Dynamic Variable Selection with Spike-and-Slab Process Priors

Veronika Ročková & Kenichiro McAlinn

Booth School of Business, University of Chicago

January 15, 2018

Abstract

The spike-and-slab methodology for variable selection has been widely used in statistical data analysis since its inception more than two decades ago. However, developments for varying coefficient models have not yet received much attention. Here, we address the problem of dynamic variable selection in time series regression, where the set of active predictors is allowed to evolve over time. To capture time-varying variable selection uncertainty, we introduce new dynamic shrinkage priors for the time series of regression coefficients. These priors are characterized by two main ingredients: smooth parameter evolutions and intermittent zeroes for modeling predictive breaks. More formally, our proposed Dynamic Spike-and-Slab (DSS) priors are constructed as mixtures of two processes: a spike process for the irrelevant coefficients and a slab autoregressive process for the active coefficients. The mixing weights are themselves time-varying and depend on a lagged value of the series. One distinguishing feature of DSS priors is that their stationary distribution is fully known and characterized by spike-and-slab marginals. This property guarantees marginal stability and probabilistic coherence. We characterize dynamic selection thresholds for MAP smoothing and implement a one-step-late EM algorithm for fast calculations. We demonstrate, through simulation and a topical high-dimensional macroeconomic dataset, that DSS priors are far more effective at finding signal and improving forecasts compared to other existing strategies.

Keywords: Autoregressive mixture processes, Dynamic sparsity, MAP smoothing, Spike and Slab, Stationarity
1 Dynamic Sparsity

For dynamic linear modeling with many potential predictors, the assumption of a static generative model with a fixed subset of regressors (albeit with time-varying regressor effects) may be misleadingly restrictive. By obscuring variable selection uncertainty over time, confinement to a single inferential model may lead to poorer predictive performance, especially when the actual effective subset at each time is sparse. The potential for dynamic model selection techniques in time series modeling has been recognized (Frühwirth-Schnatter and Wagner 2010; Groen et al. 2013; Nakajima and West 2013a; Kalli and Griffin 2014; Chan et al. 2012). One prominent practical example is inflation forecasting, where large sets of predictors are available and it is expected that different sets of predictors will predict well under different economic regimes/situations (Bańbura et al. 2010; Nakajima and West 2013a; Koop and Korobilis 2012; Groen et al. 2013; Koop 2013; Korobilis 2013; Giannone et al. 2014; Kalli and Griffin 2014; Gefang 2014; Pettenuzzo et al. 2018). For example, in recessions we might see distress related factors be effective, while having no predictive power in expansions. Motivated by such contexts, we develop a new dynamic shrinkage approach for time series models that exploits time-varying predictive subset sparsity, when it in fact exists.

We present our approach in the context of dynamic linear models (West and Harrison 1997) (or varying coefficient models with a time effect modifier (Hastie and Tibshirani 1993)) that link a scalar response \( y_t \) at time \( t \) to a set of \( p \) known regressors \( x_t = (x_{t1}, \ldots, x_{tp})' \) through the relation

\[
y_t = x_t' \beta^0_t + \varepsilon_t, \quad t = 1, \ldots, T, \tag{1.1}
\]

where \( \beta^0_t = (\beta^0_{t1}, \ldots, \beta^0_{tp})' \) is a time-varying vector of regression coefficients and where the innovations \( \varepsilon_t \) come from \( N(0, 1) \). The challenge of estimating the \( T \times p \) coefficients in (1.1), with merely \( T \) observations, is typically made feasible with a smoothness inducing state-space model that treats \( \{\beta^0_t\}_{t=1}^T \) as realizations from a (vector autoregressive) stochastic process

\[
\beta^0_t = f(\beta^0_{t-1}) + e_t, \quad e_t \sim N(0, \Lambda_t) \tag{1.2}
\]

for some \( \Lambda_t \) and \( f(\cdot) \). Nevertheless, any regression model with a large number of potential predictors will still be vulnerable to overfitting. This phenomenon is perhaps even more pronounced here, where the regression coefficients are forced to be dynamically intertwined. The major concern is that overfitted coefficient evolutions disguise true underlying dynamics and provide misleading representations with poor out-of-sample predictive performance. For long term forecasts, this concern is exacerbated by the proliferation of the state space in (1.2). As the model propagates forward, the non-sparse state innovation accumulates noise, further hindering the out-of-sample forecast ability. With many potentially irrelevant predictors, seeking sparsity is a natural remedy against the loss of statistical efficiency and forecast ability.

We shall assume that \( p \) is potentially very large, where possibly only a small portion of predictors is relevant for the outcome at any given time. Besides time-varying regressor effects, we adopt the point of view that the regressors are allowed to enter and leave the model as time progresses, rendering the subset selection problem ultimately dynamic. This anticipation can be reflected by the following sparsity manifestations in the matrix of regression coefficients \( B^0_{p \times T} = [\beta^0_1, \ldots, \beta^0_T] \):

(a) horizontal sparsity, where each individual time series \( \{\beta^0_{tj}\}_{t=1}^T \) (for \( j = 1, \ldots, p \)) allows for

\[
|\beta^0_{tj}| < 
\]
intermittent zeroes for when \( j^{th} \) predictor is not a persisting predictor at all times, (b) vertical sparsity, where only a subset of coefficients \( \beta_t^0 = (\beta_{t1}^0, \ldots, \beta_{tp}^0)' \) (for \( t = 1, \ldots, T \)) will be active at the \( t^{th} \) snapshot in time.

This problem has been addressed in the literature by multiple authors including, for example, Groen et al. (2013); Belmonte et al. (2014); Koop and Korobilis (2012); Kalli and Griffin (2014); Nakajima and West (2013a). We should like to draw particular attention to the latent threshold process of Nakajima and West (2013a,b); Zhou et al. (2014); Nakajima and West (2015, 2017), a related regime switching scheme for either shrinking coefficients exactly to zero or for leaving them alone on their autoregressive path:

\[
\beta_{tj} = b_{tj} \gamma_{tj}, \quad \text{where} \quad \gamma_{tj} = I(|b_{tj}| > d_j),
\]

\[
b_{tj} = \phi_{0j} + \phi_{1j}(b_{t-1j} - \phi_{0j}) + e_t, \quad |\phi_{1j}| < 1, \quad e_t \overset{iid}{\sim} N(0, \lambda_1).
\]

The model assumes a latent autoregressive process \( \{b_{tj}\}_{t=1}^T \), giving rise to the actual coefficients \( \{\beta_{tj}\}_{t=1}^T \) only when it meanders away from a latent basin around zero \([-d_j, d_j]\). This process is reminiscent of a dynamic extension of point-mass mixture priors that exhibit exact zeros (Mitchell and Beauchamp 1988). Other related works include shrinkage approaches towards static coefficients in time-varying models (Frühwirth-Schnatter and Wagner 2010; Bitto and Frühwirth-Schnatter 2016; Lopes et al. 2016). We approach the dynamic sparsity problem through the lens of Bayesian variable selection and develop it further for varying coefficient models. Namely, we assume the traditional spike-and-slab setup by assigning each regression coefficient \( \beta_{tj} \) a mixture prior underpinned by a binary latent indicator \( \gamma_{tj} \), which flags the coefficient as being either active or inert. While static variable selection with spike-and-slab priors has received a lot of attention (Carlin and Chib 1995; Clyde et al. 1996; George and McCulloch 1993, 1997; Mitchell and Beauchamp 1988; Rockova and George 2014), the literature on dynamic variants is far more sparse (George et al. 2008; Frühwirth-Schnatter and Wagner 2010; Nakajima and West 2013a; Groen et al. 2013). Narrowing this gap, this work proposes several new dynamic extensions of popular spike-and-slab priors.

The main thrust of this work is to introduce Dynamic Spike-and-Slab (DSS) priors, a new class of time series priors, which induce either smoothness or shrinkage towards zero. These processes are formed as mixtures of two stationary time series: one for the active and another for the negligible coefficients. The DSS priors pertain closely to the broader framework of mixture autoregressive (MAR) processes with a given lag, where the mixing weights are allowed to depend on time. Despite the reported success of MAR processes (and variants thereof) for modeling non-linear time series (Wong and Li 2000, 2001; Kalliovirta et al. 2015; Wood et al. 2011), their potential as dynamic sparsity inducing priors has been unexplored. Here, we harvest this potential within a dynamic variable selection framework. One feature of DSS priors, that sets it apart from the latent threshold model, is that it yields benchmark continuous spike-and-slab priors (such as the Spike-and-Slab LASSO of Rockova (2017)) as its marginal stationary distribution. This property guarantees marginal stability in the selection/shrinkage dynamics and probabilistic coherence.

By turning our time-domain priors into penalty constructs, we formalize the notion of prospective and retrospective shrinkage through doubly adaptive shrinkage terms that pull together past, current, and future information. We introduce asymmetric dynamic thresholding rules –extensions of existing rules for static symmetric regularizers (Fan and Li 2001; Antoniadis and Fan 2001)–
to characterize the behavior of joint posterior modes for MAP smoothing. For calculation, we implement a one-step-late EM algorithm of (Green 1990), that capitalizes on fast closed-form one-site updates. Our dynamic penalties can be regarded as natural extensions of the spike-and-slab penalty functions introduced by Rockova (2017) and further developed by Rockova and George (2016). The DSS priors here are deployed as a fast MAP smoothing catalyst rather than a vehicle for a full-blown MCMC analysis (as in Nakajima and West (2013a)). The key distinguishing feature of this deployment is that DSS priors attain sparsity through sparse posterior modes rather than auxiliary thresholding. In addition, we derive MCMC algorithms for DSS priors that leverage conditional conjugacy for forward filtering and backward sampling (Appendix).

We demonstrate the effectiveness of our introduced DSS priors with a thorough simulation study and a topical macroeconomic application. Both studies highlight the comparative improvements–in terms of inference, forecasting, and computational time– of DSS priors over conventional and recent methods in the literature. In particular, the macroeconomic application, using a large number of economic indicators to forecast inflation and infer on underlying economic structures, serves as a motivating example as to why dynamic sparsity is effective, and even necessary, in these contexts.

The paper is structured as follows: Section 2 and Section 2.1 introduce the DSS processes and their variants. Section 3 develops the penalized likelihood perspective, introducing the prospective and retrospective shrinkage terms. Section 4 provides useful characterizations of the global posterior mode. Section 5 develops the one-step-late EM algorithm for MAP smoothing. Section 6 illustrates the MAP smoothing deployment of DSS on simulated examples and Section 7 on a macroeconomic dataset. Section 8 concludes with a discussion.

2 Dynamic Spike-and-Slab Priors

In this section, we introduce the class of Dynamic Spike-and-Slab (DSS) priors that constitute a coherent extension of benchmark spike-and-slab priors for dynamic selection/shrinkage. We will assume that the p time series \( \{ \beta_t \}_{t=1}^T \) (for \( j = 1, \ldots, p \)) in (1.1) follow independent and identical DSS priors and thereby we suppress the subscript \( j \) (for notational simplicity).

We start with a conditional specification of the DSS prior. Given a binary indicator \( \gamma_t \in \{ 0, 1 \} \), which encodes the spike/slab membership at time \( t \), and a lagged value \( \beta_{t-1} \), we assume that \( \beta_t \) arises from a mixture of the form

\[
\pi(\beta_t | \gamma_t, \beta_{t-1}) = (1 - \gamma_t) \psi_0(\beta_t | \lambda_0) + \gamma_t \psi_1(\beta_t | \mu_t, \lambda_1),
\]

(2.5)

where

\[
\mu_t = \phi_0 + \phi_1 (\beta_{t-1} - \phi_0) \quad \text{with} \quad |\phi_1| < 1
\]

(2.6)

and

\[
P(\gamma_t = 1 | \beta_{t-1}) = \theta_t.
\]

(2.7)

For Bayesian variable selection, it has been customary to specify a zero-mean spike density \( \psi_0(\beta | \lambda_0) \), such that it concentrates at (or in a narrow vicinity of) zero. Regarding the slab distribution \( \psi_1(\beta_t | \mu_t, \lambda_1) \), we require that it be moderately peaked around its mean \( \mu_t \), where the amount of spread is regulated by a concentration parameter \( \lambda_1 > 0 \). The conditional DSS prior formulation (2.5) generalizes existing continuous spike-and-slab priors (George and McCulloch 1993;
Ishwaran and Rao 2005; Rockova 2017) in two important ways. First, rather than centering the slab around zero, the DSS prior anchors it around an actual model for the time-varying mean \( \mu_t \). The non-central mean is defined as an autoregressive lag polynomial of the first order with hyper-parameters \((\phi_0, \phi_1)\). While our framework can be extended to higher-order autoregressive polynomials where \( \mu_t \) may also depend on values older than \( \beta_{t-1} \), we outline our method for the first-order autoregression due to its relative simplicity and ubiquity in practice (Tibshirani et al. 2005; West and Harrison 1997; Prado and West 2010). Throughout the manuscript, however, we remark on how to suitably modify our framework for higher orders. Although estimable (subject to stationary restrictions), \( (\phi_0, \phi_1) \) will be treated as fixed. Assuming a fixed \( \phi_1 \) is not too far from the common practice in the Bayesian literature (Omori et al. 2007; Nakajima and West 2013a, to name a few) which consists of imposing a tight prior around, but below, 1 for \( \phi_1 \), to ensure stable, stationary estimation. We discuss extensions of our framework for random \( (\phi_0, \phi_1) \) in Section 8.

It is illuminating to view the conditional prior (2.5) as a “multiple shrinkage” prior (George 1986b,a) with two shrinkage targets: (1) zero (for the gravitational pull of the spike), and (2) \( \mu_t \) (for the gravitational pull of the slab). It is also worthwhile to emphasize that the spike distribution \( \psi_0(\beta_t \mid \lambda_0) \) does not depend on \( \beta_{t-1} \), only the slab does. The DSS formulation thus induces separation of regression coefficients into two groups, where only the active ones are assumed to walk on an autoregressive path.

The second important generalization is implicitly hidden in the hierarchical formulation of the mixing weights \( \theta_t \) in (2.7), which casts them as a smoothly evolving process (as will be seen in Section 2.2 below). Before turning to this formulation, we discuss several special cases of DSS priors.

### 2.1 Spike and Slab Pairings

Our recommended choice of the spike distribution is the Laplace density \( \psi_0(\beta \mid \lambda_0) = \frac{\lambda_0}{2} e^{-|\beta|\lambda_0} \) (with a relatively large penalty parameter \( \lambda_0 > 0 \)) due to its ability to threshold via sparse posterior modes, as will be elaborated on in Section 4. Under the Laplace spike distribution, the series \( \{\beta_t\}_{t=1}^T \) is thereby stationary, iid with a marginal density \( \psi_0(\beta \mid \lambda_0) \). Another natural choice, a Gaussian spike, would impose no new computational challenges due to its conditional conjugacy. However, additional thresholding would be required to obtain a sparse representation.

Regarding the slab distribution, we will focus primarily on the Gaussian slab \( \psi_1(\beta_t \mid \mu_t, \lambda_1) \) (with mean \( \mu_t \) and variance \( \lambda_1 \)) due to its ability to smooth over past/future values. Under the Gaussian slab distribution, \( \{\beta_t\}_{t=1}^T \) follow a stationary Gaussian AR(1) process

\[
\beta_t = \phi_0 + \phi_1(\beta_{t-1} - \phi_0) + e_t, \quad |\phi_1| < 1, \quad e_t \overset{iid}{\sim} \mathcal{N}(0, \lambda_1),
\]

whose stationary distribution is characterized by univariate marginals

\[
\psi_1^{ST}(\beta_t \mid \lambda_1, \phi_0, \phi_1) \equiv \psi_1 \left( \beta_t \mid \phi_0, \frac{\lambda_1}{1 - \phi_1^2} \right); \quad (2.9)
\]

a Gaussian density with mean \( \phi_0 \) and variance \( \frac{\lambda_1}{1 - \phi_1^2} \). The availability of this tractable stationary distribution (2.9) is another appeal of the conditional Gaussian slab distribution.

Rather than shrinking to the vicinity of the past value, one might like to entertain the possibility of shrinking exactly to the past value (Tibshirani et al. 2005). Such a property would be appre-
ciated, for instance, in dynamic sparse portfolio allocation models to mitigate transaction costs associated with negligible shifts in the portfolio weights (Irie and West 2016; Brodie et al. 2009; Jagannathan and Ma 2003; Puelz et al. 2016). One way of attaining the desired effect would be replacing the Gaussian slab $\psi_1(\cdot)$ in (2.5) with a Laplace distribution centered at $\mu_t$. This conditional construction, however, does not imply the Laplace distribution marginally. The univariate marginals are defined through the characteristic function given in (2.7) of Andel (1983). The lack of availability of the marginal density in a simple form thwarts the specification of transition weights needed for the implementation of our DSS framework. There are, however, avenues for constructing an autoregressive process with Laplace marginals, e.g., through the normal-gamma-autoregressive (NGAR) process by Kalli and Griffin (2014). We define the following Laplace autoregressive (LAR) process as a special case.

**Definition 1.** We define the Laplace autoregressive (LAR) process by

$$
\beta_t = \sqrt{\frac{\psi_t}{\psi_{t-1}}} \phi_1 \beta_{t-1} + \eta_t, \quad \eta_t \sim N (0, (1 - \phi_1^2) \psi_t),
$$

where $\{\psi_t\}_{t=1}^T$ follow an exponential autoregressive process specified through $\psi_t | \kappa_{t-1} \sim \text{Gamma}(1 + \kappa_{t-1}, \lambda_1^2 / [2(1 - \rho)])$ and $\kappa_{t-1} | \psi_{t-1} \sim \text{Poisson} \left( \frac{\rho_2(1 - \rho)}{\lambda_1^2 \psi_{t-1}} \right)$ with a marginal distribution $\text{Exp}(\lambda_1^2 / 2)$.

The LAR process exploits the scale-normal-mixture representation of the Laplace distribution, yielding Laplace marginals $\beta_t \sim \tilde{\psi}_{ST}(\beta_t | \lambda_1) \equiv \text{Laplace}(\lambda_1)$. This coherence property can be leveraged within our DSS framework as follows. If we replace the slab Gaussian $AR(1)$ process in (2.5) with the LAR process and deploy $\tilde{\psi}_{ST}(\beta_t | \lambda_1)$ instead of $\psi_{ST}(\beta_t | \lambda_1)$ in (2.11), we obtain a Laplace DSS variant with the Spike-and-Slab LASSO prior of Rockova (2017) as its marginal distribution (according to Theorem 1).

It is worth pointing out an alternative autoregressive construction with Laplace marginals proposed by Andel (1983), where the following $AR(1)$ scheme is considered.

$$
\beta_t = \begin{cases} 
\phi_1 \beta_{t-1} & \text{with probability } \phi_1^2, \\
\phi_1 \beta_{t-1} + \eta_t & \text{with probability } 1 - \phi_1^2, \quad \text{where } \eta_t \sim \text{Laplace}(\lambda_1).
\end{cases}
$$

(2.10)

The innovations in (2.10) come from a mixture of a point mass at zero, providing an opportunity to settle at the previous value, and a Laplace distribution. Again, by deploying this process in the slab, we obtain the Spike-and-Slab LASSO marginal distribution. While MCMC implementations can be obtained for the dynamic Spike-and-Slab LASSO method (e.g. embedding the sampler of Kalli and Griffin (2014) within our MCMC approach outlined in the Appendix), the Laplace slab extensions are ultimately more challenging for optimization. Throughout the rest of the paper, we thereby focus on the Gaussian $AR(1)$ slab process.

### 2.2 Evolving Inclusion Probabilities

A very appealing feature of DSS priors that makes them suitable for dynamic subset selection is the opportunity they afford for obtaining “smooth” spike/slab memberships. Recall that the binary indicators in (2.7) determine which of the spike or slab regimes is switched on at time $t$, where $P(\gamma_t = 1 | \beta_{t-1}) = \theta_t$. It is desirable that the sequence of slab probabilities $\{\theta_t\}_{t=1}^T$ evolves smoothly over time, allowing for changes in variable importance as time progresses and, at the same time,
avoiding erratic regime switching. Because the series \( \{ \theta_t \}_{t=1}^T \) is a key driver of the sparsity pattern, it is important that it be (marginally) stable and that it reflects all relevant information, including not only the previous value \( \theta_{t-1} \), but also the previous value \( \beta_{t-1} \). Many possible constructions of \( \theta_t \) could be considered. We turn to the implied stationary distribution as a guide for a principled construction of \( \theta_t \).

For our formulation, we introduce a *marginal* importance weight \( 0 < \Theta < 1 \), a scalar parameter which controls the overall balance between the spike and the slab distributions. Given \( (\Theta, \lambda_0, \lambda_1, \phi_0, \phi_1) \), the conditional inclusion probability \( \theta_t \) (or a transition function \( \theta(\beta_{t-1}) \)) is defined as

\[
\theta_t \equiv \theta(\beta_{t-1}) = \frac{\Theta \psi_1^{ST}(\beta_{t-1}|\lambda_1, \phi_0, \phi_1)}{\Theta \psi_1^{ST}(\beta_{t-1}|\lambda_1, \phi_0, \phi_1) + (1-\Theta)\psi_0^T(\beta_{t-1}|\lambda_0)}.
\]

(2.11)

The conditional mixing weight \( \theta_t \) can be interpreted as the posterior probability of classifying the past coefficient \( \beta_{t-1} \) as arriving from the stationary slab distribution as opposed to the (stationary) spike distribution. This interpretation reveals how the weights \( \{ \theta_t \}_{t=1}^T \) proliferate parsimony throughout the process \( \{ \beta_t \}_{t=1}^T \). Suppose that the past value \( |\beta_{t-1}| \) was large, then \( \theta(\beta_{t-1}) \) will be close to one, signaling that the *current* observation \( \beta_t \) is more likely to be in the slab. The contrary occurs when \( |\beta_{t-1}| \) is small, where \( \theta_t \) will be discouraged from the slab because the inclusion weight \( \theta(\beta_{t-1}) \) will be small (close to zero). Let us also note that the weights in (2.11) are different from the conditional probabilities for classifying \( \beta_{t-1} \) as arising from the conditional slab in (2.5). These weights will be introduced later in Section 3.

It is tempting to regard \( \Theta \) as the marginal proportion of nonzero coefficients. However, such an interpretation is misleading because, sparsity levels are ultimately determined by the \( \theta_t \) sequence, which is influenced by the component stationary distributions \( \psi_0(\cdot) \) and \( \psi_{1}^{ST}(\cdot) \), in particular by the amount of their overlap around zero. Such an interpretation is thus inapplicable for the continuous spike-and-slab mixtures considered here, where more caution is needed for calibration (Rockova 2017). This issue will be revisited in Section 3.

Now that we have elaborated on all the layers of the hierarchical model, we are ready to formally define the Dynamic Spike-and-Slab Process.

**Definition 2.** Equations (2.5), (2.6), (2.7) and (2.11) define a Dynamic Spike-and-Slab Process (DSS) with parameters \( (\Theta, \lambda_0, \lambda_1, \phi_0, \phi_1) \). We will write

\[
\{ \beta_t \}_{t=1}^T \sim \text{DSS}(\Theta, \lambda_0, \lambda_1, \phi_0, \phi_1).
\]

The DSS process relates to the Gaussian mixture of autoregressive (GMAR) process of Kalliovirta et al. (2015), which was conceived as a model for time series data with regime switches. Here, we deploy it as a prior on time-varying regression coefficients within the spike-and-slab framework, allowing for distributions other than Gaussian. The DSS, being an instance/elaboration of the GMAR process, inherits elegant marginal characterizations (as will be seen below).

The DSS construction has a strong conceptual appeal in the sense that its marginal probabilistic structure is fully known. This property is rarely available with conditionally defined non-Gaussian time series models, where not much is known about the stationary distribution beyond just the mere fact that it exists. The DSS process, on the other hand, guarantees well behaved stable marginals that can be described through benchmark spike-and-slab priors. The marginal distribution can be used as a prior for the initial vector at time \( t = 0 \), which is typically estimated with the remaining coefficients. The following theorem is an elaboration of Theorem 1 of
Similarly, we define a retrospective penalty implied by (2.5) and (2.11) as follows:

\[\mathrm{pen}(\beta | \beta_{t-1}) = \log [(1 - \theta_t) \psi_0(\beta | \lambda_0) + \theta_t \psi_1(\beta | \mu_1, \lambda_1)].\]  

(3.14)

Similarly, we define a prospective penalty \(\mathrm{pen}(\beta_{t+1} | \beta)\) as a function of the first argument \(\beta\) in (3.14). The Dynamic Spike-and-Slab (DSS) penalty is then defined as

\[\mathrm{Pen}(\beta | \beta_{t-1}, \beta_{t+1}) = \mathrm{pen}(\beta | \beta_{t-1}) + \mathrm{pen}(\beta_{t+1} | \beta) + C,\]  

(3.15)
where $C \equiv -\text{Pen}(0 \mid \beta_{t-1}, \beta_{t+1})$ is a norming constant.

**Remark 2.** Note that the dependence on the previous value $\beta_{t-1}$ in $\text{pen}(\beta \mid \beta_{t-1})$ is hidden in $\theta_t$ and $\mu_t$. Throughout the paper, we will write $\partial \theta_t / \partial \beta_{t-1}$ and $\partial \mu_t / \partial \beta_{t-1}$ without reminding ourselves of this implicit relationship.

Figure 1 portrays the prospective penalty for two choices of $\beta_{t-1}$ and two sets of tuning parameters $\phi_1, \lambda_1, \lambda_0$ and $\Theta$ (assuming $\phi_0 = 0$). Because the conditional transfer equation (2.5) is a mixture, $\text{pen}(\beta \mid \beta_{t-1})$ is apt to be multimodal. Figure 1(a) shows an obvious peak at zero (due to the Laplace spike), but also a peak around $\mu_t = 0.9 \times \beta_{t-1}$, prioritizing values in the close vicinity of the previous value (due to the non-central slab). From an implementation viewpoint, however, it is more desirable that the penalty be uni-modal, reflecting the size of the previous coefficient without ambiguity by suppressing one of the peaks. Such behavior is illustrated in Figure 1(b) and Figure 1(c), where the penalty flexibly adapts to $|\beta_{t-1}|$ by promoting either zero or a value close to $\beta_{t-1}$. This effect is achieved with a relatively large stationary slab variance, such as $\lambda_1/(1 - \phi_1^2) = 10$, a mild Laplace peak $\lambda_0 = 1$ and the marginal importance weight $\Theta = 0.9$. Smaller values $\Theta$ would provide an overwhelming support for the zero mode. The parameter $\Theta$, thus should not be regarded as a proportion of active coefficients (as is customary with point-mass mixtures), but rather an interpretation-free tuning parameter.

Figure 1 plots $\text{pen}(\beta \mid \beta_{t-1})$ prospectively as a function of $\beta$, given the previous value $\beta_{t-1}$. It is also illuminating to plot $\text{pen}(\beta_{t+1} \mid \beta)$ retrospectively as a function of $\beta$, given the future value $\beta_{t+1}$. Two such retrospective penalty plots are provided in Figure 2(a) and Figure 2(b). When the future value is relatively large ($\beta_{t+1} = 1.5$ in Figure 2(b)), the penalty $\text{pen}(\beta_{t+1} \mid \beta)$ has a peak near $\beta_{t+1}$, signaling that the value $\beta_t$ must be large too. When the future value is small ($\beta_{t+1} = 0$ in Figure 2(a)), the penalty has a peak at zero signaling that the current value $\beta_t$ must have been small. Again, this balance is achieved with a relatively large stationary slab variance and a large $\Theta$.

The behavior of the prospective and retrospective penalties is ultimately tied to the mixing weight $\theta_t \equiv \theta(\beta)$ in (2.11). It is desirable that $\theta(\beta)$ is increasing with $|\beta|$. However, Laplace tails will begin to dominate for large enough $|\beta|$, where the probability $\theta(\beta)$ will begin to drop (for $|\beta|$ greater than $\delta \equiv (\lambda_0 + \sqrt{2C/A})A$, where $A = \lambda_1/(1 - \phi_1^2)$ and $C = \log((1 - \theta)/\Theta \lambda_0/2\sqrt{2\pi A})$). However, we can make the turning point $\delta$ large enough with larger values $\Theta$ and smaller values...
\( \lambda_0 \), as indicated in Figure 2(c).

To describe the shrinkage dynamics implied by the penalty (3.15), it is useful to study the partial derivative \( \partial \text{Pen}(\beta | \beta_{t-1}, \beta_{t+1}) / \partial \beta \). This term encapsulates how much shrinkage we expect at time \( t \), conditionally on \( (\beta_{t-1}, \beta_{t+1}) \). We will separate the term into two pieces: a prospective shrinkage effect \( \lambda^*(\beta | \beta_{t-1}) \), driven by the past value \( \beta_{t-1} \), and a retrospective shrinkage effect \( \bar{\lambda}^*(\beta | \beta_{t+1}) \), driven by the future value \( \beta_{t+1} \). More formally, we write

\[
\frac{\partial \text{Pen}(\beta | \beta_{t-1}, \beta_{t+1})}{\partial \beta} \equiv -\Lambda^*(\beta | \beta_{t-1}, \beta_{t+1}),
\]

where

\[
\Lambda^*(\beta | \beta_{t-1}, \beta_{t+1}) = \lambda^*(\beta | \beta_{t-1}) + \bar{\lambda}^*(\beta | \beta_{t+1}), \tag{3.16}
\]

and

\[
\lambda^*(\beta | \beta_{t-1}) = -\frac{\partial \text{pen}(\beta | \beta_{t-1})}{\partial \beta} \quad \text{and} \quad \bar{\lambda}^*(\beta | \beta_{t+1}) = -\frac{\partial \text{pen}(\beta_{t+1} | \beta)}{\partial \beta}.
\]

### 3.1 Shrinkage “from the Past”

The prospective shrinkage term \( \lambda^*(\beta | \beta_{t-1}) \) pertains to Bayesian penalty mixing introduced by Rockova (2017) and Rockova and George (2016) in the sense that it can be characterized as an adaptive linear combination of individual spike and slab shrinkage terms. In particular, we can write

\[
\lambda^*(\beta | \beta_{t-1}) = -p_t^*(\beta) \frac{\partial \log \psi_1(\beta | \mu_t, \lambda_1)}{\partial \beta} - [1 - p_t^*(\beta)] \frac{\partial \log \psi_0(\beta | \lambda_0)}{\partial \beta}, \tag{3.17}
\]

where

\[
p_t^*(\beta) = \frac{\theta_t \psi_1(\beta | \mu_t, \lambda_1)}{\theta_t \psi_1(\beta | \mu_t, \lambda_1) + (1 - \theta_t) \psi_0(\beta | \lambda_0)}. \tag{3.19}
\]

Two observations are in order: first, by writing \( p_t^*(\beta) = P(\gamma_t = 1 | \beta_t = \beta, \beta_{t-1}, \theta_t) \), (3.19) can be viewed as a posterior probability for classifying \( \beta \) as arising from the conditional slab (versus the spike) at time \( t \), given the previous value \( \beta_{t-1} \). Second, these weights are very different
While the prospective shrinkage term promotes smooth forward proliferation, the retrospective shrinkage term from $\theta_t$ in (2.11), which are classifying $\beta$ as arising from the marginal slab (versus the spike). From (3.19), we can see how $p_t^*(\beta)$ hierarchically transmits information about the past value $\beta_{t-1}$ (via $\theta_t$) to determine the right shrinkage for $\beta_t$. This is achieved with a doubly-adaptive chain reaction. Namely, if the previous value $\beta_{t-1}$ was large, $\theta_t$ will be close to one signaling that the next coefficient $\beta_t$ is prone to be in the slab. Next, if $\beta_t$ is in fact large, $p_t^*(\beta_t)$ will be close to one, where the first summand in (3.17) becomes the leading term and shrinks $\beta_t$ towards $\mu_t$. If $\beta_t$ is small, however, $p_t^*(\beta_t)$ will be small as well, where the second term in (3.17) takes over to shrink $\beta_t$ towards zero. This gravitational pull is accelerated when the previous value $\beta_{t-1}$ was negligible (zero), in which case $\theta_t$ will be even smaller, making it even more difficult for the next coefficient $\beta_t$ to escape the spike. This mechanism explains how the prospective penalty adapts to both $(\beta_{t-1}, \beta_t)$, promoting smooth forward proliferation of spike/slab allocations and coefficients.

3.2 Shrinkage “from the Future”

While the prospective shrinkage term promotes smooth forward proliferation, the retrospective shrinkage term $\lambda^*(\beta \mid \beta_{t+1})$ operates backwards. We can write

$$
\lambda^*(\beta \mid \beta_{t+1}) = -\frac{\partial \theta_{t+1}}{\partial [\beta]} \left[ p^*_{t+1}(\beta_{t+1}) \frac{1 - p^*_{t+1}(\beta_{t+1})}{\theta_{t+1}} - p^*_{t+1}(\beta_{t+1}) \phi_1 \text{sign}(\beta) \left( \frac{\beta_{t+1} - \mu_{t+1}}{\lambda_1} \right) \right],
$$

(3.20)

where

$$
\frac{\partial \theta_{t+1}}{\partial [\beta]} = \theta_{t+1}(1 - \theta_{t+1}) \left[ \lambda_0 - \text{sign}(\beta) \left( \frac{\beta - \phi_0}{\lambda_1(1 - \phi_1^2)} \right) \right].
$$

(3.21)

For simplicity, we will write $p^*_{t+1} = p^*_{t+1}(\beta_{t+1})$. Then we have

$$
\lambda^*(\beta \mid \beta_{t+1}) = 
\left[ \lambda_0 - \text{sign}(\beta) \left( \frac{\beta - \phi_0}{\lambda_1(1 - \phi_1^2)} \right) \right] \left[ (1 - p^*_{t+1}) \theta_{t+1} - p^*_{t+1}(1 - \theta_{t+1}) \right]
$$

(3.22)

$$
- p^*_{t+1} \phi_1 \text{sign}(\beta) \left( \frac{\beta_{t+1} - \mu_{t+1}}{\lambda_1} \right).
$$

(3.23)

The retrospective term synthesizes information from both $(\beta_{t+1}, \beta_t)$ to contribute to shrinkage at time $t$. When $(\beta_{t+1}, \beta_t)$ are both large, we obtain $p^*_{t+1}(\beta_{t+1})$ and $\theta_{t+1}$ that are both close to one. The shrinkage is then driven by the second summand in (3.23), forcing $\beta_t$ to be shrunk towards the future value $\beta_{t+1}$ (through $\mu_{t+1} = \phi_0 + \phi_1(\beta_t - \phi_0)$). When either $\beta_{t+1}$ or $\beta_t$ are small, shrinkage is targeted towards the stationary mean through the dominant term (3.22).

4 Characterization of the Joint Posterior Mode

Unlike previous developments (Nakajima and West 2013a; Kalli and Griffin 2014), this paper views Bayesian dynamic shrinkage through the lens of optimization. Rather than distilling posterior samples to learn about $B = [\beta_1, \ldots, \beta_T]$, we focus on finding the MAP trajectory $\hat{B} = \arg \max \pi(B \mid y)$. MAP sequence estimation problems (for non-linear non-Gaussian dynamic models) were addressed previously with, e.g., Viterbi-style algorithms (Godsill et al. 2001). Our optimization strategy is conceptually very different. The key to our approach will be drawing upon
the penalized likelihood perspective developed in Section 3. Namely, we develop a new dynamic coordinate-wise strategy, building on existing developments for static high-dimensional variable selection.

4.1 The One-dimensional Case

To illustrate the functionality of the dynamic penalty from Section 3, we start by assuming \( p = 1 \) and \( x_t = 1 \) in (1.1). This simple case corresponds to a sparse normal-means model, where the means are dynamically intertwined. We begin by characterizing some basic properties of the posterior mode

\[
\hat{\beta} = \arg\max_{\beta} \pi(\beta | y),
\]

where \( y = (y_1, \ldots, y_T)' \) arises from (1.1) and \( \beta = (\beta_1, \ldots, \beta_T)' \) is assigned the DSS prior.

One of the attractive features of the Laplace spike in (2.5) is that \( \hat{\beta} \) has a thresholding property. This property is revealed from necessary characterizations for each \( \hat{\beta}_t \) (for \( t = 1, \ldots, T \)), once we condition on the rest of the directions through \( (\hat{\beta}_{t-1}, \hat{\beta}_{t+1}) \). The conditional thresholding rule can be characterized using standard arguments, as with similar existing regularizers (Zhang and Zhang 2012; Rockova and George 2016). While the typical sparsity-inducing penalty functions are symmetric, the penalty (3.15) is not, due to its dependence on the previous and future values \((\hat{\beta}_{t-1}, \hat{\beta}_{t+1})\). Thereby, instead of a single selection threshold, we have two:

\[
\Delta^- (x, \beta_{t-1}, \beta_{t+1}) = \sup_{\beta < 0} \left\{ \frac{\beta x^2}{2} + \frac{\text{Pen}(\beta | \beta_{t-1}, \beta_{t+1})}{\beta} \right\}, \tag{4.24}
\]

\[
\Delta^+ (x, \beta_{t-1}, \beta_{t+1}) = \inf_{\beta > 0} \left\{ \frac{\beta x^2}{2} + \frac{\text{Pen}(\beta | \beta_{t-1}, \beta_{t+1})}{\beta} \right\}. \tag{4.25}
\]

The following necessary characterization links the behavior of \( \hat{\beta} \) to the shrinkage terms characterized in Section 3.1 and Section 3.2.

**Lemma 1.** Denote by \( \bar{\beta} = (\bar{\beta}_1, \ldots, \bar{\beta}_T)' \) the global mode of \( \pi(\beta | y) \) and by \( \Delta_t^- \) and \( \Delta_t^+ \) the selection thresholds (4.24) and (4.25) with \( x = 1 \), \( \beta_{t-1} = \bar{\beta}_{t-1} \) and \( \beta_{t+1} = \bar{\beta}_{t+1} \). Then, conditionally on \((\hat{\beta}_{t-1}, \hat{\beta}_{t+1})\), we have for \( 1 < t < T \)

\[
\hat{\beta}_t = \begin{cases} 
0 & \text{if } \Delta_t^- < y_t < \Delta_t^+, \\
||y_t| - \Lambda^*(\hat{\beta}_t | \hat{\beta}_{t-1}, \hat{\beta}_{t+1})| + \text{sign}(y_t) & \text{otherwise},
\end{cases} \tag{4.26}
\]

where \( \Lambda^*(\hat{\beta}_t | \hat{\beta}_{t-1}, \hat{\beta}_{t+1}) \) was defined in (3.16).

**Proof.** We begin by noting that \( \hat{\beta}_t \) is a maximizer in \( t \)th direction while keeping \((\hat{\beta}_{t-1}, \hat{\beta}_{t+1})\) fixed, i.e.

\[
\hat{\beta}_t = \arg\max_{\beta} \left\{ -\frac{1}{2} (y_t - \beta)^2 + \text{Pen}(\beta | \hat{\beta}_{t-1}, \hat{\beta}_{t+1}) \right\}. \tag{4.27}
\]

It turns out that \( \hat{\beta}_t = 0 \) iff \( \beta \left( \frac{1}{2} + \frac{\text{Pen}(\beta | \hat{\beta}_{t-1}, \hat{\beta}_{t+1}) - y_t}{\beta} \right) < 0, \forall \beta \in \mathbb{R}\setminus\{0\} \) (Zhang and Zhang 2012). The rest of the proof follows from the definition of \( \Delta_t^- \) and \( \Delta_t^+ \) in (4.24) and (4.25). Conditionally on \((\hat{\beta}_{t-1}, \hat{\beta}_{t+1})\), the global mode \( \hat{\beta}_t \), once nonzero, has to satisfy (4.26) from the first-order necessary condition. \( \square \)
Lemma 1 formally certifies that the mode exhibits both (a) sparsity and (b) smoothness (through the prospective/retrospective shrinkage terms).

Remark 3. While Lemma 1 assumes $1 < t < T$, the characterization applies also for $t = 1$, once we specify the initial condition $\beta_{t=0}$. The value $\beta_{t=0}$ is not assumed known and will be estimated together with all the remaining parameters (Section 5). For $t = T$, an analogous characterization exists, where the shrinkage term and the selection threshold only contain the prospective portion of the penalty.

4.2 General Regression

When $p > 1$, there is a delicate interplay between the multiple series, where overfitting in one direction may impair recovery in other directions. As will be seen in Section 6, anchoring on sparsity is a viable remedy to these issues. We obtain analogous characterizations of the global mode. We will denote with $\Delta_0^{\ell}$ and $\Delta_0^+$ the selection thresholds (4.24) and (4.25) with $x = x_{tj}, \beta_{t-1} = \hat{\beta}_{t-1,j}$, and $\beta_{t+1} = \hat{\beta}_{t+1,j}$.

Lemma 2. Denote by $\hat{B} = \{\hat{\beta}_{tj}\}_{t,j=1}^{T,p}$ the global mode of $\pi(B|Y)$, by $\hat{B}_{tj}$ all but the $(t,j)^{th}$ entry in $\hat{B}$ and by $z_{tj} = y_t - \sum_{i \neq j} x_{ti}\hat{\beta}_{ti}$. Let $Z_{tj} = x_{tj}z_{tj}$. Then $\hat{\beta}_{tj}$ satisfies the following necessary condition

$$
\hat{\beta}_{tj} = \begin{cases} 
\frac{1}{x_{tj}} [Z_{tj} - \Lambda^*(\hat{\beta}_{tj} | \hat{\beta}_{t-1,j}, \hat{\beta}_{t+1,j})]_+ \text{ sign}(Z_{tj}) & \text{if } \Delta_0^{\ell} < Z_{tj} < \Delta_0^+ \\
0 & \text{otherwise.}
\end{cases}
$$

Proof. Follows from Lemma 1, noting that $\hat{\beta}_{tj}$ is a maximizer in $(t,j)^{th}$ direction while keeping $\hat{B}_{\backslash t}$ fixed, i.e.

$$
\hat{\beta}_{tj} = \arg \max_{\beta} \left\{ -\frac{1}{2} (z_j - x_{tj}\beta)^2 + \text{Pen}(\beta | \hat{\beta}_{t-1,j}, \hat{\beta}_{t+1,j}) \right\}.
$$

(4.28)

Lemma 2 evokes coordinate-wise optimization for obtaining the posterior mode. However, the computation of selection thresholds $(\Delta_{0}^{\ell}, \Delta_{0}^{+})$ (as well as the one-site maximizers (4.27)) requires numerical optimization. The lack of availability of closed-form thresholding hampers practicality when $T$ and $p$ are even moderately large. In the next section, we propose an alternative strategy which capitalizes on closed-form thresholding rules.

5 One-step-late EM Algorithm

A (local) posterior mode $\hat{B}$ can be obtained either directly, by cycling over one-site updates (4.28), or indirectly through an EM algorithm, a strategy pursued here. The direct algorithm consists of integrating out $G = [\gamma_1, \ldots, \gamma_T]'$ and solving a sequence of non-standard optimization problems (4.28), which necessitate numerical optimization. The EM algorithm, on the other hand, treats $G$ as missing data, obviating the need for numerical optimization by offering closed form one-site updates. We now describe this dynamic EM approach for MAP estimation under DSS priors.

The initial vector $\beta_{t=0} = (\beta_{01}, \ldots, \beta_{0p})'$ at time $t = 0$ will be estimated together with all the remaining coefficients $B$. We assume that $\beta_0$ comes from the stationary distribution described in
Theorem 1
\[
\pi(\beta_0, \gamma_0) = \prod_{j=1}^{p} \left[ \gamma_0 j \psi_1^{ST}(\beta_0 j \mid \lambda_1, \phi_0, \phi_1) + (1 - \gamma_0 j) \psi_0(\beta_0 j \mid \lambda_0) \right],
\]
(5.29)

where \( \gamma_0 = (\gamma_{01}, \ldots, \gamma_{0p})' \) are independent binary indicators with \( P(\gamma_{0j} = 1 \mid \Theta) = \Theta \) for \( 1 \leq j \leq p \).

Knowing the stationary distribution, thus, has advantages for specifying the initial conditions. The goal is obtaining the joint mode \([B, \beta_0]\) of the functional \( \pi(B, \beta_0 | Y) \). To this end, we proceed iteratively by augmenting this objective function with the missing data \([G, \gamma_0]\), as prescribed by Rockova and George (2014), and then maximizing w.r.t. \([B, \beta_0]\). An important observation, that facilitates the derivation of the algorithm, is that the prior distribution \( \pi(B, \beta_0 | Y) \) can be factorized into the following products

\[
\pi(\beta_0, B, \gamma_0, G) = \pi(\beta_0 | \gamma_0) \prod_{j=1}^{p} \prod_{t=1}^{T} \pi(\beta_{tj} | \gamma_{tj}, \beta_{t-1j}) \pi(\gamma_{tj} | \beta_{t-1j}),
\]

where \( \pi(\beta_{tj} | \gamma_{tj}, \beta_{t-1j}) \) and \( \pi(\gamma_{tj} | \beta_{t-1j}) \) are defined in (2.5) and (2.7), respectively. For simplicity (and without loss of generality), we will assume \( \phi_0 = 0 \) and thereby \( \mu_{tj} = \phi_1 \beta_{t-1j} \). Then, we can write

\[
\log \pi(\beta_0, B, \gamma_0, G | Y) = C + \sum_{t=1}^{T} \sum_{j=1}^{p} \left[ \gamma_{tj} \log \theta_{tj} + (1 - \gamma_{tj}) \log(1 - \theta_{tj}) \right]
\]

\[
- \sum_{t=1}^{T} \left\{ \frac{(y_t - x_t^j \beta_t)^2}{2} + \sum_{j=1}^{p} \left[ \gamma_{tj} \frac{(\beta_{tj} - \phi_1 \beta_{t-1j})^2}{2\lambda_1} + (1 - \gamma_{tj}) \lambda_0 | \beta_{tj} \right] \right\}
\]

\[
- \sum_{j=1}^{p} \left[ \gamma_{0j} \beta_{0j}^2 (1 - \phi_1^2) + (1 - \gamma_{0j}) \lambda_0 | \beta_{0j} | - \gamma_{0j} \log \Theta - (1 - \gamma_{0j}) \log(1 - \Theta) \right].
\]

We will endow the parameters \([\beta_0, B]\) with a superscript \( m \) to designate their most recent values at the \( m^{th} \) iteration. In the E-step, we compute the expectation \( E_{\gamma_0,G} \log \pi(\beta_0, B, \gamma_0, G | Y) \) with respect to the conditional distribution of \([G, \gamma_0]\), given \([B^{(m)}, \beta_0^{(m)}]\) and \( Y \). All we have to really do is obtain \( p_t^{*} = P(\gamma_{tj} = 1 | \beta^{(m)}_{tj}, \beta^{(m)}_{t-1j}, \theta_j) \) from (3.19), when \( t > 0 \), and \( p_0^{*} \equiv \theta_j \equiv \theta(\beta_{0j}) \) from (2.11), and replace all the \( \gamma_{tj}'s \) in (5.30) with \( p_t^{*} \)'s. In the M-step, we set out to maximize \( E_{\gamma_0,G} \log \pi(\beta_0, B, \gamma_0, G | Y) \) w.r.t. \([\beta_0, B]\). We proceed coordinate-wise, iterating over the following single-site updates while keeping all the remaining parameters fixed. For \( 1 < t < T \), we have

\[
\beta_{tj}^{(m+1)} = \arg \max_{\beta} Q_{tj}(\beta),
\]

where

\[
Q_{tj}(\beta) = -\frac{1}{2} (z_{tj} - x_{tj} \beta)^2 - \frac{p_{tj}^*}{2 \lambda_1} (\beta - \phi_1 \beta_{t-1j})^2 - \frac{p_{t+1j}^*}{2 \lambda_1} (\beta_{t+1j}^{(m)} - \phi_1 \beta)^2
\]

\[
- (1 - p_{tj}^*) \lambda_0 | \beta | + p_{t+1j}^* \log \theta_{t+1j} + (1 - p_{t+1j}^*) \log(1 - \theta_{t+1j}),
\]

(5.31)

and where \( z_{tj} = y_t - \sum_{i \neq j} x_{tj} \beta_{ti}^{(m)} \). From the first-order condition, the solution \( \beta_{tj}^{(m+1)} \), if nonzero, needs to satisfy \( \partial Q_{tj}(\beta) / \partial \beta |_{\beta = \beta_{tj}^{(m+1)}} = 0 \). To write the derivative slightly more concisely, we introduce the following notation: \( Z_{tj} = x_{tj} z_{tj} + \frac{p_{tj}^*}{\lambda_1} \beta_{t-1j}^{(m)} + \frac{p_{t+1j}^*}{\lambda_1} \beta_{t+1j}^{(m+1)} \) and
\[ W_{tj} = \left( x_{tj}^2 + \frac{\mu_j^2}{\lambda_1} + \frac{\mu_{t+1,j}^2}{\lambda_1} \right). \]

Then we can write for \( \beta \neq 0 \)

\[
\frac{\partial Q_{tj}(\beta)}{\partial \beta} = Z_{tj} - W_{tj} \beta - (1 - p_{tj}) \lambda_0 \text{sign}(\beta) + \frac{\partial \theta_{t+1,j}}{\partial \beta} \left[ \frac{p_{t+1,j}^*}{\theta_{t+1,j}} - \frac{1 - p_{t+1,j}^*}{1 - \theta_{t+1,j}} \right], \tag{5.32}
\]

where

\[
\frac{\partial \theta_{t+1,j}}{\partial \beta} = \theta_{t+1,j}(1 - \theta_{t+1,j}) \left[ \lambda_0 \text{sign}(\beta) - \frac{\beta(1 - \phi_j^2)}{\lambda_1} \right]
\]

is obtained from (3.21). Recall that \( \theta_{t+1,j} \), defined in (2.11), depends on \( \beta_{tj} \) (denoted by \( \beta \) above). This complicates the tractability of the M-step. If \( \theta_{t+1,j} \) was fixed, we could obtain a simple closed-form solution \( \beta_{tj}^{(m+1)} \) through an elastic-net-like update (Zou and Hastie 2005). We can take advantage of this fact with a one-step-late (OSL) adaptation of the EM algorithm (Green 1990). The OSL EM algorithm bypasses intricate M-steps by evaluating the intractable portions of the penalty derivative at the most recent value, rather than the new value. We apply this trick to the last summand in (5.32). Instead of treating \( \theta_{t+1,j} \) as a function of \( \beta \) in (5.32), we fix it at the most recent value \( \beta_{tj}^{(m)} \). The solution for \( \beta \), implied by (5.32), is then

\[
\beta_{tj}^{(m+1)} = \frac{1}{W_{tj} + (1 - \phi_j^2)/(\lambda_1 M_{tj})} \left[ Z_{tj} - \Lambda_{tj} \right] \text{sign}(Z_{tj}), \quad \text{for } 1 < t < T, \tag{5.33}
\]

where \( M_{tj} = p_{t+1,j}^*(1 - \theta_{t+1,j}) - \theta_{t+1,j}(1 - p_{t+1,j}^*) \) and \( \Lambda_{tj} = \lambda_0 [1 - (1 - p_{tj}^*) - M_{tj}] \). The update (5.33) is a thresholding rule, with a shrinkage term that reflects the size of \( \beta_{t-1,j}, \beta_{t,j}, \beta_{t+1,j}^{(m)} \). The exact thresholding property is obtained from sub-differential calculus, because \( Q_{tj}(\cdot) \) is not differentiable at zero (due to the Laplace spike). A very similar update is obtained also for \( t = T \), where all the terms involving \( p_{t+1,j}^* \) and \( \theta_{t+1,j} \) in \( \Lambda_{tj} \), \( W_{tj} \) and \( Z_{tj} \) disappear. For \( t = 0 \), we have

\[
\beta_{0j}^{(m+1)} = \frac{1}{p_{tj}^*\phi_j^2 + p_{0j}^*(1 - \phi_j^2)} \left[ p_{0j}^*\beta_{tj}\phi_1 - (1 - p_{0j}^*)\lambda_0\lambda_1 \right] \text{sign}(\beta_{tj}). \tag{5.34}
\]

The updates (5.33) and (5.34) can be either cycled-over at each M-step, or performed just once for each M-step.

**Remark 4.** For autoregression with a higher order \( h > 1 \), the retrospective penalty would be similar where \( \mu_t \) would depend on \( h \) lagged values. The prospective penalty would consist of not just one, but \( h \) terms. For the EM implementation, one would proceed analogously by evaluating the derivatives of \( \theta_{t+1}, \ldots, \theta_{t+h} \) w.r.t. \( \beta \) at the most recent update of the process from the previous iteration and keep them fixed for each one-site update.

To illustrate the ability of the DSS priors to suppress noise and recover true signal, we consider a high-dimensional synthetic dataset and a topical macroeconomic dataset.

### 6 Synthetic High-Dimensional Data

We first illustrate our dynamic variable selection procedure on a simulated example with \( T = 100 \) observations generated from the model (1.1) with \( p = 50 \) predictors. The predictor values \( x_{tj} \) are obtained independently from a standard normal distribution. Out of the 50 predictors, 46 never contribute to the model (predictors \( x_{15} \) through \( x_{150} \)), where \( \beta_{15}^0 = \beta_{16}^0 = \ldots = \beta_{150}^0 = 0 \) at all times. The predictor \( x_{11} \) is a persisting predictor, where \( \{\beta_{11}\}_{t=1}^T \) is generated according
Figure 3: The first six regression coefficients of the true and recovered time series from the high-dimensional simulated example with $p = 50$. 

to an $AR(1)$ process (2.8) with $\phi_0 = 0$ and $\phi_1 = 0.98$ and where $|\beta_{11}^0| > 0.5$. The remaining three predictors are allowed to enter and leave the model as time progresses. The regression coefficients $\{\beta_{12}^0\}_{t=1}^T, \{\beta_{13}^0\}_{t=1}^T$ and $\{\beta_{14}^0\}_{t=1}^T$ are again generated from an $AR(1)$ process ($\phi_0 = 0$ and $\phi_1 = 0.98$). However, the values are thresholded to zero whenever the absolute value of the process drops below 0.5, creating zero-valued periods. The true sparse series of coefficients are depicted in Figure 3 (black lines).

We begin with the standard DLM approach, which is equivalent to $DSS$ when the selection indicators are switched on at all times, i.e., $\gamma_{tj} = 1$ for $t = 0, \ldots, T$ and $j = 1, \ldots, p$. The autoregressive parameters are set to the true values $\phi_0 = 0$ and $\phi_1 = 0.98$. Plots of the estimated trajectories of the first 6 series (including the 4 active ones) are in Figure 3 (orange lines). With the absence of the spike, the estimated series of coefficients cannot achieve sparsity. By failing to discern the coefficients as active or inactive, the state process confuses the source of the signal, distributing it across the redundant covariates. This results in loss of efficiency and poor recovery.

With the hope to improve on this recovery, we deploy the $DSS$ process with a sparsity inducing spike. For now, the hyper-parameters are chosen as $\phi_0 = 0, \phi_1 = 0.98, \lambda_0 = 0.9, \lambda_1 = 25(1 - \phi_1^2)$, and $\Theta = 0.98$. This hyper-parameter choice corresponds to a very mild separation between the stationary spike and slab distributions, and unimodal retrospective and prospective penalties. Later, we explore the sensitivity of $DSS$ to this choice. We deploy the one-step-late EM algorithm outlined in Section 5, initializing the calculation with a zero matrix. We assume that the initial vector $\beta_{t=0}$ is drawn from the stationary distribution and we estimate it together with all the other parameters, as prescribed in Section 5.

The recovered series have a strikingly different pattern compared to the non-sparse solution (Figure 3, blue lines). First, the MAP series is seen to track closely the periods of predictor importance/irrelevance, achieving dynamic variable selection. Second, by harvesting sparsity, the
Figure 4: Mixing weights $\theta_{ij}$, defined in (2.11), for the simulated examples in Section 6 for the first six series.

DSS priors alleviate bias in the nonzero directions, outputting a cleaner representation of the true underlying signal.

We also compare the performance to the NGAR process of Kalli and Griffin (2014) and the LASSO method. The latter does not take into account the temporal nature of the problem. For NGAR, we use the default settings, $b^* = s^* = 0.1$, with 1,000 burn-in and 1,000 MCMC iterations. For LASSO, we sequentially run a static regression in an extending window fashion, where the LASSO regression is refit using $1 : t$ for each $t = 1 : T$ to produce a series of quasi-dynamic coefficients; a common practice for using static shrinkage methods for time series data (Bai and Ng 2008; De Mol et al. 2008; Stock and Watson 2012; Li and Chen 2014), choosing $\lambda$ via 10-fold cross-validation.

For the first series, the only persistent series, we see that both NGAR and DSS succeed well in tracing the true signal. This is especially true in contrast to DLM and LASSO, which both significantly under-valuate the signal. The estimated coefficient evolutions for DLM and LASSO become inconclusive for assessing variable importance, where the coefficient estimates for the relevant variables have been reduced by the increased estimates for the irrelevant variables. For the second to fourth series with intermittent zeros, we see that DSS is the only method able to separate the true zero/nonzero signal (noted by the flat coefficient estimates during inactive periods). On the other hand, NGAR seems to be not shrinking enough, instead smoothing the series as if curve-fitting. The LASSO method produces sparse estimates, however the variable selection is not linked over time and thereby erratic. For the two zero series (series five and six), it is clear that DSS is the only method that truly shrinks noise to zero. The DSS priors mitigate overfitting by eliminating noisy coefficients and thereby leaving enough room for the true predictors to capture the trend. As a revealing byproduct, we also obtain the evolving mixing weights determining the relevance of each coefficient at each time. The evolutions of rescaled
Table 1: Performance evaluation of the methods compared for the high-dimensional simulated example with $p = 50$. The results are split for the signal parameters ($x_{1:4}$) and noise parameters ($x_{5:50}$). Hamming distance is in percentages. Number in brackets for $DSS$ is using the hyperparameter set $\{\phi_1, \phi_0, \lambda_1/(1 - \phi_1^2), \Theta\}$. Best 5 results in $DSS$ are in bold.

<table>
<thead>
<tr>
<th>$p = 50$</th>
<th>Time (s)</th>
<th>$x_{1:50}$</th>
<th>$x_{1:4}$</th>
<th>$x_{5:50}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLM</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>NGAR</td>
<td>97.1</td>
<td>97.1</td>
<td>97.1</td>
<td>97.1</td>
</tr>
<tr>
<td>LASSO</td>
<td>532.3</td>
<td>532.3</td>
<td>532.3</td>
<td>532.3</td>
</tr>
<tr>
<td><strong>DSS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{.95, .7, 10, .9}</td>
<td>19.7</td>
<td>123.0</td>
<td>123.0</td>
<td>123.0</td>
</tr>
<tr>
<td>{.95, .7, 10, .95}</td>
<td>19.3</td>
<td>89.5</td>
<td>89.5</td>
<td>89.5</td>
</tr>
<tr>
<td>{.95, .7, 10, .98}</td>
<td>17.0</td>
<td>108.2</td>
<td>108.2</td>
<td>108.2</td>
</tr>
<tr>
<td>{.95, .9, 10, .9}</td>
<td>19.9</td>
<td>206.3</td>
<td>206.3</td>
<td>206.3</td>
</tr>
<tr>
<td>{.95, .9, 10, .95}</td>
<td>19.5</td>
<td>127.6</td>
<td>127.6</td>
<td>127.6</td>
</tr>
<tr>
<td>{.95, .9, 10, .98}</td>
<td>19.0</td>
<td>91.8</td>
<td>91.8</td>
<td>91.8</td>
</tr>
<tr>
<td>{.95, .7, 25, .9}</td>
<td>21.8</td>
<td>425.1</td>
<td>425.1</td>
<td>425.1</td>
</tr>
<tr>
<td>{.95, .7, 25, .95}</td>
<td>21.0</td>
<td>341.8</td>
<td>341.8</td>
<td>341.8</td>
</tr>
<tr>
<td>{.95, .7, 25, .98}</td>
<td>21.0</td>
<td>233.9</td>
<td>233.9</td>
<td>233.9</td>
</tr>
<tr>
<td>{.95, .9, 25, .9}</td>
<td>19.0</td>
<td>468.8</td>
<td>468.8</td>
<td>468.8</td>
</tr>
<tr>
<td>{.95, .9, 25, .95}</td>
<td>17.4</td>
<td>404.4</td>
<td>404.4</td>
<td>404.4</td>
</tr>
<tr>
<td>{.95, .9, 25, .98}</td>
<td>16.6</td>
<td>393.7</td>
<td>393.7</td>
<td>393.7</td>
</tr>
<tr>
<td>{.98, .7, 10, .9}</td>
<td>18.2</td>
<td>130.6</td>
<td>130.6</td>
<td>130.6</td>
</tr>
<tr>
<td>{.98, .7, 10, .95}</td>
<td>22.8</td>
<td>215.1</td>
<td>215.1</td>
<td>215.1</td>
</tr>
<tr>
<td>{.98, .7, 10, .98}</td>
<td>14.5</td>
<td>290.8</td>
<td>290.8</td>
<td>290.8</td>
</tr>
<tr>
<td>{.98, .9, 10, .9}</td>
<td>20.0</td>
<td>79.7</td>
<td>79.7</td>
<td>79.7</td>
</tr>
<tr>
<td>{.98, .9, 10, .95}</td>
<td>20.9</td>
<td>81.5</td>
<td>81.5</td>
<td>81.5</td>
</tr>
<tr>
<td>{.98, .9, 10, .98}</td>
<td>9.2</td>
<td>148.0</td>
<td>148.0</td>
<td>148.0</td>
</tr>
<tr>
<td>{.98, .7, 25, .9}</td>
<td>17.7</td>
<td>162.9</td>
<td>162.9</td>
<td>162.9</td>
</tr>
<tr>
<td>{.98, .7, 25, .95}</td>
<td>13.7</td>
<td>94.5</td>
<td>94.5</td>
<td>94.5</td>
</tr>
<tr>
<td>{.98, .7, 25, .98}</td>
<td>17.1</td>
<td>75.2</td>
<td>75.2</td>
<td>75.2</td>
</tr>
<tr>
<td>{.98, .9, 25, .9}</td>
<td>12.3</td>
<td>254.1</td>
<td>254.1</td>
<td>254.1</td>
</tr>
<tr>
<td>{.98, .9, 25, .95}</td>
<td>12.0</td>
<td>152.7</td>
<td>152.7</td>
<td>152.7</td>
</tr>
<tr>
<td>{.98, .9, 25, .98}</td>
<td>10.6</td>
<td>91.3</td>
<td>91.3</td>
<td>91.3</td>
</tr>
</tbody>
</table>

weights (so that $\theta(0) = 0$) are plotted in Figure 4. These companion plots are helpful visualizations of the time-varying sparsity profile.

We repeat this experiment 10 times, generating different responses and regressors using the same set of coefficients. We compare the sum of squared error (SSE) and the Hamming distance between the MAP estimate $\hat{B}$ and the true series $B_0$. Table 1 reports average performance metrics over the 10 experiments. The performance of $DSS$ is compared to the full DLM model (West and Harrison 1997), NGAR (Kalli and Griffin 2014) and LASSO. For DLM and NGAR, we use the same set of coefficients. We compare the sum of squared error (SSE) and the Hamming distance for the block of noisy coefficients with $\phi_1 = \{0.95, 0.98\}$, $\lambda_0 = \{0.7, 0.9\}$, $\lambda_1 = \{10(1 - \phi_1^2), 25(1 - \phi_1^2)\}$, and $\Theta = \{0.9, 0.95, 0.98\}$. All these parameters are in the mild sparsity range; not over-emphasizing the spike. We initialize the calculation with a zero matrix.

Looking at Table 1, $DSS$ performs better in terms of both SSE and Hamming distance compared to DLM, NGAR, and LASSO for the majority of the hyperparameters considered. To gain more insights, the table is divided into three blocks: overall performance on $\beta_{1:50}$, active coefficients $\beta_{1:4}$ and noise coefficients $\beta_{5:50}$. The Hamming distance is reported in percentages. Because DLM and NGAR only shrink (and do not select), the Hamming distance for the block of noisy coefficients is 100%. The number of false positives (and thereby the overall Hamming distance) is seen to increase with $\Theta$, where large values of $\Theta$ have to be compensated with a larger spike parameter $\lambda_0$ to shrink the noisy coefficients to zero. The stationary slab variance $\lambda_1/(1 - \phi_1^2)$
Additionally, we note that DLM (only dynamics) and LASSO (only shrinkage) underperform. Performance. Regarding the choice of $\phi_1$, larger values seem beneficial for the signal coefficients, where borrowing more strength enhances stability in predictive periods.

Although there are some settings where DSS underperforms, it is clear that DSS has the potential to greatly improve over existing methods for a wide range of hyperparameters. In terms of SSE, DSS outperforms all other methods in estimating $\beta_{5:50}$, demonstrating great success in suppressing unwanted parameters. Regarding the choice of $\phi_1$, larger values seem beneficial for the signal coefficients, where borrowing more strength enhances stability in predictive periods.

Comparing the results with DLM and LASSO, DSS showcases the benefits of combining dynamics and shrinkage, since DLM (only dynamics) and LASSO (only shrinkage) underperform significantly. Additionally, we note that DSS is substantially faster, in terms of computation time, compared to NGAR and LASSO, which is appealing for practical applications.

Now, we explore a far more challenging scenario, repeating the example with $p = 1000$ instead

Table 2: Performance evaluation of the methods compared for the high-dimensional simulated example with $p = 1000$. The results are split for the signal parameters ($x_{1:4}$) and noise parameters ($x_{5:1000}$). Hamming distance is in percentages. Number in brackets for DSS is using the hyperparameter set $\{\phi_1, \lambda_0, \lambda_1/(1 - \phi_1^2), \Theta\}$. Best 5 results in DSS are in bold.

<table>
<thead>
<tr>
<th>$p = 1000$</th>
<th>Time</th>
<th>$x_{1:1000}$</th>
<th>$x_{1:4}$</th>
<th>$x_{5:1000}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLM</td>
<td>6.2</td>
<td>949.5</td>
<td>99.7</td>
<td>936.9</td>
</tr>
<tr>
<td>LASSO</td>
<td>717.1</td>
<td>589.4</td>
<td>1.7</td>
<td>537.2</td>
</tr>
<tr>
<td>DSS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(.95, 7, 10, .9)</td>
<td>854.2</td>
<td>294.7</td>
<td>2.1</td>
<td>293.6</td>
</tr>
<tr>
<td>(.95, 7, 10, .95)</td>
<td>899.4</td>
<td>273.2</td>
<td>3.7</td>
<td>272.2</td>
</tr>
<tr>
<td>(.95, 7, 10, .98)</td>
<td>782.8</td>
<td>302.3</td>
<td>6.7</td>
<td>300.6</td>
</tr>
<tr>
<td>(.95, 9, 10, .9)</td>
<td>658.7</td>
<td>376.0</td>
<td>1.1</td>
<td>373.9</td>
</tr>
<tr>
<td>(.95, 9, 10, .95)</td>
<td>738.9</td>
<td>289.3</td>
<td>1.7</td>
<td>288.1</td>
</tr>
<tr>
<td>(.95, 9, 10, .98)</td>
<td>785.4</td>
<td>288.9</td>
<td>3.6</td>
<td>287.8</td>
</tr>
<tr>
<td>(.95, 7, 25, .9)</td>
<td>490.7</td>
<td>597.2</td>
<td>0.7</td>
<td>590.0</td>
</tr>
<tr>
<td>(.95, 7, 25, .95)</td>
<td>630.9</td>
<td>511.7</td>
<td>1.0</td>
<td>507.7</td>
</tr>
<tr>
<td>(.95, 7, 25, .98)</td>
<td>814.4</td>
<td>375.8</td>
<td>1.7</td>
<td>373.5</td>
</tr>
<tr>
<td>(.95, 9, 25, .9)</td>
<td>388.8</td>
<td>663.2</td>
<td>0.5</td>
<td>652.2</td>
</tr>
<tr>
<td>(.95, 9, 25, .95)</td>
<td>527.4</td>
<td>574.8</td>
<td>0.5</td>
<td>567.3</td>
</tr>
<tr>
<td>(.95, 9, 25, .98)</td>
<td>563.4</td>
<td>439.5</td>
<td>0.9</td>
<td>436.0</td>
</tr>
<tr>
<td>(.95, 7, 10, .9)</td>
<td>1018.1</td>
<td>344.2</td>
<td>10.4</td>
<td>343.3</td>
</tr>
<tr>
<td>(.95, 7, 10, .95)</td>
<td>466.6</td>
<td>397.5</td>
<td>13.5</td>
<td>389.2</td>
</tr>
<tr>
<td>(.95, 7, 10, .98)</td>
<td>300.7</td>
<td>475.8</td>
<td>16.5</td>
<td>450.6</td>
</tr>
<tr>
<td>(.95, 9, 10, .9)</td>
<td>836.6</td>
<td>235.6</td>
<td>3.4</td>
<td>235.0</td>
</tr>
<tr>
<td>(.95, 9, 10, .95)</td>
<td>740.9</td>
<td>251.4</td>
<td>6.4</td>
<td>250.6</td>
</tr>
<tr>
<td>(.95, 9, 10, .98)</td>
<td>383.3</td>
<td>413.6</td>
<td>11.5</td>
<td>409.0</td>
</tr>
<tr>
<td>(.95, 7, 25, .9)</td>
<td>837.2</td>
<td>341.6</td>
<td>1.5</td>
<td>340.2</td>
</tr>
<tr>
<td>(.95, 7, 25, .95)</td>
<td>953.6</td>
<td>242.6</td>
<td>2.4</td>
<td>241.8</td>
</tr>
<tr>
<td>(.95, 7, 25, .98)</td>
<td>787.0</td>
<td>218.1</td>
<td>4.4</td>
<td>217.3</td>
</tr>
<tr>
<td>(.95, 9, 25, .9)</td>
<td>701.9</td>
<td>413.6</td>
<td>0.8</td>
<td>410.9</td>
</tr>
<tr>
<td>(.95, 9, 25, .95)</td>
<td>760.3</td>
<td>320.4</td>
<td>1.2</td>
<td>318.9</td>
</tr>
<tr>
<td>(.95, 9, 25, .98)</td>
<td>781.3</td>
<td>245.4</td>
<td>2.3</td>
<td>244.5</td>
</tr>
</tbody>
</table>

also affects variable selection, where larger values increase the selection threshold and produce less false discoveries. The true is true for the signal coefficients. In terms of SSE, DSS outperforms all other methods in estimating $\beta_{5:50}$, demonstrating great success in suppressing unwanted parameters. Regarding the choice of $\phi_1$, larger values seem beneficial for the signal coefficients, where borrowing more strength enhances stability in predictive periods.
of 50. The coefficients and data generating process are the same with \( p = 50 \), but now instead of 46 noise regressors, we have 996. This high regressor redundancy rate is representative of the “\( p >> n \)” paradigm (“\( p >> T \)” for time series data) and can test the limits of any sparsity inducing procedure. Under this setting, we will be able to truly evaluate the efficacy of DSS and compare it to other methods when there is a large number of predictors with sparse signals.

The results are collated in Table 2. The same set of hyperparameters that performed best for \( p = 50 \) also does extremely well for \( p = 1000 \), dramatically reducing SSE over DLM and LASSO. We have not reported comparisons to NGAR, because it did not seem to produce reliable/sensible results in this high-dimensional scenario. Due to a long running time, we had to consider only short MCMC chains which produced unstable and un-reportable results. In addition, we did not attempt to tune the hyper-parameters and used the default options in the matlab program provided by the authors. On the other hand, DSS produces reasonable estimates, improving upon DLM and LASSO (for the majority of settings). We also note that, while LASSO does perform well in terms of the Hamming distance, it does not do so well in terms of SSE. Because LASSO lacks dynamics, the pattern of sparsity is not smooth over time, leading to erratic coefficient evolutions. Because of the smooth nature of its sparsity, DSS harnesses the dynamics to discern signal from noise, improving in both SSE and Hamming distance.

7 High-Dimensional Macroeconomic Data

We further illustrate the effectiveness of DSS through a topical macroeconomic dataset from McCracken and Ng (2016), which consists of 119 monthly macroeconomic variables over the span of 1986/1 to 2015/12. There are eight primary groups that comprise this dataset: Output and Income, Labor Market, Consumption and Orders, Orders and Inventories, Money and Credit, Interest Rate and Exchange Rates, Prices, and Stock Market. Details of the descriptions and transformations of the predictors can be found in the Appendix of McCracken and Ng (2016). For this example, we aim to forecast and infer on monthly US inflation (Figure 5), a context of topical interest (Cogley and Sargent 2005; Primiceri 2005; Koop et al. 2009; Nakajima and West 2013a). One of the dual mandates of the central bank is to control inflation. To conduct effective economic policy, it is imperative for the policy maker to obtain accurate forecasts and infer on the effects of different indicators on inflation using a large set of available economic indicators. The benefit of shrinkage for this example, as well as in other contexts, is to recover sparse signals from the underlying economic indicators that are pertinent to inflation. However, because the economy is dynamic, it is natural to assume certain indicators to be effective during a certain period but useless during another. For example, one might expect financial indicators to play a significant role in the economy during a financial crisis. The dataset from 2001/1 to 2015/12 is of particular interest, as it includes the subprime mortgage crisis of the late 2000s.

To evaluate our method, we first measure its forecasting ability by conducting one month ahead forecasts and comparing the mean squared cumulative forecast error (MSFE). The analysis is done by cutting the data in half, training the methods using the first half of the data from 1986/1 to 2000/12. We then sequentially update the forecasts through the second half from 2001/1 to 2015/12, updating and rerunning estimation to produce 1-month ahead forecasts every time we observe a new data at each \( t \) (using data from \( 1 : t \) to forecast \( t + 1 \) for \( t = 1 : T - 1 \), where \( t = 1 \) is 1986/1, and \( t = T - 1 \) is 2015/11). Out-of-sample forecasting is thus con-
ducted and evaluated in a way that mirrors the realities facing decision and policy makers. At the end of the analysis (2015/12), we estimate the retrospective coefficients throughout 2001/1 to 2015/12 in order to infer on the recovered signals, given all the data used in the analysis. As with Section 7, we compare DSS against the full DLM, now with discount stochastic volatility to learn the observation and state variance (see, e.g., West and Harrison 1997; Prado and West 2010; Dangl and Halling 2012; Koop and Korobilis 2013; Gruber and West 2016, 2017; Zhao et al. 2016), NGAR, and LASSO (expanding window). For DSS, we use the hyperparameters \( \phi_0 = 0, \phi_1 = 0.98, \lambda_0 = 0.9, \lambda_1 = 25(1 - \phi_1^2), \Theta = 0.98 \), since it produced decent results in the simulation study in Section 7.

On the comparison of forecast ability (Figure 6), it is clear that DSS dramatically improves forecasts over the methods compared (note that the axes are on a different scales). The exact values of MSFE at 2015/12 are DLM: 0.4076; NGAR: 261.7716; LASSO: 0.4307; DSS: 0.0204. DSS significantly improves over the full DLM, which is unsurprising since the full DLM model is plagued with overfitting and false signals. A surprising result is that the forecasting performance of LASSO is slightly worse than the full DLM. This indicates that, even though LASSO might achieve shrinkage, it does not capture the dynamics of signals entering and leaving the threshold. The fact that the forecasting results of LASSO and DLM are similar suggests that the gain coming from either dynamics or shrinkage are similar. DSS, capturing and capitalizing on both dynamics and shrinkage, achieves significant forecast gains that either one or the other alone cannot. For NGAR, the estimation is challenged by the high-dimensionality of the data, producing results that are not as good/reliable. Note that, for NGAR, we used the default settings with 1,000 burn-in and 1,000 iterations.

Figure 7 displays the recovered coefficients from DSS at 2015/12, the end of the analysis, taking into account the full dataset available. Out of the 119 indicators (not including the intercept) only 19 were recovered, suggesting that many of these indicators are noise. As we see from the list of the 19 indicators (Table 3), the recovered predictors span six of the eight groups, covering many different aspects of the economy.

The two largest signals are the two labor indicators (Avg Weekly Hours; both of which are very

Figure 5: Monthly US inflation from 2001/1 to 2015/12.
Figure 6: Mean squared cumulative forecast error from 2001/1 to 2015/12 for DLM (top left), NGAR (top right), LASSO (bottom left), and \textit{DSS} (bottom right). Vertical axes are on a different scale.

Table 3: Economic predictors recovered by \textit{DSS} in the analysis of US inflation from 2001/1 to 2015/12.

<table>
<thead>
<tr>
<th>Group</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output and Income</td>
<td>CUMFNS</td>
<td>Capacity Utilization: Manufacturing</td>
</tr>
<tr>
<td>Labor Market</td>
<td>HWI</td>
<td>Help-Wanted Index for United States</td>
</tr>
<tr>
<td></td>
<td>UEEMPMEAN</td>
<td>Average Duration of Unemployment (Weeks)</td>
</tr>
<tr>
<td></td>
<td>CES0600000007</td>
<td>Avg Weekly Hours : Goods-Producing</td>
</tr>
<tr>
<td></td>
<td>AWHMAN</td>
<td>Avg Weekly Hours : Manufacturing</td>
</tr>
<tr>
<td>Consumption and Orders</td>
<td>HOUST</td>
<td>Housing Starts: Total New Privately Owned</td>
</tr>
<tr>
<td></td>
<td>HOUSTNE</td>
<td>Housing Starts, Northeast</td>
</tr>
<tr>
<td></td>
<td>HOUSTMW</td>
<td>Housing Starts, Midwest</td>
</tr>
<tr>
<td></td>
<td>HOUSTS</td>
<td>Housing Starts, South</td>
</tr>
<tr>
<td></td>
<td>HOUSTW</td>
<td>Housing Starts, West</td>
</tr>
<tr>
<td></td>
<td>PERMIT</td>
<td>New Private Housing Permits (SAAR)</td>
</tr>
<tr>
<td></td>
<td>PERMITNE</td>
<td>New Private Housing Permits, Northeast (SAAR)</td>
</tr>
<tr>
<td></td>
<td>PERMITMW</td>
<td>New Private Housing Permits, Midwest (SAAR)</td>
</tr>
<tr>
<td></td>
<td>PERMITS</td>
<td>New Private Housing Permits, South (SAAR)</td>
</tr>
<tr>
<td></td>
<td>PERMITW</td>
<td>New Private Housing Permits, West (SAAR)</td>
</tr>
<tr>
<td>Interest Rate and</td>
<td>AAAPFM</td>
<td>Moodies Aaa Corporate Bond Minus FEDFUNDS</td>
</tr>
<tr>
<td>Exchange Rates</td>
<td>BAAFFM</td>
<td>Moodies Baa Corporate Bond Minus FEDFUNDS</td>
</tr>
<tr>
<td>Orders and Inventories</td>
<td>UMCSENTx</td>
<td>Consumer Sentiment Index</td>
</tr>
<tr>
<td>Stock Market</td>
<td>VXOCLSx</td>
<td>CBOE S&amp;P 100 Volatility Index</td>
</tr>
</tbody>
</table>

similar), which can be seen as a proxy of the labor market and its effect on inflation. Interestingly, conventional indices of the labor market (including unemployment) are either not recovered or have a very low signal. Our analysis suggests that the number of hours worked is the major indicator in the labor market as it is relevant to inflation.

A large group of the retrieved variables is from the housing market, including all regions and total of housing starts and new private housing permits. This is understandable, as the real-estate market is directly tied to the real economy, including inflation. During the subprime crisis, we
observe an interesting pattern, where most of the coefficients are shrunk to zero. This is most likely due to the crash of the real-estate market, which was significantly greater than that of overall consumption prices. The characteristics of these coefficients demonstrate how DSS is successful in dynamically shrinking coefficients to zero during a regime change (recession) as we would expect to happen.

Looking at financial indices, we first note that the volatility index is persistent throughout the analysis. The volatility index is considered to be a reflection of the investor sentiment, so the persistence is unsurprising. The financial market is undoubtedly linked to the real economy, and the sentiment towards one should also reflect a sentiment to the other. An interesting pattern emerges with the coefficients for the Baa and (to a lesser extent) Aaa corporate bond rates. The coefficients shrink to zero right before the subprime mortgage crisis and reemerge during the crisis, persistent thereafter. We can perhaps see this as an indicator of the impending crisis, since the disjoint of the real economy and risky assets can be considered as a signal of a bubble. Again, this demonstrates the effectiveness of DSS in recovering interesting signals that can be useful for policy makers, as we might want to prepare for a crisis if we observe similar patterns in the future.

We further our analysis by comparing the forecast performance between DSS estimated using the EM algorithm and MCMC (description of the MCMC algorithm can be found in Appendix A). Note that the MCMC version employs a Gaussian spike rather than a Laplace. This is done for computational simplicity, as a Gaussian spike and slab will ensure conjugate updating throughout. The mean squared cumulative forecast error are reported in Figure 8, including the results from the EM (mode) and MCMC (mean and median of the MC sampled forecast error; ran with 1,000 burn-in and 1,000 iterations). We first note that all three methods are practically equivalent, in terms of predictive performance—especially when considering the difference between DSS and other methods compared earlier—demonstrating that using either EM or MCMC will produce nearly equivalent results. In terms of computation speed, the EM is far superior, with EM clocking.
Figure 8: Mean squared cumulative forecast error from 2001/1 to 2015/12 for DSS estimated using the EM algorithm and MCMC. For MCMC, we report both the mean and median of the forecast error.

in at 4 minutes while MCMC taking 56 minutes on the entire dataset. Improving computation time in the order of $14 \times$ is vast, and allows for flexibility to change parameters and specifications without hesitation, as well as conducting sequential predictive analysis over a large dataset. On the other hand, MCMC benefits from added flexibility in including estimation of parameters and more complex components, such as stochastic volatility. Additionally, obtaining full posterior/predictive distributions allows for the assessment of uncertainty, leading to better informed decisions in some contexts.

Through this macroeconomic example, we were able to demonstrate the efficacy of DSS, in terms of forecast ability and interpretability, on a real, topical macroeconomic dataset. With DSS, we were able to significantly outperform competing methods and successfully recover interesting signals previously unattainable. Mirroring the results from the synthetic data, DSS is able to be effective under datasets with large number of predictors, where other methods might fail.

8 Discussion

This paper introduces a new class of dynamic shrinkage priors, where the stationary distribution is fully known and characterized by spike-and-slab marginals. A key to obtaining this stabilizing property is the careful hierarchical construction of adaptive mixing weights that allows them to depend on the lagged value of the process, thereby reflecting sparsity of past coefficients. We propose various versions of DSS, using Laplace/Gaussian spike/slab distributions. For implementation, we resort to both optimization as well as posterior sampling. We implement a one-step late EM algorithm for MAP estimation which iterates over one-site closed-form thresholding rules. Through simulation and a macroeconomic dataset, we demonstrate that DSS are well suited for
the dual purpose of dynamic variable selection (through thresholding to exact zero) and smoothing (through an autoregressive slab process) for forecasting and inferential goals.

Many variants and improvements are possible on the DSS prototype constructions. While these go beyond the scope of the present paper, we would like to make the reader aware of generalizations that might be important for future deployment of these priors. The first will be linking the time series priors across the different coordinates. This can be achieved by endowing the importance weight $\Theta$ with a prior distribution, thereby allowing to adapt to the overall sparsity pattern. The second very important extension will be treating the residual variances $\sigma^2$ as random and possibly time-varying. In this paper, we focused primarily on the priors on the regression coefficients. However, the DSS priors can be deployed in tandem with a stochastic process prior on $\{\sigma^2_t\}_{t=1}^T$ (as e.g. in Kalli and Griffin (2014)). We have noted one possible variant in our MCMC algorithm in the Appendix.

Third, the autoregressive parameters $(\phi_0, \phi_1)$ have been assumed fixed and shared across coordinates. It is a standard practice to assign a strongly informative prior for $\phi_1$ that sharply concentrates around one (Prado and West 2010, Chapter 5), which is not too far from setting $\phi_1$ equal to a constant. Lopes, McCulloch and Tsay (2016) consider a grid of possible values for $\phi_1$ through a discretized Gaussian distribution centered at one with a small variance. We could incorporate a similar discrete prior within our EM and MCMC schemes. For the EM algorithm, we would add a conditional maximization step in the M-step, updating the states, given $\phi_1$, and updating $\phi_1$, given the states. For the update of $\phi_1$, one would pick the point on the grid that maximizes the expected complete data log-likelihood, given $\beta_{ij}$’s. One might also consider a continuous prior on $\phi_1$, such as a Gaussian truncated on the unit (Prado and West 2010, Chapter 7) or a Beta distribution centered near one (Omori et al. 2007; Nakajima and West 2013a), and perform univariate optimization in the M-step, conditionally on $\beta_{ij}$. Regarding $\phi_0$, we set it equal to 0 by default in all our examples. However, one could apply, e.g., the approach of Lopes, McCulloch and Tsay (2016) and assign a conditionally conjugate prior. Regarding $\phi_1$ and $(\lambda_1, \lambda_0)$, we performed an extensive sensitivity analysis showing that the procedure is fairly robust as long as the parameters are in the right range.

A matlab code is available from the authors upon request.

References


Puelz, D., Hahn, P. R., Carvalho, C., 2016. Sparse mean-variance portfolios: A penalized utility approach. Submitted manuscript.


A Appendix: Summary of MCMC for Dynamic Spike-and-Slab process

A.1 Overview and Initialization

This appendix summarizes algorithmic details of the MCMC computations for dynamic spike-and-slab process priors. The prior specification of the DSS process outlined here departs slightly from our previous setup. In the paper, the spike is a Laplace, in order to achieve sparsity in posterior modes. Here, the spike is assumed to be Gaussian in order to enjoy simple conjugacy for posterior updating. We remark on how to extend the sampling routine for the Laplace spike at the end of this section. This leads to the spike density defined as

$$\psi_0(\beta_t \mid 0, \lambda_0) \equiv N(\beta_t \mid 0, \lambda_0).$$  \hspace{1cm} (A.35)

The MCMC example has a fixed observation variance, which can be easily modified to be learnt with, e.g., discount stochastic volatility (as remarked at the end of this section).

The MCMC algorithm consists of a block Gibbs sampler, where one block entails updating the time-varying slab locations, and another one the inclusion probabilities. The former involves the FFBS algorithm central to MCMC in all conditionally normal DLMs (Frühwirth-Schnatter 1994; West and Harrison 1997, Sect 15.2; Prado and West 2010, Sect 4.5). The MCMC is run iteratively as follows.

Initialization: To begin, we set the initial values for the inclusion indicators $$\gamma_{tj}$$ for $$0 \leq t \leq T$$. Anticipating sparsity, one could sample a binary matrix with a small success probability or use the EM output for initialization. Setting sensible $$\gamma_{tj}$$ can improve speed of convergence. We note that the initial vector at time zero $$\beta_0$$ is subject to estimation as well. As its prior, we use the stationary distribution (5.29) characterized by the spike-and-slab marginals, i.e.

$$\beta_0 \mid \gamma_0 \sim N(m_0, C_0),$$

where $$m_0$$ and $$C_0$$ are obtained from (5.29) with a Gaussian spike.
A.1.1 Per MCMC Iterate Step 1: Sampling DLM parameters

Conditional on the inclusion indicators $\gamma_{tj}$ for $t = 1:T$ and $j = 1:p$, we are in the setting of a conditionally normal DLM. The conjugate DLM form has

$$y_t = x_t'\beta_t + \varepsilon_t, \quad \varepsilon_t \sim N(0, v),$$
$$\beta_t = H_t + G_t(\beta_{t-1} - H_t) + e_t, \quad e_t \sim N(0, W_t),$$

with known elements $x_t = (x_{t1}, x_{t2}, ..., x_{tp})'$ and known variance $v = 1$, for $t \geq 1$, and

$$W_t = \text{diag}\left\{ \gamma_{tj}\lambda_1 + (1 - \gamma_{tj})\lambda_0 \right\}_{j=1}^p$$
$$G_t = \text{diag}\left\{ \gamma_{tj}\phi_1 \right\}_{j=1}^p$$
$$H_t = \text{diag}\left\{ \gamma_{tj}\phi_0 \right\}_{j=1}^p$$

The implied conditional posterior $\pi(\beta_{1:T}|x_{1:T}, y_{1:T}, \gamma_{1:T})$ is simulated from using the standard FFBS algorithm. In detail, this proceeds as follows.

**Forward filtering:** Starting from $t = 1$, we move to $t = T$, performing the filtering updates to compute and save the sequence of sufficient statistics for the on-line posteriors $\pi(\beta_t|y_{1:t}, x_{1:t}, \gamma_t)$ at each $t$.

1. **Time $t-1$ posterior:**
$$\beta_{t-1}|x_{1:t-1}, y_{1:t-1}, \gamma_{1:t-1} \sim N(m_{t-1}, C_{t-1}),$$

with a mean vector $m_{t-1}$.

2. **Update to time $t$ prior:**
$$\beta_t|x_{1:t-1}, y_{1:t-1}, \gamma_t \sim N(a_t, R_t),$$

with $a_t = H_t + G_t(m_{t-1} - H_t)$ and $R_t = W_t$.

3. **1-step predictive distribution:** $y_t|x_{1:t}, y_{1:t-1} \sim N(f_t, q_t)$ where
$$f_t = x_t'a_t \quad \text{and} \quad q_t = x_t'R_t x_t + v.$$

4. **Filtering update to time $t$ posterior:**
$$\beta_t|x_{1:t}, y_{1:t}, \gamma_{1:t} \sim N(m_t, C_t),$$

with defining parameters $m_t = a_t + A_te_t$ and $C_t = R_t$, based on $e_t = y_t - f_t$ and $A_t = R_te_t/q_t$.

**Backward sampling:** Having run the forward filtering analysis up to time $T$, the backward sampling proceeds as follows.

a. **At time $T$:** Simulate $\beta_T$ from the normal posterior: $N(m_T, C_T)$. 

2
b. **Recursively sample backwards in time** $t = T - 1, T - 2, \ldots, 1$ : At time $t \geq 1$ sample $\beta_t$ from the conditional normal posterior $\pi(\beta_t | \beta_{t+1}, x_{1:T}, y_{1:T}, \gamma_{1:T})$ with mean vector $m_t + B_t(\beta_{t+1} - m_t)$ and variance matrix $C_t$, where $B_t = C_t G_t R_{t+1}^{-1}$.

c. **Update prior $\beta_0$** : Given the sample of $\beta_t$ from step b., compute the conditional normal posterior $\pi(\beta_0 | \beta_1, x_{1:T}, y_{1:T}, \gamma_{1:T})$ with mean vector $m_0 + B_0(\beta_1 - m_0)$ and variance matrix $C_0$. Here, $m_{0j} = \gamma_{0j} \phi_0$ and $C_0 = \text{diag}(\gamma_0 \lambda_1 + (1 - \gamma_0) \lambda_0)$.

### A.1.2 Per MCMC Iterate Step 2: Sampling the inclusion indicator $\gamma_{0:T, 1:p}$

Conditionally on most recently sampled values of the DLM parameters $\beta_{0:T, 1:p}$, the MCMC iteration completes with sampling the inclusion indicators $\gamma_{0:T, 1:p}$ from their full conditional posterior $\pi(\gamma_{0:T, 1:p} | \beta_{0:T, 1:p})$. The sampling of each entry $\gamma_{tj}$ is independent of the other entries, making it very easy to distribute the calculations, if needed. For each $1 \leq t \leq T$ and $1 \leq j \leq p$ we have:

1. **Compute inclusion probability $\theta_{tj}$** : We first compute the p.d.f. for both the spike and stationary slab distributions, with,

   \[\psi_0(\beta_{t-1,j} | 0, \lambda_0) \equiv N(\beta_{t-1,j} | 0, \lambda_0),\]
   \[\psi_1^{ST}(\beta_{t-1,j} | \lambda_1, \phi_0, \phi_1) \equiv N\left(\beta_{t-1,j} | \phi_0, \frac{\lambda_1}{1 - \phi_1^2}\right).\]

   From this, and given $\Theta$, we can compute the inclusion probability $\theta_{tj}$ for each $j = 1:p$ as,

   \[\theta_{tj} \equiv \theta(\beta_{t-1,j}) = \frac{\Theta \psi_1^{ST}(\beta_{t-1,j} | \lambda_1, \phi_0, \phi_1)}{\Theta \psi_1^{ST}(\beta_{t-1,j} | \lambda_1, \phi_0, \phi_1) + (1 - \Theta) \psi_0(\beta_{t-1,j} | 0, \lambda_0)}. \tag{A.36}\]

2. **Update $\mu_{tj}$ and $p_{tj}^*(\beta_{tj})$** : Update $\mu_{tj}$,

   \[\mu_{tj} = \phi_0 + \phi_1(\beta_{t-1,j} - \phi_0),\]

   then compute the spike and slab p.d.f. as

   \[\psi_0(\beta_{tj} | 0, \lambda_0) \equiv N(\beta_{tj} | 0, \lambda_0),\]
   \[\psi_1(\beta_{tj} | \lambda_1, \phi_0, \phi_1) \equiv N\left(\beta_{tj} | \phi_0, \lambda_1\right).\]

   From this, we can compute $p_{tj}^*(\beta_{tj})$ for each $j = 1:p$ as

   \[p_{tj}^*(\beta_{tj}) \equiv \frac{\theta_{tj} \psi_1(\beta_{tj} | \mu_{tj}, \lambda_1)}{\theta_{tj} \psi_1(\beta_{tj} | \mu_{tj}, \lambda_1) + (1 - \theta_{tj}) \psi_0(\beta_{tj} | \lambda_0)}.\]

3. **Compute and sample from posterior $\gamma_{tj}$ for each $j = 1:p$** : Given $p_{tj}^*(\beta_{tj})$, we can sample from the posterior of $\gamma_{tj}$ from

   \[\gamma_{tj} \sim Bin(p_{tj}^*(\beta_{tj})).\]

For $t = 0$ and given $\beta_{0j}$, we sample $\gamma_{0j}$ from $\gamma_{0j} \sim Bin(\theta_{0j})$, where $\theta_{1j}$ was defined in (A.36) as

\[\theta_{1j} \equiv \theta(\beta_{0,j}) = \frac{\Theta \psi_1^{ST}(\beta_{0,j} | \lambda_1, \phi_0, \phi_1)}{\Theta \psi_1^{ST}(\beta_{0,j} | \lambda_1, \phi_0, \phi_1) + (1 - \Theta) \psi_0(\beta_{0,j} | 0, \lambda_0)}.\]
A.2 Additional comments

A.2.1 Other spike/slab densities

We note that the above framework can be easily modified to include other densities, in particular when that density is a mixture of normal(s). For example, as with the paper, in the case of a Laplace spike, the following modification to the MCMC

$$\psi_0(\beta_t | 0, \lambda_0^*) \equiv N(\beta_t | 0, \lambda_0^*),$$

$$\lambda_0^* \sim \text{Exp} \left( \frac{\lambda_0}{2} \right)$$

will lead to the Laplace/Gaussian spike/slab, by first sampling from the exponential and using that sample as the variance of the spike distribution.

A.2.2 Learning the observation variance via stochastic volatility

Learning the observation variance can be done relatively easily using discount stochastic volatility (West and Harrison 1997; Prado and West 2010). With the following observation equation,

$$y_t = x_t' \beta_t + \varepsilon_t, \quad \varepsilon_t \sim N(0, v_t),$$

and denoting $s_t$ as the point estimate of $v_t$ and $\delta$ as the discount (or "forgetting") factor, we add the following additional steps. In step 4. of the forward filtering step, we recursively update the point estimate of the variance by

$$s_t = r_t s_{t-1},$$

where $r_t = (\delta n_{t-1} + e_t^2 / q_t) / n_t$ and $n_t$ is the degree of freedom. Then, in the backwards step, prior to sampling $\beta_{T:1}$, draw $v_T^{-1}$ from $G(n_T / 2, n_T s_T / 2)$, and recursively draw $v_{T-1:1}$ via $v_t^{-1} = \delta v_{t+1}^{-1} + \eta_t$ where $\eta_t$ is an independent draw from $\eta_t \sim G((1-\delta)n_t/2, n_t s_t/2)$. Note that this leads to a predictive distribution $y_t | x_{1:t}, y_{1:t-1} \sim T_{3n_{t-1}}(f_t, q_t)$. 

4