterman, 1984). The interpretation on stationarity and spectral analyses (Burg, 1967) can be smoothly applied to the latent VAR processes. Moreover, the latent VAR pro-

Appearing in Proceedings of the 17^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2014, Reykjavik, Iceland. JMLR: W&CP volume 33.

cess can be easily extended to cover variants of the AR models such as ARMA (Box et al., 1994), Autoregressive Conditional Heteroskedasticity (ARCH) (Engle, 1982), and Generalized ARCH (GARCH) processes.

The goal of this paper is to estimate the maximum likelihood parameters, $\boldsymbol{\theta} = \{\mathbf{c}, \boldsymbol{\Phi}\}$, of the following model, Latent Vector-Autoregressive model for Categorical Time series (LAVA-Cat):

$$\text{(Latent VAR)} \qquad \mathbf{x}_t = \mathbf{c} + \boldsymbol{\Phi}\mathbf{x}_{t-1} + \boldsymbol{\varepsilon}_t \tag{1}$$

$$\text{(Observation)} \qquad p(y_t = k \mid \mathbf{x}_t) = f_k(\mathbf{A}_k \mathbf{x}_t) \tag{2}$$

$$\text{(Noise model)} \qquad \boldsymbol{\varepsilon}_t \sim \text{Normal}(0, \boldsymbol{\Sigma}) \tag{3}$$

where $\mathbf{y} = [y_t]_{t=1}^T$ with $y_t \in \{1, 2, \ldots, K\}$ and $p(y_t = k \mid \mathbf{x}_t)$ is the probability that the kth category is observed at time t. The dimensions of the parameters are $\mathbf{c},\ \mathbf{x}_t,\ \boldsymbol{\varepsilon}_t \in \mathbb{R}^{(K-1)}$, $\boldsymbol{\Phi},\ \boldsymbol{\Sigma} \in \mathbb{R}^{(K-1)\times(K-1)}$ where $\boldsymbol{\Sigma}$ is a semi-positive definite matrix. The link function $f_k(\mathbf{A}_k\mathbf{x}_t)$ connects a real vector to a categorical value as in the generalized linear model. Some potential applications of the LAVA-Cat are: modeling (recurring) purchases of items, tracking a patient's disease history, and predicting a customer's life events. Note that the model parameters in this paper are *fixed but unknown*.

Several challenges need to be addressed to estimate the parameters from a categorical sequence. First, unlike continuous time series, categorical time series contain only finite bits of information. Categorical outputs can be viewed as lossy-compression from an information theoretic perspective, thus the reconstruction of the continuous latent variables suffers from a low signal-to-noise ratio. Furthermore, this noisy reconstruction increases the uncertainty of the estimated parameters, especially in the alternating minimization framework. As a result, classical alternating minimization techniques, such as the Expectation-Maximization algorithm, become susceptible to various factors such as noisy reconstruction and multiple local optima of the log-likelihood function.

We have frequently observed that the estimated parameters from the EM algorithm are inconsistent with the stationary and spectral properties of a categorical time series. In fact, this research was initially motivated to calibrate the estimated parameters to be more consistent with the asymptotic properties. While searching for theoretically sound calibration methods, two techniques were found to be easy to apply and effective in more accurate estimation: *asymptotic mean regularization* and *low-resolution augmentation*. We summarize the contributions of this paper as follows: First, we propose a novel regularization technique that improves the consistency of estimated parameters. Second, we augment the original time series using a low-resolution time series to reduce the variance of estimated parameters. These proposed techniques are efficiently integrated with the EM algorithm using a Bayesian linear regression update equation.

The rest of this paper is organized as follows: In Section 2, we cover the basics of particle methods and parameter estimation techniques in state-space models. In Section 3, we introduce the model of our interest, and then illustrate a brute-force particle filter implementation of the EM algorithm for the model. The two auxiliary techniques and their implementation details are described in Section 4. Empirical results from simulated and two real-life datasets are illustrated in Section 5. Finally, we discuss the limitation of the proposed methods and future work in Section 6.

2 Preliminaries

In this section, we cover related work on particle methods and parameter estimation techniques in state-space models.

2.1 Particle Methods

If both observation and latent variables are normally distributed, the optimal filtering is solved by the Kalman Filter (KF) (Kalman, 1960). For non-linear systems, several approximation techniques based on linearization, such as Extended KF (first-order approximation) and Unscented KF (second-order approximation), can be applied. However, such linearization usually causes non-diminishing bias, and even worse, those algorithms are typically difficult to implement and tune correctly (Julier and Uhlmann, 2004).

Algorithm 1: Bootstrap Particle Filter

Data: $\mathbf{y}$, $\boldsymbol{\theta}$
Result: $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}_{t,i}$
for $t \in 1 : T$ **do**
  $\{\mathbf{x}_{t-1}^{(i)}\}_{i=1}^P$ to $\{\tilde{\mathbf{x}}_t^{(i)}\}_{i=1}^P$ via $p_{\boldsymbol{\theta}}(\mathbf{x}_t \mid \mathbf{x}_{t-1})$;
  $\{\mathbf{x}_t^{(i)}\}_{i=1}^P$ from $\{\tilde{\mathbf{x}}_t^{(i)}\}_{i=1}^P$ with $w_t^{(i)} \propto p(y_t \mid \tilde{\mathbf{x}}_t^{(i)})$;
end

Particle methods (Gordon et al., 1993) use a different kind of approximation technique, Monte Carlo simulation. Unlike those variants of KF, the state estimates from particle methods can be made arbitrarily accurate with enough particles. Particle methods are based on a sequence of importance sampling steps. Resampling techniques (Liu and Chen, 1998) are typically adopted to decelerate the degeneracy of particles. Also, Auxiliary Particle Filter has been developed to prevent the degeneracy of the Sequential Monte Carlo mechanism (Pitt and Shephard, 1999). Particle methods are powerful and general state-space estimation techniques that are widely applicable to non-linear evolution and observation processes (Doucet and Johansen, 2008). Algorithm 1 illustrates one of the most popular particle methods, the Bootstrap Particle Filter (BPF) algorithm.

2.2 Parameter Estimation in SSMs

Parameter estimation techniques for SSMs fall into three main groups: Bayesian online, maximum-likelihood offline, and maximum-likelihood online settings (Kantas et al., 2009). In the Bayesian online setting, model parameters are assumed to be *dynamic* over time series, and the model parameters are sequentially estimated. Some of the successful algorithms are Liu-West filter (Liu and West, 2001), Storvik filter (Storvik, 2002), and Particle learning (Carvalho et al., 2010). Recall that the parameters in this paper are fixed but unknown; our setting is different from the Bayesian online setting.

In the offline (or batch) maximum-likelihood setting, two approaches have been popular: Fisher's scoring and Expectation-Maximization (EM). The Fisher's scoring algorithm is a variant of Newton-Raphson algorithm based on the log-likelihood function. However, obtaining the log-likelihood of a time series is typically intractable. Doucet and Tadic (2003) proposed a general approach for approximating the log-likelihood using particle methods. Although this Fisher's scoring algorithm is generally applicable to several settings, it is difficult to scale the gradients for high dimensional parameters (Kantas et al., 2009).

The EM algorithm is numerically more stable and usually computationally cheaper for high dimensional parameters. For a Gaussian SSM, the EM algorithm can be implemented using Kalman Filter and Smoother (Shumway and Stoffer, 1982). For non-linear systems, the EM-PF (EM using Particle Filter) algorithm was introduced in (Zia et al., 2008), but many of the assumptions are not applicable in our setting. As will be seen later in this paper, a generic combination of EM and PF algorithms fails for a categorical time series.

For the online setting, Andrieu et al. (2005) have demonstrated an online estimation algorithm using block time series and pseudo-likelihood. We will discuss the possibility and limits of extending our methods to the online estimation setting in Section 6.

3 LAVA-Cat model

In this section, we describe a Latent Vector-Autoregressive model for Categorical Time series (LAVA-Cat), and illustrate the formulation of a brute-force BPF-implementation of the EM algorithm.

Let us consider the LAVA-Cat model defined in Equation (1), (2), and (3). Although other kinds of link functions, such as probit, can be applied to this LAVA-Cat model, we primarily focus on a multinomial logistic (softmax) link function as follows:

$$f_k(\mathbf{x}_t) = \begin{cases} \exp(x_{tk})/h(\mathbf{x}_t) & k \in 1, 2, \dots, (K-1) \\ 1/h(\mathbf{x}_t) & k = K \end{cases}$$

where $h(\mathbf{x}_t) = \sum_{l=1}^{K-1} \exp(x_{tl}) + 1$, x_{tk} represents the kth entry of $\mathbf{x}_t$. In other words, $p(y_k = K)$ is the reference probability for the other categorical outcomes .

The log-likelihood of the LAVA-Cat model is decomposed as follows:

$$\begin{aligned} \max_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{y}) &= \max_{\boldsymbol{\theta}} \log \int_{\mathbf{X}} p_{\boldsymbol{\theta}}(\mathbf{y}, \mathbf{X}) d\mathbf{X} \\ &= \max_{\boldsymbol{\theta}} \log \int_{\mathbf{X}} \prod_t f(y_t \mid \mathbf{x}_t) p_{\boldsymbol{\theta}}(\mathbf{x}_t \mid \mathbf{x}_{t-1}) d\mathbf{X} \end{aligned}$$

where $\mathbf{X} = [\mathbf{x}_t]_{t=1}^T$. As can be seen, the maximization of the log-likelihood is intractable. Instead, we derive the lower bound of the log-likelihood, and then maximize the lower bound i.e. the Expectation-Maximization algorithm (Neal and Hinton, 1998). The lower-bound is obtained using Jensen's inequality as follows:

$$\log \int_{\mathbf{X}} p_{\boldsymbol{\theta}}(\mathbf{y}, \mathbf{X}) d\mathbf{X} \geq \int_{\mathbf{X}} q(\mathbf{X}) \log \frac{p_{\boldsymbol{\theta}}(\mathbf{y}, \mathbf{X})}{q(\mathbf{X})} d\mathbf{X}$$

The lower-bound is maximized by iteratively solving two sub-problems:

$$q = \arg\max_q \int q \log(p_{\boldsymbol{\theta}}/q)$$

$$\boldsymbol{\theta} = \arg\max_{\boldsymbol{\theta}} \int q \log(p_{\boldsymbol{\theta}}/q)$$

The first maximization problem has a closed-form solution, $q(\mathbf{X}) = p_{\boldsymbol{\theta}}(\mathbf{X} \mid \mathbf{y})$. However, obtaining the exact $p_{\boldsymbol{\theta}}(\mathbf{X} \mid \mathbf{y})$ is intractable for the LAVA-Cat model. Instead of having the exact distribution, we approximate the target distribution using particle methods, $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}_{t=1:T}^{i=1:P}$ where P is the number of particles. Then, the second maximization step is derived as:

$$\begin{aligned} &\max_{\boldsymbol{\theta}} \int_{\mathbf{X}} q(\mathbf{X}) \log p_{\boldsymbol{\theta}}(\mathbf{y}, \mathbf{X}) d\mathbf{X} \\ &= \max_{\boldsymbol{\theta}} \int_{\mathbf{X}} q(\mathbf{X}) \log \prod_t f(y_t \mid \mathbf{x}_t) p_{\boldsymbol{\theta}}(\mathbf{x}_t \mid \mathbf{x}_{t-1}) d\mathbf{X} \\ &\approx \max_{\boldsymbol{\theta}} \sum_{i,t} w_t^{(i)} (\log f(y_t \mid \mathbf{x}_t^{(i)}) + \log p_{\boldsymbol{\theta}}(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})) \end{aligned}$$

Note that the distribution $q(\mathbf{X})$ is approximated in the E-step, not the expectation of the latent vectors.

The mean and variance of the latent vectors, $\mathrm{E}[\mathbf{X}]$ and $\mathrm{Var}[\mathbf{X}]$, are no longer sufficient statistics for the non-Gaussian distribution $q(\mathbf{X})$.

A Brute-force Particle-filter implementation of the EM algorithm (BPEM) is illustrated in Algorithm 2. Although the BPEM algorithm is simple, this algorithm is not applicable in practice because of two issues: First, the algorithm stores at least $2 \times T \times P$ particles and weights. As an illustrative example, if we use 5000 particles for a time series with $T = 100$, then one million particles need to be stored and flushed at every iteration. Second, the algorithm relies on the SMC approximation of the E-step. In other words, this implementation does not fully satisfy the convergence requirements of the EM algorithm, thus the algorithm may not even converge. These issues are much more noticeable for categorical time series; for continuous time series, the BPEM algorithm provides reasonably accurate estimators.

Algorithm 2: Brute-force Particle EM

Data: $\mathbf{y}$, $\boldsymbol{\theta}_{\text{init}}$
Result: $\boldsymbol{\theta}$
while *until converge* **do**
  (E-step) $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\} = \mathrm{BPF}(\mathbf{y}, \boldsymbol{\theta})$;
  (M-step)
  $\boldsymbol{\theta} = \max_{\boldsymbol{\theta}} \sum_{i,t} w_t^{(i)} (\log f(y_t \mid \mathbf{x}_t^{(i)}) + \log p_{\boldsymbol{\theta}}(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}))$
end

4 LAMORE algorithm

In this section, we introduce three novel techniques to address the scalability and stability issues of the BPEM algorithm, then combine these three components to obtain our **L**ow-resolution augmented **A**symptotic Mean (**Mo**ment) **R**egularized **E**M (LAMORE) algorithm.

4.1 Asymptotic Mean Regularization

Asymptotic Mean Regularization is a novel regularization technique based on the asymptotic property of a categorical time series. Without loss of generality, we assume that the LAVA-Cat model in this paper is stable. In other words, the root z of $\det(\mathbf{I} - \boldsymbol{\Phi} z) = 0$ lies outside the complex unit circle. Given that this condition is met, the latent VAR process is stationary, thus the categorical series is also stationary. However, two problems arise when we enforce the intermediate parameters in the EM algorithm to satisfy the stationarity condition. First, as long as the parameters are properly initialized, the intermediate parameters usually do not violate the stability condition, but the converged parameters are still far from the true parameters. Second, if the intermediate parameters violate the stability condition, several iterations are needed to obtain a solution for the constrained maximization step.

The asymptotic mean regularization is an auxiliary condition for the stationarity condition. The stationarity information can be incorporated through an indicator random variable $\mathbb{I}_s$ that is one if $\mathbf{y}$ is stationary, and zero otherwise. The joint log-likelihood including the stationarity indicator can be written as follows:

$$\begin{aligned}&\max_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{y}, \mathbb{I}_s = 1)\\ &= \max_{\boldsymbol{\theta}} \underbrace{\log p_{\boldsymbol{\theta}}(\mathbb{I}_s = 1 \mid \mathbf{y})}_{\text{Asymptotic Mean Regularization}} + \log p_{\boldsymbol{\theta}}(\mathbf{y})\end{aligned}$$

where the additional term $\log p_{\boldsymbol{\theta}}(\mathbb{I}_s = 1 \mid \mathbf{y})$ is the Asymptotic Mean (Moment) Regularization (AMOR) term. *The AMOR term explains the likelihood of being stationary given the parameter* $\boldsymbol{\theta}$. The stationary condition is smoothly integrated in the likelihood maximization process; we now maximize the likelihood of parameters as well as the likelihood of being stationary. However, the exact AMOR term is not trivial to obtain, and we approximate it as follows:

$$\begin{aligned}\log p_{\boldsymbol{\theta}}(\mathbb{I}_s = 1 \mid \mathbf{y}) &\propto -\gamma_{\text{AMOR}} \|\mathrm{E}[\mathbf{y}_t] - \frac{\sum \mathbf{y}_t}{T}\|^2\\ &\propto -\lambda_{\text{AMOR}} \|\mathrm{E}[\mathbf{x}_t] - \frac{\sum \mathbf{x}_t}{T}\|^2\end{aligned}$$

where $\mathrm{E}[\mathbf{x}_t]$ and $\sum \mathbf{x}_t / T$ denote the true and sample means of the latent vectors, respectively. This form was motivated from the fact that the sample mean of a stationary time series becomes very close to the true mean with enough samples.

The AMOR term does not need particle methods; the true and sample means can be directly obtained from the observations. The theoretical stationary mean of the latent VAR process can be obtained by taking expectation on both sides. As $\mathrm{E}[\mathbf{x}_{t-1}] = \mathrm{E}[\mathbf{x}_t]$ and $\mathrm{E}[\boldsymbol{\varepsilon}_t] = 0$, we obtain $\boldsymbol{\mu} = E[\mathbf{x}_t] = (\mathbf{I} - \boldsymbol{\Phi})^{-1}\mathbf{c}$. The stationary distribution of the categorical output becomes:

$$f_k(\boldsymbol{\mu}) = \mathrm{E}[\mathbb{I}(y = k)] \approx \frac{\sum_t \mathbb{I}(y_t = k)}{T} = \hat{y}_k$$

To make notation simple, let us define a variable $\hat{\mathbf{y}} = \begin{pmatrix} \hat{y}_1 & \dots & \hat{y}_{K-1} \end{pmatrix}^\top$. For the softmax link function, the empirical stationary mean $\hat{\boldsymbol{\mu}}$ for the latent process can be obtained as follows:

$$\hat{\mathbf{y}} = \frac{\exp \hat{\boldsymbol{\mu}}}{\mathbf{1}^\top \exp \hat{\boldsymbol{\mu}} + 1} \Rightarrow \quad \hat{\boldsymbol{\mu}} = \log((\mathbf{I} - \hat{\mathbf{y}} \mathbf{1}^\top)^{-1} \hat{\mathbf{y}})$$

According to the law of large numbers and the ergodicity of a stable VAR process, we have $\hat{\boldsymbol{\mu}} \to \boldsymbol{\mu}$ as $T \to$

∞. Thus, we formulate a practical AMOR term for the LAVA-Cat model as follows:

$$\begin{aligned}
&\log p_{\boldsymbol{\theta}}(\mathbb{I}_s = 1 \mid \mathbf{y}) \propto -\lambda_{\text{AMOR}} \|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}\|^2 \\
&= -\lambda_{\text{AMOR}} \|(\mathbf{I} - \boldsymbol{\Phi})^{-1}\mathbf{c} - \hat{\boldsymbol{\mu}}\|^2 \\
&= -\lambda_{\text{AMOR}} (\mathbf{I} - \boldsymbol{\Phi})^{-2} \|\mathbf{c} - (\mathbf{I} - \boldsymbol{\Phi})\hat{\boldsymbol{\mu}}\|^2
\end{aligned}$$

The M-step of the BPEM algorithm is now modified with this AMOR term:

$$\min_{\mathbf{c},\boldsymbol{\Phi}} \sum_{i,t} -w_t^{(i)} \log p_{\boldsymbol{\theta}}(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}) + \lambda_{\text{AMOR}} \|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}\|^2$$

where $\mathbf{x}_t^{(i)}$ represents the ith particle and $w_t^{(i)}$ is the corresponding weight at time t, respectively.

The asymptotic mean regularization is a regularization technique that utilizes additional stationarity information. This is, in fact, a natural extension of the likelihood function for stationary time series. Of course, our technique cannot be used for estimating the parameters for non-stationary time series. The intuition behind this method is to combine the EM algorithm with a Method of Moments approach, the Yule-Walker (YW) approach. Essentially, the asymptotic moment idea from the YW equation is imposed as a regularization term on the state variables.

4.2 Pseudo-Bayesian Update

The AMOR-appended maximization step presents two challenges. First, the number of particles can be fairly large (scalability issue). Second, the solution does not have a closed form. Our pseudo-Bayesian update addresses these two issues. In the Bayesian linear regression, a sequence of data is processed by updating the posterior distribution. The parameters of a linear regression can be learned sequentially as more data points are observed. This sequential update can be used to resolve the scalability issue. Furthermore, the non-linearity issue also can be addressed by slightly modifying the form of the sequential update.

The first step of the pseudo-Bayesian update is to introduce two auxiliary variables, $\mathbf{d}_t$ for $\mathbf{c}$ and $\boldsymbol{\Psi}_t$ for $\boldsymbol{\Phi}$, respectively. At each t, these auxiliary variables are sequentially updated by solving the following equation, namely *BAM* (Bayesian update with Asymptotic Mean regularization):

$$\begin{aligned}
\min_{\mathbf{d}_t,\boldsymbol{\Psi}_t} &\sum_i w_t^{(i)} \|\mathbf{x}_t^{(i)} - \boldsymbol{\Psi}_t \mathbf{x}_{t-1}^{(i)} - \mathbf{d}_t\|^2 \\
&+ \lambda_{\text{AMOR}} (\mathbf{I} - \boldsymbol{\Psi}_{t-1})^{-2} \|\mathbf{d}_t - (\mathbf{I} - \boldsymbol{\Psi}_t)\hat{\boldsymbol{\mu}}\|^2 \\
&+ \lambda_{\text{Bayes}} \|\mathbf{d}_t - \mathbf{d}_{t-1}\|^2 + \lambda_{\text{Bayes}} \|\boldsymbol{\Psi}_t - \boldsymbol{\Psi}_{t-1}\|^2
\end{aligned}$$

where λ_{AMOR} and λ_{Bayes} control the strengths of the asymptotic mean regularization and the Bayesian sequential update. The essence of the pseudo-Bayesian update is disintegrating the double summations $\sum_{i,t}$ to a single summation $\sum_i$ by sequentially updating the time-indexed auxiliary parameters. The term $(\mathbf{I} - \boldsymbol{\Psi}_{t-1})^{-2}$ is now indexed by $t-1$, and this is the key trick for obtaining an approximate closed-form solution. To get a closed form solution, we rearrange the terms as follows:

$$\begin{aligned}
\min_{\mathbf{d}_t,\boldsymbol{\Psi}_t} &\sum_i w_t^{(i)} \|\mathbf{x}_t^{(i)} - \begin{pmatrix}\mathbf{d}_t & \boldsymbol{\Psi}_t\end{pmatrix} \begin{pmatrix}1 \\ \mathbf{x}_{t-1}^{(i)}\end{pmatrix} \|^2 \\
&+ \underbrace{\lambda_{\text{AMOR}} (\mathbf{I} - \boldsymbol{\Psi}_{t-1})^{-2}}_{\lambda'_{\text{AMOR}}} \|\hat{\boldsymbol{\mu}} - \begin{pmatrix}\mathbf{d}_t & \boldsymbol{\Psi}_t\end{pmatrix} \begin{pmatrix}1 \\ \hat{\boldsymbol{\mu}}\end{pmatrix} \|^2 \\
&+ \lambda_{\text{Bayes}} \|\mathbf{d}_t - \mathbf{d}_{t-1}\|^2 + \lambda_{\text{Bayes}} \|\boldsymbol{\Psi}_t - \boldsymbol{\Psi}_{t-1}\|^2
\end{aligned}$$

Let us define $\mathbf{B}_t = \begin{pmatrix}\mathbf{d}_t & \boldsymbol{\Psi}_t\end{pmatrix} \in \mathbb{R}^{(K-1)\times K}$:

$$\begin{aligned}
&\min_{\mathbf{B}_t} \sum_i \| \sqrt{w_t^{(i)}} \mathbf{x}_t^{(i)} - \sqrt{w_t^{(i)}} \mathbf{B}_t \begin{pmatrix}1 \\ \mathbf{x}_{t-1}^{(i)}\end{pmatrix} \|^2 \\
&\quad + \lambda'_{\text{AMOR}} \|\hat{\boldsymbol{\mu}} - \mathbf{B}_t \begin{pmatrix}1 \\ \hat{\boldsymbol{\mu}}\end{pmatrix} \|^2 + \lambda_{\text{Bayes}} \|\mathbf{B}_t - \mathbf{B}_{t-1}\|^2 \\
\Rightarrow &\min_{\mathbf{B}_t} \| \begin{pmatrix}\mathbf{w}_t \mathbf{X}_t \\ \hat{\boldsymbol{\mu}}^\top\end{pmatrix} - \begin{pmatrix}\mathbf{w}_t^\top & \mathbf{w}_t \mathbf{X}_{t-1} \\ \lambda'_{\text{AMOR}} & \lambda'_{\text{AMOR}} \hat{\boldsymbol{\mu}}^\top\end{pmatrix} \mathbf{B}_t^\top \|^2 \\
&\quad + \lambda_{\text{Bayes}} \|\mathbf{B}_t - \mathbf{B}_{t-1}\|^2
\end{aligned}$$

where $\mathbf{w}_t = \begin{pmatrix}w_t^1 & \cdots & w_t^P\end{pmatrix}$. This form further reduces to:

$$\min_{\mathbf{B}_t} \| \underbrace{\begin{pmatrix}\mathbf{w}_t \mathbf{X}_t \\ \hat{\boldsymbol{\mu}}^\top \\ \lambda_{\text{Bayes}} \mathbf{B}_{t-1}^\top\end{pmatrix}}_{=\mathbf{S}} - \underbrace{\begin{pmatrix}\mathbf{w}_t^\top & \mathbf{w}_t \mathbf{X}_{t-1} \\ \lambda'_{\text{AMOR}} & \lambda'_{\text{AMOR}} \hat{\boldsymbol{\mu}}^\top \\ \lambda_{\text{Bayes}} & \mathbf{0} \\ \mathbf{0} & \lambda_{\text{Bayes}} \mathbf{I}\end{pmatrix}}_{=\mathbf{R}} \mathbf{B}_t^\top \|^2$$

Thus, the solution of this least square problem is given as follows:

$$\mathbf{B}_t^* = ((\mathbf{R}^\top \mathbf{R})^{-1} \mathbf{R}^\top \mathbf{S})^\top$$

where $\mathbf{R} \in \mathbb{R}^{(P+K+1)\times K}$ and $\mathbf{S} \in \mathbb{R}^{(P+K+1)\times(K-1)}$.

4.3 Low-resolution Augmentation

More observations usually help in reducing the variance of estimated parameters. The structure of the LAVA-Cat model can be utilized to obtain another set of observations, a low-resolution time series:

$$\mathbf{x}_t = \mathbf{c}_L + \boldsymbol{\Phi}_L \mathbf{x}_{t-2} + \boldsymbol{\xi}_t$$

where $\mathbf{c}_L = \mathbf{c}(\mathbf{I} + \boldsymbol{\Phi})$ and $\boldsymbol{\Phi}_L = \boldsymbol{\Phi}^2$. The transformed parameters $\mathbf{c}_L$ and $\boldsymbol{\Phi}_L$ can be estimated by maximizing $\log p_{h(\boldsymbol{\theta})}(g(\mathbf{y}), \mathbb{I}_s)$, where $g(\mathbf{y})$ and $h(\boldsymbol{\theta})$ are low-resolution transformed time series and parameters. If $\boldsymbol{\Phi} > 0$, the estimated parameters from this low-resolution signal are combined with the original parameters as follows:

$$\begin{aligned}
\boldsymbol{\Phi} &\approx (\sqrt{\hat{\boldsymbol{\Phi}}_L} + 2 \times \hat{\boldsymbol{\Phi}})/3 \\
\mathbf{c} &\approx (\hat{\mathbf{c}}_L (\mathbf{I} + \boldsymbol{\Phi})^{-1} + 2 \times \hat{\mathbf{c}})/3
\end{aligned}$$

where $\mathbf{c}_L$ and $\mathbf{\Phi}_L$ are inverse-transformed to match the original representation. The effects of combining the low-resolution parameters are two-fold; this procedure can reduce the Monte-Carlo sampling bias in the E-step, and the parameters are more consistent with the low-resolution property. This low-resolution augmentation is analogous to the multiresolution analysis that has been widely used in image processing and computer vision (Willsky, 2002). Although it is possible to drop the indices that are multiple of, say, 3 or 4, the derived signals would have 3 or 4 times less observations. For a finite time series that has less than 100 observations, obtaining such low-resolution signals does not add much to parameter estimation.

4.4 LAMORE Implementation

These three components form one iteration of the LAMORE algorithm, (see Algorithm 3). *Empirical-AM* is a procedure that computes the empirical mean of the latent variables, and *BAM* denotes the pseudo-Bayesian update for Asymptotic Mean Regularization. As can be seen, auxiliary variables, $\mathbf{d}_t, \boldsymbol{\psi}_t$ for the original sequence and $\mathbf{e}_t, \boldsymbol{\zeta}_t$ for the low-resolution sequence, are introduced to distribute the M-step across the particle filtering (MCMC E-step). Note that $\mathbf{e}_t$ and $\boldsymbol{\zeta}_t$ are updated every two time steps, then combined with the other auxiliary parameters to calculate the model parameters for the next iteration $\boldsymbol{\theta}$.

Algorithm 3: LAMORE algorithm

Data: $\mathbf{y}$, $\boldsymbol{\theta}_{\text{init}}$
Result: $\boldsymbol{\theta}$
$\hat{\boldsymbol{\mu}} = \text{Empirical-AM}(\mathbf{y})$;
while *until converge* **do**
  Initialize $\{\mathbf{d}_0, \boldsymbol{\zeta}_0\} = \boldsymbol{\theta}$ and $\{\mathbf{e}_0, \boldsymbol{\zeta}_0\} = \boldsymbol{\theta}$;
  for $t \in$ *1:T* **do**
    $\{x_t^{(i)}, w_t^{(i)}\} = \text{1Step-BPF}(y_t, \boldsymbol{\theta}, \{x_{t-1}^{(i)}, w_{t-1}^{(i)}\})$;
    $\mathbf{d}_t, \boldsymbol{\psi}_t = \text{BAM}(\{x_t^{(i)}, w_t^{(i)}\}, \mathbf{d}_{t-1}, \boldsymbol{\psi}_{t-1})$;
    if *t % 2 == 0* **then**
      $\mathbf{e}_t, \boldsymbol{\zeta}_t = \text{BAM}(\{x_t^{(i)}, w_t^{(i)}\}, \mathbf{e}_{t-2}, \boldsymbol{\zeta}_{t-2})$;
    end
  end
  Set $\mathbf{c} = (2\mathbf{d}_T + \mathbf{e}_T)/3$ and $\mathbf{\Phi} = (2\boldsymbol{\psi}_T + \boldsymbol{\zeta}_{T/2})/3$;
  Set $\boldsymbol{\theta} = \{\mathbf{c}, \mathbf{\Phi}\}$;
end

5 Empirical Evaluation

In this section, three experimental results are provided. The first experiment uses simulated data to verify whether the estimated parameters are more reliable than the brute-force EM algorithm. The other two experiments illustrate potential applications of the LAVA-Cat model and the proposed LAMORE algorithm.

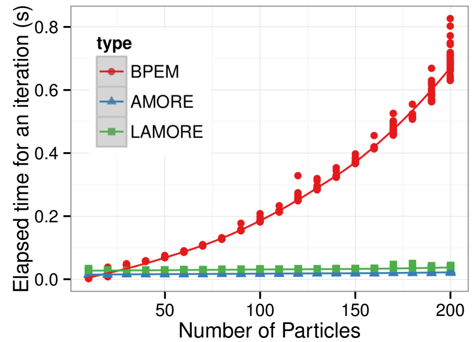

Figure 1: Run-time vs. Number of Particles. The BPEM algorithm does not scale with the number of particles.

5.1 Simulation Study

Simulated time series data are generated from the LAVA-Cat model with random parameters $\mathbf{c}$ and $\mathbf{\Phi}$. First, we measured the run-times of one EM iteration from three algorithms that are implemented in the `R` programming language: *BPEM*, *AMORE* (LAMORE without low-resolution augmentation), and *LAMORE*[1]. Figure 1 shows the results from the experiment. As can be seen, the BPEM algorithm scales poorly with respect to the number of particles, whereas the other two algorithms maintain almost constant run-times. In fact, the BPEM algorithm does not run with a thousand particles, facing memory allocation issues in a 4GB-memory machine . For the rest of the paper, the *Baseline* algorithm refers to a LAMORE without both asymptotic mean regularization and low-resolution augmentation.

Next, we check whether AMORE and LAMORE can stabilize estimated parameters. Figure 2 and 3 show the estimated parameters over the three different EM algorithms. The E-step using particle methods is inherently noisy, and the baseline algorithm often diverges from the true parameter values. On the other hand, although the estimators fluctuate, those two asymptotic mean regularized EM algorithms converge near to the true values. Although we only showed the result for a binary time series, the stability of the LAMORE algorithm is more noticeable when a time series becomes tertiary and more ($K > 2$) i.e. high-dimensional parameters.

The (L)AMORE algorithms provide more accurate

[1]The LAMORE implementation in `R` is available upon request. A GitHub URL for the LAMORE code is not provided to keep anonymity of the authors.

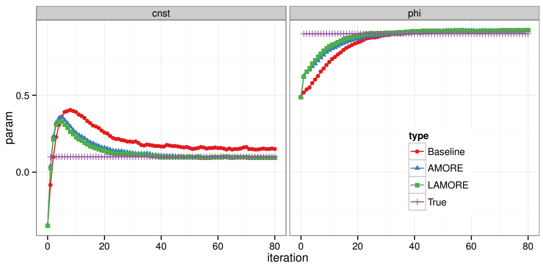

Figure 2: Estimated Parameters (c, ϕ) vs. EM iteration on a simulated binary time series. The true values are shown in purple lines.

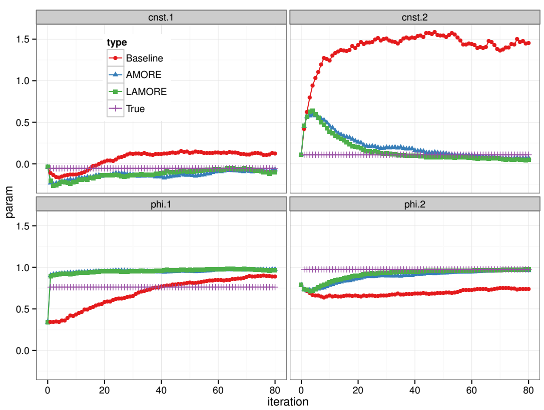

Figure 3: Estimated Parameters $(\mathbf{c}, \mathbf{\Phi})$ vs. EM iteration on a simulated tertiary time series, where $\mathbf{\Phi}$ is a diagonal matrix.

and stable estimators than the baseline algorithm. Simulated tertiary categorical time series ($K = 3$) were generated with random parameter initialization, and then we measured the Mean Squared Errors between the estimated parameters and the true parameters. Figure 4 shows the results from 30 different runs per fixed length categorical time series. As can be seen, the estimators from (L)AMORE are the closest to the true value and also exhibit the smallest variances. Noticeably, AMORE also provides better estimators than the baseline.

The proposed two techniques are, in fact, general techniques that can be applied to non-categorical data. This is now illustrated using a continuous multivariate time series that is generated as $\mathbf{y}_t = \mathbf{x}_t + \boldsymbol{\eta}_t$, where $\mathbf{x}_t$ is generated by the same latent vector autoregressive model. As $\mathrm{E}[\mathbf{y}_t] = \mathrm{E}[\mathbf{x}_t]$, the asymptotic mean regularization can be directly applied. Figure 5 shows the results from 30 different runs per fixed length continuous time series. Note that the optimal EM solution can be obtained by using KF, not BPF, and the goal of this example is to show the effectiveness of the two methods in a different setting.

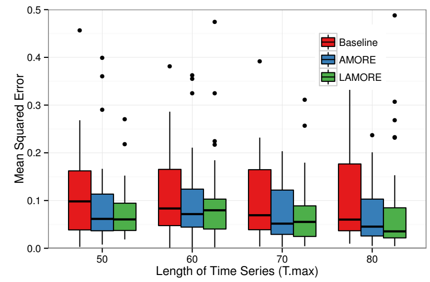

Figure 4: Estimation Performance of (L)AMORE vs. length of finite categorical time series. Box-plots are drawn based on 30 different simulation trajectories.

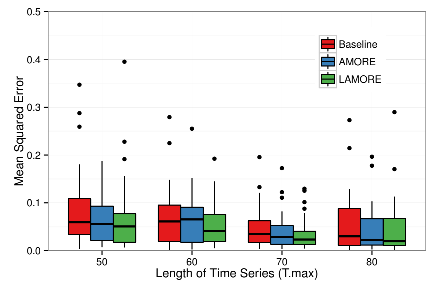

Figure 5: [Continuous Time Series] Estimation Performance of (L)AMORE vs. length of finite categorical time series. .

5.2 Case Study I: Medicare Part-D

The following two subsections show potential applications of the LAVA-Cat model and the LAMORE algorithm. We first attempt to measure the inertia of drug re-purchases in the Medicare part-D program. The target population of the Medicare part-D program tends to have at least one chronic condition, and they regularly purchase drugs with the aid of the insurance program. For this experiment, we use the synthetic Medicare part-D claim records[2] that are generated based on 2007 and 2010 5% US population data. From our data exploration, we found that two major pharmaceutical companies are dominant in the Medicare drug market; Novartis and Teva. For each beneficiary, we extracted a drug purchase sequence with three categories: Novartis, Teva, and the others. For the ease of interpretation, we restricted the form of $\mathbf{\Phi}$ to be a diagonal matrix as $\mathbf{\Phi} = \begin{pmatrix} \phi_{\text{Novartis}} & 0 \\ 0 & \phi_{\text{Teva}} \end{pmatrix}$. The diagonal component of $\mathbf{\Phi}$ can be interpreted as the inertia to re-purchase the same brand drug.

We separately estimated the LAVA-Cat parameters from each individual. Each individual in the dataset has a different $\mathbf{\Phi}$ (re-purchase inertia), and Figure 6 shows the distribution of the estimated parameters $\mathbf{\Phi}$

[2] http://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/SynPUFs/

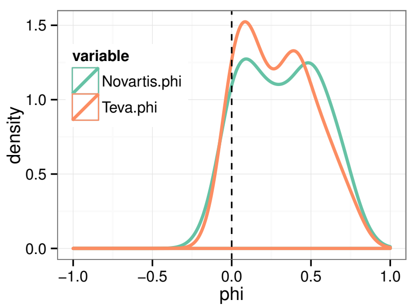

Figure 6: Purchase inertia parameters in the Medicare part-D program.

using LAMORE. The distributions indicate a mixture of two groups; one group who are loyal to their previous purchases (loyal beneficiaries), and the other group who almost randomly change their choices regardless of their previous purchases (random-choice beneficiaries). As can be seen, Novartis has more loyal beneficiaries than Teva (interestingly, Novartis has a slightly bigger market share than Teva).

In addition to the interpretability of the LAVA-Cat model, the model provides better predictive performance than traditional lagged variable models. Without using latent variables, multinomial logistic regression can be used to predict future events:

$$\log \frac{p(y_t = k)}{p(y_t = K)} = \beta_0^k + \beta_1^k y_{t-1} + \ldots + \beta_L^k y_{t-L}$$

where L is the maximum lagging interval (in our experiment, $L = 2$). $K - 1$ binomial logistic regression models are built to predict future events. As lagged features are highly correlated, we use an elastic-net multinomial logistic regression using the `glmnet` package (Friedman et al., 2010). Figure 7 shows the performance comparison with this regularized multinomial logistic regression. For a drug purchase time series, the initial 80 % of the observations are used as a training set, and the rest as a test set. As can be seen, the LAVA-Cat model with LAMORE provides reasonable predictive accuracies, while the logistic regression almost fails to predict future drug purchases.

5.3 Case Study II: MIMIC-II

In this example, we attempt cardiac condition prediction using the MIMIC-II dataset (Saeed et al., 2011). The MIMIC-II is the most extensive publicly available intensive care unit resource. Our data exploration found that two cardiac conditions are observed frequently: Sinus Tachycardia (SinusTachy) and Atrial Fibrillation (AtrialFib). We compare the predictive performance of the LAVA-Cat model with the baseline logistic model (see Figure 8). The LAVA-Cat model

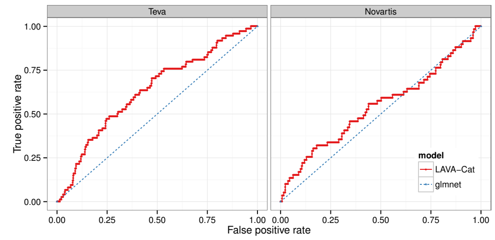

Figure 7: Medicare Drug purchase predictive performance of LAVA-Cat using LAMORE. Each cell shows Area under Receiver Operating Characteristic curves (AUROC) for one-versus-all settings. The blue dotted-lines are the performance curves from the `glmnet` algorithm using lagged features.

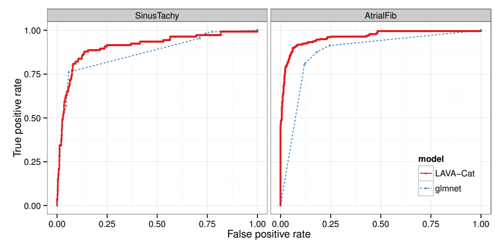

Figure 8: Future cardiac condition predictive performance of LAVA-Cat using LAMORE. Each cell shows AUROCs for one-versus-all settings. The blue dotted-lines indicate the performance of the `glmnet` algorithm.

can predict the future cardiac condition more accurately than the logistic regression model.

6 Discussion

This paper introduced two auxiliary parameter estimation techniques for a state-space categorical time series model: asymptotic mean regularization and low-resolution augmentation. These two methods have shown their effectiveness in the simulation experiments. The experiments with the real datasets showed various potential applications of the proposed methods.

We assumed that a time series is stationary, and the model parameters are unknown but fixed over time. For dynamically changing parameters, Markov switching models can potentially be applied to the LAVA-Cat model. Extensions of our approaches to non-stationary time series are left as future work.

Acknowledgment

We thank Joyce C. Ho, Kang Bok Lee, and Russell Zaretzki for helpful discussions. This research is funded by NSF IIS-1017614.

References

C. Andrieu, A. Doucet, and V. B. Tadic. Online parameter estimation in general state-space models. In *Proc. IEEE CDC/ECC*, 2005.

G. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis: Forecasting and Control*. Prentice-Hall, third edition, 1994.

J. P. Burg. Maximum entropy spectral analysis. In *Proc. 37th Meeting of the Society of Exploration Geophysicists*, 1967.

F. Canova and M. Cicarelli. Panel vector autoregressive models: A survey. *European Central Bank: Working Paper Series*, 2013.

C. M. Carvalho, M. S. Johannes, H. F. Lopes, and N. G. Polson. Particle learning and smoothing. *Statistical Science*, 25(1):88–106, 2010.

A. Doucet and A. M. Johansen. A Tutorial on Particle Filtering and Smoothing: Fifteen years later, Dec. 2008.

A. Doucet and V. B. Tadic. Parameter estimation in general state-space models using particle methods. *Annals of the Institute of Statistical Mathematics*, 55(Issue 2):409–422, 2003.

R. F. Engle. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica*, 50(4):987–1007, 1982.

K. Fokianos and B. Kedem. Regression theory for categorical time series. *Statistical Science*, 18(3):357–376, 2003.

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33 (1):1–22, 2010.

S. Gadzhanova, I. Iankov, J. R. Warren, J. Stanek, G. M. Misan, Z. Baig, and L. Ponte. Developing high-specificity anti-hypertensive alerts by therapeutic state analysis of electronic prescribing records. *Journal of American Medical Informatics Association*, 14:100–109, 2007.

N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140:107–113, 1993.

P. A. Jacobs and P. A. W. Lewis. Stationary discrete autoregressive-moving average time series generated by mixtures. *Journal of Time Series Analysis*, 4(1): 19–36, 1983.

B. H. Juang and L. R. Rabiner. Hidden Markov Models for speech recognition. *Technometrics*, 33(3): 251–272, 1991.

S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

N. Kantas, A. Doucet, S. S. Singh, and J. M. Maciejowski. An overview of sequential monte carlo methods for parameter estimation in general state-space models. In *15th IFAC Symposium on System Identification*, 2009.

H. Kaufmann. Regression models for nonstationary categorical time series: asymptotic estimation theory. *The Annals of Statistics*, 15(1):79–98, 1987.

Y. Kou, C.-T. Lu, S. Sirwongwattana, and Y.-P. Huang. Survey of fraud detection techniques. In *Proc. IEEE International Conference on Networking, Sensing and Control*, volume 2, pages 749–754, 2004.

R. B. Litterman. Specifying vector autoregressions for macroeconomic forecasting. *Federal Reserve Bank of Minneapolis Staff report*, (92), 1984.

J. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of American Statistical Association*, 93:1032–1044, 1998.

J. Liu and M. West. Combined parameters and state estimation in simulation-based filtering. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*, 2001.

R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 355–368, 1998.

M. K. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of American Statistical Association*, 94:590–599, 1999.

A. E. Raftery. A model for high-order markov chains. *Journal of the Royal Statistical Society. Series B (Methodological)*, 47(3):528–539, 1985.

M. Saeed, M. Villarroel, A. T. Reisner, G. Clifford, L.-W. Lehman, G. Moody, T. Heldt, T. H. Kyaw, B. Moody, and R. G. Mark. Multiparameter Intelligent Monitoring in Intensive Care II: A public-access intensive care unit database. *Critical Care Medicine*, 39(5):952–960, May 2011.

R. H. Shumway and D. S. Stoffer. An appoach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, 3(4): 253–264, 1982.

G. Storvik. Particle filters in state-space models with the presence of unknown static parameters. *IEEE Trans. Signal Processing*, 50:281–289, 2002.

A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Information Theory*, 13(2):260–269, 1967.

A. S. Willsky. Multiresolution markov models for signal and image processing. *Proceedings of the IEEE*, 90(8), 2002.

S. L. Zeger, K.-Y. Liang, and P. S. Albert. Models for longituidinal data: A generalized estimating equation approach. *Biometrics*, 44:1049–1060, 1988.

X. Zhen and I. V. Basawa. Observation-driven generalized state space models for categorical time series. *Statistics and Probability Letters*, 79:2462–2468, 2009.

A. Zia, T. Kirubarajan, J. P. Reilly, D. Yee, K. Punithakumar, and S. Shirani. An EM algorithm for nonlinear state estimation with model uncertainties. *IEEE Trans. Signal Processing*, 56:921–936, 2008.

W. Zucchini and I. L. MacDonald. *Hidden Markov Models for Time Series: An Introduction Using R*. Chapman and Hall/CRC, 2009.