Sequential Demand Forecasting of Bursty Count Data

Kaoru Irie*, Chris Glynn†, and Tevfik Aktekin‡

Abstract

Existing methods for sequentially analyzing count data typically utilize a discounting strategy, where the contribution of past observations to parameter updates diminishes with elapsed time. Discount factor techniques offer an intuitive approach to sequentially updating parameters with weighted contributions from all previously observed data; however, when the time series undergoes a sudden change and the observed count significantly jumps, parameter estimates are slow to adapt, as they are heavily informed by data observed prior to the structural break. Immediately following such bursts, predictive performance degrades. In this study, we introduce an augmented Poisson-gamma state-space (PGSS) model whose state evolution structure is flexible and responsive to sudden changes in the level of counts, focusing on consumer demand settings where sequential and online learning/forecasting are of great interest. Such adaptability is achieved by augmenting the state vector of the PGSS model with an additional state variable for a time-varying discount factor. We develop an efficient particle-based estimation procedure that is suitable for sequential analysis, allowing us to estimate dynamic state variables and static parameters via closed form conditional sufficient statistics. To illustrate how the state-augmented PGSS model performs with data that exhibit bursts, we present results from a simulation study and two case studies to monitor and forecast web traffic and ridesharing demand. We show that – in the presence of structural breaks – our proposed approach yields superior sequential model fit and predictive performance compared to viable alternatives.

1 Introduction

Structural changes in count-valued time series are pervasive in the digital economy. For instance, demand for Uber rides may exhibit a sudden burst at uncommon times due to the conclusion of a sporting event or concert. Surges in web traffic on Facebook, Instagram, Twitter, and Google – while often due to predictable intraday variation – are occasionally driven by unanticipated news events. At call centers and online help desks for insurance companies, unexpected natural disasters and severe weather may result in dramatic increases in the number of customers requiring service. When business operations of these web platforms depend on accurate short-term predictions of consumer demand, the ability to quickly identify structural breaks and adjust forecasts of customer counts is critical.

Markov switching models have traditionally been used to model regime changes in time series data; however, on e-commerce platforms, consumer demand changes rapidly, and resources are re-allocated frequently. In high-frequency applications, the computational cost of fitting Markov switching models is prohibitive. Balancing model complexity with timely decision-making necessitates novel modeling and computational strategies for high-frequency, bursty count data.

Forecasting bursty count data in high-frequency settings poses a number of statistical and computational challenges. The primary challenge is to flexibly model temporal dependence in a way that facilitates rapid and online estimation of model parameters. There are two main approaches for modeling count time series: The first assumes that time-varying counts are generated by a stationary stochastic process

*Department of Economics, University of Tokyo, irie@e.u-tokyo.ac.jp
†Paul College of Business and Economics, University of New Hampshire, christopher.glynn@unh.edu
‡Paul College of Business and Economics, University of New Hampshire, tevfik.aktekin@unh.edu

(Freeland and McCabe, 2004); the second approach models temporal dependence via state-space models and allows for the possibility that counts are non-stationary (Harvey and Fernandes, 1989; Fruhwirth-Schnatter and Wagner, 2006; Gamerman et al., 2013; Chen et al., 2018b,a; Aktekin et al., 2018; Berry and West, 2018; Glynn et al., 2018). The state-space approach exploits the conditional independence of counts given that state parameters themselves follow a stochastic process, inducing temporal dependence in counts marginally; see Prado and West (2010) and Davis et al. (2015) for recent reviews of state-space models and time series of counts.

The Poisson-gamma state-space (PGSS) model (Aktekin et al., 2013; Chen et al., 2018b) is a popular choice for modeling time-varying count data, since the Poisson-gamma conjugacy admits online, closed-form calculation of posterior and forecast distributions. The PGSS model is one in a broader class of gamma-beta random walk models for Poisson rates. The gamma-beta state transition was first introduced by Smith and Miller (1986) for state-space models with exponential likelihoods and was later utilized to model stochastic volatility in financial markets by Uhlig (1994, 1997). Recently, the same state transition structure has been used to model a general class of non-Gaussian state space models (Gamerman et al., 2013). One attractive feature common to gamma-beta random walk models is that the beta-distributed innovations in the state equation yield a state variable that is marginally gamma-distributed (assuming that the initial state prior is also gamma-distributed), leading to closed-form updates of posterior and forecast distributions in the PGSS model. While online, analytically available posterior and predictive distributions are attractive features, the single-process PGSS model is unable to capture sudden bursts or regime switches in counts. The lack of flexibility in the PGSS model stems from the static discount parameter used in defining state transitions.

In this paper, we develop the state-augmented PGSS (sa-PGSS) model, an integer-valued state-space model whose structure flexibly adapts to newly observed counts and admits a sequential Monte Carlo algorithm for online updates of posterior and one-step-ahead predictive distributions. It advances the literature on Poisson-gamma state-space (PGSS) models by introducing a mechanism to sequentially adapt model structure as called for by data. Specifically, we augment the state-variable in the PGSS model with a dynamic discount factor that enables rapid model adaption to structural changes in observed counts. The methodological novelty of our approach stems from this state variable augmentation, which increases the flexibility of the PGSS model while allowing us to develop a fast estimation algorithm suitable for sequential parameter learning and demand forecasting. The sa-PGSS model adapts to streaming counts so that during bursts in demand, Bayesian prior distributions are more diffuse and forecasts rely more heavily on recently observed data.

The sa-PGSS model requires fast and efficient computational strategies for online updates of posterior and predictive distributions. As pointed out by Storvik (2002) and Carvalho et al. (2010a), traditional Markov chain Monte Carlo (MCMC) methods, especially the forward filtering backward sampling (FFBS) algorithm of Carter and Kohn (1994) and Fruhwirth-Schnatter (1994), are computationally expensive, as state variables must be re-estimated each time new data is observed. With this in mind, we develop a particle-based algorithm that allows us to update static as well as the dynamic (state) variables in a fast sequential manner. The initial idea of particle filtering (PF) dates back to the work of Gordon et al. (1993). Since then there have been several successful applications of the PF algorithm in various settings such as those discussed in Carvalho et al. (2010b) for general mixtures, Gramacy and Polson (2011) for Gaussian process models in sequential optimization, Lopes and Polson (2016) for fat-tailed distributions, and Prado and Lopes (2013) for estimating parameters in autoregressive time series models. One of challenges common to all PF applications is the particle degeneracy issue that arises in learning static parameters. We overcome this issue by obtaining the conditional sufficient statistics for the static parameters in a similar vein to the methods proposed by Storvik (2002); Carvalho et al. (2010a); and Prado and Lopes (2013). For recent surveys of particle-based methods, we refer readers to the works of

Lopes and Tsay (2011) and Singpurwalla et al. (2018).

We illustrate the utility of the sa-PGSS model with three case studies. First, we investigate the model's ability to adapt to structural changes in simulated data (Section 5.1). Second, we fit the sa-PGSS model to traffic on the Fox News website, demonstrating improved one-step-ahead forecasts compared to the static discount factor in existing PGSS implementations. Third, we use sa-PGSS to forecast Uber ride requests in New York City. While we discuss these case studies in detail, it is important to note that our approach is general and can be applied to many other settings where forecasts of relatively high-frequency and bursty count data are of interest.

The remainder of our paper is structured as follows. In Section 2, we summarize key components of the PGSS model with a static discount factor. We highlight model properties and illustrate the inability of the base PGSS model to rapidly adapt to structural breaks with the Fox News web-traffic data. In Section 3, we introduce the state augmented sa-PGSS model with an autoregressive process for the dynamic discount factor. In Section 4, we develop a custom particle-based algorithm for estimating both static and dynamic parameters in the sa-PGSS model. Section 5 discusses the numerical analysis of simulated data, web traffic from Fox News, and Uber demand in New York City. Section 6 concludes with a brief summary and discussion of future directions.

2 Poisson-Gamma State Space (PGSS) Model

In this section, we introduce necessary notation and the conjugacy preliminaries for the standard PGSS model, which yields tractable filtering as well as one-step-ahead predictive densities. Let N_t for $t = 1, \ldots, T$ represent a univariate time series of counts and $\mathcal{D}_t = \{N_1, \ldots, N_t\}$ a collection of these counts until time t. The likelihood (observational equation) is defined by the Poisson distribution,

$$(N_t|\theta_t) \sim Po(\theta_t), \tag{1}$$

where, given θ_t, N_t is assumed to be conditionally independent of N_{t-1}. Temporal dependence of N_t on N_{t-1} is governed by the stochastic evolution of θ_{t-1} to θ_t. The state transition (evolution) equation follows a multiplicative gamma-beta random walk. Conditional on θ_{t-1} and $\mathcal{D}_{t-1}$,

$$\theta_t = \theta_{t-1}\eta_t/\gamma, \qquad \eta_t \sim Beta(\gamma\alpha_{t-1}, (1-\gamma)\alpha_{t-1}), \tag{2}$$

which implies a state transition equation given by

$$(\theta_t|\theta_{t-1}, \gamma, \mathcal{D}_{t-1}) \sim ScaledBeta(\gamma\alpha_{t-1}, (1-\gamma)\alpha_{t-1}), \tag{3}$$

for $\theta_t \in (0, \theta_{t-1}/\gamma)$, $\alpha_{t-1} > 0$, and $0 < \gamma < 1$. The shape parameter, α_{t-1}, is the function of the past observations $\mathcal{D}_{t-1}$ in general, and its specific functional form is given later. We note here that the state transition density (3) is a function of the past observations, $\mathcal{D}_{t-1}$, unlike traditional linear state space models. Here, γ is referred to as the discount factor and controls the persistence of the state variables. For instance, when $\gamma \uparrow 1$, θ_t and θ_{t-1} will be similar (strong dependence and persistence). Whereas, when $\gamma \downarrow 0$, θ_t and θ_{t-1} will likely be less similar, implying more volatile state dynamics (weak dependence and persistence).

Various versions of the PGSS model have been considered in the literature. Gamerman et al. (2013) consider a general class of non-Gaussian state-space models where the Poisson sampling model appears as a special case. Aktekin et al. (2013) consider it for modelling mortgage default counts, Chen et al. (2018b) utilize it to model web traffic in network flow data, and Aktekin et al. (2018) extend it to account for multivariate time series of counts. Further details of the PGSS model can be found in these papers and the references therein. In what follows, we provide a summary of some of the relevant results of the

PGSS model. Given the initial state prior of $\theta_0 \sim Ga(\alpha_0, \beta_0)$, we can show (i) the time $t-1$ posterior distribution $(\theta_{t-1}|\gamma, \mathcal{D}_{t-1})$ is gamma-distributed

$$(\theta_{t-1}|\gamma, \mathcal{D}_{t-1}) \sim Ga(\alpha_{t-1}, \beta_{t-1}); \tag{4}$$

(ii) the prior distribution for the state variable at time t is a discounted version of equation (4), inflating the prior variance of θ_t relative to the posterior at $t-1$,

$$(\theta_t|\gamma, \mathcal{D}_{t-1}) \sim Ga(\gamma\alpha_{t-1}, \gamma\beta_{t-1}); \tag{5}$$

(iii) the time t posterior is also gamma-distributed

$$(\theta_t|\gamma, \mathcal{D}_t) \sim Ga(\alpha_t, \beta_t), \tag{6}$$

where $\alpha_t = \gamma\alpha_{t-1} + N_t = \sum_{s=0}^{t-1} \gamma^s N_{t-s} + \gamma^t \alpha_0$ and $\beta_t = \gamma\beta_{t-1} + 1 = \frac{1-\gamma^t}{1-\gamma} + \gamma^t\beta_0$; observe that α_t combines a γ-discounted shape parameter from the posterior of θ_{t-1} and the most recently observed N_t, while β_t increments the γ-discounted rate parameter from the posterior of θ_{t-1} by one to reflect an additional data point; (iv) the one-step-ahead predictive distribution is Negative Binomial,

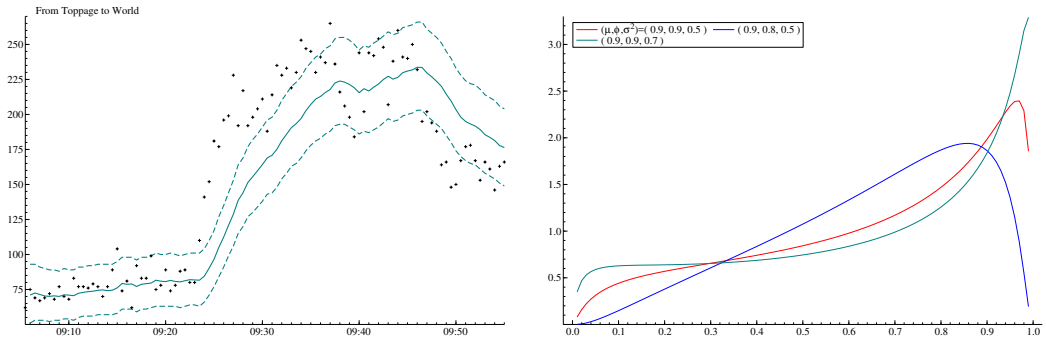
$$(N_t|\gamma, \mathcal{D}_{t-1}) \sim NegBin(\gamma\alpha_{t-1}, \frac{\gamma\beta_{t-1}}{\gamma\beta_{t-1}+1}). \tag{7}$$

Conditional on γ, the filtering density $p(\theta_t|\gamma, \mathcal{D}_t)$ and the one-step-ahead predictive density $p(N_t|\gamma, \mathcal{D}_{t-1})$ are available in closed form, which makes the PGSS model attractive for practical applications of streaming count data. Another noteworthy property of the PGSS model is the closed form availability of the marginal likelihood that can be used to estimate static model parameters like γ. Typically these marginal likelihoods cannot be obtained analytically outside of linear and Gaussian models such as the well-known dynamic linear model (West and Harrison, 1986, 1997). With the negative binomial one-step ahead densities in 7, we can construct the marginal likelihood from the product

$$p(\mathcal{D}_T|\gamma) = \prod_{t=1}^{T} p(N_t|\gamma, \mathcal{D}_{t-1}) = \prod_{t=1}^{T} \frac{\Gamma(\gamma\alpha_{t-1} + N_t)}{N_t!\Gamma(\gamma\alpha_{t-1})} \left(\frac{\gamma\beta_{t-1}}{\gamma\beta_{t-1}+1}\right)^{\gamma\alpha_{t-1}} \left(\frac{1}{\gamma\beta_{t-1}+1}\right)^{N_t}. \tag{8}$$

If we do not fix γ but treat it as a parameter to be estimated, the sequential analysis of posterior and predictive distributions becomes more complicated. For any given continuous prior choice of γ, it is not possible to obtain an analytically tractable posterior analysis. However, given (8) we can obtain a discrete posterior distribution for γ if we assume a discrete prior defined over the region $(0, 1)$. Alternatively, we can compute a point estimate of γ by maximizing (8). In both cases, the computations are straightforward and fast.

When γ is static, regardless of whether it is treated as a tuning parameter or a parameter to be estimated, the PGSS model is slow to adapt to structural changes in counts. Such a structural change is illustrated in Figure 1a, where a sudden surge in web traffic on the Fox News website occurs at approximately 9:25 AM (a similar graphic is presented in Figure 14 of Chen et al. (2018b), which contains a full PGSS analysis of the Fox News data). Observe that the median of the one-step-ahead predictive distribution (solid green line) fails to rapidly adapt to the surge. In fact, the predictive distribution from the PGSS model effectively smooths the web traffic data due to the discount structure in (7). When volume surges at 9:25, the PGSS model underpredicts traffic, and when the number of visitors drops after 9:45 AM, the PGSS model overpredicts traffic. In both directions, the predictions are sluggish in responding to rapid changes in observed data. This is largely due to the static treatment of γ. During the stable period from 9:00 to 9:25, the data provides evidence for a reasonably high value of γ, and past counts



(a) Static γ (b) Prior for γ_t

Figure 1: Left: The one-step ahead predictive distribution of N_t (web traffic flow data with 30 second time intervals) with static γ. The optimal value of γ is computed by the Empirical Bayes method that makes use of the closed form availability of the marginal likelihood. The solid line shows the median of the one-step-ahead predictions and the dashed lines represent the 90% predictive intervals. Right: The stationary marginal density of γ_t for different choices of hyperparameters: $(\mu, \phi, \sigma^2) = (\text{logit}(0.9), 0.9, 0.5)$ (red), $(\mu, \phi, \sigma^2) = (\text{logit}(0.9), 0.8, 0.5)$ (blue) and $(\mu, \phi, \sigma^2) = (\text{logit}(0.9), 0.9, 0.7)$ (green).

significantly contribute to forecasts $N_{t+1}|\gamma, \mathcal{D}_t$. This feature – a strength from 9:00 to 9:25 – becomes a weakness when a surge in traffic occurs. The high value of γ gives significant weight to past counts in one-step-ahead forecasts, but the forecasts fail to adequately adapt to the structural change in the time series. At 9:25, a small γ is needed so that less information is inherited from past counts and forecasts rapidly adapt to recently observed data. We view this static γ as a major shortcoming of the PGSS model. In Section 3, we augment the state variable with a dynamic discount factor γ_t that adaptively weights previous information based on predictive errors, providing increased model flexibility when structural changes occur.

3 The state-augmented PGSS model

In this section, we extend the PGSS model to account for dynamic changes in the discount factor, γ. In doing so, we preserve the properties of the base PGSS model conditional on the dynamic discount factor. The motivation for modeling γ as dynamic stems from the lack of adaptability of the PGSS model to sudden shifts in the time series of counts as evidenced by the behavior from Figure 1a. This adaptability can be achieved by allowing γ_t to be relatively large in stable regions and small in regions where sudden shifts occur without the need for prospective intervention as in the case of Chen et al. (2018b).

Assuming the same Poisson observation equation (1), we define the state evolution conditional on $\gamma_{1:t} = \{\gamma_1, \ldots, \gamma_t\}$, as $p(\theta_t|\theta_{t-1}, \mathcal{D}_{t-1}, \gamma_{1:t})$ which will be

$$(\theta_t|\theta_{t-1}, \mathcal{D}_{t-1}, \gamma_{1:t}) \sim ScaledBeta(\gamma_t\alpha_{t-1}, (1-\gamma_t)\alpha_{t-1}) \text{ where } \theta_t \in (0, \theta_{t-1}/\gamma_t). \tag{9}$$

We note that the state equation depends on all the past observations $\mathcal{D}_{t-1}$ and discount factors $\gamma_{1:(t-1)}$ whose contributions are embedded in α_{t-1}. Assuming the same initial state prior as before, $\theta_0 \sim Ga(\alpha_0, \beta_0)$, the online state updating conditional on $\gamma_{1:t}$ will be $(\theta_t|\gamma_{1:t}, \mathcal{D}_t) \sim Ga(\alpha_t, \beta_t)$ where

$$\begin{aligned} \alpha_t &= \gamma_t\alpha_{t-1} + N_t, \\ \beta_t &= \gamma_t\beta_{t-1} + 1 \end{aligned} \tag{10}$$

Similarly, the one-step ahead predictive density can be shown to follow

$$(N_t|\gamma_{1:t}, \mathcal{D}_{t-1}) \sim NegBin\left(\gamma_t\alpha_{t-1}, \frac{\gamma_t\beta_{t-1}}{\gamma_t\beta_{t-1}+1}\right). \tag{11}$$

The dynamic nature of the discount factors can be described by any Markovian process such as

$$(\gamma_t|\gamma_{1:(t-1)}) \sim p(\gamma_t|\gamma_{t-1}),$$

which needs to be selected carefully such that it facilitates sequential estimation but at the same time is flexible enough to increase the adaptability of the PGSS model. With this in mind and the fact that γ_t's are defined between 0 and 1, we consider a logistic transformation of the following form

$$g_t = \text{logit}(\gamma_t) = \log\frac{\gamma_t}{1-\gamma_t},$$

where the transformed series, $g_1, g_2, \ldots, g_{t-1}, g_t, \ldots$, follows a first order autoregressive model as in

$$g_t = (1-\phi)\mu + \phi g_{t-1} + N(0, \sigma^2). \tag{12}$$

We take a fully Bayesian point of view and assume priors on the above AR(1) triplet, (μ, ϕ, σ^2), and allow them to be updated sequentially in the face of new count data, substantially increasing temporal adaptability of the PGSS model. To be more specific, for another parametrization $\phi_0 = (1-\phi)\mu$, $\phi_1 = \phi$ and $w = \sigma^{-2}$, we assume the following normal-inverse gamma distribution as the hyper-prior

$$p(\phi_0, \phi_1, w|\mathcal{D}_0) = N(\phi_0, \phi_1|m_0, C_0/w)\; Ga(w|a_0/2, b_0/2),$$

As we see in the next section, this prior is conditionally conjugate in our model.

If the distribution of g_0 is $g_0 \sim N(\mu, \sigma^2/\sqrt{1-\phi^2})$, the g_t process is stationary with marginal distribution $g_t \sim N(\mu, \sigma^2/\sqrt{1-\phi^2})$, which indirectly implies the stationary distribution of γ_t by inverse logistic transformation. We show the implications of various values of (μ, ϕ, σ^2) on γ and what they represent in Figure 1b; the random variables are generated from the stationary normal distribution and transformed into the exponential scale to draw the histograms.

While the discount factor is now allowed to be dynamically changing, it is preferable in many cases that the value of γ_t is high and stable over time. Such a process is realized by choosing high μ, high ϕ and small σ^2, as in the red density function in Figure 1b. With μ decreased, we have the blue density in the figure which is shifted, or less skewed, toward zero. The green density is obtained with larger noise variance σ^2 and more skewed; while concentrated around one, the probability mass also remains around zero. Once posterior estimates of $\hat{g}_t^i$ are obtained, $\hat{\gamma}_t^i$ can be easily computed using the inverse-logit transformation, $\hat{\gamma}_t^i = \text{logit}^{-1}(\hat{g}_t^i)$.

4 Sequential Estimation Using Particle Filtering (PF) Methods

In our proposed extension of the PGSS model, the new state vector consists of the (θ_t, γ_t) pair and the static parameters vector is defined by $\vartheta = (\mu, \phi, \sigma)$. The full joint density of all model parameters can be summarized via $p(\theta_{1:t}, \gamma_{1:t}, \vartheta|\mathcal{D}_t)$. However, as our main target is to sequentially update the relevant parameters and to obtain one-step-ahead forecasts, our goal reduces to generating samples from $p(\theta_t, \gamma_t, \vartheta|\mathcal{D}_t)$, which is not available in analytical form. Markov chain Monte Carlo (MCMC) and particle filtering (PF) methods are the two options for generating samples from this density. As pointed out by Storvik (2002), MCMC requires restarting each simulation as new data is observed, increasing the

computational burden significantly as the dimension t increases in state space models. As our goal is fast sequential online updating and prediction, we consider PF algorithms that are based on re-balancing of a finite number of particles of the state posterior distributions proportional to the likelihood. As pointed out by Carvalho et al. (2010a), estimating static parameters in sequential models is surprisingly difficult due to potential particle degeneracy. A potential remedy is to use conditional sufficient statistics of the static parameters when they are analytically available, as considered by Storvik (2002); Fearnhead (2002); Carvalho et al. (2010a). As these conditional sufficient statistics for ϑ can be obtained analytically in our model, we can devise a fast PF algorithm to generate samples from $p(\theta_t, \gamma_t, \vartheta|\mathcal{D}_t)$. In developing the PF algorithm, we exploit three major features of our model: 1) Closed-form availability of the state filtering density conditional on the dynamic discount parameters, 2) Closed-form availability of the marginal likelihoods, and 3) Analytical tractability of the conditional sufficient statistics for the static parameters.

Our goal is to eventually obtain samples from $p(\theta_t, \gamma_t, \vartheta|\mathcal{D}_t)$ which can be achieved by augmenting the density by adding α_t, β_t as

$$\begin{aligned} p(\theta_t, \gamma_t, \alpha_t, \beta_t, \vartheta|\mathcal{D}_t) &= p(\theta_t|\gamma_t, \alpha_t, \beta_t, \vartheta, \mathcal{D}_t)p(\alpha_t, \beta_t|\gamma_t, \vartheta, \mathcal{D}_t)p(\gamma_t, \vartheta|\mathcal{D}_t) \\ &= p(\theta_t|\alpha_t, \beta_t, \mathcal{D}_t)p(\alpha_t, \beta_t|\gamma_t, \mathcal{D}_t)p(\gamma_t, \vartheta|\mathcal{D}_t) \end{aligned} \tag{13}$$

where $p(\theta_t|\alpha_t, \beta_t, \mathcal{D}_t)$ is a Gamma distribution with parameters α_t, β_t and $p(\alpha_t, \beta_t|\gamma_t, \mathcal{D}_t)$ is not a known density but can be computed via

$$p(\alpha_t, \beta_t|\gamma_t, \mathcal{D}_t) = \int p(\alpha_t, \beta_t|\gamma_t, \alpha_{t-1}, \beta_{t-1}, \mathcal{D}_t)p(\alpha_{t-1}, \beta_{t-1}|\mathcal{D}_t)d\alpha_{t-1}d\beta_{t-1},$$

where $p(\alpha_t, \beta_t|\gamma_t, \alpha_{t-1}, \beta_{t-1}, \mathcal{D}_t)$ is a degenerate density with deterministic parameter updating given by (10). We note here that to sequentially compute α_t and β_t, we would need samples from $p(\alpha_{t-1}, \beta_{t-1}|\mathcal{D}_t)$ which we discuss in the sequel (See the paragraph after the algorithm on page 10.).

The next step is to sample from $p(\gamma_t, \vartheta|\mathcal{D}_t)$ which can be decomposed using a similar augmentation approach via

$$\begin{aligned} p(\gamma_t, \vartheta|\mathcal{D}_t) &= \int p(\gamma_t, \gamma_{t-1}, \alpha_{t-1}, \beta_{t-1}, \vartheta|\mathcal{D}_t)d\gamma_{t-1}d\alpha_{t-1}d\beta_{t-1} \\ &\propto \int p(N_t|\gamma_t, \alpha_{t-1}, \beta_{t-1}, \mathcal{D}_{t-1})p(\gamma_t|\gamma_{t-1}, \vartheta, \mathcal{D}_{t-1})p(\vartheta|\gamma_{t-1}, \mathcal{D}_{t-1}) \times \dots \\ &\times p(\gamma_{t-1}, \alpha_{t-1}, \beta_{t-1}|\mathcal{D}_{t-1})d\gamma_{t-1}d\alpha_{t-1}d\beta_{t-1}, \end{aligned}$$

where $p(N_t|\gamma_t, \alpha_{t-1}, \beta_{t-1}, \mathcal{D}_{t-1})$ is a negative binomial density given by (11), and $p(\gamma_t|\gamma_{t-1}, \vartheta, \mathcal{D}_{t-1})$ is the state transition for γ_t given by (12). In addition, we can approximate the online posterior $p(\gamma_{t-1}, \alpha_{t-1}, \beta_{t-1}|\mathcal{D}_{t-1})$ at $t-1$ by S particles as

$$p(\gamma_{t-1}, \alpha_{t-1}, \beta_{t-1}|\mathcal{D}_{t-1}) \approx \sum_{i=1}^{S} w_{t-1}^i \delta_{\{\gamma_{t-1}^i, \alpha_{t-1}^i, \beta_{t-1}^i\}}(\gamma_{t-1}, \alpha_{t-1}, \beta_{t-1}),$$

where $\delta_{\{x\}}(\cdot)$ is the point-mass distribution at x and $\{w_{t-1}^i\}_{i=1:S}$ are non-negative mixture weights whose sum over i must be equal to one. The final step is to generate from the density $p(\vartheta|\gamma_{t-1}, \mathcal{D}_{t-1})$ to fully implement the above sequential scheme. To do so, we utilize the conditional sufficient statistics updating of ϑ which is available analytically in our model. After the reparametrization of the hyperparameters as $\phi_0 = (1-\phi)\mu$, $\phi_1 = \phi$ and $w = \sigma^{-2}$, we can use a bivariate normal-gamma prior as

$$p(\phi_0, \phi_1, w|\mathcal{D}_0) = N(\phi_0, \phi_1|m_0, C_0/w)\ Ga(w|a_0/2, b_0/2),$$

for a given collection of prior parameters $\mathcal{S}_0 = \{m_0, C_0, a_0, b_0\}$. The likelihood function for the triplet ϕ_0, ϕ_1, w is obtained via the AR(1) model

$$g_t = \phi_0 + \phi_1 g_{t-1} + N(0, \sigma^2), \tag{14}$$

and for all t, the conditional posterior would be

$$p(\phi_0, \phi_1, w|\gamma_{1:t}, \mathcal{D}_t) = p(\phi_0, \phi_1, w|\mathcal{S}_t) = N(\phi_0, \phi_1|m_t, C_t/w)\ Ga(w|a_t/2, b_t/2), \tag{15}$$

where the set of conditional sufficient statistics are given by $\mathcal{S}_t = \{m_t, C_t, a_t, b_t\}$ updated as a function of $\mathcal{S}_{t-1}, g_t$, and g_{t-1} via

$$\begin{aligned} m_t &= m_{t-1} + A_t e_t \qquad & C_t &= C_{t-1} - q_t A_t A_t' \\ a_t &= a_{t-1} + 1 & b_t &= b_{t-1} + e_t^2/q_t, \end{aligned} \tag{16}$$

and

$$\begin{aligned} G_t &= \left[1, g_{t-1}\right] & e_t &= g_t - G_t' m_{t-1} \\ q_t &= 1 + G_t' C_{t-1} G_t & A_t &= C_{t-1} G_t / q_t. \end{aligned} \tag{17}$$

The above approach can be implemented with a minor modification under the constraint on ϕ for stationarity as in (Prado and Lopes, 2013). Namely, the prior and posterior distributions are truncated such that the generated particles of ϕ that do not fall in the region $(-1, 1)$ (or $(0, 1)$) are rejected in sampling. Consequently, we can obtain samples from $p(\phi_0, \phi_1, w|\mathcal{S}_{t-1})$ and in turn from $p(\vartheta|\gamma_{t-1}, \mathcal{D}_{t-1})$ that is required for updating (13).

In what follows, we present our PF algorithm that is based on the sequential decomposition of model parameters of interest summarized by (13). Our approach can be viewed as a combination of the auxiliary particle filter (APF) of Pitt and Shephard (1999) with conditional sufficient statistics updating of static parameters. Our PF algorithm can be summarized via the following steps:

Given a particle set $(\theta_{t-1}^i, \gamma_{t-1}^i, \alpha_{t-1}^i, \beta_{t-1}^i, \vartheta^i|\mathcal{D}_{t-1})$ with weights w_{t-1}^i,
repeat the following step 1-6 for each $j \in 1{:}S$.

1. Resample an auxiliary index $i(j)$ with probability
$w_{t-1|t}^{i(j)} \propto p(N_t|\hat{\gamma}_t^i, \alpha_{t-1}^i, \beta_{t-1}^i, \mathcal{D}_{t-1}) w_{t-1}^i$ for each i.

2. Propagate g_t^j from the state transition density,
$N(\ (1-\phi^{i(j)})\mu^{i(j)} + \phi^{i(j)} g_{t-1}^{i(j)}\ ,\ (\sigma^2)^{i(j)}\ )$ and set $\gamma_t^j = \text{logit}^{-1}(g_t^j)$.

3. Resample using normalized weights
$w_t^j \propto p(N_t|\gamma_t^j, \alpha_{t-1}^{i(j)}, \beta_{t-1}^{i(j)}, \mathcal{D}_{t-1}) / p(N_t|\hat{\gamma}_t^{i(j)}, \alpha_{t-1}^{i(j)}, \beta_{t-1}^{i(j)}, \mathcal{D}_{t-1})$.

4. Compute $\alpha_t^j = \gamma_t^j \alpha_{t-1}^j + N_t$ and $\beta_t^j = \gamma_t^j \beta_{t-1}^j + 1$ and sample θ_t^j from $Ga(\alpha_t^j, \beta_t^j)$.

5. Update $\mathcal{S}_t^j = f(\mathcal{S}_{t-1}, \gamma_t^j, \gamma_{t-1}^{i(j)})$ via (16) and (17).

6. Sample ϑ^j from $p(\vartheta|\mathcal{S}_t^j)$ given by (15).

Use the particle set $(\theta_t^j, \gamma_t^j, \alpha_t^j, \beta_t^j, \vartheta^j|\mathcal{D}_t)$ for the next time period $t+1$.

We note here that in step 1, $\hat{\gamma}_t^i$ is set equal to γ_{t-1}^i as an estimator for γ_t (similar to the APF approach). At the end of step 3, as a consequence of resampling, we obtain samples from $p(\gamma_t, \alpha_{t-1}, \beta_{t-1}|\mathcal{D}_t)$ that are used in updating α_t and β_t in step 4. We do not need to propagate θ_t from θ_{t-1} as the conditional filtering density is available analytically. An important feature of our sa-PGSS model is that the vector of past discount terms, $\gamma_{1:t}$, can be summarized by a lower dimensional vector $(\gamma_t, \alpha_{t-1}, \beta_{t-1})$, thus reducing the dimension of the state vector for γ_t's to 3 from t. This avoids the need to generate from the t dimensional state vector (can be achieved using a forward filtering and backward sampling (FFBS) step) and reduces the computational burden significantly.

Hyperparameter Selection

The selection of hyperparameters in ϑ control the implied stationary distribution for γ_t. Assuming a relatively strong prior on ϑ has practical advantages in realizing our prior belief that the discount factor should be almost constant to avoid being overly flexible and overfitting, while allowing the decrease of discount factor to be more adaptive only when absolutely necessary. For all of our numerical examples, the hyperparameters of normal-inverse gamma prior in equation (15), or the initial values of sufficient statistics for ϑ, are set at $m_0 = [(1 - 0.9)\text{logit}(0.9), 0.9]'$, $C_0 = (0.05)^2 I_2$, $a_0 = 10$ and $b_0 = 5$. This prior reflects our preference on the choice of $(\mu, \phi, \sigma^2) = (\text{logit}(0.9), 0.9, 0.5)$ whose implied stationary density of γ_t is shown in Figure 1b. Observe in Figure 1b that with this set of hyperparameter choices, the prior distribution favors γ_t values near one – implying persistent counts – but still allowing for the possibility of γ_t values close to zero – implying less dependence in N_t on previously observed counts $\mathcal{D}_{t-1}$. We briefly comment on the implications of hyperparameters on the overall estimation path in our numerical examples.

Particle Dimension and Effective Sample Size

Our experiments with the sa-PGSS model typically suggest that a particle size of $N = 5,000$ was more than sufficient in all the numerical examples. We also investigated the implications of using smaller particle sizes (1,000, 2,000, and 3,000) on the estimation paths of both the state and static parameters of our model and found no clear differences. We omit the details of these experiments to preserve space in the narrative and use $N = 5,000$ as a very conservative particle size in all our subsequent numerical examples.

To assess the existence of potential particle degeneracy in the estimates obtained using our PF algorithm, we also keep track of the the so called effective sample size (ESS) via

$$ESS_t = \frac{1}{\sum_{i=1}^{N}(w_t^i)^2}$$

where w_t^i represents the weight of particle i at t before the resampling step (if any). We note there that $1 \leq ESS_t \leq N$ where lower values indicate evidence in favor of degeneration, and vice versa. The ESS estimates can be used as a monitoring tool for assessing the need to resample at each point in time and to detect anomalies (such as structural breaks or sudden bursts in data). We investigate the implications of monitoring the ESS over time and how it can be used as a practical tool in our numerical examples.

5 Numerical Illustrations

In this section, we present three cases studies to illustrate the advantages of our sa-PGSS model: the first case study presented in Section 5.1 is a simulated time series of counts that includes sudden bursts; the

second case presented in Section 5.2 utilizes web traffic data from the Fox News Website; and the third case presented in Section 5.3 forecasts demand for Uber rides. In these three case studies, we compare online learning and forecasting results for various PGSS models from the literature. A brief description of each model included in our comparison follows:

1. *sa-PGSS*: The state-augmented Poisson-gamma state space model where the dynamic discount factor, γ_t evolves over time via the transition equation defined in equation (12).

2. *PGSS-random*: The Poisson-gamma state space model where γ is assumed to be static but random. We assume that the prior distribution of γ is a uniform discrete distribution defined over $\{0.01, 0.02, \ldots, 0.99\}$, an approach considered in Aktekin et al. (2013). The posterior distribution of γ is then obtained via
$$p(\gamma|\mathcal{D}_t) \propto p(\gamma) \prod_{s=1}^{t} p(N_s|\mathcal{D}_{s-1}, \gamma),$$
where $p(N_s|\mathcal{D}_{s-1}, \gamma)$ is the negative binomial marginal likelihood from (8).

3. *PGSS-deterministic*: The Poisson-gamma state space model where the discount factor γ_t evolves dynamically but in a deterministic manner as considered by Chen et al. (2018b). More specifically, γ_t is assumed to exhibit the following functional form
$$\gamma_t = d + (1 - d)\exp(-k\alpha_{t-1}),$$
where d represents the baseline, k is a tuning parameter controlling the speed of the information decay, and α_{t-1} is the shape parameter of the time $t-1$ posterior distribution from (4). The motivation of using the above specification stems from scenarios with zero counts and to mitigate the numerical issues caused by extremely small α_{t-1}'s. When α_{t-1} is large, the exponential term approaches zero and $\gamma_t \approx d$, leading to an approximately constant discount factor. In Chen et al. (2018b) and our study, the decay parameter is set to $k = 1$. Formally, the optimal value of d can be estimated using an empirical Bayes approach as in
$$d^* = \arg\max\{p(d|\mathcal{D}_T)\} = \arg\max\left\{p(d)\prod_{t=1}^{T} p(N_t|\mathcal{D}_{t-1}, d)\right\},$$
with some constraint on the support of d, such as $d \in (0.9, 1)$. In the case study of Fox News dataset, $d = 0.9$ is obtained by following this procedure with the training dataset. We choose $d = 0.9$ for the other datasets, the simulated and UBER data, where the training dataset is not available.

Performance Measures

In assessing the model performance in each case study, we consider three performance measures: (i) mean absolute percent error (MAPE) for assessing the predictive performance; (ii) the posterior model probability for assessing/monitoring the online model fit performance; and (iii) the marginal log-likelihood. MAPE is a standard measure of predictive performance and is defined as
$$\text{MAPE}_t = \frac{100}{t}\sum_{s=1}^{t}\frac{|N_s - f_s|}{N_s},$$
where f_t is the point forecast of N_t at $t-1$; in our study, f_t is the posterior mean or median of the one-step ahead predictive distribution $p(N_t|\mathcal{D}_{t-1})$ for simplicity, although the optimal point forecast for the standard of MAPE can also be considered (e.g., Berry et al. 2018, Section 3.3.2).

The posterior model probability $p(\mathcal{M}|\mathcal{D}_t)$ for model $\mathcal{M} \in \{\text{sa-PGSS, PGSS-random, PGSS-deterministic}\}$ is used to monitor the online model fit. Particularly, $p(\mathcal{M}|\mathcal{D}_t)$ can help us identify when and why a particular model outperforms others. For instance, in count data with sudden bursts and/or structural breaks, $p(\mathcal{M}|\mathcal{D}_t)$ can provide simple to interpret visual guidance. In addition, one can also consider $p(\mathcal{M}|\mathcal{D}_t)$ to assess different choices of hyperparameters, computational methodologies, and particle sizes.

In order to compute $p(\mathcal{M}|\mathcal{D}_t)$, the marginal likelihood is needed and analytically available through (the sum of) (8) in the PGSS family of models. For instance, in the sa-PGSS model, the log marginal likelihood can be computed as a mixture as in

$$\begin{aligned}\log p(N_t|\mathcal{D}_{t-1}) &= \int \log p(N_t|\mathcal{D}_{t-1}, \gamma_t, \alpha_{t-1}, \beta_{t-1}) p(\gamma_t, \alpha_{t-1}, \beta_{t-1}|\mathcal{D}_{t-1}) d(\gamma_t, \alpha_{t-1}, \beta_{t-1}) \\ &= \frac{1}{S}\sum_{i=1}^{S} \log p(N_t|\mathcal{D}_{t-1}, \gamma_t^i, \alpha_{t-1}^i, \beta_{t-1}^i)\end{aligned}$$

where the density of $(N_t|\mathcal{D}_{t-1}, \gamma_t, \alpha_{t-1}, \beta_{t-1})$ is the negative binomial distribution given by equation (11). In the above, the particle set, $(\gamma_t^i, \alpha_{t-1}^i, \beta_{t-1}^i)$, is obtained by augmenting $(\gamma_{t-1}^i, \alpha_{t-1}^i, \beta_{t-1}^i, \vartheta^i)$ with γ_t^i through $p(\gamma_t|\gamma_{t-1}^i, \vartheta^i)$. We remark here that the state variable, θ_t, is integrated out of $(N_t|\mathcal{D}_{t-1}, \gamma_t, \alpha_{t-1}, \beta_{t-1})$ ("Rao-Blackwellized") which reduces the overall computational burden significantly.

Computational Details and Performance

The computations for all three case studies are implemented in Ox (Doornik, 2007) on a laptop computer with Intel Core i7-7500U CPU 2.70GHz, 2.90GHz, RAM-8GB specifications. Table 1 summarizes the actual time (in seconds) of sampling the online joint posterior distribution, $p(\theta_t, \gamma_t, \vartheta|\mathcal{D}_t)$ using PF and MCMC methods. For instance, in the Fox News example, the time for completing the update of $N = 5,000$ particles from time period t to $t+1$ for the PF algorithm (without any explicit parallelization) is 0.27 seconds on average, and 0.328 at maximum for $t = 0 : T-1$. Conversely, the estimation of the online posterior at time $t = T$ using an MCMC algorithm with an independent Metropolis-Hastings step is approximately equal to 65.85 seconds. The details of the MCMC algorithm (5000 iterations after a 500 burn-in period) can be found in the Appendix. We note here that, for the Fox News example, each time period is 30 seconds long, and the MCMC approach far exceeds this threshold. As the dimension of T gets larger, the computational burden for the MCMC method exponentially increases while the PF algorithm stays around the same as evidenced by the Uber example with $T = 287$.

Table 1: Summary of computational performance in seconds.

	Simulation ($T = 99$)	Fox News ($T = 99$)	UBER ($T = 287$)
PF (Avg)	0.264	0.270	0.274
PF (Max)	0.297	0.328	0.547
MCMC	64.947	65.858	577.542

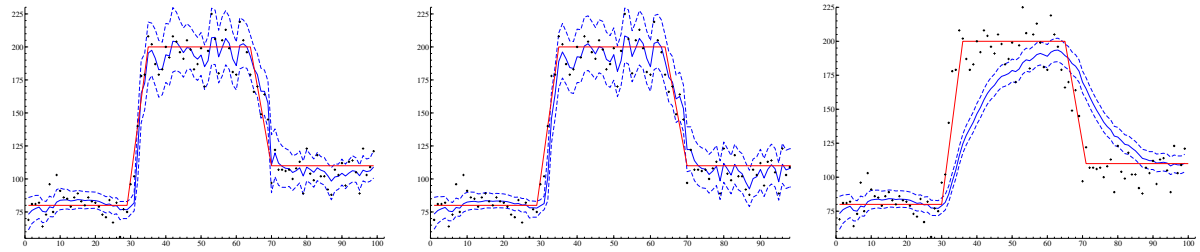
5.1 Case 1: Simulated Data

To assess the performance of the sa-PGSS model, we considered a simulated set that clearly exhibits structural breaks. The data are generated from a non-homogeneous Poisson model via $N_t \sim Po(\theta_t^*)$

independently, where

$$\theta_t^* = \begin{cases} 80 & t \in 1{:}30 \\ 100, 120, 140, 160, 180 & t = 31, 32, 33, 34, 35, \text{ resp.} \\ 200 & t \in 36{:}65 \\ 185, 170, 155, 140, 125 & t = 66, 67, 68, 69, 70, \text{ resp.} \\ 110 & t \in 71{:}100 \end{cases} \tag{18}$$

In the simulation design, two relatively slow-to-build structural breaks are represented at time points $t = 31$ and $t = 66$ with structural shifts occurring shortly after (see the red lines on Figures 2a, 2b, and 2c). The overall pattern of the simulated set roughly mimics that of the Fox News example with steeper and clearer breaks. The simulation design allows us to investigate the flexibility of the sa-PGSS model in adopting to sudden surges in the count data without the need for more complex models (such as hidden Markov models) that are computationally expensive and thus are not suitable for fast online learning/forecasting.



(a) sa-PGSS (b) PGSS-random (c) PGSS-deterministic

Figure 2: Online posterior distributions of θ_t with mean (blue, solid) and 95% credible intervals (blue, dashed) and the true values of θ_t^* (red, solid) with the observed counts (+). The three figures, (a), (b) and (c), corresponds to sa-PGSS, PGSS-random and PGSS-deterministic models, respectively. Compared with the posterior of sa-PGSS in (a), that of PGSS-random in (b) is volatile and overly adaptive in $t \geq 71$. The posterior of PGSS-determinisic in (c) is too persistent for the changes of true Poisson rates.

Figures 2a, 2b, and 2c display the online state posterior distributions with the respective 95% credibility intervals for all three models where the straight red line represents the level of the true state variable, θ_t. The posterior uncertainty provided by the sa-PGSS model in Figure 2a exhibits a fairly quick adaptive behavior to the sudden changes on the level. In Figure 2b, the PGSS-random model also seems to provide flexible coverage at first glance, with some excessive overfitting right around the second state change at $t = 71$. In contrast, the posterior coverage provided by the PGSS-deterministic model from Figure 2c clearly shows the shortcomings of the base PGSS model with a deterministic discount factor as evidenced by its inability to adopt to the sudden changes in the level.

To further investigate the online fit performances around and at the inflation points, we computed the cumulative mean squared error (MSE) estimates over time via

$$MSE_t = \frac{1}{t}\sum_{s=1}^{t}(E[\theta_s|\mathcal{D}_s] - \theta_s^*)^2,$$

where θ_s^* represents the true value of the Poisson rate at time s given in equation (18). The overall pattern of the MSEs for all three models are shown in Figure 3. Right after the first change-point, the PGSS-deterministic model provides the worst coverage with respect to the other two models with random

discount factors. The encouraging finding here is that the sa-PGSS model consistently outperforms the PGSS-random model strictly after the first change-point (around $t = 31$). This may be explained by plotting the estimated paths of discount factor γ for both models. For lower values of γ, the PGSS model tends to over-fit the data (i.e., posterior mean estimates will follow recently observed data too closely).

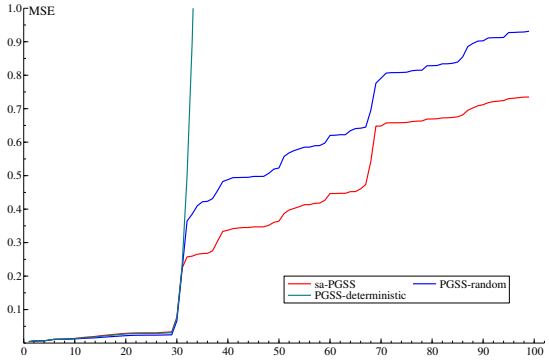


Figure 3: The mean squared errors (scaled by 10^4) for the sa-PGSS (red), PGSS-random (blue), and PGSS-deterministic (green) models. The MSE estimates for the PGSS-deterministic model are extremely large with respect to the other two models and after $t = 31$ (the first switching point) are beyond the borders of the figure in this scale.

Figures 4 and 5 show the estimated paths of the posterior means and the respective 95% credible intervals of the discount factors of the sa-PGSS (via γ_t) and PGSS-random (via γ) models. For the sa-PGSS model, the initial γ_t estimates are high (close to 1) followed by a steep drop right after the first change point ($t = 31$). Another drop can be observed at the second change point ($t = 70$), beyond which the discount factor gradually increases back to higher levels. The path of γ for the PGSS-random model tells a similar story during the first 31 time points with a steep decrease at the change point. However, after the second change point, the PGSS-random model is unable to push γ back to the region of 0.9 to 1, unlike the sa-PGSS model. We believe that this sheds light on the dominance previously observed in the MSE estimates from Figure 3, as the PGSS-random model is unable to recover the true value of γ, especially after the second break point. The dynamic nature of γ_t in the sa-PGSS model allows the posterior distribution to shift between high values (when θ_t's are similar or close to identical) and low values (when θ_t's are not similar, which occurs at the breaks). These structural breaks can also be identified by the sudden dips in the ESS estimates from Figure 4, once again occurring at $t = 31$ and $t = 70$. Severe and sudden drops in ESS estimates can be used as a formal monitoring tool for identifying structural breaks in automated machine learning settings, alerting the potential need for human intervention.

In terms of online model fit and predictive performance (marginal likelihoods, model probabilities, and MAPE estimates), the sa-PGSS model mostly outperforms the other two models. Figure 6 shows the posterior model probabilities for all three models, where equal prior probabilities are assumed. One noteworthy observation is that the PGSS-random is found to be the best model during the initial 30 observations. This is expected since there is no need for a dynamically changing discount factor until $t = 30$, as the simulation design implies that γ should be equal to 1 in this epoch (e.g., $\theta_{1:30}$ are the same). After the first change point, sa-PGSS becomes the dominant model and continues to outperform others due to its ability to rapidly adapt to the new level of the generated counts. A similar argument can be made around the second change point where the sa-PGSS model gradually starts dominating the other two models with a steep increase in model probabilities after a few observations. In a similar vein, the results from the MAPE estimates from Figure 7 also confirms the findings implied by the model

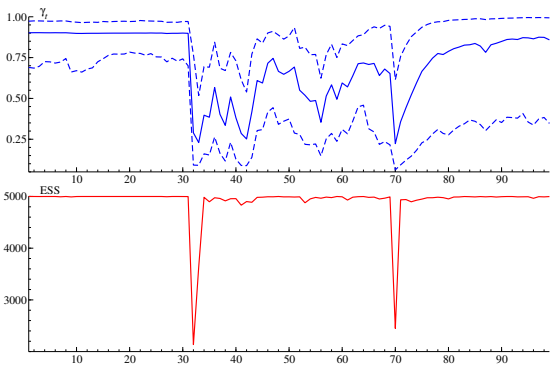


Figure 4: The means and credible intervals of the posterior of dynamic discount factor, $p(\gamma_t|\mathcal{D}_t)$ (blue), and the ESS over time (red). Both discount factor and ESS are lowered when the true Poisson rates started to change. The posterior of discount factors starts to increase after $t \geq 71$.

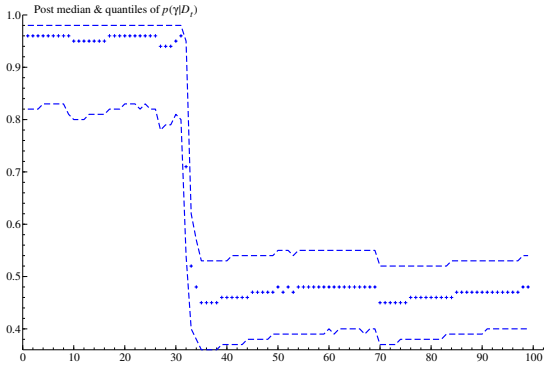


Figure 5: The median (+ symbols) and credible intervals of the posterior of constant discount factor, $p(\gamma|\mathcal{D}_t)$. Unlike the postrior results of sa-PGSS in Figure 4, once the discount factor is lowered, it remains to be aounrd 0.4-0.5 and never increases.

probabilities where the sa-PGSS model consistently outperforms the other models after the first change point. It is worth mentioning here that, in the first 30 time points, the difference between the three models is small. The difference becomes visually clearer at the two time points of structural change. A summary of the mean and the median MAPEs are shown in Table 2 which fails to highlight the superior performance of the sa-PGSS model at the structural breaks but provides a general overall summary.

In summary, our goal was to develop a highly adaptable PGSS model suitable for sequential parameter learning and online demand forecasting of counts with structural breaks. In doing so, we focused on developing a fast and efficient particle based algorithm while avoiding traditional MCMC methods that are found to increase computational burden significantly. The summary of results discussed previously based on the simulated study confirms that our proposed sa-PGSS model performs extremely well in terms of online model fit and predictive performances when compared against two other modeling strategies from the PGSS literature.

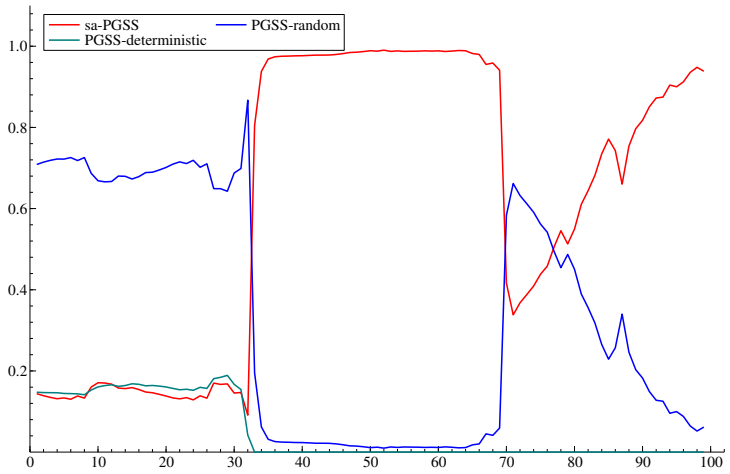


Figure 6: The posterior model probabilities of sa-PGSS (red), PGSS-random (blue), and PGSS-deterministic (pink) with equal prior probabilities. After the first change of Poisson rate, sa-PGSS becomes the best model for its adaptivity and predictive performance. The second change favors PGSS-random temporaly, but soon sa-PGSS increase its model probabillity for its high discount factor that is more suitable for the stable process of counts. PGSS-deterministic is outperformed by the other models.

Table 2: Overall summary of predictive performances

MAPE (%)	Simulation		Fox News		UBER	
	median	mean	median	mean	median	mean
sa-PGSS	9.55	9.60	10.38	10.34	30.71	30.83
PGSS-random	9.98	10.06	9.03	9.00	30.82	31.29
PGSS-deterministic	17.63	17.64	14.94	14.90	55.08	55.87

5.2 Case 2: Fox News Web Traffic Demand

To illustrate the implementation of the sa-PGSS model in a setting where fast online learning, monitoring, and prediction are essential, we consider web traffic data from the Fox News website. Robust forecasting

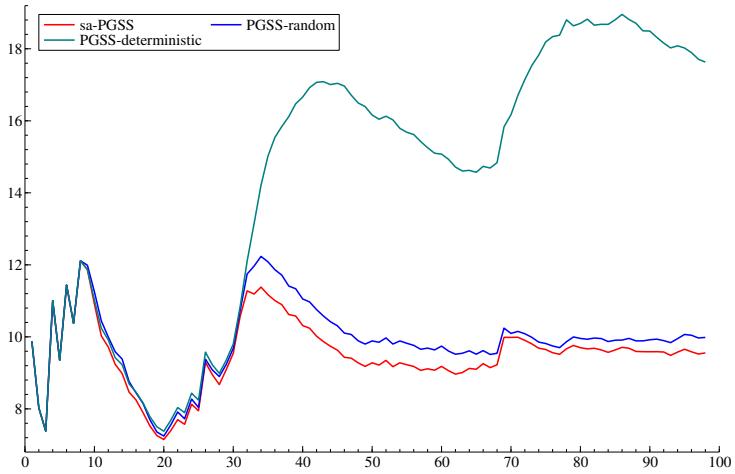


Figure 7: The mean absolute percentage errors (MAPEs) of of sa-PGSS (red), PGSS-random (blue), and PGSS-deterministic (green). In the first 30 observations, there is no clear difference between the three models. The PGSS-deterministic is clearly outperformed by the other two models. The proposed sa-PGSS model is slightly better in predictions than the PGSS-random model.

and real-time monitoring systems of web traffic are of great interest to many e-commerce firms, as optimal online ad placement and efficient web server maintenance are top priorities. Chen et al. (2018b) present a thorough analysis of the Fox News data set utilizing the PGSS-dterministic model, and in this section we benchmark the sa-PGSS model's forecasting performance and model fit to both the PGSS-random and PGSS-deterministic models. The data itself was obtained from the raw access log of the Fox News website, which is a collection of individual URL access logs (date and time) and is the flow (number of accesses) from one category of news articles to another. The counts are observed at 30 second intervals, which precludes standard forward filtering backward sampling techniques. For our illustration, we consider one particular flow from the top (main) page of the website to the category titled "World" between 9:05 and 9:55 AM on February 23rd 2015. The first observation at 9:05 is omitted from the series, as it is set equal to the hyperparameter of the initial state prior α_0. The total length of the time series is $T = 99$. Our goal is to forecast the number of visitors navigating to the "World" section 30 seconds in advance, allowing advertising impressions to be optimally allocated across sections.

It is visually evident in Figure 8 that the sa-PGSS model yields significantly better one-step-ahead predictions when there is a surge in the web traffic around 9:25. Forecasts from the PGSS-deterministic model do not quickly adapt to such a sudden change in counts, highlighting the need for more flexible discounting strategies. The estimation paths of γ_t and ESS are shown in Figure 9a, where the drop in the posterior mean of γ_t and the ESS coincide with the sudden shifts in the web traffic counts, a property that the PGSS-deterministic model fails to capture. The MAPE summary from Table 2 confirms that the sa-PGSS model has better predictive performance with respect to the PGSS-deterministic model. The PGSS-random outperforms the sa-PGSS model in the MAPE for the Fox News dataset; this is because the sa-PGSS takes a few observations before decreasing its discount factor for the drop of counts in 9:40-9:50 to avoid being overfitting, as evident in Figure 9a.

The AR model structure and informative prior distributions mimic a constant discount factor when called for by the data, enabling us to sharply estimate the dynamic discount factor γ_t in stable epochs. The effect of using informative priors on the hyperparameters, $\vartheta = (\mu, \phi, \sigma^2)$, in the AR model of γ_t can be observed in Figure 9b. The posterior distribution paths of μ, ϕ, and σ^2 are quite stable, especially

during the first 20 minutes before the sudden surge in the traffic. While the prior is informative, it is sufficiently diffuse, placing moderate prior mass on lower values of γ_t (refer to Figure 1b). Observe the drop in location parameter μ and increased variance σ^2 from 9:20-9:23. The changes in posteriors for μ, ϕ, and σ^2 translate to a drop in γ_t from 9:20-9:23 but with increased uncertainty (see Figure 9a). Our analysis shows that, despite the relatively strong priors on the AR parameters $\vartheta = (\mu, \phi, \sigma^2)$, the posterior distributions quickly respond to changes in the level of the web traffic data.

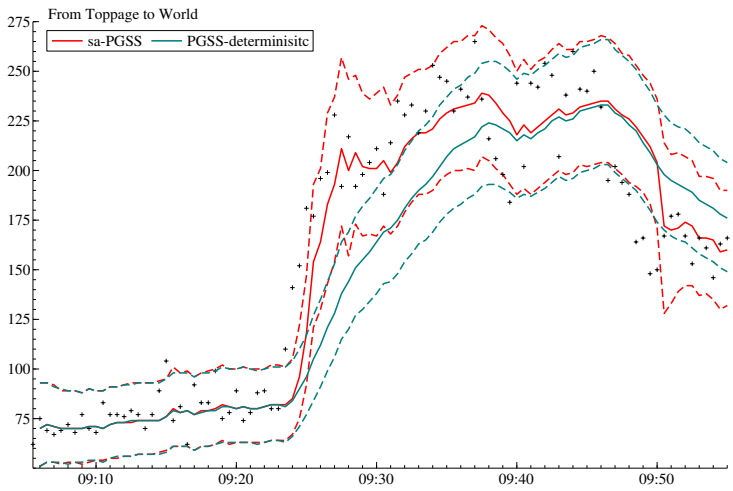
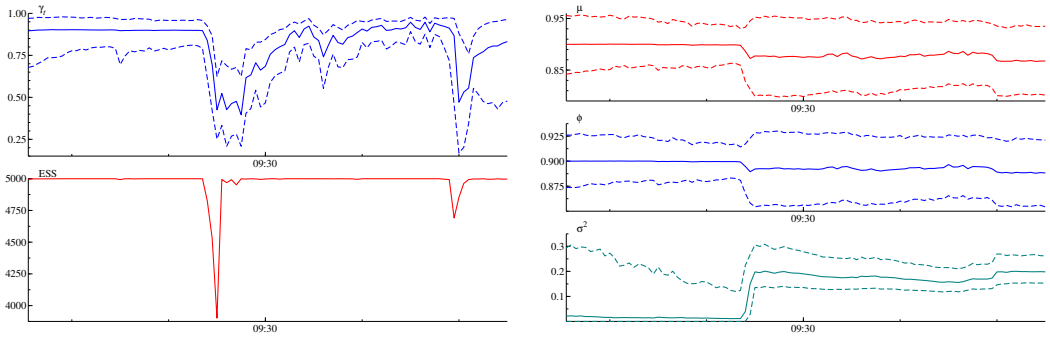


Figure 8: One step ahead predictive distributions of N_t for the sa-PGSS (red) and PGSS-deterministic (green) models, revisiting the dataset used in Figure 1a. The sa-PGSS model is able to change its predictive location flexibly in 9:25-9:30 and 9:50-9:55, while it makes stable predictions that are almost identical to those of the PGSS-deterministic model in 9:05-9:25 and 9:40-9:45.



(a) Posterior of γ_t and ESS.

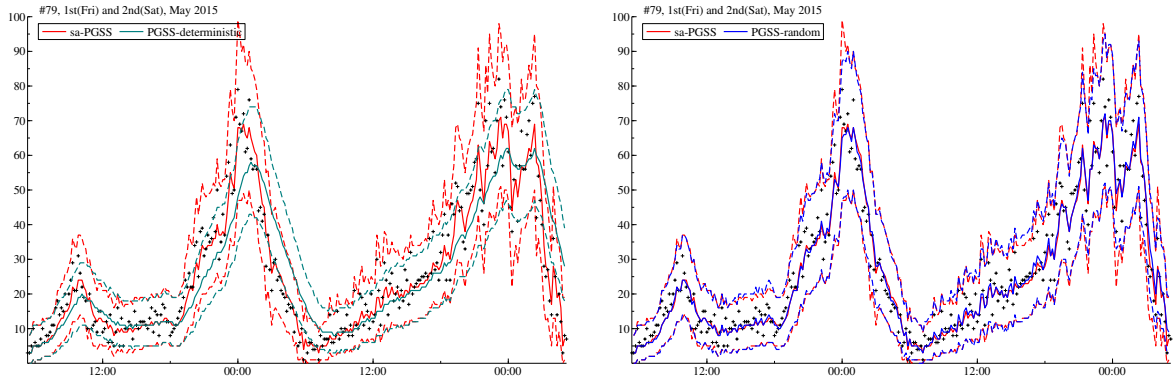
(b) Posteriors of ϑ

Figure 9: Left: Online posterior median and 95% credibility intervals for γ_t (top) and the ESS (bottom). The drop of discount factors can be seen only in the time of sudden changes in observed counts. Right: Online posterior of AR(1) parameters, i.e., $p(\text{logit}^{-1}(\mu)|\mathcal{D}_t)$, $p(\phi|\mathcal{D}_t)$ and $p(\sigma^2|\mathcal{D}_t)$. The informative prior chosen for this analysis is affected only by the sudden burst in 9:25.

5.3 Case 3: Uber Demand

The final online learning and prediction example that we consider is from the ridesharing platform Uber. In July 2016, Uber completed an average of 5.5 million rides per day (Dickey, 2017). As the global market for ridesharing has expanded, reliable forecasts of demand have become increasingly important in dynamic pricing algorithms. Here, we generate ten-minute-ahead forecasts of the number of requested Uber pickups using the sa-PGSS, PGSS-deterministic, and PGSS-random models. The data in our analysis is created from the Uber call log in the state of New York, which was made publicly available as part of a Freedom of Information Law request by the website Five Thirty Eight. The data can be accessed online at the GitHub page of Five Thirty Eight, `https://github.com/fivethirtyeight/uber-tlc-foil-response`.

We focus on pickups in the East Village of Manhattan (location ID 79) on Friday May 1st and Saturday May 2nd in 2015. The calls are binned in 10 minute time intervals, and the observations start at 5:00AM on May 1st and end at 4:55AM on Sunday May 3rd. As before, the first observation is excluded from the study in order to initialize parameters, which yields a total of 287 observations. Observe in Figure 10a that dynamics in the Uber demand data are relatively smoother and more cyclical when compared to the simulated and Fox News data. While this particular set of Uber ride requests does not exhibit clear structural breaks or sudden surges, it is expected that Uber does confront sharp and unexpected increases in demand.



(a) sa-PGSS vs PGSS-deterministic

(b) sa-PGSS vs PGSS-random

Figure 10: Top: The posterior medians (solid) and 95% credible intervals (dashed) for the sa-PGSS (green) and PGSS-deterministic (grey) models. Bottom: The posterior medians (solid) and 95% credible intervals (dashed) for the sa-PGSS (green) and PGSS-random (blue) models. Again, the predictions by the PGSS-deterministic are not adaptive to the trend of counts. In the absence of clear bursts, the prediction by sa-PGSS is almost identical to that of PGSS-deterministic, requiring no further adjustment.

The intraday and interday dynamics of pick-up calls are illustrated in Figures 10a and 10b, which also present the median and 95% credible interval for one-step-ahead predictive distributions. There are three surges in demand observed over the 48 hour span: the first increase in demand occurs from 6:00-10:00 AM on May 1st can be attributed to the morning commute on a weekday; the second surge in demand peaks at 0:00 (midnight on Friday) and is attributed to the increase in late night activity on a Friday night; a third peak starts late Saturday morning and continues to build throughout the day and into the evening, peaking at midnight on Saturday. Most of these observable and deterministic trends can be captured by incorporating covariates into our sa-PGSS model; however, the inclusion of covariates in the PGSS framework presents significant computational difficulties for online learning and forecasting that are beyond the scope of this paper. Our current aim is to investigate how well the PGSS family of

models perform in the absence of covariate effects.

Predictions from the sa-PGSS and PGSS-random models provide similar coverage in tracking these trends where most observations are within the 95% credible intervals. In contrast, predictions from the PGSS-deterministic model appear to lag behind in following the intraday Uber demand dynamics. This is visually evident in Figure 10a and clear from the MAPE estimates in Table 2. The similar predictive performance of the sa-PGSS and PGSS-random models is explained by the evolution of the posterior distribution of γ_t, shown in Figure 11. Unlike the simulated and web traffic examples, the online posterior distribution is fairly stable over time with no sudden dips or surges (as there are no structural breaks or sudden bursts in the Uber data). The posterior distribution of γ of the PGSS-random model (not shown here) also concentrates on values between 0.6 to 0.7. In summary, sa-PGSS model performs comparably to the PGSS-random model in terms of its predictive performance when the count level build-up is smooth and cyclical in nature. We remark here that structural breaks in ride sharing demand data can be observed at uncommon times such as the end of a concert, sports game, conference, or some other event, where the sa-PGSS model would deliver improved predictions of future demand.

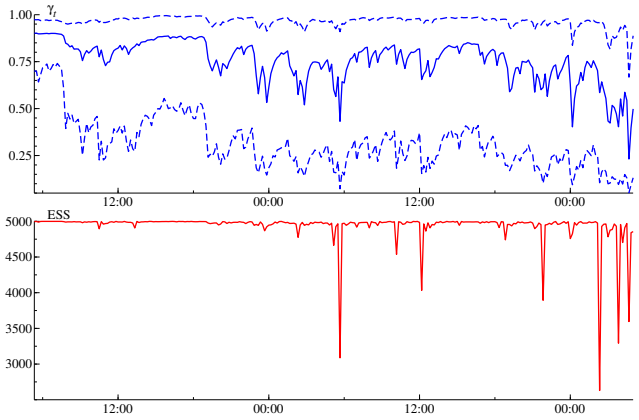


Figure 11: Evolution of γ_t (top) and ESS (bottom) over time for the sa-PGSS model. More uncertainty can be seen in the posterior of discount factors, for the time series of counts keep changing its location over time.

6 Discussion

Many e-commerce platforms continually allocate resources to meet sudden bursts in consumer demand. Bursty phenomena arise in ridesharing (Uber, Lyft), online advertising (Facebook, Google, ...), customer call centers (Liberty Mutual, GEICO), and rapid-delivery online retailing (Amazon's Prime Now and Fresh services). In each case, consumer demand is a count of the number of units (rides, ad impressions, products) requested at a single point in time, and it is natural for demand to fluctuate throughout the hour, day, or week. The challenge is to statistically model sudden bursts in observed counts while facilitating rapid, online estimation of model parameters. In the e-commerce settings considered, resource allocation decisions are made on the order of seconds to minutes, and balancing model complexity and computational speed is imperative. Markov switching models, the standard approach to modeling regime changes in time series data, require intensive computational algorithms not amenable to rapid, online learning and forecasting.

In this paper, we introduced a Poisson-gamma state-space model whose state evolution structure is flexible and responsive to sudden changes in the level of demand. This is achieved by augmenting the state vector of the PGSS model class with a dynamic discount factor, whose temporal evolution is modeled with an autoregressive process. Modeling the discount factor as a dynamic state variable is methodologically novel, as current approaches treat the discount factor as either a fixed tuning parameter, a random (static) parameter, or a deterministically time-varying quantity. Through the dynamic discount factor, the contribution of previously observed data to parameter updates varies according to recent volatility in observed counts. When counts are stable and persistent, the discount factor is close to one, and variance in the one-step-ahead predictive distribution tightens. When counts undergo large sudden

changes, the discount factor drops toward zero, and variance in the one-step-ahead predictive distribution increases. Data-driven tightening and inflating of variance in forecasts is a critical feature of the sa-PGSS model, allowing it to (i) quickly respond to bursts in observed counts, (ii) reduce prediction errors, and (iii) improve sequential model fit.

We developed a particle-based algorithm that harnesses closed-form conditional sufficient statistics to rapidly estimate dynamic state variables and static parameters. We find that the PF algorithm is $\sim$250 times faster than comparable MCMC methods when the time series has 100 observations (see Table 1). As the time series lengthens, the relative speed gap between our PF algorithm and MCMC significantly widens. To illustrate the advantages of our proposed model, we considered simulated as well as real case studies in web traffic and ride sharing demand. In the presence of structural breaks, the sa-PGSS model outperforms the base PGSS model class in terms of both model fit and predictive performance. An important finding of our study is that when level changes in counts are gradual, the dynamic discount factor embedded in the sa-PGSS offers no practical advantage over the base PGSS model where a constant discount factor is estimated from data (see Figure 10b). The comparative advantage of the sa-PGSS model is in applications where observed counts undergo sudden bursts.

Many modern applications involve analysis of multiple time series that exhibit auto and cross-sectional correlations. For example, Uber rides requested at nearby locations likely exhibit rich temporal and cross-series structure. Not only are the time series of pick-up requests spatially related, but the pick-up locations themselves may have defining characteristics that explain variation in the number of requests. These applied challenges call for a multivariate extension of the sa-PGSS model that includes covariates; however, extending the sa-PGSS model to a multivariate setting with covariates presents significant technical difficulties beyond the scope of our current paper. While we recognize the current limits of the sa-PGSS model, we believe that it offers significant promise for scaling online learning, monitoring, and forecasting of bursty count data to higher dimensions.

References

Aktekin, T., Polson, N., Soyer, R., et al. (2018). "Sequential Bayesian Analysis of Multivariate Count Data." Bayesian Analysis, 13(2): 385–409.

Aktekin, T., Soyer, R., and Xu, F. (2013). "Assessment Of Mortgage Default Risk Via Bayesian State Space Models." Annals of Applied Statistics, 7(3): 1450–1473.

Berry, L. R., Helman, P., and West, M. (2018). "Probabilistic forecasting of heterogeneous consumer transaction-sales time series." Manuscript. ArXiv:1808.04698.

Berry, L. R. and West, M. (2018). "Bayesian forecasting of many count-valued time series." Manuscript. ArXiv:1805.05232.

Carter, C. K. and Kohn, R. (1994). "On Gibbs sampling for state space models." Biometrika, 81(3): 541–553.

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

Carvalho, C. M., Lopes, H. F., Polson, N. G., and Taddy, M. A. (2010b). "Particle learning for general mixtures." Bayesian Analysis, 5(4): 709–740.

Chen, X., Banks, D., and West, M. (2018a). "Bayesian dynamic modeling and monitoring of network flows." Manuscript. ArXiv:1805.04667.

Chen, X., Irie, K., Banks, D., Haslinger, R., Thomas, J., and West, M. (2018b). "Scalable Bayesian modeling, monitoring and analysis of dynamic network flow data." Journal of the American Statistical Association, 113: 519–533.

Davis, R., Holan, S., Lund, R., and Ravishanker, N. (2015). Handbook of Discrete-Valued Time Series. Chapman and Hall/CRC.

Dickey, M. R. (2017). "Lyft is now completing one million rides a day." TechCrunch. [Online; accessed 03/06/2019].
URL https://techcrunch.com/2017/07/05/lyft-is-now-completing-one-million-rides-a-day/

Doornik, J. A. (2007). Object-Oriented Matrix Programming Using Ox, 3rd ed.. London: Timberlake Consultants Press and Oxford, 3rd edition.

Fearnhead, P. (2002). "Markov chain Monte Carlo, sufficient statistics, and particle filters." Journal of Computational and Graphical Statistics, 11(4): 848–862.

Freeland, R. K. and McCabe, B. P. M. (2004). "Analysis of Low Count Time Series Data by Poisson Autocorrelation." Journal of Time Series Analysis, 25(5): 701–722.

Fruhwirth-Schnatter, S. (1994). "Data Augmentation and Dynamic Linear Models." Journal of Time Series Analysis, 15(2): 183–202.

Fruhwirth-Schnatter, S. and Wagner, H. (2006). "Auxiliary mixture sampling for parameter-driven models of time series of counts with applications to state space modelling." Biometrika, 93(4): 827–841.

Gamerman, D., Dos-Santos, T. R., and Franco, G. C. (2013). "A non-Gaussian family of state-space models with exact marginal likelihood." Journal of Time Series Analysis, 34(6): 625–645.

Glynn, C., Tokdar, S. T., Howard, B., and Banks, D. L. (2018). "Bayesian Analysis of Dynamic Linear Topic Models." Bayesian Analysis. Advance publication.

Gordon, N. J., Salmond, D. J., and Smith, A. F. (1993). "Novel approach to nonlinear/non-Gaussian Bayesian state estimation." In IEE Proceedings F (Radar and Signal Processing), volume 140, 107–113. IET.

Gramacy, R. B. and Polson, N. G. (2011). "Particle learning of Gaussian process models for sequential design and optimization." Journal of Computational and Graphical Statistics, 20(1): 102–118.

Harvey, A. C. and Fernandes, C. (1989). "Time Series Models for Count or Qualitative Observations." Journal of Business and Economic Statistics, 7(4): 407–417.

Lopes, H. F. and Polson, N. G. (2016). "Particle learning for fat-tailed distributions." Econometric Reviews, 35(8-10): 1666–1691.

Lopes, H. F. and Tsay, R. S. (2011). "Particle filters and Bayesian inference in financial econometrics." Journal of Forecasting, 30: 168–209.

Pitt, M. K. and Shephard, N. (1999). "Filtering via simulation: Auxiliary variable particle filter." Journal of the American Statistical Association, 94: 590–599.

Prado, R. and Lopes, H. F. (2013). "Sequential parameter learning and filtering in structured autoregressive state-space models." Statistics and Computing, 23: 43–57.

Prado, R. and West, M. (2010). Time Series: Modeling, Computation and Inference. Chapman and Hall/CRC Press.

Singpurwalla, N. D., Polson, N. G., and Soyer, R. (2018). "From least squares to signal processing and particle filtering." Technometrics, 60(2): 146–160.

Smith, R. and Miller, J. E. (1986). "A Non-Gaussian State Space Model and Application to Prediction of Records." Journal of the Royal Statistical Society, Series B, 48(1): 79–88.

Storvik, G. (2002). "Particle filters for state-space models with the presence of unknown static parameters." IEEE Transactions on Signal Processing, 50(2): 281–289.

Uhlig, H. (1994). "On singular Wishart and singular multivariate beta distributions." The Annals of Statistics, 395–405.

— (1997). "Bayesian vector autoregressions with stochastic volatility." Econometrica: Journal of the Econometric Society, 59–73.

West, M. and Harrison, P. J. (1986). "Monitoring and adaptation in Bayesian forecasting models." Journal of the American Statistical Association, 81: 741–750.

— (1997). Bayesian Forecasting and Dynamic Models. Springer Verlag, 2nd edition.

Appendix: Markov chain Monte Carlo Algorithm for the sa-PGSS Model

In what follows, we present a summary of steps of the MCMC algorithm that is an alternative for the proposed PF algorithm. The goal is to generate samples from the full joint posterior distribution of state as well as static parameters, $p(\theta_{1:t}, \gamma_{1:t}, \vartheta|\mathcal{D}_t)$, in a sequential manner. This can be achieved via the following steps:

1. Sampling $\theta_{1:t}$

   Given $\gamma_{1:t}$ and ϑ, sampling from the conditional posterior $p(\theta_{1:t}|\gamma_{1:t}, \vartheta, \mathcal{D}_t)$ can be done by forward filtering and backward sampling. First, we compute $(a_{1:t}, b_{1:t})$ by forward filtering. Next, we sample from $\theta_t \sim Ga(a_t, b_t)$. Recursively, at each $s < t$, we sample θ_s based on the distributional relation $\theta_s = \gamma_s\theta_{s+1} + Ga((1-\gamma_s)a_s, b_s)$.

2. Sampling ϑ

   The conditional posterior of $p(\vartheta|\theta_{1:t}, \gamma_{1:t}, \mathcal{D}_t)$ is given in Section 3 where the normal-inverse gamma distribution for the transformed parameters are shown. Same approach can be followed here.

3. Sampling $\gamma_{1:t}$

   This is the hardest part of the MCMC algorithm to implement. We take the single-mover sampler approach and consider the sampling of each γ_s for $s = 1{:}t$. The conditional posterior is written as (e.g., for $0 < s < t$)

$$p(\gamma_s|\theta_{1:t}, \gamma_{1:t\setminus s}, \vartheta, \mathcal{D}_t) \propto p(g_{s+1}|g_s, \vartheta)p(g_s|g_{s-1}, \vartheta) \prod_{u=s}^{t} p(\theta_u|\theta_{u-1}, \gamma_{1:u}, \mathcal{D}_{u-1}) \tag{19}$$

   where the transition density of states is that of the scaled-beta distribution,

$$\begin{aligned} p(\theta_u|\theta_{u-1}, \gamma_{1:u}, \mathcal{D}_{u-1}) = & \frac{1}{Be(\gamma_u a_{u-1}, (1-\gamma_u)a_{u-1})} \\ & \left(\frac{\gamma_u}{\theta_{u-1}}\right)^{\gamma_u\alpha_{u-1}} \theta_u^{\gamma_u\alpha_{u-1}-1} \left(1 - \frac{\gamma_u}{\theta_{u-1}}\theta_u\right)^{(1-\gamma_u)\alpha_{u-1}-1} \end{aligned} \tag{20}$$

   Note that γ_s is involved implicitly in $p(\theta_u|\theta_{u-1}, \gamma_{1:u}, \mathcal{D}_{u-1})$ for not only $u = s$ but also $u > s$ through the sufficient statistics α_u that is sequentially updated by, for example, $\alpha_{s+1} = \gamma_s\alpha_s + N_s$.

The sampling from equation (19) is the key in the implementation of the MCMC algorithm. The common approach is to use a random-walk Metropolis Hastings step where t tuning parameters are required in addition to many iterations, making this is an unattractive solution. As an alternative, an independent Metropolis Hastings step with a Gaussian proposal density can be considered which requires the computation of the gradient and the Hessian of the density in (19) in the log scale. To speed up the estimation, we propose to sample from

$$q(g_s) \propto p(g_{s+1}|g_s, \vartheta)p(g_s|g_{s-1}, \vartheta)$$

and accept the generated particle g_t^{new} with acceptance probability

$$P[g_s^{old} \to g_t^{new}] = \max\left\{1, \prod_{u=s}^{t} \frac{p(\theta_u|\theta_{u-1}, \gamma_{1:u\setminus s}, \gamma_s^{new}, \mathcal{D}_{u-1})}{p(\theta_u|\theta_{u-1}, \gamma_{1:u\setminus s}, \gamma_s^{old}, \mathcal{D}_{u-1})}\right\}.$$