A mia madre Sabrina e mio padre Giuliano

Contents

List of Algorithms

Introduction

Central banks, statistical institutes, intergovernmental organizations, financial markets and online sources produce large amounts of data everyday. Such immense availability of data has offered many opportunities for macroeconometric modelling, but it has also posed new challenges. The passage from classical econometrics to big data econometrics has been characterized by two phenomena: the inclusion of variable selection strategies inside models and a preference for Bayesian statistics. An emblematic attempt in this direction is the model developed by Scott and Varian (2013) which merges Bayesian Structural Time Series (BSTS) with Spike-and-Slab regression. The authors, economists at Google Inc, realized immediately the importance of big data in economic time series analysis and forecasting and they built this flexible model able to accommodate structural time series components, such as trend and seasonality, together with a large set of predictors. However, their model presents two limitations: static regression coefficients and constant variance. Such assumptions are unduly restrictive in time series analysis since they do not allow to fully explore the dynamics of the system under investigation. A wide literature indeed points out that the relationships among economic and financial variables are likely to change over time and therefore they are better described using regression models with time-varying parameters. Nevertheless, while models with stochastic variation in the parameters can be even more prone to overfitting, classical variable selection strategies are designed to induce shrinkage and sparsity in a static framework and not a dynamic one. For this reason, dynamic shrinkage has become an hot-topic in time series analysis and an increasing number of literature is addressing this issue by developing new strategies which are able not only to identify which predictors are truly

relevant for the matter in question, but also in which moment in time. Pioneers in this new literature are Rockova and McAlinn (2021a), who developed the Dynamic Spike-and-Slab Process Priors that captured our attention for both flexibility and performances.

In this thesis we present this approach with two major extensions: we improve estimates with a stochastic volatility model for the residual variance and we replace the Forward Filtering Backward Sampling (FFBS) strategy with the precision sampler of Chan and Jeliazkov (2009) to boost the computational efficiency. Moreover, we propose a novel Bayesian Structural Time Series model with dynamic shrinkage and stochastic volatility that can be regarded as the natural evolution of BSTS model aforementioned. The proposed model is designed to perform time series analysis and forecasting especially in the field of macroeconomics, however it may find applications also in other research areas. The structural components along with time-varying regression coefficients and the stochastic volatility process allow this model to capture several features which are common in macroeconomic time series such as trend, seasonality, structural breaks and change points. This new class of models will be discussed in details in Chapter 3, after a preliminary review of Bayesian variable selection methods and BSTS (Chapter 1) and a comprehensive analysis of Dynamic Spike-and-Slab Process Priors (Chapter 2).

Then, driven by the enthusiasm of the results obtained with dynamic shrinkage priors, we extended the dynamic variable selection approach to the field of multivariate time series and, in particular, in Vector Autoregressive (VAR) models. Indeed, multivariate time series models suffer even more harshly of the the curse of dimensionality. For this reason, attempts have been made to introduce variable selection strategies in these models. Nevertheless, inducing dynamic shrinkage in Time-Varying VAR models is still a open problem because of the intrinsic complexity of the task. In Chapter 4 we provide a possible way to address this problem by exploring the potentiality of Dynamic Spike-and-Slab Process Priors in Time-Varying VAR models.

Every algorithm described in this thesis has been implement in R. The replication code is publicly available on Github at the personal page of the author [1]. The code is organized as

[1] url: https://github.com/edoardo-marcelli

a preliminary version of an R-package (to be possibly submitted to CRAN). All the details about the latter are provided in Chapter 5 and in Appendix.

0.1 Notation

Before starting with the heart of this thesis, let me introduce the basic notation that will be employed henceforth. A time series will be denoted as $(Y_t)_{t\geq 1}$ or simply (Y_t) where the subscript t denotes the time of observation; capital letter are used for denoting random variables, while lower case letters denote the realizations. Bold upper case letters or upper case Greek letters identify matrices, while bold lower case letters identify vectors. Note, however, that finite sequences will be also denoted with the colon notation, e.g. $y_{1:T} = (y_1, ..., y_T)$. This notation allows indeed to underline the temporal nature of some objects, which implies that the elements have to be considered in a precise chronological order. In the same manner, when dealing with time-varying matrices, the subscript will be explicated, for example $\boldsymbol{y}_{1:T} = (\boldsymbol{y}_1, ..., \boldsymbol{y}_T)'$ indicates a $T \times n$ matrix. However, for some very specific cases, the notation might be simplified, for instance we use $\boldsymbol{X}$ to define a $T \times p$ matrix of explanatory variables in regression models:

$$\boldsymbol{X} = \begin{pmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,p} \\ x_{2,1} & x_{2,2} & \dots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{T,1} & x_{T,2} & \dots & x_{T,p} \end{pmatrix}$$

Finally, bold upper case Greek letters indicates large matrices, such as block diagonal matrices.

Objects that do not respect this notation will be accurately introduced in order to make the reading as smoothly as possible.

Chapter 1

Bayesian Variable Selection Methods in Time Series Regression Models

"frustra fit per plura quod potest fieri per pauciora"

— Novacula Occami

The emergence of big data in the contemporary era has meant the development of new tools and methodologies to handle and menage efficiently such great sources of information. The aim of variable selection is to retain inside the model only those predictors that significantly contribute at explaining a given phenomenon to a certain degree, removing unwanted noise variables that would bias estimation or lead to overfitting in prediction. Therefore, by selecting only the relevant features we expect not only an higher model's explanatory power but also better out-of-sample predictions. However, such methodologies are not intended to be a substitute to experts' knowledge and they are able to express their potentiality only if combined with a robust theory.

In this chapter, we decided to focus on the problem of variable selection in linear regression models adopting a Bayesian perspective. This issue however was raised for the first time in the frequentist literature because of an intrinsic weakness of traditional (frequentist) estimators.

Consider the classical linear model

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I}_n).$$

The least squared estimator, $\boldsymbol{b} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}$, requires to invert the matrix $\boldsymbol{X}'\boldsymbol{X}$, which becomes unattainable when the number of predictors exceed the number of available observations. To overcome this fragility, in the frequentist framework, some techniques have been developed to control for high dimensionality. Leaving aside stepwise procedures that are intrinsically weak and clearly unfeasible when the number of predictors is large, these techniques are mainly based on the penalization of the likelihood and the resolution of optimization problems. However, model selection techniques merely based on likelihood penalization (AIC, BIC etc.) require to compare 2^p models and they do not provide coefficient's estimates, whereas frequentist shrinking estimators such as the Least Absolute Shrinkage and Selection Operator (LASSO) provides shrunk coefficients' estimates, however the optimization step on which they rely might not be that trivial.

On the other hand, the Bayesian approach offers appealing methods for variable selection which are straightforward in principle. In principle, the researcher formalizes her information on each model by assigning them a probability law and then a prior distribution on the model-specific parameters; both the information on the parameters and on the models are updated through the Bayes rule as information from the data becomes available. This process is however not without difficulties; the main difficulties arise when it comes to: (i) specify prior distributions for the models' parameters, (ii) specify prior distributions on the models' space and (iii) compute the posterior distributions. For these purposes, classes of informative and uninformative priors provide a wide range of potential solutions for (i) and (ii), whereas recent developments in computational statistics offer powerful tools for (iii). In fact, Bayesian variable selection techniques rely on stochastic simulations rather than optimization problems. We should remark that the Bayesian approach to variable selection is also regarded as a generalization of the frequentist approach rather than an alternative to the latter. Indeed, the prior distribution provides a penalization of the likelihood, which makes Bayesian methods relevant also from a frequentist standpoint. Some of these methods

will be reviewed in Section 1.2 of this chapter, after a brief introduction to Bayesian inference in linear regression models (Section 1.1). Then, all the considerations made for the standard linear regression model will be translated in the framework of State-State Models (SSM).
SSM are a class of stochastic models that describe the probabilistic dependence between a latent state process and the observable measurements. Originated in the early sixties in the field of control engineering, they have gained popularity in time series analysis thanks to their flexibility. They allow to model both univariate and multivariate time series, also in presence of non-stationarity, structural breaks and irregular patterns.
Let $(\boldsymbol{Y}_t)_{t\geq 1}$ and $(\boldsymbol{\theta}_t)_{t\geq 0}$ be discrete time processes, where $\boldsymbol{Y}_t$ and $\boldsymbol{\theta}_t$ are respectively $n \times 1$ and $p \times 1$ vectors, with $\boldsymbol{Y}_t$ and $\boldsymbol{\theta}_t$ taking values respectively in spaces $\mathcal{Y}$ and Θ (these spaces can be multi-dimensional Euclidean spaces, discrete spaces or also less standard), and let $\boldsymbol{\psi}$ be a vector of unknown model's parameters, characterized by a prior distribution $\pi(\boldsymbol{\psi})$. Then the process $\big((\boldsymbol{Y}_t, \boldsymbol{\theta}_t)\big)_{t\geq 1}$ starting at $\boldsymbol{\theta}_0 \sim \pi_0(\cdot)$ is a SSM if

(A.1) $(\boldsymbol{\theta}_t)|\boldsymbol{\psi}$ is a Markov Chain

(A.2) Conditionally on $(\boldsymbol{\theta}_t)$ and $\boldsymbol{\psi}$, the $\boldsymbol{Y}_t$'s are independent and $\boldsymbol{Y}_t$ only depends on $\boldsymbol{\theta}_t$ and $\boldsymbol{\psi}$

Therefore, for any t, the joint distribution of $(\boldsymbol{\theta}_{0:t}, \boldsymbol{y}_{1:t}, \boldsymbol{\psi})$ is given by

$$\pi(\boldsymbol{\theta}_{0:t}, \boldsymbol{y}_{1:t}, \boldsymbol{\psi}) = \pi_0(\boldsymbol{\theta}_0|\boldsymbol{\psi})\pi(\boldsymbol{\psi}) \prod_{j=1}^{t} \pi_j(\boldsymbol{y}_j|\boldsymbol{\theta}_j, \boldsymbol{\psi})\pi_j(\boldsymbol{\theta}_j|\boldsymbol{\theta}_{j-1}, \boldsymbol{\psi})$$

and it is thus fully specified by the initial distribution $\pi_0(\boldsymbol{\theta}_0 \mid \boldsymbol{\psi})$, the parameters' distribution $\pi(\boldsymbol{\psi})$, the transition distribution $\pi_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \boldsymbol{\psi})$ describing the evolution of the latent process, and the emission distribution $\pi_t(\boldsymbol{y}_t|\boldsymbol{\theta}_t, \boldsymbol{\psi})$ illustrating how data are generated.
A linear regression model can be thus regarded as a special case of SSM where the latent process is assumed to be static and the observations $\boldsymbol{y}_t$ are linearly related to the state $\boldsymbol{\theta}_t$. An in-depth discussion on the ways in which regressors are handled in SSM follows in Section 1.3, where a recent class of models combining SSM and Bayesian variable selection methods is introduced.

1.1 Bayesian Inference in Linear Regression

Regression models are meant to describe the relationship between a random variable Y and a set of explanatory variables $\boldsymbol{x} = (x_1, ..., x_p)$. The classic linear regression model is

$$Y'_{1:T} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \mid \Sigma \sim \mathcal{N}(\boldsymbol{0}, \Sigma) \tag{1.1}$$

This formulation provides a full specification of the joint probability of the vector $Y_{1:T} = (Y_1, ..., Y_T)$ given the matrix regressors $\boldsymbol{X} = (\boldsymbol{x}'_1, ..., \boldsymbol{x}'_T)'$ along with the model's parameters $\boldsymbol{\beta}$ and Σ. The covariance matrix Σ can be specified using any symmetric positive-definite matrix, however considering $\Sigma = \sigma^2 \boldsymbol{I}_T$ preserves the i.i.d errors property. In a Bayesian perspective, the unknown parameters are random quantities and model (1.1) expresses the conditional distribution $Y'_{1:T} \mid (\boldsymbol{\beta}, \Sigma) \sim \mathcal{N}(\boldsymbol{X}\boldsymbol{\beta}, \Sigma)$. The model is thus completed by assigning a prior distribution on $(\boldsymbol{\beta}, \Sigma)$, and inference is solved by computing their posterior distribution given the data. However, the posterior distribution can be analytically involved; and a useful way to proceed is by using conjugate priors. Considering both $(\boldsymbol{\beta}, \Sigma)$ unknown, the conjugate model is a Normal-Gamma with parameters $(\beta_0, \Omega_0, n_0, d_0)$, which means assuming

$$\begin{aligned} \boldsymbol{\beta} \mid \sigma^2 &\sim \mathcal{N}(\boldsymbol{\beta}_0, \sigma^2 \Lambda_0) \\ \frac{1}{\sigma^2} &\sim \mathcal{G}(n_0, d_0) \end{aligned}$$

where $\mathcal{G}(\alpha, \beta)$ is a Gamma distribution with mean α/β and variance α/β^2. Therefore, the posterior of $(\boldsymbol{\beta}, \Sigma)$ given $y_{1:T}$ is still a Normal-Gamma with updated parameters $(\tilde{\boldsymbol{\beta}}, \Lambda, n_T, d_T)$ where

$$\begin{aligned} \tilde{\boldsymbol{\beta}} &= \boldsymbol{\beta}_0 + \Lambda_0 \boldsymbol{X}'(\boldsymbol{X}\Lambda_0\boldsymbol{X}' + \boldsymbol{I})^{-1}(y'_{1:T} - \boldsymbol{X}\boldsymbol{\beta}_0), \\ \Lambda &= \Lambda_0 - \Lambda_0 \boldsymbol{X}'(\boldsymbol{X}\Lambda_0\boldsymbol{X}' + \boldsymbol{I})^{-1}\boldsymbol{X}\Lambda_0, \\ n_T &= n_0 + \frac{T}{2}, \\ d_T &= d_0 + \frac{1}{2}(\boldsymbol{\beta}_0\Lambda_0^{-1}\boldsymbol{\beta}_0 + y_{1:T}y'_{1:T} - \tilde{\boldsymbol{\beta}}'\Lambda\tilde{\boldsymbol{\beta}}) \end{aligned}$$

Another useful result that is worth to be mentioned since it will be resumed multiple time in this thesis concerns the updating rule of a non-conjugate model provided by theorem (8.1)

of Kroese and Chan (2014). They consider the following prior densities

$$\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{\beta}_0, \Lambda_0)$$
$$\frac{1}{\sigma^2} \sim \mathcal{G}(n_0, d_0)$$

and

$$\boldsymbol{\beta}|\sigma^2, y_{1:T} \sim \mathcal{N}(\tilde{\boldsymbol{\beta}}, \Lambda)$$
$$\frac{1}{\sigma^2}|\boldsymbol{\beta}, y_{1:T} \sim \mathcal{G}\left(n_0 + \frac{T}{2}, d_0 + \frac{(y'_{1:T} - \boldsymbol{X}\boldsymbol{\beta})'(y'_{1:T} - \boldsymbol{X}\boldsymbol{\beta})}{2}\right)$$

where $\tilde{\boldsymbol{\beta}} = \Lambda(\boldsymbol{X}' y'_{1:T}/\sigma^2 + \Lambda_0^{-1}\boldsymbol{\beta_0})$ and $\Lambda = (\boldsymbol{X}'\boldsymbol{X}/\sigma^2 + \Lambda_0^{-1})^{-1}$. This result is important since it provides the foundation of the precision sampler of Chan and Jeliazkov (2009). The latter is a fast posterior sampling strategy for Dynamic Linear Models that can be used in alternative to the FFBS. The precision sampler is illustrated in details in Section 2.2.2.

So far, all the regression coefficients were treated as influential, however one could reasonably questions the relevance of some predictors. This belief can be taken into account in the Bayesian approach through a different specification of the density priors as illustrated in the next paragraphs.

1.2 Hierarchical Models for Variable Selection

Bayesian variable selection methods are usually built on hierarchical mixture priors. With reference to the linear regression framework described in the previous section, this translates in introducing auxiliary binary variables γ_j taking values $\gamma_j = 1$ if the j-th coefficient is significantly different from zero and $\gamma_j = 0$ otherwise. Therefore, the vector $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_p)'$ indicates which covariates enter the regression model. For example, $\boldsymbol{\gamma} = (0, 1, 0, ..., 0)'$ identifies a model including only x_2. We denote with $\boldsymbol{X}_\gamma \boldsymbol{\beta}_\gamma$ the model associated with a generic $\boldsymbol{\gamma}$. The specification of such hierarchical model is completed by setting the priors $\pi(\boldsymbol{\gamma})$, $\pi(\boldsymbol{\beta}_\gamma|\boldsymbol{\gamma})$ and by specifying a likelihood $\pi(y_{1:T}|\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma})$. Every strategy here presented starts from this setup and considers different prior specifications and posterior inference strategies. For brevity, we decide to focus on only one technique for each class of methods.

For a more complete overview, we refer to O'Hara and Sillanpää (2009), Rockova (2013) and Fan and Sisson (2010).

1.2.1 Spike-and-Slab Priors

A popular approach in Bayesian analysis to generate sparsity consists in using mixture density priors of the class *Spike-and-Slab*. The name attached to these priors derives from their peculiarity of combining a diffuse density for $\beta_j|\gamma_j = 1$ and a concentrated distribution with mean zero on $\beta_j|\gamma_j = 0$. In the original formulation of Mitchell and Beauchamp (1988), the idea was to assign to every coefficients a mixture prior with a Dirac measure on zero and a uniform density elsewhere; while in more recent versions the point mass prior is replaced by a continuous distribution concentrated on zero since it facilitates computations. The Stochastic Search Variable Selection (SSVS) by George and McCulloch (1993) belong to this latter class. This strategy will be relevant in the developments of the thesis, therefore we analyse it in details. The approach uses a scale mixture of two normal distributions for the model's coefficients. Globally, the hierarchical setup is the following

$$
\begin{aligned}
Y_t|\boldsymbol{x}_t, \boldsymbol{\beta}, \sigma^2 &\overset{ind}{\sim} \mathcal{N}(\boldsymbol{x}_t'\boldsymbol{\beta}, \sigma^2), \quad t = 1, ..., T, \\
\beta_j|\gamma_j &\sim (1-\gamma_j)\mathcal{N}(0, \lambda_j) + \gamma_j \mathcal{N}(0, c_j^2\lambda_j), \quad j = 1, ..., p, \qquad (1.2) \\
\sigma^2|\boldsymbol{\gamma} &\sim \mathcal{IG}\left(\frac{n_\gamma}{2}, \frac{d_\gamma}{2}\right), \qquad (1.3) \\
P(\gamma_j = 1) &= 1 - P(\gamma_j = 0) = \omega_j \qquad (1.4)
\end{aligned}
$$

where $\mathcal{IG}(a, b)$ is an Inverse-Gamma distribution with mean $\frac{b}{a-1}$ and variance $\frac{b^2}{(a-1)^2(a-2)}$. Thus, by setting a (positive) small value for λ_j when $\gamma_j = 0$ then β_j would be shrunk to zero. On the other hand, fixing c_j large (and at least greater than 1) when $\gamma_j = 1$ then the prior gives high probability to β_j being substantially far from zero. According to this interpretation, ω_j can be regarded as the prior probability of x_j being included in the model.

Equation (1.2) gives the conditional distribution of β_j; the joint conditional distribution of

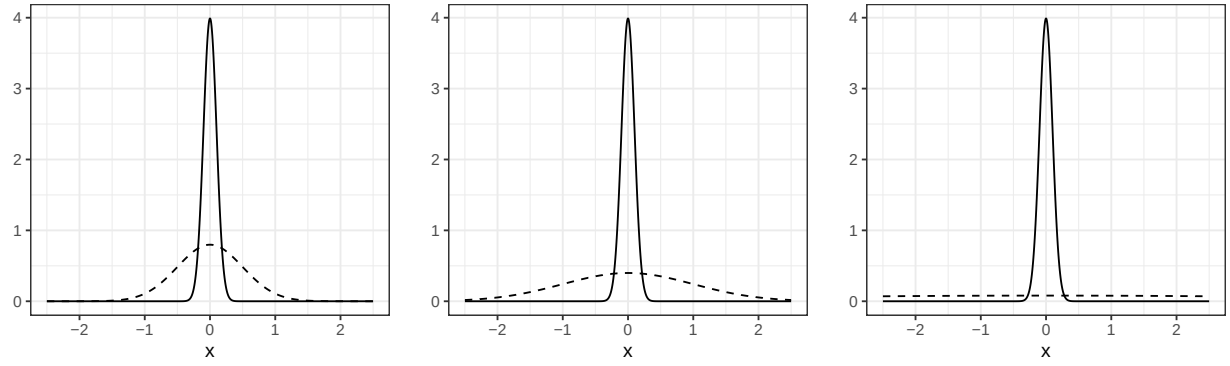


Figure 1.1: Spike (solid line) and Slab (dashed line) distributions with hyperparameters: $\lambda = 0.1$ and $c = 5, 10, 50$.

β_j can be rewritten in a general multivariate form as

$$\boldsymbol{\beta}|\boldsymbol{\gamma} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{D}_\gamma \boldsymbol{R}_\gamma, \boldsymbol{D}_\gamma)$$

where $\boldsymbol{D}_\gamma \equiv diag(\alpha_1\lambda_1, ..., \alpha_p\lambda_p)$ with $\alpha_j = 1$ if $\gamma_j = 0$ and $\alpha_j = c_j$ if $\gamma_j = 1$, and $\boldsymbol{R}_\gamma$ is prior correlation matrix when the model is characterized by gamma. A convenient specification for the latter is $\boldsymbol{R}_\gamma = \boldsymbol{I}$ if the β_j can be considered independent or the Zellner (1986) *g-prior* that assumes $\boldsymbol{R}_\gamma^{-1} = g(\boldsymbol{X'X})/n$. Regarding the priors on the residual variance σ^2 and on $\boldsymbol{\gamma}$, it is reasonable to assume that σ^2 would decrease when the number of active coefficient increases. Thus, the parameters of the Inverse-Gamma distribution in equation (1.3) can be set in function of the expected dimensionality. For instance, interpreting n_γ as the prior sample size and $d_\gamma/(n_\gamma - 1)$ as the prior estimate of σ^2. One may let $d_\gamma/(n_\gamma - 1)$ be a decreasing function of $|\boldsymbol{\gamma}|$. Finally, George and McCulloch (1993) recognize the difficulty in choosing an informative prior for $\boldsymbol{\gamma}$ especially when p is large, thus they suggest the following specification which assumes independent indicators marginally distributed as in equation (1.4),

$$\pi(\boldsymbol{\gamma}) = \prod_{j=1}^{p} \omega_j^{\gamma_j} (1 - \omega_j)^{1-\gamma_j}$$

This prior basically implies that the inclusion probability of each regressor is independent of the inclusion of the others. Despite its simplicity, this prior choice is very effective since it facilitates Gibbs sampling; George and McCulloch (1993) found it to work well in various situations. Note also that, considering an equal probability of each regressor to enter the

model and thus setting $\omega_j = \frac{1}{2}$, the prior becomes a uniform $\pi(\boldsymbol{\gamma}) \equiv 2^{-p}$.

Obviously, the priors must be chosen with care, and they must strike a balance between reliability and functionality. The prior distributions in SSVS are precisely configured to produce closed form full conditional distributions, allowing for rapid and efficient simulations using Gibbs sampling, for example.

Alternatively to equation (1.2.1), George and McCulloch (1997) propose a conjugate design for $(\boldsymbol{\beta}, \sigma^2)$, i.e. $\boldsymbol{\beta}|\sigma^2, \boldsymbol{\gamma} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{D}_\gamma \boldsymbol{R}_\gamma, \boldsymbol{D}_\gamma)$. This formulation simplifies analytical calculations since $(\boldsymbol{\beta}, \sigma^2)$ can be integrated out from the full posterior $\pi(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\gamma}|y_{1:T})$ and thus it enables to easily obtain the posterior $\pi(\boldsymbol{\gamma}|y_{1:T})$, which is known up to a proportionality constant, and it can be carried analytically for small p or numerically otherwise with MCMC methods. George and McCulloch (1997) show that this conjugate design facilitates MCMC exploration.

1.2.2 Bayesian Regularization

Rather than using auxiliary variables to generate spike and slab densities, shrinking towards zero can alternatively be achieved by directly assigning a continuous prior density to the model's parameters. Such priors are usually exponential density priors of the form $\pi(\beta_j|\lambda_j) \propto \exp(-\lambda_j|\beta_j|^i)$, with $i > 0$ and where λ_j is a function of the scale parameter. For example, Tibshirani (1996) uses a double-exponential (Laplace) prior with $i = 1$ and assumes that the regression coefficients are i.i.d. distributed according to

$$\pi(\beta_j|\lambda) = \frac{\lambda}{2} \exp(-\lambda|\beta_j|)$$

for $j = 1, ..., p$. Assuming a Laplace rather than a Normal distribution has the effect of increasing the probability mass around zero, resulting in a more severe shrinking towards zero. Tibshirani (1996) reconciles this Bayesian approach to variable selection with the frequentist LASSO by showing that the latter can be regarded as the mode of the posterior distribution of β. However, while in the frequentist LASSO the regression coefficients are obtained by solving a constrained optimization problem, in the Bayesian version we rely on

simulation algorithms, primarily MCMC, approximating the posterior distribution which is usually analytically intractable. This implies that contrary to the frequentist LASSO that produces sparse models by forcing coefficients to zero, in the Bayesian LASSO the point estimates are never exactly equal to zero.

The approach of Tibshirani (1996) was interestingly developed into the Bayesian LASSO by Park and Casella (2008). In this case the regression coefficients are modeled with a conditional Laplace prior distribution of the type

$$\pi(\boldsymbol{\beta}|\sigma^2) = \prod_{j=1}^{p} \frac{\lambda}{2\sqrt{\sigma^2}} \exp\left(- \frac{\lambda|\beta_j|}{\sqrt{\sigma^2}} \right)$$

with an improper density $\pi(\sigma^2) \propto \frac{1}{\sigma^2}$ on the variance or an Inverse-Gamma prior that maintains the conjugacy. The effect of conditioning on σ^2 has significant consequences in terms of convergence of the MCMC strategy and more meaningful point estimates since it guarantees an unimodal full posterior density. Interestingly (Andrews and Mallows 1974), a scale mixture of Normal densities with an exponential mixing density can be used to represent the Laplace distribution:

$$\frac{\lambda}{2} \exp(-\lambda|\beta|) = \int_0^{\infty} \frac{1}{\sqrt{2\pi s}} \exp\left(- \frac{\beta^2}{2s} \right) \frac{\lambda}{2} \exp\left(- \frac{\lambda^2 \beta}{2} \right) ds$$

Therefore, the full model can be represented with this hierarchical scheme

$$\begin{aligned}
\boldsymbol{\beta}|\sigma^2, \tau_1^2, ..., \tau_p^2 \sim & \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{D}_\tau), \\
\boldsymbol{D}_\tau = & diag(\tau_1^2, ..., \tau_p^2), \\
\sigma^2, \tau_1^2, ..., \tau_p^2 \sim & \pi(\sigma^2) \prod_{j=1}^{p} \frac{\lambda^2}{2} \exp\left(- \lambda^2 \tau_j^2 / 2 \right), \\
\sigma^2, \tau_1^2, ..., \tau_p^2 > & 0
\end{aligned}$$

which can be simplified by assuming $\tau_1^2, ..., \tau_p^2 = \tau^2$. The Bayesian LASSO parameter, λ, can be estimated in several ways; Park and Casella (2008) suggest empirical Bayes via marginal maximum likelihood or the use of hyperpriors. We refer to the original article for further details.

1.2.3 Reversible Jump MCMC

The approaches mentioned so far presume that the collection of predictors is fixed and that individual coefficients are given shrinkage priors. Alternatively, one can wonder if the number of explanatory variables in the model is appropriate, or if it is too high or too low. The Bayesian way to model such uncertainty is straightforward and it consists of treating also N_p as a random variable and assigning a prior to it. In this way the degree of sparseness depends on two factors: the coefficients shrunk towards zero and the dimension of the regression matrix. The computational challenges of this approach can be addressed by using reversible jump MCMC algorithms (Green (1995)) which allow to simultaneously explore spaces of different dimensions. The idea is to generate a Markov Chain in order to simulate from $\pi(\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma}|y_{1:T}, \boldsymbol{X}) \propto \pi(y_{1:T}|\boldsymbol{X}, \boldsymbol{\beta})\pi(\boldsymbol{\beta}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma})$. This is commonly achieved through a Metropolis-Hastings algorithm. The difficulty stands in the fact that in order to explore different spaces we need a strategy that allow the transition from state $(\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma})$ to the state $(\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^*)$ where $N_p \neq N_{p^*}$. Such "jump" from one dimension to an other is possible through the concept of *dimension matching*. The idea is the following. Let's say for instance that the jump occurs from $(\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma})$ to $(\boldsymbol{\beta}^*_{\gamma^*}, \boldsymbol{\gamma}^*)$ where $N_p < N_{p*}$. To make this jump possible a random vector $\boldsymbol{u}$ with density $h_{\gamma\to\gamma^*}(\boldsymbol{u})$ and length $(N_{p^*} - N_p)$ is generated from a known density. The aim of this vector is to fill the dimensional gap between $\boldsymbol{\gamma}$ and $\boldsymbol{\gamma}^*$. In other words, considering a one-to-one mapping function $g_{\gamma\to\gamma^*} : \mathbb{R}^{N_p} \times \mathbb{R}^{N_{p^*}-N_p} \to \mathbb{R}^{N_{p^*}}$, the relationship between the current and the new state is given by $\boldsymbol{\beta}^*_{\gamma^*} = g_{\gamma\to\gamma^*}(\boldsymbol{\beta}_\gamma, \boldsymbol{u})$. Once we generate a random candidate state $(\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^*)$, its acceptance probability must take into account also the change in dimensions, thus it is necessary to compute the Jacobian of g:

$$\alpha((\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma}), (\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^*)) = \min\left\{1, \frac{\pi(\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^*|y_{1:T}, \boldsymbol{X})q(\boldsymbol{\gamma}^* \to \boldsymbol{\gamma})}{\pi(\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma}|y_{1:T}, \boldsymbol{X})q(\boldsymbol{\gamma} \to \boldsymbol{\gamma}^*)h_{\gamma\to\gamma^*}(\boldsymbol{u})}\left|\frac{\partial g_{\gamma\to\gamma^*}(\boldsymbol{\beta}_\gamma, \boldsymbol{u})}{\partial(\boldsymbol{\beta}_\gamma, \boldsymbol{u})}\right|\right\}$$

where $q(\boldsymbol{\gamma}^* \to \boldsymbol{\gamma})$ is the the probability of proposing a transaction from $(\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma})$ to $(\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^*)$. In the same way, the acceptance ratio of the reverse move proposal (from $\boldsymbol{\gamma}^* \to \boldsymbol{\gamma}$) is given by $\alpha((\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma}), (\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^*))$. The mechanism just introduced, however, requires a wise evaluation of the functions h and g, since their choice could affect the performances of the simulation

strategy.

Moreover, one may relax the assumptions on the size of $\boldsymbol{u}$. In this case, the dimensionality can still be matched by letting the length of $\boldsymbol{u}$ to be equal to l_p such that $N_p + l_p = N_{p^*} + l_{p^*}$. However, now $\boldsymbol{u}^*$ cannot be generated by the inverse function of $h_{\gamma \to \gamma^*}$ since it is no more available in deterministic terms. Therefore, $\boldsymbol{u}^*$ will be generated from a new density $h_{\gamma^* \to \gamma} = (\boldsymbol{u}^*)$. Finally, in addition to the mapping $\boldsymbol{\beta}^*_{\gamma^*} = g_{\gamma \to \gamma^*}(\boldsymbol{\beta}_\gamma, \boldsymbol{u})$ is added a reverse mapping $\boldsymbol{\beta}_\gamma = g_{\gamma^* \to \gamma}(\boldsymbol{\beta}^*_{\gamma^*}, \boldsymbol{u}^*)$. The new acceptance probability is thus

$$\alpha((\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma}), (\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^*)) = \min\left\{1, \frac{\pi(\boldsymbol{\beta}^*_\gamma, \boldsymbol{\gamma}^* | y_{1:T}, \boldsymbol{X}) q(\boldsymbol{\gamma}^* \to \boldsymbol{\gamma}) h_{\gamma^* \to \gamma}(\boldsymbol{u}^*)}{\pi(\boldsymbol{\beta}_\gamma, \boldsymbol{\gamma} | y_{1:T}, \boldsymbol{X}) q(\boldsymbol{\gamma} \to \boldsymbol{\gamma}^*) h_{\gamma \to \gamma^*}(\boldsymbol{u})} \left| \frac{\partial g_{\gamma \to \gamma^*}(\boldsymbol{\beta}_\gamma, \boldsymbol{u})}{\partial(\boldsymbol{\beta}_\gamma, \boldsymbol{u})} \right| \right\}.$$

In general, Reversible Jump MCMC have proved to perform well for variable selection and they allow for a rapid mixing time. However, we do not explore this approach further. See Green (1995) for furthers details and examples.

1.3 Bayesian Structural Time Series

After Varian and Choi (2009a), the awareness of scientists about the effectiveness of web data (and in particular Google Trends) in improving forecasting increased. However, because of the high dimensionality, dealing with such great amount of data poses new modelling challenges. In order to meet the demand for an ensemble method able to average over different combinations of predictors in time series analysis, Steven L. Scott and Hal Varian, respectively Senior Economist Analyst and Chief Economist at Google Inc, developed a new methodology that combines State-Space Models and Bayesian Variable Selection Methods. The authors refer to this new class of models as *Bayesian Structural Time Series* (BSTS). The latter is based on three pillars:

- Structural Time Series Models
- Spike and Slab Regression
- Markov Chain Monte Carlo methods

We develop the discussion in the next three sections.

1.3.1 Structural Time Series Models

Structural Time Series Models can be regarded as a subclass of Dynamic Linear Models (DLM) where the state process describes the evolution of some unobserved time series features, precisely the trend, the seasonal and, possibly, the cycle components. Intuitively, they remind of the classical time series decomposition, that describes a time series as composed by some systematic components (trend, seasonality, cycle) and a random component (noise). However, while classical decomposition is just a descriptive tool, Structural Time Series Models are probabilistic models, and allow prediction and a proper quantification of uncertainty - assuming that we are able to identify the components from the noisy background. Formulating the problem in the form of general multivariate DLM, the observable $\mathbb{R}^n$-valued time series $(\boldsymbol{Y}_t)_{t\geq 1}$ is related to the latent $\mathbb{R}^p$-valued state process $(\boldsymbol{\theta}_t)_{t\geq 0}$ through the following system of equation, for each $t \geq 1$,

$$\boldsymbol{Y}_t = \boldsymbol{F}_t\boldsymbol{\theta}_t + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}_n(\mathbf{0}, \Sigma_{\epsilon,t}) \tag{1.5}$$

$$\boldsymbol{\theta}_t = \boldsymbol{G}_t\boldsymbol{\theta}_{t-1} + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \sim \mathcal{N}_p(\mathbf{0}, \Sigma_{\eta,t}) \tag{1.6}$$

with $\boldsymbol{\theta}_0 \sim \mathcal{N}_p(\boldsymbol{m}_0, \boldsymbol{C}_0)$. $\boldsymbol{F}_t$ and $\boldsymbol{G}_t$ are respectively a $n \times p$ and a $p \times p$. The disturbances, $(\boldsymbol{\epsilon}_t)$ and $(\boldsymbol{\eta}_t)$, are assumed to be two sequences of serially uncorrelated Gaussian random vectors with mean zero and $p \times p$ covariance matrices $\Sigma_{\epsilon,t}$ and $\Sigma_{\eta,t}$. Moreover,it is assumed that $(\boldsymbol{\epsilon}_t) \perp\!\!\!\perp (\boldsymbol{\eta}_t) \perp\!\!\!\perp \boldsymbol{\theta}_0$. Equation (1.5) s refereed to as the observation equation and equation (1.6) as the state equation. State-space models naturally arise in problems of fintering, where interest is in extracting the signal, $(\boldsymbol{\theta}_t)_{t\geq 1}$, from an incomplete, potentially noisy, set of observations $(\boldsymbol{y}_t)_{t\geq 1}$. Very briefly, this is possible through a recursive procedure that exploits the Markovian structure of the state process and the assumption of conditionally independent observations (see (A.1) and (A.2)) in order to compute the filter and predictive densities at any time t starting from $\boldsymbol{\theta}_0 \sim p_0$. The process of filtering can be summarized in three steps. Let $(\boldsymbol{Y}_t, \boldsymbol{\theta}_t)_{t\geq 1}$ be a discrete time stochastic process satisfying (A.1) and (A.2), and $\boldsymbol{\psi}$ a vector of model's parameters, for $t \geq 1$ compute

(i) The one-step-ahead predictive distribution for the states using the filtering density at

time $t-1$, $\pi(\boldsymbol{\theta}_{t-1}|\boldsymbol{y}_{1:t-1})$:

$$\pi(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t-1}) = \int \int \pi(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \boldsymbol{\psi})\pi(\boldsymbol{\theta}_{t-1}|\boldsymbol{y}_{1:t-1}, \boldsymbol{\psi})\pi(\boldsymbol{\psi}|\boldsymbol{y}_{1:t-1})\ d\boldsymbol{\theta}_{t-1}\ d\boldsymbol{\psi}$$

(ii) The one-step-ahead predictive distribution for the observations using the density computed in step (i):

$$\pi(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}) = \int \int \pi(\boldsymbol{y}_t|\boldsymbol{\theta}_t, \boldsymbol{\psi})\pi(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t-1}, \boldsymbol{\psi})\pi(\boldsymbol{\psi}|\boldsymbol{y}_{1:t-1})\ d\boldsymbol{\theta_t}\ d\boldsymbol{\psi}$$

(iii) The filtered distribution at time t using the densities computed at step(i) and step (ii):

$$\pi(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t}) = \frac{\pi(\boldsymbol{y}_t|\boldsymbol{\theta}_t)\pi(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t-1})}{\pi(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1})}$$

Such densities are available in closed form and they are easy to compute in linear Gaussian State-Space Models only when model's parameters are known. In this latter case, the joint distribution for $(\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_T, \boldsymbol{Y}_1, ..., \boldsymbol{Y}_T)$ is a multivariate Normal distribution, which implies that also the marginal and conditional distributions are normally distributed. Therefore they are characterized by the first two moments, which are obtained by the celebrated Kalman Filter which is based on the process described below. Let $\boldsymbol{\theta}_0 \sim \mathcal{N}(\boldsymbol{m}_0, \boldsymbol{C}_0)$, compute for $t = 1, ..., T$

$$\begin{aligned}
\boldsymbol{a}_t &= \mathbb{E}(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t-1}) = \boldsymbol{G}_t\boldsymbol{m}_{t-1}, \quad \boldsymbol{R}_t = \mathbb{V}(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t-1}) = \boldsymbol{G}_t\boldsymbol{C}_{t-1}\boldsymbol{G}_t' + \Sigma_{\eta,t}, \\
\boldsymbol{f}_t &= \mathbb{E}(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}) = \boldsymbol{F}_t\boldsymbol{a}_t, \qquad \boldsymbol{Q}_t = \mathbb{V}(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}) = \boldsymbol{F}_t\boldsymbol{R}_t\boldsymbol{F}_t' + \Sigma_{\epsilon,t}, \\
\boldsymbol{m}_t &= \mathbb{E}(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t}) = \boldsymbol{a}_t + \boldsymbol{A}_t\boldsymbol{e}_t, \qquad \boldsymbol{C}_t = \mathbb{V}(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t}) = \boldsymbol{R}_t - \boldsymbol{A}_t\boldsymbol{Q}_t\boldsymbol{A}_t'
\end{aligned} \tag{1.7}$$

where $\boldsymbol{A}_t = \boldsymbol{R}_t\boldsymbol{F}_t'\boldsymbol{Q}_t^{-1}$ and $\boldsymbol{e}_t = \boldsymbol{y}_t - \boldsymbol{f}_t$. The problem of smoothing, that is, the computation of the distribution of $\boldsymbol{\theta}_{0:T} \mid \boldsymbol{y}_{1:T}$ is similarly solved by Kalman smoothing. Given the filtering distribution at time T, we trace the states' history up T through backward recursive formulae. Let $\boldsymbol{\theta}_{t+1}|\boldsymbol{y}_{1:T} \sim \mathcal{N}(\boldsymbol{s}_{t+1}, \boldsymbol{S}_{t+1})$, then $\boldsymbol{\theta}_t|\boldsymbol{y}_{1:T} \sim \mathcal{N}(\boldsymbol{s}_t, \boldsymbol{S}_t)$

$$\boldsymbol{s}_t = m_t + \boldsymbol{B}_t(\boldsymbol{s}_{t+1} - \boldsymbol{a}_{t+1}), \quad \boldsymbol{S}_t = \boldsymbol{C}_t - \boldsymbol{B}_t(\boldsymbol{R}_{t+1} - \boldsymbol{S}_{t+1})\boldsymbol{B}_t' \tag{1.8}$$

where $\boldsymbol{B}_t = \boldsymbol{C}_t\boldsymbol{G}_{t+1}'\boldsymbol{R}_{t+1}^{-1}$. Jointly, Kalman Filter (also known as *forward step*) and Kalman Smoother (or *backward step*) provide the posterior distribution of the states assuming that the matrices of the model are known. Unfortunately, the model's parameters are usually unknown. Nevertheless, the process described in (1.7) and (1.8) to obtain moments can be

used in Markov Chain Monte Carlo methods to compute the full conditional distribution $p(\boldsymbol{\theta}_t|\boldsymbol{y}_{1:t}, \boldsymbol{\psi})$. Further details on MCMC are provided in Section 1.3.3.

Let us now consider the BSTS model's specification. We restrict our attention to univariate time series, as in Scott and Varian (2013), but multivariate extensions are available (see Ning and Qiu (2021)). As already mentioned, the time series structural components enter the model through the state vector or, in other words, $\boldsymbol{\theta}_t = (\mu_t, \delta_t, \tau_t)'$ where μ_t is the level of the series at time t, δ_t is its the time-varying growth and τ_t is the seasonal factor. A convenient way to deal with seasonality is to consider the seasonal factors as the dynamic coefficients of a set of S dummy variables and constrain them to have zero expectation over a full cycle of S seasons. Finally, augmenting the observation equation with a regression component, we obtain a Bayesian Structural Time Series model

$$\begin{aligned} Y_t &= \mu_t + \tau_t + \boldsymbol{x}_t'\boldsymbol{\beta} + \epsilon_t \\ \mu_t &= \mu_{t-1} + \delta_{t-1} + u_t \\ \delta_t &= \delta_{t-1} + v_t \\ \tau_t &= -\sum_{s=1}^{S-1} \tau_{t-s} + w_t \end{aligned} \tag{1.9}$$

With reference to equation (1.6), $\boldsymbol{\eta}_t = (u_t, v_t, w_t)'$ and $\boldsymbol{\eta}_t \sim N_3(\mathbf{0}, \Sigma_\eta)$ where $\Sigma_{\eta,t} = diag(\sigma_u^2, \sigma_v^2, \sigma_w^2)$, while $\Sigma_{\epsilon,t} = \sigma_\epsilon^2$ in equation (1.5), for $t \geq 1$. The unknown parameters of this model are $(\boldsymbol{\beta}, \sigma_\epsilon^2, \sigma_u^2, \sigma_v^2, \sigma_w^2)$. We assign a prior to the unknown parameters, that is updated as new observations become available. More insights on this are provided in Section 1.3.3. In the next paragraph (Section 1.3.2) we focus instead on the regression part and the problem of Bayesian variable selection.

1.3.2 Static Shrinkage in Bayesian Structural Time Series

A static regression component is frequently included in Bayesian Structural Time Series. The latter can be incorporated into the baseline state space model in a variety of ways. The method envisaged by Scott and Varian (2013) is conceptually the simplest and operationally the most efficient. It consists in appending a constant 1 to the state vector $\boldsymbol{\theta}_t$ and $\boldsymbol{x}_t'\boldsymbol{\beta}$ to

Z_t in equation (1.9). In this way the dimension of the state vector increases just by one, regardless the number of predictors, and since the computational complexity of the Kalman filter is linear in the length of the data and quadratic in the dimension of the state vector, the computational savings are substantial. Considering that BSTS have been developed to deal with internet data, the number of predictors entering our model might be potentially huge and, in particular, much larger than the number of available observations (in other words $p >> T$). Therefore, estimation issues are likely to arise. To cope with this, spike-and-slab regression is used to introduce sparsity among the regression coefficients. Especially when using web data, it is indeed safe to assume that many predictors would not play any role in the regression and thus they must be shrunk to zero. For this purpose, Scott and Varian (2013) rely on the SSVS we discussed in Section 1.2.1. Let γ_j be an auxiliary binary indicator assuming two values

$$\gamma_j = 1 \quad \text{if} \quad \beta_j \neq 0$$
$$\gamma_j = 0 \quad \text{if} \quad \beta_j = 0$$

where β_j is the coefficient of the j-th predictor. Let $\boldsymbol{\beta}_\gamma$ be the subset of active coefficients, i.e $\beta_j \neq 0$. A spike-and-slab prior can be written

$$\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}, \sigma_\epsilon^2) = \pi(\boldsymbol{\beta}_\gamma | \boldsymbol{\gamma}, \sigma_\epsilon^2)\pi(\sigma_\epsilon^2 | \boldsymbol{\gamma})\pi(\boldsymbol{\gamma})$$

where $\pi(\boldsymbol{\gamma})$ is the spike and, in practice, it usually coincides with a Bernoulli prior

$$\boldsymbol{\gamma} \sim \prod_{j=1}^{p} \pi_j^{\gamma_j}(1 - \pi_j)^{1-\gamma_j}$$

which can be simplified by assuming that the probability of the indicator to be 0 or 1 is the same for each predictor or, in other words, by imposing $\pi_j = \pi$. Moreover, there are multiple ways to set the hyperparameter's values. Coherently with the Bayesian approach, a simple strategy is to fix π according to the "expected model size." Therefore the idea is to make a guess on how many coefficients we expect to be active, p_0, and then set $\pi = \frac{p_0}{p}$. On the other hand, a convenient way to model $\boldsymbol{\beta}$ and σ_ϵ^2 is through a conjugate prior law, namely the Normal-Gamma prior distribution

$$\boldsymbol{\beta}_\gamma | \boldsymbol{\gamma}, \sigma_\epsilon^2 \sim \mathcal{N}(\boldsymbol{b}_\gamma, \sigma_\epsilon^2(\Omega_\gamma^{-1})^{-1}) \tag{1.10}$$

$$\frac{1}{\sigma_\epsilon^2}|\boldsymbol{\gamma} \sim \mathcal{G}\left(\frac{n_0}{2}, \frac{d_0}{2}\right) \tag{1.11}$$

where $\boldsymbol{b}_\gamma$ is a vector of prior guesses for non null predictors, Ω is a symmetric matrix and Ω_γ denote the rows and columns of Ω for which $\gamma_j = 1$. Furthermore, n_0 represents the prior sample size and d_0 is the prior sum of squares. They can be set properly by making a guess on the number of observations worth of weight and the expected R^2. Therefore, equations (1.10) and (1.11) represent the slab density, which can be more or less informative according to how the hyperparameters have been set. Finally, for effective shrinkage, a possible prior distribution for $\boldsymbol{\beta}_\gamma|\boldsymbol{\gamma}, \sigma_\epsilon^2$ is the "g-prior" proposed by Zellner (1986). The latter is characterized by a null vector $\boldsymbol{b}_\gamma$ and $\Omega^{-1} = g\boldsymbol{X}'\boldsymbol{X}/n$.

Let $y_t^\star = y_t - \boldsymbol{F}_t^\star \boldsymbol{\theta}_t$ where $\boldsymbol{F}_t^\star$ is the observation matrix of equation (1.5) with the regression component $\boldsymbol{\beta}'\boldsymbol{x}_t$ set to zero and let $y_{1:T}^\star = (y_1^\star, ..., y_t^\star)$. Thanks to the conjugate nature of the model, posterior inference can be carried out. The Bayesian updating process of a Normal-Gamma prior leads to a Normal-Gamma posterior distribution with parameters:

$$\boldsymbol{\beta}_\gamma|\boldsymbol{\gamma}, \sigma_\epsilon^2, y_{1:T}^\star \sim \mathcal{N}(\tilde{\boldsymbol{\beta}}_\gamma, \sigma_\epsilon^2(\Lambda_\gamma^{-1})^{-1})$$
$$\frac{1}{\sigma_\epsilon^2}|\boldsymbol{\gamma}, y_{1:T}^\star \sim \mathcal{G}\left(\frac{n_T}{2}, \frac{d_T}{2}\right)$$

where

$$\Lambda_\gamma^{-1} = (\boldsymbol{X}'\boldsymbol{X})_\gamma + \Omega_\gamma^{-1}$$
$$n_T = n_0 + T$$
$$\tilde{\boldsymbol{\beta}}_\gamma = (\Lambda_\gamma^{-1})^{-1}(\boldsymbol{X}'_\gamma y_{1:T}^\star + \Omega_\gamma^{-1}\boldsymbol{b}_\gamma)$$
$$d_T = d_0 + y_{1:T}^{\star\prime} y_{1:T}^\star + \boldsymbol{b}'_\gamma \Omega_\gamma^{-1}\boldsymbol{b}_\gamma - \tilde{\boldsymbol{\beta}}_\gamma{}' \Lambda_\gamma^{-1}\tilde{\boldsymbol{\beta}}_\gamma$$

Moreover, it is possible to marginalize out $\boldsymbol{\beta}_\gamma$ and σ_ϵ^2 from the full posterior distribution $\pi(\boldsymbol{\beta}_\gamma, \sigma_\epsilon^2, \boldsymbol{\gamma})$ and obtain the marginal posterior distribution of the indicator

$$\boldsymbol{\gamma}|y_{1:T}^\star \sim C(y_{1:T}^\star)\frac{|\Omega_\gamma^{-1}|^{\frac{1}{2}}\pi(\boldsymbol{\gamma})}{|\Lambda_\gamma^{-1}|^{\frac{1}{2}} d_T^{\frac{n_T}{2}-1}}$$

where $C(y_{1:T}^\star)$ is a normalizing constant that depends on $\boldsymbol{y}^\star$ but not on $\boldsymbol{\gamma}$. This equation can be used to draw from the posterior distribution of every γ_j given $\boldsymbol{\gamma}_{-j}$, which represents all the elements of $\boldsymbol{\gamma}$ except γ_j. However, the MCMC algorithms used to fit the model is such

that it does not require $C(y^{\star}_{1:T})$ to be computed explicitly.

1.3.3 MCMC methods for posterior inference

Let $\boldsymbol{\psi} = (\boldsymbol{\beta}, \sigma^2_\epsilon, \sigma^2_u, \sigma^2_v, \sigma^2_w)$ be the vector of model parameters and y_{t+h} the h-step ahead forecast, the Bayesian approach to parameter learning and forecasting is based on providing their respective posterior distribution starting from an assigned prior distribution. The prior densities of $(\boldsymbol{\beta}, \sigma^2_\epsilon)$ have already been described in the previous paragraph; we also assign an Inverse-Gamma prior to $(\sigma^2_u, \sigma^2_v, \sigma^2_w)$ so that the update rule is analytically available. Since model's parameters are unknown, and here treated as random quantities, Kalman Filter and Kalman Smoother cannot be directly used to obtain a full description of these densities, but an approximate solution can be obtained through Markov Chain Monte Carlo (MCMC) methods. The algorithm below is the Gibbs sampling presented in Scott and Varian (2013).

Algorithm 1: Parameter learning in BSTS

Step 1: Sample states and States' variances

1 Sample $\theta_{0:T}$ from $p(\theta_{0:T}|\boldsymbol{\beta}, \sigma^2_\epsilon, \Sigma_\eta, y_{1:T})$ using the method proposed by Durbin and Koopman (2002);

2 Sample indipendently the diagonal elements of Σ_η from $p(\sigma^2_i|\theta_{0:T}, \boldsymbol{\beta}, \sigma^2_\epsilon, y_{1:T})$ for $i \in \{u, v, w\}$;

Step 2: SSVS

3 Sample the coefficient vector $\boldsymbol{\beta}$ from $p(\boldsymbol{\beta}|\theta_{0:T}, \sigma^2_\epsilon, \boldsymbol{\gamma}, y_{1:T})$;

4 Sample σ^2_ϵ from $p(\sigma^2_\epsilon|\theta_{0:T}, \boldsymbol{\beta}, \boldsymbol{\gamma}, y_{1:T})$;

5 Sample the indicator vector $\boldsymbol{\gamma}$ componentwise by sampling consecutively from $p(\gamma_j|\theta_{0:T}, \boldsymbol{\beta}, \boldsymbol{\gamma}_{-j}, y_{1:T})$

The first step of Algorithm 1 can be carried out in a variety of way. In Scott and Varian (2013), the authors propose the Simulation Smoother developed by Durbin and Koopman

(2002). A brief description of this method is reported separately in Algorithm 2. Some of the other existing strategies that are used in Chapter 2 are the Forward Filter Backward Sampling (FFBS) algorithm by Carter and Kohn (1994), Fruhwirth-Schnatter (1994), Shephard (1994) and the more recent sparse matrix method of Chan and Jeliazkov (2009).

Algorithm 2: Simulation Smoother by Durbin and Koopman (2002)

1 Draw $\boldsymbol{\xi}^+ \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\xi)$, where $\boldsymbol{\xi} = (\boldsymbol{\epsilon}_1', \boldsymbol{\eta}_1', ..., \boldsymbol{\epsilon}_T', \boldsymbol{\eta}_T')'$ and $\boldsymbol{\Sigma}_\xi = diag(\Sigma_{\epsilon,1}, \Sigma_{\eta,1}, ..., \Sigma_{\epsilon,T}, \Sigma_{\eta,T})$;

2 Draw $\boldsymbol{\theta}_0 \sim \mathcal{N}(\boldsymbol{m}_0, \boldsymbol{C}_0)$;

3 Compute $\boldsymbol{\theta}_t^+$ and $\boldsymbol{y}_t^+$, using equation (1.6) and 1.5 for $t = 1, ..., T$;

4 Apply Kalman Filter and Kalman Smoother to $\boldsymbol{y}_{1:T}$ and $\boldsymbol{y}_{1:T}^+$, respectively the original and the simulated time series, and compute $\boldsymbol{m}_t$ and $\boldsymbol{m}_t^+$ for $t = 0, ..., T$;

5 Compute the mean correction $\tilde{\boldsymbol{\theta}}_t = \boldsymbol{m}_t + (\boldsymbol{\theta}_t^+ - \boldsymbol{m}_t^+)$ for $t = 0, ..., T$ and obtained the corrected trajectory of the latent states $\tilde{\boldsymbol{\theta}}_{0:T}$;

Forecasting and parameter learning are two interconnected issue. Let $\tilde{\boldsymbol{y}} = (y_{T+1}, ..., y_{T+h})$ the random vector of future observations, forecasting coincides with making inference on the posterior density $\pi(\tilde{\boldsymbol{y}}|y_{1:T})$. Such distributions is easy to sample once we have knowledge of the parameters, by integrating out the latter:

$$\pi(\tilde{\boldsymbol{y}}|y_{1:T}) = \int_\Psi \pi(\tilde{\boldsymbol{y}}|\boldsymbol{\psi}, y_{1:T})\pi(\boldsymbol{\psi}|y_{1:T})\ d\boldsymbol{\psi} \tag{1.12}$$

where $\pi(\tilde{\boldsymbol{y}}|\boldsymbol{\psi}, y_{1:T})$ is obtain through the following recursion. For $h = 1, 2, ...$

(i) $\pi(\boldsymbol{\theta}_{t+h}|\boldsymbol{\psi}, y_{1:T}) = \int \pi(\boldsymbol{\theta}_{t+h}|\boldsymbol{\psi}, \boldsymbol{\theta}_{t+h-1})\pi(\boldsymbol{\theta}_{t+h-1}|\boldsymbol{\psi}, y_{1:T})d\boldsymbol{\theta}_{t+h-1}$

(ii) $\pi(y_{t+h}|\boldsymbol{\psi}, y_{1:T}) = \int \pi(y_{t+h}|\boldsymbol{\psi}, \boldsymbol{\theta}_{t+h})\pi(\boldsymbol{\theta}_{t+k}|\boldsymbol{\psi}, y_{1:T})d\boldsymbol{\theta}_{t+h}$

The draws generated from the density of equation (1.12) can be used to compute quantities of interest such as the expected values, the median, the variance and the quantiles. Moreover, an interesting feature of SSVS algorithm is that each draw $(\boldsymbol{\theta}_{0:T}^{(i)}, \boldsymbol{\psi}^{(i)})$, $i = 1, ..., N$, where N is the number of draws, depends on a different $\boldsymbol{\gamma}^{(i)}$ or, in other words, on a different

model. Therefore the posterior predictive distribution can be thought as a weighted average of model-specific predictive distributions where the weights are given by $\pi(\boldsymbol{\gamma}|y_{1:T})$. This mechanism is better known as Bayesian Model Averaging.

Finally, the intrinsic flexibility of Bayesian Structural Time Series models allows to insert a dynamic regression component into the system of equation (1.9). Although Scott and Varian (2013) do not explore this further, the insight is to enlarge the state vector of as many dimensions as the number of dynamic coefficients, which are assumed to have this evolution

$$\beta_{i,t} = \beta_{i,t-1} + \xi_t, \quad \xi_t \sim \mathcal{N}\left(0, \frac{\sigma_i^2}{\mathbb{V}(\boldsymbol{X}_i)}\right)$$

for i^{th} coefficient. Thus, each coefficient is supposed to evolve independently as a random walk and its variance is given by σ_i^2 scaled by the variance of the i^{th} column of $\boldsymbol{X}$. Moreover, a Gamma prior is assigned to the precision

$$\frac{1}{\sigma^2} \sim \mathcal{G}(a, b)$$

and two independent Gamma priors are also assigned to the hyperparameters $\sqrt{a/b}$ and a. The authors do not discourage the use of dynamic regressors, however they recommend to be parsimonious with them. As remarked in Section 1.3.2, the computational complexity of Kalman Filter is quadratic in the size of the state vector, hence the introduction of many dynamic regressors would dramatically increases the computational efforts. Moreover, no shrinking methods has been envisaged so far. Therefore, the inclusion of a set of dynamic regressors (that is meant to be large in a big data context) may lead to a slow and inefficient algorithm, entailing overfitting and inaccurate states' estimates. In Chapter 3 we propose an innovative approach to overcome these problems by introducing sparsity in the Bayesian Structural Time Series framework.

Chapter 2

Dynamic Shrinkage with Spike-and-Slab Process Priors

In time-series analysis, the assumption of static regression coefficients may be overly limiting. Moreover, the lack of dynamic variable selection methods posed a significant barrier to using large sets of predictors in dynamic regression models. Recently, a growing literature has been addressing this issue (see Kalli and Griffin 2014, Nakajima and West 2013, Belmonte, Koop, and Korobilis 2014, Bitto and Frühwirth-Schnatter 2018, and others). In this chapter we want to illustrate the dynamic variable selection approach of Rockova and McAlinn (2021a) is a significant contribution in this research area. Before we continue, a small premise on what is dynamic sparsity is necessary. Let $\boldsymbol{\beta}_{1:T} = (\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_T)$ be the matrix of regression coefficients. Sparsity can be induced: (a) *horizontally*, when the j^{th} predictor is not persistently relevant or, in other words, the j^{th} coefficient $\beta_{1:T,j} = (\beta_{1,j}, \dots, \beta_{T,j})$ presents intermittent zeros, (b) *vertically*, when at period t only a subset of coefficients $\boldsymbol{\beta}_t = (\beta_{t,1}, \dots, \beta_{t,p})$ is active. With the term *dynamic sparsity* we refer to a two-pronged issue, which is given by the combination of horizontal and vertical sparsity.

Therefore, consider the following large Time-Varying Parameter (TVP) regression model

characterized by a latent process having a Markovian structure

$$\begin{aligned} y_t &= \boldsymbol{x}'_t \boldsymbol{\beta}_t + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2_{\epsilon,t}) \\ \boldsymbol{\beta}_t &= f(\boldsymbol{\beta}_{t-1}) + \boldsymbol{\xi}_t, \quad \boldsymbol{\xi}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}_t) \end{aligned} \tag{2.1}$$

for $t = 1, ..., T$, and assume that only a small fraction of the total set of explanatory variables is truly relevant and that their importance changes over time. Assuming f to be a linear function, this TVP model coincides with the DLM illustrated in Section 1.3.1 with $\boldsymbol{F}_t = \boldsymbol{x}_t$ and $\boldsymbol{\theta}_t$ being a vector of time-varying regression coefficients $\boldsymbol{\beta}_t = (\beta_{t,1}, ..., \beta_{t,p})'$. It is the Markovian structure of the state process that makes possible to estimate the $T \times p$ coefficients of equation (2.1) with just T observations. Standard filtering approaches, on the other hand, become less accurate at providing state estimations as p grows larger, dissolving the signal across redundant predictors. Therefore, the necessity of a dynamic variable selection method arises. *Dynamic Spike-and-Slab (DSS) priors* (Rockova and McAlinn (2021a)) offer a solution to this problem. Before we proceed, it is worth to make a remark about the trade-off between modeling flexibility in the (conditional) mean and in the residuals, and possible poor identifiability or overfitting. Time-varying regression coefficients allow to capture non linear features of the data; thus, the model might actually succeed in capturing the predictable dependence through the (flexible) conditional mean, so that "what is left" are indeed i.i.d. residuals. However, there are situations where some form of structural dependence still remains. In particular, model mispecification might imply time-varying residual variance despite the large number of predictors. Also, especially in macroeconomics, idiosyncratic shocks are of interest and they are usually time persistent, for this reason in the macroeconomic literature they are usually modeled using a stochastic volatility model. The gain of switching from a constant variance model to a time-varying volatility model for macroeconomic forecasting have been empirically shown by Clark and Ravazzolo (2015).

2.1 Dynamic Spike-and-Slab priors

Dynamic Spike-and-Slab priors (DSS) can be regarded as an extension of the Spike-and-Slab approach, illustrated in paragraph 1.2, to the framework of Mixture Autoregressive (MAR) processes (Wong and Li 2000), where the mixing weights are allowed to change over time. We present this approach in a TVP regression model where every coefficients $\beta_{t,j}$ are independently and identically distributed as a DSS process prior. In this case we can simplify the notation by suppressing the subscript j.

Let us start with the conditional specification of the DSS prior. Given β_{t-1} and an auxiliary binary indicator $\gamma_t \in \{0, 1\}$ that denotes whether the variable fall in the spike $\psi_0(\beta_t|\lambda_0)$ or in the slab $\psi_1(\beta_t|\mu_t, \lambda_1)$ distribution, then β_t has a mixture density

$$\pi(\beta_t|\gamma_t, \beta_{t-1}) = (1 - \gamma_t)\psi_0(\beta_t|\lambda_0) + \gamma_t\psi_1(\beta_t|\mu_t, \lambda_1) \tag{2.2}$$

where

$$\mu_t = \phi_0 + \phi_1(\beta_{t-1} - \phi_0) \quad \text{with} \quad |\phi_1| < 1 \tag{2.3}$$

and

$$P(\gamma_t = 1|\beta_{t-1}) = \omega_t \tag{2.4}$$

In order to achieve variable selection, the spike and slab distributions have to be choosen such that the first is concentrated on zero (λ_0 is small) while the latter is more diffuse ($\lambda_1 > \lambda_0$) and it may eventually have a different mean. Note that this hierarchical setup (equations (2.2)–(2.4)) introduces two important innovations with respect to the one described in Section 1.2.1.

Firstly, instead of centering both the spike and the slab densities at zero, the latter depends on the previous value of the coefficients through μ_t. This feature enables to regard the mixture prior of equation (2.2) as a *multiple shrinkage prior* (George 1986a and George 1986b) which has two gravitational pulls, 0 and μ_t, and only the slab density depends on β_{t-1}. Therefore, with Dynamic Spike-and-Slab priors we are assuming that the regression coefficients can be classified into two groups: irrelevant coefficients, which fall in the spike and are anchored to zero, and active coefficients, which follow an autoregressive process.

In this thesis we conveniently assume that the spike and slab distributions are Gaussian, i.e. $\psi_0(\beta_t|\lambda_0) \equiv \mathcal{N}(0, \lambda_0)$ and $\psi_1(\beta_t|\mu_t, \lambda_1) \equiv \mathcal{N}(\mu_t, \lambda_1)$. Therefore, when the coefficient is active it follows a stationary Gaussian autoregressive process of order one:

$$\beta_t = \phi_0 + \phi_1(\beta_{t-1} - \phi_0) + \xi_t, \quad \xi_t \overset{iid}{\sim} \mathcal{N}(0, \lambda_1) \tag{2.5}$$

with $|\phi_1| < 1$, which is characterized by the following Gaussian stationary distribution

$$\psi_1^{ST}(\beta_t|\lambda_1, \phi_0, \phi_1) \equiv \psi_1\left(\beta_t|\phi_0, \frac{\lambda_1}{1-\phi_1^2}\right). \tag{2.6}$$

The second advancement is the use of time-varying mixing weights. The sequence of probabilities $(\omega_t)_{t\geq 1}$ is indeed able to evolve smoothly over time. The way the conditional inclusion probabilities are modeled by Rockova and McAlinn (2021a) allows to make them marginally stable, that is, their marginal distribution is constant over time, while also including all the relevant information. The rule is

$$\omega_t \equiv \omega(\beta_{t-1}) = \frac{\Omega\ \psi_1^{ST}(\beta_{t-1}|\lambda_1\phi_0, \phi_1)}{\Omega\ \psi_1^{ST}(\beta_{t-1}|\lambda_1\phi_0, \phi_1) + (1-\Omega)\ \psi_0(\beta_{t-1}, \lambda_0)} \tag{2.7}$$

where $\Omega \in (0, 1)$ is a scalar hyperparameter whose role is to weight the chances to fall in the spike or in the slab distribution. ω_t is therefore the conditional inclusion probability of β_{t-1} to belong to the stationary slab distribution or to the stationary spike distribution. The intuition is the following: when $|\beta_{t-1}|$ is large, then $\omega(\beta_{t-1})$ approaches one, pointing out that β_t is likely to belong to the slab distribution. In the same manner, a small coefficient $|\beta_{t-1}|$ means a small mixing weight $\omega(\beta_{t-1})$ and thus it suggests that β_t is in the spike.

Together, equations (2.2)–(2.4) and (2.7) define a Dynamic Spike-and-Slab (DSS) process with hyperparameters $(\Omega, \lambda_0, \lambda_1, \phi_0, \phi_1)$

$$(\beta_t) \sim DSS(\Omega, \lambda_0, \lambda_1, \phi_0, \phi_1) \tag{2.8}$$

Again, we remark that this formulation entails an interesting consequence which consists in having stable marginal distributions. Theorem 1 in Rockova and McAlinn (2021a) proves that if $(\beta_t) \sim DSS(\Omega, \lambda_0, \lambda_1, \phi_0, \phi_1)$ with $|\phi_1| < 1$, then the probability law of (β_t) is has marginal distributions

$$\pi^{ST}(\beta_t|\Omega, \lambda_0, \lambda_1, \phi_0, \phi_1) = \Omega\ \psi_1^{ST}(\beta_t|\lambda_1, \phi_0, \phi_1) + (1-\Omega)\ \psi_0(\beta_t|\lambda_0) \tag{2.9}$$

for $t \geq 1$.

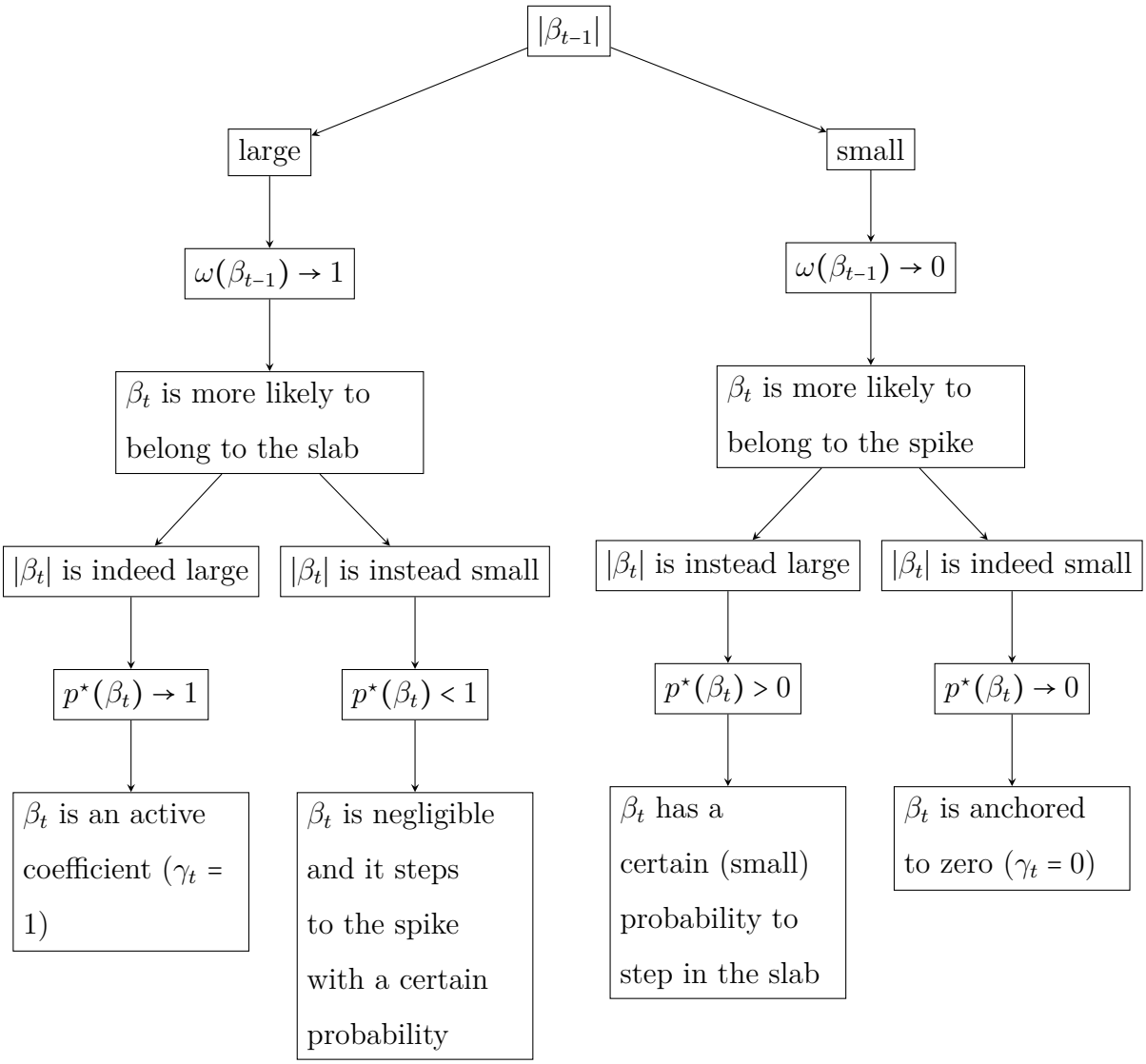
In fact, the mechanism that induces shrinkage counts two steps. After establishing whether the coefficients belong to the the *stationary* slab distribution, then it should be assessed whether the coefficients belong also to the *conditional* slab density. In other words, it is necessary to compute $p_t^\star(\beta_t) = \mathbb{P}(\gamma_t = 1 \mid \beta_t, \beta_{t-1}, \omega_t)$. As shown in Rockova and McAlinn (2021a), the posterior inclusion probability results to be

$$p_t^\star(\beta_t) \equiv \frac{\omega_t \psi_1(\beta_t|\mu_t, \lambda_1)}{\omega_t \psi_1(\beta_t|\mu_t, \lambda_1) + (1-\omega_t)\psi_0(\beta_t|\beta_0)} \tag{2.10}$$

Therefore, if β_{t-1} is large, then $\omega_t(\beta_{t-1})$ tends to one, signalling that β_t is likely to belong to the slab. However, depending on the value of β_t estimated, there are two possible scenarios. If the latter is large in absolute value then also $p_t^\star(\beta_t)$ is close to one, which signals that there are no more doubts about β_t being active and to belong to an autoregressive path. On the other hand, if $|\beta_t|$ is small, $p_t^\star(\beta_t)$ will be close to zero. In the latter case, the predictor is considered no more important and its coefficient is shrunk to zero. Clearly, shrinkage towards zero is even more accentuated if also $|\beta_{t-1}|$ is small, anchoring the coefficient to the spike. The full information transmission mechanism is described in the flowchart on the next page.

Assuming Gaussian spike and slab distributions thus leads to shrinkage towards μ_t or zero. Alternatively, rather than shrinking around those values, one may prefer to bring the coefficients to these exact values. This is possible within the Dynamic Spike-and-Slab framework by assuming $\psi_0(\beta_t|\lambda_0)$ and $\psi_1(\beta_t|\mu_t, \lambda_1)$ to be two Laplace priors centered respectively in 0 and μ_t, i.e. $\psi_0(\beta|\lambda_0) \equiv \frac{\lambda_0}{2}\exp\{-|\beta|\lambda_0\}$ and $\psi_1(\beta|\mu_t, \lambda_1) \equiv \frac{\lambda_1}{2}\exp\{-|\beta - \mu_t|\lambda_1\}$. However, while using a Laplace spike does not introduce further complexities to the model, modelling the slab density as a Laplace density entails a whole range of considerations. We discuss them in Appendix.

Operationally, once the full system has been specified, inference can be carried out using Markov Chain Monte Carlo methods. In the next sections we illustrate some useful algorithms to efficiently simulate from the posterior distributions and to compute Maximum A Posteriori (MAP) estimates.



2.2 Dynamic Stochastic Search Variable Selection

Efficient posterior simulation of parameters of Time-Varying Parameter regression models with Dynamic Spike-and-Slab process priors is made possible thanks to a MCMC algorithm developed by Rockova and McAlinn (2021a). The latter consists of a Gibbs sampler which develops in three stages:

1. Simulate the regression coefficients from $\pi(\boldsymbol{\beta}_{0:T}|\boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,0:T}, y_{1:T})$ using Forward Filter

Backward Sampling (FFBS), as illustrated in *Step: 1* of Algorithm 3,

2. Simulate the inclusion indicators from $\pi(\boldsymbol{\gamma}_{0:T}|\boldsymbol{\beta}_{0:T}\sigma^2_{\epsilon,0:T}, y_{1:T})$ using results of Equations 2.7 and 2.10 to compute conditional mixing weights and posterior inclusion probabilities,

3. Simulate the residual variances from $\pi(\sigma^2_{\epsilon,0:T}|\boldsymbol{\gamma}_{0:T}, \boldsymbol{\beta}_{0:T}, y_{1:T})$ using a FFBS strategy as the one illustrated in *Step: 3* of Algorithm 3.

Since this strategy reminds the SSVS of George and McCulloch (1993), then the procedure takes the label *Dynamic Stochastic Search Variable Selection* (or Dynamic SSVS). Here we illustrate the original version of the Dynamic SSVS developed by the authors, which constitute the starting point for the algorithms developed in the next sections. In particular, we refer to the Gaussian Spike-and-Slab case introduced previously. Nevertheless, the procedure described here can be rearranged easily to include a Laplacian spike instead of a Gaussian one. The Gaussian specification allow a straightforward and elegant representation of the latent process, which is the following

$$\boldsymbol{\beta}_t = \boldsymbol{H}_t + \boldsymbol{G}_t(\boldsymbol{\beta_{t-1}} - \boldsymbol{H}_t) + \boldsymbol{\xi}_t, \quad \boldsymbol{\xi}_t \sim \mathcal{N}(\mathbf{0}, \Lambda_t) \tag{2.11}$$

where $\boldsymbol{H}_t = \phi_0\boldsymbol{\gamma}'_t$, $\boldsymbol{G}_t = diag\{\gamma_{t,j}\phi_1\}_{j=1}^p$, $\Lambda_t = diag\{\gamma_{t,j}\lambda_1 + (1-\gamma_{t,j})\lambda_0\}_{j=1}^p$ and the initial state vector $\boldsymbol{\beta}_0 \sim \mathcal{N}(\boldsymbol{m}_0, \boldsymbol{C}_0)$ with $\boldsymbol{m}_0 = \phi_0\boldsymbol{\gamma}_0$ and $\boldsymbol{C}_0 = diag\{\gamma_{0,j}\lambda_1/(1-\phi_1^2) + (1-\gamma_{0,j})\lambda_0\}_{j=1}^p$. This model's hyperparameters are $(\Omega, \lambda_0, \lambda_1, \phi_0, \phi_1)$, which can be calibrated or, eventually, treated as random variables by placing a prior density to them. In this thesis we fix a priori ϕ_1 and $\phi_0 = 0$, and we usually assume that $\phi_0 = 0$ and ϕ_1 ranges between 0.98 and 0.9. This of course entails great simplifications, but seems to work well in all cases we explored. On the other hand, in its original version the author suggest to treat ϕ_1 as a Beta random variable with support $[0.8, 1)$. We see that the gain due to the latter specification and the drawbacks in terms of implementation of adding a Metropolis-Hasting step in the Gibbs sampling scheme compensate each other, that's why we opted for we opted for a predetermined choice. The hyperparameters λ_0 and λ_1 are set a-priori also in the original model Rockova and McAlinn (2021a).

The residual variances are modeled through discount factors. This simple strategy, developed by West and Harrison (1997), allows temporal fluctuations of the model's variance while preserving conjugacy. Briefly, consider the precision sequence $\nu_t = \frac{1}{\sigma^2_{\epsilon,t}}$ for $t = 1, ..., T$. The latter is assumed to follow a stochastic evolution affected by independent random shocks c_t/δ, thus

$$\nu_t = \frac{c_t}{\delta}\nu_{t-1} \tag{2.12}$$

where δ is the discount factor. It takes values in $(0, 1]$ and it can be regarded as the decay of precision from period $t-1$ to t. Therefore, the precision in every period depends on the previous precision and on the new information that becomes recursively available. The dynamic of equation (2.12) is completed by assuming $c_t \sim \mathcal{B}(\frac{\delta n_{t-1}}{2}, \frac{(1-\delta)n_{t-1}}{2})$ and $n_t = \delta n_{t-1} + 1$. This implies $\mathbb{E}(c_t|y_{1:t-1}) = \delta$ or, equivalently, that $\mathbb{E}(c_t/\delta|y_{1:t-1}) = 1$. Assigning a Gamma prior on $\nu_0 \sim \mathcal{G}(n_0, d_0)$, one can solve the updating process and solutions in closed form. For $t > 0$, the one-step-ahead predictive distribution is

$$\nu_t|y_{1:t-1}, \boldsymbol{\beta}_{1:t-1} \sim \mathcal{G}\left(\frac{\delta n_{t-1}}{2}, \frac{\delta d_{t-1}}{2}\right)$$

which updates into

$$\nu_t|y_{1:t}, \boldsymbol{\beta}_{1:t} \sim \mathcal{G}\left(\frac{n_t}{2}, \frac{d_t}{2}\right) \tag{2.13}$$

with $n_t = \delta n_{t-1} + 1$ and $d_t = \delta d_{t-1} + r_t^2$, where $r_t = y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t$. Posterior sampling of model's precision can be carried out via a FFBS algorithm (more details in West and Harrison, 1997). Basically, the filtering steps follow the updating rule just mentioned, while the backward sampling consists in drawing from $\nu_T \sim \mathcal{G}(n_T/2, d_T/2)$, and then drawing from $\eta_t \sim \mathcal{G}((1-\delta)n_t/2, d_t/2)$ and setting $\nu_t = \eta_t + \delta\nu_{t+1}$ for $t = T-1, ..., 0$. This mechanism is outlined in *Step 3* of Algorithm 3. Note that the discount factor strategy for variances estimation can be reconciled to the standard Bayesian estimation strategy for the unknown model's variance. Consider for example

$$y_t|\boldsymbol{x}'_t\boldsymbol{\beta}_t, \sigma^2_\epsilon \overset{ind}{\sim} \mathcal{N}(\boldsymbol{x}'_t\boldsymbol{\beta}_t, \sigma^2_\epsilon)$$
$$\frac{1}{\sigma^2_\epsilon} \sim \mathcal{G}\left(\frac{n}{2}, \frac{d}{2}\right)$$

Then we have

$$\begin{aligned}
\pi(\nu|\boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, y_{1:T}) &\propto \pi(\nu, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, y_{1:T}) \\
&\propto \pi(y_{1:T}|\nu, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T})\pi(\boldsymbol{\beta}_{0:T}|\nu, \boldsymbol{\gamma}_{0:T})\pi(\boldsymbol{\gamma}_{0:T}|\nu)\pi(\nu) \\
&\propto \pi(y_{1:T}|\nu, \boldsymbol{\beta}_{0:T})\pi(\nu) \\
&\propto \prod_{t=1}^{T} \pi(y_t|\nu, \boldsymbol{\beta}_t)\pi(\nu) \\
&\propto \pi(\nu)(\nu)^{T/2} \exp\left\{ -\frac{\nu}{2}\sum_{t=1}^{T}(y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t)^2 \right\} \\
&\propto \nu^{\frac{n}{2}+T/2+1} \exp\left\{ -\nu\left[\frac{d}{2} + \frac{1}{2}\sum_{t=1}^{T}(y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t)^2\right]\right\}.
\end{aligned}$$

Therefore, the full conditional posterior distribution is

$$\nu|\boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, y_{1:T} \sim \mathcal{G}\left(\frac{n+T}{2}, \frac{d}{2} + \frac{1}{2}\sum_{t=1}^{T}(y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t)^2\right)$$

which is equivalent to the one obtained using the discount factor model at time T for $\delta = 1$. Consequently, no further challenges are required to step from a stochastic to a constant volatility model.

The Dynamic SSVS illustrated in this section provides satisfactory results, however there is room for improvements. Trials performed on simulated data show indeed that misspecification concerning residual variances may seriously affect coefficients' estimates. The reason lies on an artificial bias we detected in the signal-to-noise ratio, which is decreasing in $\sigma^2_{\epsilon,t}$. We label this phenomenon the "Spike trap." In fact, some irrelevant predictors may activate after some periods whereas other predictors might leave and re-enter the model multiple times as time progresses. However, the Dynamic Spike-and-Slab approach is basically distrustful and it allows steps from the spike to the slab distribution only if there is enough evidence supporting this change. This is legitimate and it aims at avoiding abrupt changes from the spike to the slab densities. However, in certain cases some coefficients may be forced to the spike even when it is incorrect. On the other hand, the discounted factor model for the observational variance is an estimation strategy that heavily relies on cumulated residuals through $d_t = d_{t-1} + r_t^2$. However, if the system fails to recognize some active coefficients in the first iterations of the Markov Chain, then the

residuals $r_t = y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t$ become larger and thus the variances inflate. In other words, the algorithm mistakes the fluctuations of the observations for variations of the volatility process rather than changes of the regression coefficients. Eventually, such a bias can exacerbate in a vicious cycle whereby: the system erroneously assign a variable to the spike, then the estimated residual variances inflate and the signal-to-noise ratio decreases. This leads the algorithm to become even more distrustful of the observations and it continues to assign those coefficients to the spike.

Nevertheless, this undesirable mechanism can be fixed. In particular, our proposal is the following. Since the first iterations are the most critical we recommend to fix the variance at a very low level for the first loops in such a way that the algorithm strives to understand which variable can produce the fluctuation observed and then continue with the MCMC as described. Even though this solution is quite heuristic, we notice that forcing $\sigma^2_{\epsilon,t} = 0.25$ for $t = 1, ..., T$ for just the first 10 loops improves significantly the estimates. In addition, we developed an alternative Dynamic SSVS which is based on another strategy for estimating the volatility process and that we found out to return better performances in terms of accuracy of the estimates and running time. More details about this method are provided in the next paragraph.

2.2.1 Introducing stochastic volatility with Dynamic Spike-and-Slab process priors

Interestingly, the nature of the MCMC described in Algorithm 3 allows for a great flexibility in the way the volatility can be modeled. Indeed, observational variances are sampled only in *Step 3* and thus our focus will be on the latter, leaving the first two steps unchanged. Our proposal is to replace the discount factor model for the variance with a Stochastic Volatility (SV) model. In the discount factor model, the precision is thought as affected by a random impulse which is modeled in order to maintain the stability of the Gamma function. On the other hand a stochastic volatility model assumes the volatility to be a latent stationary

Algorithm 3: Dynamic SSVS by Rockova and McAlinn (2021)

Initialize $\gamma_{j,t}$ and $\sigma^2_{\epsilon,t}$ for $0 \leq t \leq T$ and $1 \leq j \leq p$ and set n_0 and d_0;

Step 1: Sample Regression Coefficients

for $1 \leq t \leq T$ **do**

Compute $\boldsymbol{a}_t = \boldsymbol{H}_t + \boldsymbol{G}_t(\boldsymbol{m}_{t-1} - \boldsymbol{H}_t)$;

Compute $\boldsymbol{R}_t = \boldsymbol{G}_t \boldsymbol{C}_{t-1} \boldsymbol{G}_t' + \boldsymbol{\Lambda}_t$;

Compute $f_t = \boldsymbol{x}_t' \boldsymbol{a}_t$ and $e_t = y_t - f_t$;

Compute $q_t = \boldsymbol{x}_t' \boldsymbol{R}_t \boldsymbol{x}_t + \sigma^2_{\epsilon,t}$;

Compute $\boldsymbol{m}_t = \boldsymbol{a}_t + \boldsymbol{A}_t e_t$ and $\boldsymbol{C}_t = \boldsymbol{R}_t - \boldsymbol{A}_t \boldsymbol{A}_t' q_t$ with $\boldsymbol{A}_t = \boldsymbol{R}_t \boldsymbol{x}_t / q_t$

Draw $\boldsymbol{\beta}_t \sim \mathcal{N}(\boldsymbol{m}_T, \boldsymbol{C}_t)$;

for $t = T-1, ..., 0$ **do**

Compute $\boldsymbol{a}_T(t-T) = \boldsymbol{m}_t + \boldsymbol{B}_t(\boldsymbol{\beta}_{t+1} - \boldsymbol{a}_{t+1})$;

Compute $\boldsymbol{R}_t(t-T) = \boldsymbol{C}_t - \boldsymbol{B}_t \boldsymbol{R}_{t+1} \boldsymbol{B}_t'$ where $\boldsymbol{B}_t = \boldsymbol{C}_t \boldsymbol{G}_{t+1}' \boldsymbol{R}_{t+1}^{-1}$;

Draw $\boldsymbol{\beta}_t \sim \mathcal{N}(\boldsymbol{a}_T(t-T), \boldsymbol{R}_t(t-T))$;

Step 2: Sample Indicators

for $j = 1, ..., p$ **do**

for $1 \leq t \leq T$ **do**

Compute $\omega_{j,t} = \omega(\beta_{j,t-1})$ from ;

Compute $p^\star_{t,j} = p^\star_{t,j}(\beta_{j,t})$ from ;

Draw $\gamma_{j,t} \sim Bernoulli(p^\star_{j,t}(\beta_{j,t}))$;

Compute $p^\star_{j,0} = \omega(\beta_{j,0})$;

Draw $\gamma_{j,0} \sim Bernoulli(p^\star_{j,0}(\beta_{j,0}))$;

Step 3: Sample Volatility

for $t = 1, ..., T$ **do**

Compute $n_t = \delta n_{t-1} + 1$ and $d_t = \delta d_{t-1} + r_t^2$, where $r_t = y_t - \boldsymbol{x}_t' \boldsymbol{\beta}_t$;

Draw $\nu_T \sim \mathcal{G}(n_T/2, d_T/2)$;

for $t = T-1, ..., 0$ **do**

Draw $\eta_t \sim \mathcal{G}((1-\delta)n_t/2, d_t/2)$;

Set $\nu_t = \eta_t + \delta \nu_{t+1}$;

Compute $\sigma^2_{\epsilon,t} = \frac{1}{\nu_t}$;

autoregressive process of unknown parameters $(\alpha_0, \alpha_1, \sigma_\zeta^2)$, with $|\alpha_1| < 1$, which are usually considered as random variables. These class of models has been widely explored and a great amount of literature has been produced. Given its popularity in the financial econometrics literature we believe that a stochastic volatility model is of interest and it is particularly appropriate for the applications discussed in this thesis. In details, the specification for the variance here considered is the canonical one reported in the influential article of Kim, Shephard, and Chib (1998)

$$\begin{aligned} \sigma_{\epsilon,t}^2 &= \exp^{h_t}, \\ h_t &= \alpha_0 + \alpha_1 h_{t-1} + \zeta_t, \quad \zeta_t \overset{iid}{\sim} \mathcal{N}(0, \sigma_\zeta^2) \end{aligned}$$

Therefore, the new hierarchical setup is

$$\begin{aligned} y_t|\boldsymbol{x}'_t\boldsymbol{\beta}_t, h_t &\overset{ind}{\sim} \mathcal{N}(\boldsymbol{x}'_t\boldsymbol{\beta}_t, \exp h_t) \\ \beta_{t,j}|\gamma_{t,j}, \beta_{t-1,j} &\overset{ind}{\sim} \mathcal{N}(\gamma_{t,j}\mu_{t,j}, \gamma_{t,j}\lambda_1 + (1-\gamma_{t,j})\lambda_0) \\ \gamma_{t,j}|\beta_{t-1,j} &\overset{ind}{\sim} Bernoulli(\omega(\beta_{t-1,j})) \\ \beta_{0,j} &\overset{iid}{\sim} \mathcal{N}(m_0, C_0) \\ h_t|h_{t-1}, \alpha_0, \alpha_1, \sigma_\zeta^2 &\sim \mathcal{N}(\alpha_0 + \alpha_1(h_{t-1} - \alpha_0), \sigma_\zeta^2) \\ h_0|\alpha_0, \alpha_1, \sigma_\zeta^2 &\sim \mathcal{N}\left(\alpha_0, \frac{\sigma_\zeta^2}{1-\alpha_1}\right) \end{aligned} \tag{2.14}$$

Note that by taking the residuals, the resulting system is

$$\begin{aligned} r_t &= e^{\frac{h_t}{2}}\epsilon_t, \quad \epsilon_t \overset{iid}{\sim} \mathcal{N}(0,1) \\ h_t &= \alpha_0 + \alpha_1(h_{t-1} - \alpha_0) + \zeta_t, \quad \zeta_t \overset{iid}{\sim} \mathcal{N}(0,1) \end{aligned} \tag{2.15}$$

which is a standard stochastic volatility model.

Over the years, several methods have been developed to estimate this model. Here, for pragmatism and efficiency, we use the MCMC method developed of Kastner and Frühwirth-Schnatter (2014). The steps described henceforth replace the steps illustrated in *Step 3* of the original DSS algorithm. The latter considers the following prior specifications for the unknown stochastic volatility model's parameters:

$$\alpha_0 \sim \mathcal{N}(a_\alpha, A_\alpha),$$

$$(\alpha_1 + 1)/2 \sim \mathcal{B}(a_0, b_0),$$
$$\pm\sqrt{\sigma_\zeta^2} \sim \mathcal{N}(0, A_{\sigma_\zeta})$$

A recommended choice for the hyperparameters in our problem is too set them in such a way that the stochastic volatility process would not show abrupt changes. For example we can guess: $\alpha_0 \sim \mathcal{N}(-2, 10)$, $(\alpha_1 + 1)/2 \sim \mathcal{B}(20, 1.5)$ such that $\mathbb{E}(\alpha_1) = 0.86$ and $\mathbb{V}(\alpha_1) = 0.11$, and $A_{\sigma_\zeta} = 1$ or even less. Draws from the posterior distributions are obtained using a fast simulation procedure provided by the `stochvol` package (Kastner 2016). The rapidity of the algorithm is guaranteed by sampling volatilities jointly "all without a loop" (AWOL), a techinque which is exhaustively described in McCausland, Miller, and Pelletier (2011), and by using the "ancillarity-sufficiency interweaving strategy" (ASIS) developed by Yu and Meng (2011). In a nutshell, the ASIS strategy is based on a simple intuition, i.e. every SV models can be written in a *centered parametrization (C)* form, which is the one of equation (2.15), or alternatively in a *non-centered parametrization (NC)* form, that is

$$\begin{aligned} r_t &\sim \mathcal{N}(0, e^{\mu} e^{\sigma \tilde{h}_t}), \\ \tilde{h}_t &= \alpha_1 \tilde{h}_{t-1} + \sigma_\zeta \zeta_t, \quad \zeta_t \overset{iid}{\sim} \mathcal{N}(0, 1) \end{aligned} \tag{2.16}$$

where $\tilde{h}_t = (h_t - \alpha_0)/\sigma_\zeta$. It is immediate to notice that $h_{1:T}$ is a sufficient statistic for α_0 and α_1 in C, while $\tilde{h}_{1:T}$ in NC is ancillary for the same parameters. Therefore, sampling can be improved by interwaving between the two parametrizations. Yu and Meng (2011) link this results to the Basu's theorem, which demonstrates independence among complete sufficient and ancillary statistics. Operationally, the strategy consists in sampling $\alpha_0, \alpha_1, \sigma_\zeta^2$ twice: once in C and once in NC. Therefore, let $\tilde{r}_t = \log r_t{}^2$ and let's approximate $\tilde{\epsilon} = \log \epsilon_t^2$ with a mixture of normal distributions such that $\log \epsilon_t^2 | i_t \overset{iid}{\sim} \mathcal{N}(m_{i_t}, s_{i_t}^2)$, where $i_t \in \{1, ..., 10\}$ is a mixture component indicator at period t, then consider the following linear and conditionally Gaussian approximation of a SV model:

$$\tilde{r}_t = m_{i_t} + h_t + \tilde{\epsilon}_t, \quad \tilde{\epsilon}_t \overset{iid}{\sim} \mathcal{N}(0, s_{i_t}^2)$$

A MCMC algorithm to simulate from

$$\tilde{r}_t = m_{i_t} + h_t + \tilde{\epsilon}_t, \quad \tilde{\epsilon}_t \overset{iid}{\sim} \mathcal{N}(0, s_{i_t}^2),$$

$$h_t = \alpha_0 + \alpha_1(h_{t-1} - \alpha_0) + \zeta_t, \quad \zeta_t \overset{iid}{\sim} \mathcal{N}(0,1),$$

can be performed in this way:

1. Sample $h_{1:T}$ AWOL from $h_{1:T} \mid r_t, i_t, \alpha_0, \alpha_1, \sigma_\zeta^2 \sim \mathcal{N}(\Xi^{-1}\boldsymbol{c}, \Xi^{-1})$ in C, where

$$\Xi = \begin{pmatrix} \frac{1}{s^2_{i_1}} + \frac{1}{\sigma^2_\zeta} & \frac{-\alpha_1}{\sigma^2_\zeta} & 0 & \dots & 0 \\ \frac{-\alpha_1}{\sigma^2_\zeta} & \frac{1}{s^2_{i_1}} + \frac{1+\alpha_1^2}{\sigma^2_\zeta} & \frac{-\alpha_1}{\sigma^2_\zeta} & \ddots & \vdots \\ 0 & \frac{-\alpha_1}{\sigma^2_\zeta} & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \frac{1}{s^2_{i_{T-1}}} + \frac{1+\alpha_1}{\sigma^2_\zeta} & \frac{-\alpha_1}{\sigma^2_\zeta} \\ 0 & \cdots & 0 & \frac{-\alpha_1}{\sigma^2_\zeta} & \frac{1}{s^2_{i_T}} + \frac{1}{\sigma^2_\zeta} \end{pmatrix}$$

and

$$\boldsymbol{c} = \begin{pmatrix} \frac{1}{s^2_{i_1}}(\tilde{r}_1 - m_{i_1}) + \frac{\alpha_0(1-\alpha_1)}{\sigma^2_\zeta} \\ \frac{1}{s^2_{i_2}}(\tilde{r}_2 - m_{i_2}) + \frac{\alpha_0(1-\alpha_1)^2}{\sigma^2_\zeta} \\ \vdots \\ \frac{1}{s^2_{i_{T-1}}}(\tilde{r}_{T-1} - m_{i_{T-1}}) + \frac{\alpha_0(1-\alpha_1)^2}{\sigma^2_\zeta} \\ \frac{1}{s^2_{i_T}}(\tilde{r}_2 - m_{i_T}) + \frac{\alpha_0(1-\alpha_1)}{\sigma^2_\zeta} \end{pmatrix}$$

and sample h_0 from $\pi(h_0 \mid h_1, \alpha_1, \sigma_\zeta^2)$.

2. Sample $\alpha_0, \alpha_1, \sigma_\zeta^2$. In C, it is possible to draw the parameters jointly from $\pi(\alpha_0, \alpha_1, \sigma_\zeta^2 \mid h_{1:T})$ using a single Metropolis-Hastings step. Let $\rho = (1-\alpha_1)\alpha_0$ and $\boldsymbol{\psi} = (\alpha_0, \alpha_1, \sigma_\zeta^2)$, the proposal density can be wisely chose as

$$\pi_{aux}(\boldsymbol{\psi}^{(\text{new})} \mid h_{0:T}) = \pi_{aux}(\rho^{(\text{new})}, \alpha_1^{(\text{new})} \mid h_{0:T}, \sigma_\zeta^{2(\text{new})})\pi_{aux}(\sigma_\zeta^{2(\text{new})} \mid h_{0:T})$$

and $\pi_{aux}(\sigma_\zeta^2) \propto \sigma_\zeta^{-1}$ and $\pi_{aux}(\rho, \alpha_1 \mid \sigma_\zeta^2) \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{B}_0)$ with $\boldsymbol{B}_0 = diag(B_0^{11}, B_0^{22})$. Therefore, the posteriors are

$$\rho, \alpha_1 \mid h_{0:T}, \sigma_\zeta^2 \sim \mathcal{N}(\boldsymbol{b}_T, \sigma_\zeta^2\boldsymbol{B}_T)$$

with $\boldsymbol{B}_T = (\boldsymbol{H}'\boldsymbol{H} + \boldsymbol{B}_0^{-1})^{-1}$ and $\boldsymbol{b}_T = \boldsymbol{B}_T\boldsymbol{H}'h'_{1:T}$, where $\boldsymbol{H}$ is a $T \times 2$ matrix $\boldsymbol{H} = (\boldsymbol{1}'_T, h'_{0:T-1})$, and

$$\sigma_\zeta^2 \mid h_{0:T} \sim \mathcal{IG}(c_T, C_T)$$

with $c_T = (T-1)/2$ and $C_T = \frac{1}{2}(\sum_{t=1}^{T} h_t^2 - \boldsymbol{b}_T' \boldsymbol{H}' h_{1:T}')$. The acceptance probability is $\min(1, R)$ where

$$R = \frac{\pi(h_0 \mid \boldsymbol{\psi}^{(\text{new})})\pi(\rho^{(\text{new})} \mid \alpha_1^{(\text{new})})\pi(\sigma_\zeta^{2(\text{new})})}{\pi(h_0 \mid \boldsymbol{\psi}^{(\text{old})})\pi(\rho^{(\text{old})} \mid \alpha_1^{(\text{old})})\pi(\sigma_\zeta^{2(\text{old})})} \times \frac{\pi_{aux}(\alpha_1^{(\text{old})}, \rho^{(\text{old})} \mid \sigma_\zeta^{2(\text{old})})}{\pi_{aux}(\alpha_1^{(\text{new})}, \rho^{(\text{new})} \mid \sigma_\zeta^{2(\text{new})})}.$$

Alternatively, one can use a 2-block sampler that draw from $\pi(\sigma_\zeta^2 \mid h_{1:T}, \alpha_0, \alpha_1)$ and $\pi(\alpha_0, \alpha_1 \mid h_{1:T}, \sigma_\zeta^2)$, or sample them individually from their full conditional distributions.

3. Shift to NC using the transformation $\tilde{h}_t = (h_t - \alpha_0)/\alpha_1$ for $t = 1, ..., T$.

4. Draw again $\alpha_0, \alpha_1, \sigma_\zeta^2$. In NC, use Metropolis-Hastings for sampling from $\pi(\alpha_1 \mid \tilde{h}_{1:T})$ and then sample α_0 and σ_ζ^2 jointly from $\pi(\alpha_0, \sigma_\zeta^2 \mid r_{1:T}, \tilde{h}_{1:T}, i_{1:T})$. In details, to sample α_1, which is the only parameter in the state equation, one can use an improper auxiliary prior $\pi_{aux}(\alpha_1) \propto c$, with posterior

$$\alpha_1 \mid \tilde{h}_{0:T} \sim \mathcal{N}\left(\frac{\sum_{t=0}^{T-1} \tilde{h}_t \tilde{h}_{t+1}}{\sum_{t=0}^{T-1} \tilde{h}_t^2}, \frac{1}{\sum_{t=0}^{T-1} \tilde{h}_t^2}\right)$$

and acceptance probability $\min(1, R)$, where

$$R = \pi(\tilde{h}_0 \mid \alpha_1^{(\text{new})})\pi(\alpha_1^{(\text{new})})/\pi(\tilde{h}_0 \mid \alpha_1^{(\text{old})})\pi(\alpha_1^{(\text{old})}).$$

Then, samples of α_0 and σ_ζ^2 are obtained starting from the regression model

$$\hat{\boldsymbol{r}} = \boldsymbol{H}\begin{pmatrix}\alpha_0 \\ \sigma_\zeta^2\end{pmatrix} + \boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_T)$,

$$\hat{\boldsymbol{r}} = \begin{pmatrix}(\tilde{r}_1 - m_{i_1})/s_{i_1} \\ \vdots \\ (\tilde{r}_T - m_{i_T})/s_{i_T}\end{pmatrix}$$

and

$$\boldsymbol{H} = \begin{pmatrix}\tilde{h}_1/s_{i_1} & 1/s_{i_1} \\ \vdots & \\ \tilde{h}_T/s_{i_T} & 1/s_{i_T}\end{pmatrix}.$$

The posterior distribution $\pi(\alpha_0, \sigma_\zeta^2 \mid r_{1:T}, \tilde{h}_{0:T}, i_{1:T})$ is thus $\mathcal{N}(\boldsymbol{b}_T, \boldsymbol{B}_T)$ with $\boldsymbol{B}_T = (\boldsymbol{H}'\boldsymbol{H} + \boldsymbol{B}_0^{-1})^{-1}$ and $\boldsymbol{b}_T = \boldsymbol{B}_T(\boldsymbol{B}_0^{-1}\boldsymbol{\beta}_0 + \boldsymbol{H}'\boldsymbol{r})$, where $\boldsymbol{b}_0 = (b_{\alpha_0}, 0)'$ and

$\boldsymbol{B}_0 = diag(B_{\alpha_0}, B_{\sigma_\zeta})$. Alternatively, one can sample $(\alpha_0, \sigma_\zeta^2)$ individually form $\pi(\alpha_0 \mid r_{1:T}, \tilde{h}_{1:T}, i_{1:T}, \sigma_\zeta^2)$ and $\pi(\sigma_\zeta^2 \mid r_{1:T}, \tilde{h}_{1:T}, i_{1:T}, \alpha_0)$.

5. Return to C by computing $h_t = \alpha_0 + \sigma_\zeta \tilde{h}_t$ for $t = 1, ..., T$

6. Draw i_t from $\pi(i_t \mid r_{1:T}, h_{1:T})$ in C. Given that $\tilde{r}_t - h_t = \epsilon_t^*$, with $\epsilon_t^* \sim \mathcal{N}(m_{i_t}, s_{i_t}^2)$, then the posterior probability of $i_{1:T} \mid r_{1:T}, h_{0:T}$ is

$$\mathbb{P}(i_t = k | r_{1:T}, h_{0:T}) \propto \mathbb{P}(i_t = k) \frac{1}{s_k} \exp\left\{ -\frac{(\epsilon^* - m_k)^2}{2s_k^2} \right\}$$

for $k \in \{1, ..., 10\}$ and $t \in \{1, ..., T\}$, where $\mathbb{P}(i_t = k)$ are the mixture weights of the kth component.

So far, we described the procedure starting in C, however it is also possible to start in NC, then move to C and finally return back to NC. For more details on the AWOL-ASIS strategy, the reference is Kastner and Frühwirth-Schnatter (2014).

2.2.2 Speeding up MCMC with a precision sampler

In order to speed up the Dynamic SSVS, we also propose an alternative estimation and simulation technique for the regression coefficients. As already mentioned in Chapter 1, the computational complexity of the Kalman filter is linear in the length of the data and quadratic in the dimension of the state vector. This turns out to be a downside in the context we are considering which is characterized by several latent processes, as many as the predictors inside the model. The AWOL method to sample from the posterior distributions of the SV model allows to slightly reduce the running time, however the major computational effort comes from the FFBS step for the regression coefficients. Therefore, we decide to try a different approach for sampling the $\boldsymbol{\beta}_{0:T}$, namely the precision sampler of Chan and Jeliazkov (2009). The latter is based on sparse matrix routines and it has been developed for multivariate time series. We describe it using the notation of the authors which will be resumed in Chapter 4. In particular we follow Chan and Eisenstat (2018). In fact, the Gibbs sampler proposed here extends the original precision sampler, in taking into account the dynamic shrinkage

Algorithm 4: Dynamic Shrinkage in the Precision Sampler of Chan and Jeliazkov (2009)

1 Sample from $\pi(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{h}, \boldsymbol{\gamma}, \boldsymbol{\theta}_0)$;

2 Sample from $\pi(\boldsymbol{\theta}_0|\boldsymbol{y}, \boldsymbol{h}, \boldsymbol{\gamma}, \boldsymbol{\theta})$;

3 Sample individually and independently from $\pi(\gamma_{j,t}|\boldsymbol{y}, \boldsymbol{h}, \boldsymbol{\theta}, \boldsymbol{\theta}_0, \gamma_{-j,t})$;

4 Compute $\boldsymbol{\Lambda} = diag\{\boldsymbol{\gamma}\lambda_1 + (\mathbf{1} - \boldsymbol{\gamma})\lambda_0\}$;

5 Compute $\boldsymbol{D}$ as in equation (2.17);

6 Sample from $\pi(\boldsymbol{h}|\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{\gamma}, \boldsymbol{\theta}_0, \boldsymbol{\Psi})$;

expressed through the DSS process prior. Let $\boldsymbol{y} = (\boldsymbol{y}_1', ..., \boldsymbol{y}_T')'$, where $\boldsymbol{y}_t = (y_{t,1}, ..., y_{t,n})'$, and $\boldsymbol{\theta} = (\boldsymbol{\theta}_1', ..., \boldsymbol{\theta}_T')'$, where in the TVP Regression model $\boldsymbol{\theta} = \boldsymbol{\beta}$ and $\boldsymbol{\theta}_t = (\theta_{t,1}, ..., \theta_{t,p})'$ The Gibbs sampling follows the scheme 4.

Note: in Algorithm 4, the sampling of $\boldsymbol{\Psi} = \{\alpha_{0,i}, \alpha_{1,i}, \sigma^2_{\zeta,i}\}_{i=1}^n$ is implicit, but it consists of the same steps described in Section 2.2.1. Because of the large number of parameters to be estimated in multivariate time series, it seems a reasonable simplifying assumption to use a random walk for the volatility, i.e. $\boldsymbol{h}_t = \boldsymbol{h}_{t-1} + \boldsymbol{\zeta}_t$ with $\boldsymbol{\zeta}_t \sim \mathcal{N}(\mathbf{0}, \Sigma_\zeta)$ and $\Sigma_\zeta = diag\{\sigma_i^2\}_{i=1}^n$. Let's focus on each step.

- Step 1; Let us recall the state space representation of a TVP-VAR model

$$\underset{(T\times n)\times 1}{\boldsymbol{y}} = \underset{(T\times n)\times p}{\boldsymbol{X}} \quad \underset{p\times 1}{\boldsymbol{\theta}} + \underset{(T\times n)\times 1}{\boldsymbol{\epsilon}}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\left(\underset{(T\times n)\times 1}{\mathbf{0}}, \underset{(T\times n)\times(T\times n)}{\boldsymbol{\Sigma}}\right)$$

  where $\boldsymbol{\epsilon} = (\boldsymbol{\epsilon}_1', ..., \boldsymbol{\epsilon}_T')'$, $\boldsymbol{\Sigma} = diag(\Sigma_{\epsilon,1}, ..., \Sigma_{\epsilon,T})$ and $\boldsymbol{X} = diag(\boldsymbol{X}_1, ..., \boldsymbol{X}_T)$, and the latent process of the stochastic coefficients evolves as

$$\boldsymbol{\theta}_t = \boldsymbol{G}_t\boldsymbol{\theta}_{t-1} + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \Lambda_t)$$

  where in the DSS scheme $\boldsymbol{G}_t = diag\{\gamma_{j,t}\phi_1\}_{j=1}^p$ and $\Lambda_t = diag\{\gamma_{j,t}\lambda_1 + (1 - \gamma_{j,t})\lambda_0\}_{j=1}^p$. Define the matrix

$$\boldsymbol{D} = \begin{pmatrix} \boldsymbol{I}_k & 0 & ... & 0 \\ -\boldsymbol{G}_1 & \boldsymbol{I}_k & ... & 0 \\ ... & ... & ... & 0 \\ 0 & ... & -\boldsymbol{G}_T & \boldsymbol{I}_k \end{pmatrix} \tag{2.17}$$

Therefore we can write

$$\boldsymbol{D\theta} = \tilde{\boldsymbol{\alpha}}_0 + \boldsymbol{\xi}, \quad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{S_\theta})$$

where $\tilde{\boldsymbol{\alpha}}_0 = (\boldsymbol{\theta}_0', \mathbf{0}, ...\mathbf{0})'$ and $\boldsymbol{S_\theta} = diag(\Lambda_1, ..., \Lambda_T)$. Equivalently we write

$$\boldsymbol{\theta}|\boldsymbol{\theta}_0, \boldsymbol{\gamma} \sim \mathcal{N}(\boldsymbol{D}^{-1}\tilde{\boldsymbol{\alpha}}_0, (\boldsymbol{D}'\boldsymbol{S_\theta}^{-1}\boldsymbol{D})^{-1})$$

and we label $\boldsymbol{\alpha}_0 = \boldsymbol{D}^{-1}\tilde{\boldsymbol{\alpha}}_0$. Thanks to Corollary 8.1 of Theorem 8.1 of Kroese and Chan (2014), that we mentioned in Section 1.1, we can sample from

$$\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{h}, \boldsymbol{\gamma}, \boldsymbol{\theta}_0, \boldsymbol{h}_0 \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \boldsymbol{K}_{\boldsymbol{\theta}}^{-1})$$

where $\hat{\boldsymbol{\theta}} = \boldsymbol{K}_{\boldsymbol{\theta}}^{-1}\boldsymbol{d}_{\boldsymbol{\theta}}$, $\boldsymbol{K_\theta} = \boldsymbol{D}'\boldsymbol{S_\theta}^{-1}\boldsymbol{D} + \boldsymbol{X}'\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{X}$ and $\boldsymbol{d_\theta} = \boldsymbol{D}'\boldsymbol{S_\theta}^{-1}\boldsymbol{D\alpha}_0 + \boldsymbol{X}'\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{y}$.

- Step 2; Sample $\boldsymbol{\theta}_0$ from the full conditional distribution

$$(\boldsymbol{\theta}_0|\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{h}, \boldsymbol{\Lambda}_0) \sim \mathcal{N}(\hat{\boldsymbol{\theta}}_0, \boldsymbol{K_{\theta_0}}^{-1})$$

where $\boldsymbol{K_{\theta_0}} = \boldsymbol{C}_0^{-1} + \Lambda_0^{-1}$ and $\hat{\boldsymbol{\theta}}_0 = \boldsymbol{K_{\theta_0}}^{-1}(\boldsymbol{C}_0^{-1}\boldsymbol{m}_0 + \Lambda_0^{-1}\boldsymbol{\theta}_1)$, with $\boldsymbol{m}_0$ and C_0 respectively the prior mean and the prior variance of the state vector.

- Step 3; Sample individually and independently the indicators $\gamma_{j,t}$ from their full conditional distribution as in Step 2 of Algorithm 3.

- Step 6; Compute $\boldsymbol{r} = \boldsymbol{y} - \boldsymbol{X\theta}$ and use the residuals $\boldsymbol{r}_i = (r_{1,i}, ..., r_{T,i})'$ (for $i = 1, ..., n$) to perform the AWOL-ASIS strategy of Kastner (2016).

In alternative to Kastner and Frühwirth-Schnatter (2014), the authors of the precision sampler use the computational strategy of Kim, Shephard, and Chib (1998). In this case it is also necessary to sample $\boldsymbol{h}_0$ from the full conditional distribution

$$(\boldsymbol{h}_0|\boldsymbol{y}, \boldsymbol{h}, \boldsymbol{\theta}, \boldsymbol{\Lambda}_0, \Sigma_\zeta) \sim \mathcal{N}(\hat{\boldsymbol{h}_0}, \boldsymbol{K_{h_0}}^{-1})$$

where $\boldsymbol{K_{h_0}} = \boldsymbol{V}_h^{-1} + \Sigma_\zeta^{-1}$ and $\hat{\boldsymbol{h}_0} = \boldsymbol{K_{h_0}}^{-1}(\boldsymbol{V}_h^{-1}\boldsymbol{a}_h + \Sigma_\zeta^{-1}\boldsymbol{h}_1)$, and $\boldsymbol{a}_h$ and $\boldsymbol{V}_h$ are the prior mean and variance of the initial volatility state.

2.3 Dynamic Expectation-Maximization Variable Selection

A valuable alternative to simulation algorithms come from Rockova and McAlinn (2021a). They propose an innovative optimization approach to variable selection based on an Expectation-Maximization (EM) algorithm which is labelled by the authors Dynamic Expectation-Maximization Variable Selection or, more briefly, Dynamic EMVS. This method aims at providing the Maximum A Posteriori (MAP) estimate $\hat{\boldsymbol{\beta}}_{1:T} = \arg\max \pi(\boldsymbol{\beta}_{1:T}|y_{1:T})$ by reducing this challenging maximization problem into a sequence of simpler maximization steps. Each iteration is indeed decomposed into two stages. In the first step, known as E-Step, the conditional expectations of the unknown model's parameters is computed given the observations and the other parameters of interest. The second step, or M-step, consists in maximizing the joint conditional expectation of the model's parameters given the observations, where the other unknown model's parameters are treated as missing values and are replaced by their conditional expectation. In our specific case, these two passages become:

- *E-step:* Compute $\mathbb{E}(\boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,1:T} \mid \boldsymbol{\beta}^{(m)}_{0:T}, y_{1:T})$, where $\boldsymbol{\beta}^{(m)}_{0:T}$ indicates the regression coefficients sequence at the m^{th} iteration.
- *M-step:* Maximize $\mathbb{E}_{\boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,1:T}|\cdot} \log \pi(\boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,1:T} \mid y_{1:T})$ with respect to $\boldsymbol{\beta}_{0:T}$.

In details, since $\boldsymbol{\gamma}_{0:T}$ and $\sigma^2_{\epsilon,1:T}$ are conditionally independent, we compute $\mathbb{E}(\boldsymbol{\gamma}_{0:T} \mid \boldsymbol{\beta}_{0:T}, y_{1:T})$ and $\mathbb{E}(\sigma^2_{\epsilon,1:T} \mid \boldsymbol{\beta}_{0:T}, y_{1:T})$. The first boils down in computing $p^\star_{t,j} = \mathbb{P}(\gamma_{t,j} = 1 \mid \beta^{(m)}_{t,j}, \beta^{(m)}_{t-1,j}, \Omega)$ for $t = 0, ..., T$ and $j = 1, ..., p$. The latter depends on the specification of the variances used in the model. For example, considering the discount factor model for the volatility of equation (2.12), it can be shown (West and Harrison 1997) that

$$\mathbb{E}(\nu_t \mid \boldsymbol{\beta}^{(m)}_{0:T}, y_{1:T}) = (1-\delta)n_t/d_t + \delta\mathbb{E}(\nu_{t+1} \mid \boldsymbol{\beta}^{(m)}_{0:T}, y_{1:T})$$

for $1 \leq t < T$, and $\mathbb{E}(\nu_t \mid \boldsymbol{\beta}^{(m)}_{0:T}, y_{1:T}) = n_T/d_T$. On the other hand, the non-linearity introduced by specifying the volatility as a log-Normal AR(1) process does not allow to obtain first

moments in closed form. For this reason, we developed a novel approach characterized by particle filtering and smoothing in the E-step. At the cost of a slight increase in the running time, the new algorithm provides with better volatility estimates. More details of this methods are provided in the next section.

The M-step requires some considerations too. Here, we describe the Gaussian Spike-and-Slab case; for the Laplace case we refer to Rockova and McAlinn (2021a). The initial state vector, $\boldsymbol{\beta}_0$, is estimated with the whole state sequence $\boldsymbol{\beta}_{1:T}$ and, according to the DSS specification, it is assumed to have a stationary distribution

$$\pi(\boldsymbol{\beta}_0|\boldsymbol{\gamma}_0) = \prod_{j=1}^{p}[\gamma_{0,j}\psi_1^{ST}(\beta_{0,j}|\lambda_1,\phi_0,\phi_1) + (1-\gamma_{0,j})\psi_0(\beta_{0,j}|\lambda_0)] \tag{2.18}$$

where $\gamma_{0,j}|\Omega \sim Bernoulli(\Omega)$ for $j = 1,...,p$. Notice that the Markov structure of the models parameters and their independence allow the following factorization

$$\pi(\boldsymbol{\beta}_{0:T},\boldsymbol{\gamma}_{0:T},\sigma^2_{\epsilon,1:T}) = \pi(\boldsymbol{\beta}_0|\boldsymbol{\gamma}_0)\pi(\boldsymbol{\gamma}_0)\prod_{t=1}^{T}\left[\pi(\sigma^2_{\epsilon,t}|\sigma^2_{\epsilon,t-1})\prod_{j=1}^{p}\pi(\beta_{t,j}|\gamma_{t,j},\beta_{t-1,j})\pi(\gamma_{t,j}|\beta_{t-1,j})\right] \tag{2.19}$$

Then we can write

$$\begin{aligned}
&\log\pi(\boldsymbol{\beta}_{0:T},\boldsymbol{\gamma}_{0:T},\sigma^2_{\epsilon,1:T}|y_{1:T}) \\
&= C + \sum_{t=1}^{T}\sum_{j=1}^{p}[\gamma_{t,j}\log\theta_{t,j} + (1-\gamma_{t,j})\log(1-\theta_{t,j})] \\
&- \sum_{t=1}^{T}\left\{\frac{(y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t)^2}{2\sigma^2_{\epsilon,t}} + \sum_{j=1}^{p}\left[\gamma_{t,j}\frac{(\beta_{t,j}-\phi_1\beta_{t-1,j})^2}{2\lambda_1} + (1-\gamma_{t,j})\frac{\beta^2_{t,j}}{2\lambda_0}\right] + \log\pi(\sigma^2_{\epsilon,t}|\sigma^2_{\epsilon,t-1})\right\} \\
&- \sum_{j=1}^{p}\left\{\gamma_{0,j}\frac{\beta^2_{0,j}(1-\phi_1^2)}{2\lambda_1} + (1-\gamma_{0,j})\frac{\beta^2_{0,j}}{2\lambda_0} - \gamma_{0,j}\log\Omega - (1-\gamma_{0,j})\log(1-\Omega)\right\}
\end{aligned} \tag{2.20}$$

where C incorporates all the constant components. Let $\nu_t = \frac{1}{\sigma^2_{\epsilon,t}}$, the objective function we want maximize in the M-step is $\mathbb{E}_{\boldsymbol{\gamma}_{0:T},\nu_{1:T}|\cdot}\log\pi(\boldsymbol{\beta}_{0:T},\boldsymbol{\gamma}_{0:T},\nu_{1:T}\mid y_{1:T}) = Q(\boldsymbol{\Xi}\mid y_{1:T})$. Therefore, we replace $(\boldsymbol{\gamma}_{0:T},\nu_{1:T})$ in equation (2.20) with their conditional expectations $(\boldsymbol{p}^\star_{0:T},\nu^\star_{1:T})$ computed in the E-step. Then, we find the maximum of $Q(\boldsymbol{\Xi}\mid y_{1:T})$ with respect to $\boldsymbol{\beta}_{0:T}$. Below we report the first order conditions:

$$\begin{aligned}
\frac{\partial Q(\boldsymbol{\Xi}\mid y_{1:t})}{\partial\boldsymbol{\beta}_t} = 0 \iff \boldsymbol{\beta}_t &= \boldsymbol{D}_t^{-1}\left\{\nu_t^\star y_t\boldsymbol{x}_t + \frac{\phi_1}{\lambda_1}\boldsymbol{\beta}_{t-1}\odot\boldsymbol{p}_t^\star + \frac{\phi_1}{\lambda_1}\boldsymbol{\beta}_{t+1}\odot\boldsymbol{p}_{t+1}^\star\right\} \\
\frac{\partial Q(\boldsymbol{\Xi}\mid y_{1:t})}{\partial\boldsymbol{\beta}_0} = 0 \iff \boldsymbol{\beta}_0 &= \boldsymbol{D}_0^{-1}\frac{\phi_1}{\lambda_1}\boldsymbol{\beta_1}\odot\boldsymbol{p}_{\boldsymbol{1}}^\star
\end{aligned}$$

where

$$\boldsymbol{D}_t = \nu_t^\star \boldsymbol{x}_t \boldsymbol{x}_t' + diag\left\{ \frac{p_{t,j}^\star}{\lambda_1} + \frac{1 - p_{t,j}^\star}{\lambda_0} + \frac{\phi_1^2 p_{t+1,j}^\star}{\lambda_1} \right\}_{j=1}^{p}$$

and

$$\boldsymbol{D}_0 = diag\left\{ \frac{(1-\phi_1^2) p_{0,j}^\star}{\lambda_1} + \frac{1 - p_{0,j}^\star}{\lambda_0} + \frac{\phi_1^2 p_{1,j}^\star}{\lambda_1} \right\}_{j=1}^{p}.$$

The notation $\odot$ stands for the element-wise vector multiplication.

The passages so far illustrated are resumed in algorithm (5). Note that the inversion of $\boldsymbol{D}_t$ is facilitated by the Woodburry formula (William 1989).

Labelling $\Delta = diag\left\{ \frac{p_{t,j}^\star}{\lambda_1} + \frac{1-p_{t,j}^\star}{\lambda_0} + \frac{\phi_1^2 p_{t+1,j}^\star}{\lambda_1} \right\}_{j=1}^{p}$, then it yields

$$\boldsymbol{D}_t^{-1} = \Delta_t^{-1} - \nu_t^\star \Delta_t^{-1} \frac{\boldsymbol{x}_t \boldsymbol{x}_t'}{1 + \nu_t^\star \boldsymbol{x}_t' \Delta_t^{-1} \boldsymbol{x}_t} \Delta_t^{-1}$$

Thanks to this shortcut, the overall running time of the algorithm reduces drastically.

2.3.1 Particle Smoothing for Dynamic EMVS

For the particle smoothing strategy implemented in the E-step we refer to the scheme proposed by Godsill, Doucet, and West (2004). In this section, we describe the latter with reference to model (2.14). Therefore, a stochastic volatility model is assumed for the regression residuals $r_t = y_t - \boldsymbol{x}_t' \boldsymbol{\beta}_t$ for $t = 1, \ldots, T$.

The markovian structure of the stochastic volatility model allows the following factorization

$$\begin{aligned} \pi(h_t \mid h_{t+1:T}, r_{1:T}, \boldsymbol{\psi}) &= \pi(h_t \mid h_{t+1}, r_{1:T}, \boldsymbol{\psi}) \\ &\propto \pi(h_t \mid r_{1:T}, \boldsymbol{\psi}) \pi(h_{t+1} \mid h_t, \boldsymbol{\psi}) \end{aligned} \tag{2.21}$$

where $\boldsymbol{\psi}$ indicates a vector containing other unknown model's parameters. In equation (2.21), $\pi(h_t \mid r_{1:T}, \boldsymbol{\psi})$ can be approximated by the empirical distribution computed using a particle filter. In our implementation, for example, we use a Bootstrap Particle Filter (Chopin and Papaspiliopoulos 2020). Consequently, $\pi(h_t \mid h_{t+1:T}, r_{1:T}, \boldsymbol{\psi})$ can be approximated by the empirical distribution

$$\pi(h_t \mid h_{t+1:T}, r_{1:T}, \boldsymbol{\psi}) \approx \sum_{i=1}^{N} w_{t|t+1}^{(i)} \delta_{h_t^{(i)}}$$

Algorithm 5: Dynamic EMVS by Rockova and McAllin (2021)

Initialize $\beta_{t,j}$ for $t = 0, ..., T$ and $j = 1, ..., p$;

E-Step

for $j = 1, ..., p$ **do**

  for $1 \leq t \leq T$ **do**

    Compute $\omega_{t,j} = \omega(\beta_{t-1,j})$ from ;

    Compute $p^{\star}_{t,j} = p^{\star}_{t,j}(\beta_{t,j})$ from ;

  Compute $p^{\star}_{0,j} = \omega(\beta_{0,t})$ from ;

for $t = 1, ..., T$ **do**

  Compute $n_t = \delta n_{t-1} + 1$ and $d_t = \delta d_{t-1} + r_t^2$, where $r_t = y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t$;

Set $\nu^{\star}_T = n_T/d_T$;

for $t = T - 1, ..., 0$ **do**

  Set $\nu^{\star}_t = (1-\delta)n_t/d_t + \delta\nu^{\star}_{t+1}$;

M-Step

for $t = 1, ..., T$ **do**

  Compute $\boldsymbol{D}_t = \nu^{\star}_t \boldsymbol{x}_t\boldsymbol{x}_t' + diag\left\{\frac{p^{\star}}{\lambda_1} + \frac{1-p^{\star}}{\lambda_0} + \mathbb{I}(t<T)\frac{\phi_1^2 p^{\star}_{t+1,j}}{\lambda_1}\right\}_{j=1}^{p}$;

  Compute $\boldsymbol{\beta}_t = \boldsymbol{D}_t^{-1}\left\{\nu^{\star}_t y_t \boldsymbol{x}_t + \frac{\phi_1}{\lambda_1}\boldsymbol{\beta}_{t-1} \odot \boldsymbol{p}^{\star}_t + \mathbb{I}(t<T)\frac{\phi_1}{\lambda_1}\boldsymbol{\beta}_{t+1} \odot \boldsymbol{p}^{\star}_{t+1}\right\}$;

  Compute $\boldsymbol{D}_0 = diag\left\{\frac{(1-\phi_1^2)p^{\star}_{0,j}}{\lambda_1} + \frac{1-p^{\star}_{0,j}}{\lambda_0} + \frac{\phi_1^2 p^{\star}_{1,j}}{\lambda_1}\right\}_{j=1}^{p}$;

  Compute $\boldsymbol{\beta}_0 = \boldsymbol{D}_0^{-1}\frac{\phi_1}{\lambda_1}\boldsymbol{\beta_1} \odot \boldsymbol{p}^{\star}_1$;

with modified weights

$$w_{t|t+1}^{(i)} = \frac{w_t^{(i)} \pi(h_{t+1} \mid h_t^{(i)}, \boldsymbol{\psi})}{\sum_{j=1}^{N} w_t^{(j)} \pi(h_{t+1} \mid h_t^{(j)}, \boldsymbol{\psi})}.$$

For practical reasons, we decided to implement a Bootstrap Particle Filter where the parameters $(\alpha_1, \alpha_2, \sigma_\zeta^2)$ are known. The whole filtering and smoothing process used for the applications of this thesis is presented below.

- Generate N particle $(h_0^{(1)}, ..., h_0^{(N)})$ from $\mathcal{N}(\alpha_0, \sigma_\zeta^2/(1-\alpha_1^2))$ and N weights such that $w_0^{(i)} = N^{-1}$ for $i = 1, ..., N$

- For $t = 1, ..., T$:

  1. Draw $h_t^{(i)} \sim \mathcal{N}(\alpha_0 + \alpha_1 h_{t-1}^{(i)}, \sigma_\eta^2) \quad i = 1, ..., N$
  2. Set $\tilde{w}_t^{(i)} = w_{t-1}^{(i)} \, \mathcal{N}(r_t; 0, e^{h_t^{(i)}}) \quad i = 1, ..., N$
  3. Normalize the weights: $w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^{N} (\tilde{w}_t^{(j)})}$
  4. Compute the effective sample size (ess): $ess = \left(\sum_{i=1}^{N} (w_t^{(i)})^2 \right)^{-1}$
  5. if $ess < N/2$ then
     (a) Draw a sample of size N, $(h_t^{(1)}, ..., h_t^{(N)})$, from the discrete distribution $\mathbb{P}(h_t = h_t^{(i)}) = w_t^{(i)}, \quad i = 1, ..., N$
     (b) Reset the weights: $w_t^{(i)} = N^{-1}$, $i = 1, ..., N$.

- For $t = T-1, ..., 0$

  1. Set $w_{t|t+1}^{(i)} \propto w_t \, \mathcal{N}(\tilde{h}_{t+1} | h_t^{(i)}), \quad i = 1, ..., N$
  2. Draw a sample of size N, $(\tilde{h}_t^{(1)}, ..., \tilde{h}_t^{(N)})$, from the discrete distribution $\mathbb{P}(\tilde{h}_t = h_t^{(i)}) = w_{t|t+1}^{(i)}, \quad i = 1, ..., N$

The ess-based resampling step is meant to avoid the degeneracy of the particles that may occurs in a Sequential Importance Sampling. In addition, it allows to save computational time by resampling only whether necessary. In alternative to the Bootstrap Particle Filter

we propose, it would be possible to adopt the Liu and West filter (Liu and West 2001) which is a modified version of the filtering strategy described above that provides estimates also for the unknown SV model's parameter. This is made possible by resampling them over time from a continuous distribution, that in our case coincides with a Normal distribution for α_0 and α_1 and an Inverse Gamma distribution for σ_η^2. In this way, at every time, the support of the parameters is not limited to the N initially sampled values. Since the resampling of the parameters involves additional computational efforts we decide not to take into account the Liu and West (2001) strategy here.

2.4 Simulation Study

Synthetic data are used in order to assess the validity of the algorithms illustrated in the previous sections. A sequence of $T = 144$ observations has been generated from model (2.1) with $p = 50$ explanatory variables and $\sigma_{\epsilon,t}^2 = 0.25$ for $t = 1, ..., T$. Each $x_{t,j}$ is drawn from a standard Normal distribution. Only the first four predictors contribute to explaining $y_{1:T}$, whereas the remaining forty-six are uncorrelated to the outcome. Hence, the actual coefficients associated to relevant predictors, namely $\beta_{1:T,j}^0$ for $j = 1, ..., 4$, are simulated from the $AR(1)$ process defined in equation (2.5) with hyperparameters $\lambda_1 = 0.1$, $\phi_0 = 0$ and $\phi_1 = 0.98$. The rest of the coefficients are instead forced to zero at every time: $\beta_{1:T,5}^0 = ... = \beta_{1:T,50}^0 = \mathbf{0}$. As in the simulation study presented in Rockova and McAlinn (2021a), the values are rescaled and thresholded to zero if the absolute value of the process falls below 0.5, resulting in zero-valued periods. Below we report the results obtained by fitting the data using the Dynamic SSVS and Dynamic EMVS strategies illustrated in the previous paragraphs. Posterior sampling in the Dynamic SSVS is performed via MCMC with $N = 1000$ iterations of which 200 are burn-in. We force $\sigma_{\epsilon,t}^2 = 0.25$ for the first ten iterations since we empirically notice that this simple trick avoids the degeneracy of the Markov Chain. Similarly, $N = 1000$ iterations are used for the Dynamic EMVS.

The large number of predictors and the dynamic evolution of their relevance over time make

it extremely challenging for traditional techniques to extrapolate the actual source of the signal. Evidence of this is provided in Figure 2.1, where data are fitted using a standard DLM without shrinkage. The latter can be easily obtained by fixing $\Omega = 1$, which is in practice equivalent to set $\gamma_{t,j} = 1 \ \forall \ t \in \{0, \dots, T\}, \ \forall \ j \in \{1, \dots, p\}$. As in Rockova and McAlinn (2021a), the volatility process is estimated using a discount factor model with hyperparameters $n_0 = 10$, $d_0 = 10$ and $\delta = 0.9$. The plots in Figure 2.1 show that the noise induced by unnecessary covariates produces biased point estimates with high uncertainty which results in huge credible intervals. The latter are computed as

$$\left[\mathbb{E}(\beta_{t,j}|y_{1:T}) - z_{1-\alpha/2}\sqrt{\mathbb{V}(\beta_{t,j}|y_{1:T})} \ , \ \mathbb{E}(\beta_{t,j}|y_{1:T}) + z_{1-\alpha/2}\sqrt{\mathbb{V}(\beta_{t,j}|y_{1:T})}\right].$$

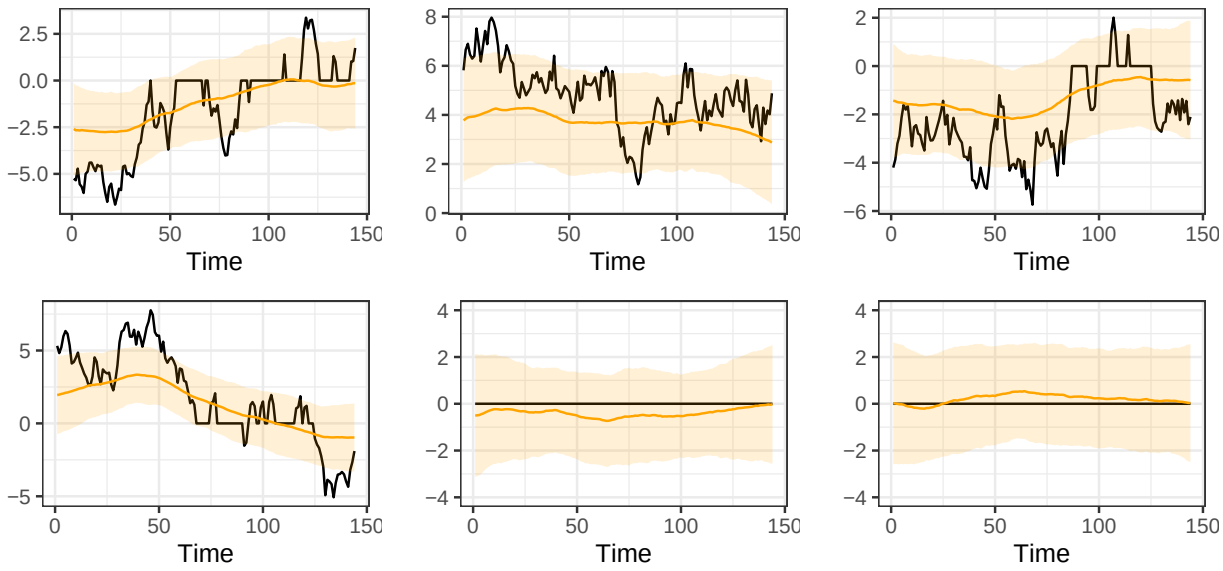


Figure 2.1: Dynamic SSVS with $\Omega = 1$ (no shrinkage) and discount factor model for the residuals; true values of $\beta_{1:T,j}$, $j = 1, \dots, 6$, (black) and smoothing estimates with 95 percent credible intervals (yellow) of the first six regression coefficients.

Sparsity is induced by lowering the value of Ω. Indeed, Dynamic SSVS with $\Omega = 0.2$ seems to lead to improved results. As shown in Figure 2.2, the new model is able to capture more features of the data. The smoothed values follow quite closely the true ones and they present smaller credible intervals. Results may eventually improve by choosing appropriate values

for the hyperparameters $(\Omega, \lambda_1, \lambda_0, n_0, d_0, \delta)$. In this specific case, we chose $\lambda_0 = 0.01$ and $\lambda_1 = 0.1$ in order to maintain a good ratio between the spike and slab variances and hence to facilitate the recognition of the active coefficients. Moreover, by setting $n_0 = 40$ and $d_0 = 10$ we want to express our prior belief of a small residual variance.

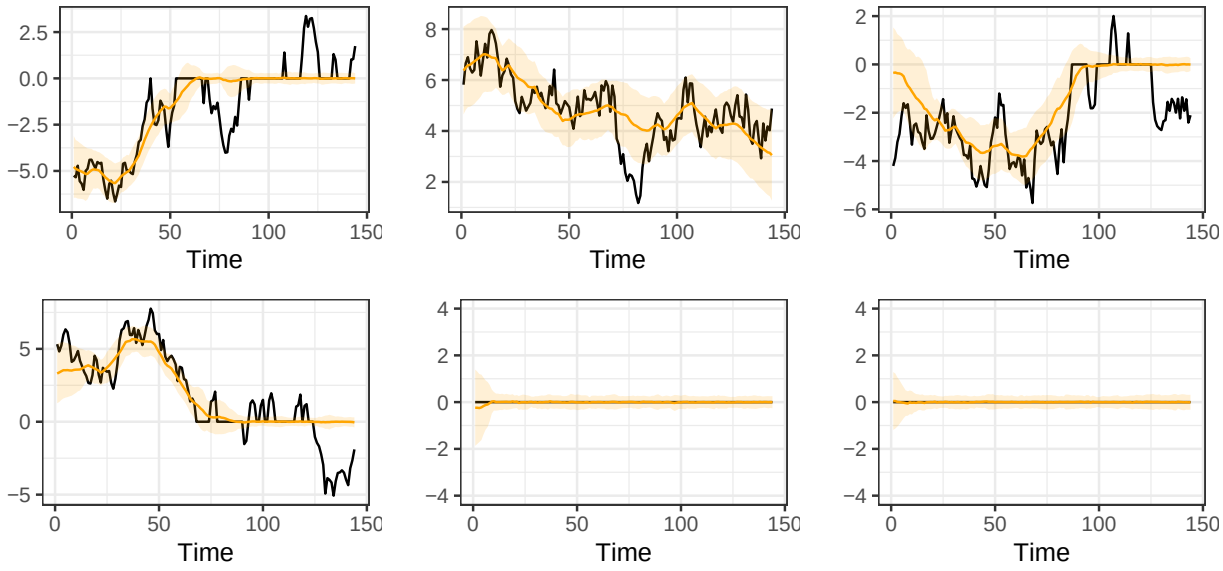


Figure 2.2: Dynamic SSVS with $\Omega = 0.2$ and discount factor model for the residuals; true values of $\beta_{1:T,j}$, $j = 1, \ldots, 6$, (black) and smoothing estimates with 95 percent credible intervals (yellow) of the first six regression coefficients.

As anticipated at the end of Section 2.2, estimates can benefit from switching from a discount factor model to a stochastic volatility model for the residual variances. We prove this statement by comparing the Dynamic SSVS with these two alternative specifications.

Therefore, we replace the third step of Algorithm 3 with the simulation strategy for stochastic volatility models discussed in Section 2.2.1. The parameters' priors are $\alpha_0 \sim \mathcal{N}(-10, 100)$, $\alpha_2 \sim \mathcal{B}(20, 1.5)$ and $\sigma_\zeta^2 \sim \mathcal{IG}(0.5, 0.5)$, and we set the initialization values $h_0^{(0)} = -2$, $\alpha_0^{(0)} = -2$, $\alpha_1^{(0)} = 0.9$ and $\sigma_\zeta^{(0)} = 0.1$. Figure 2.3 shows that there is an actual improvement when using a stochastic volatility model for the residual variances. Indeed, this model is less sensible to the "spike trap" we discussed in Section 2.2, as shown in Figure 2.4.

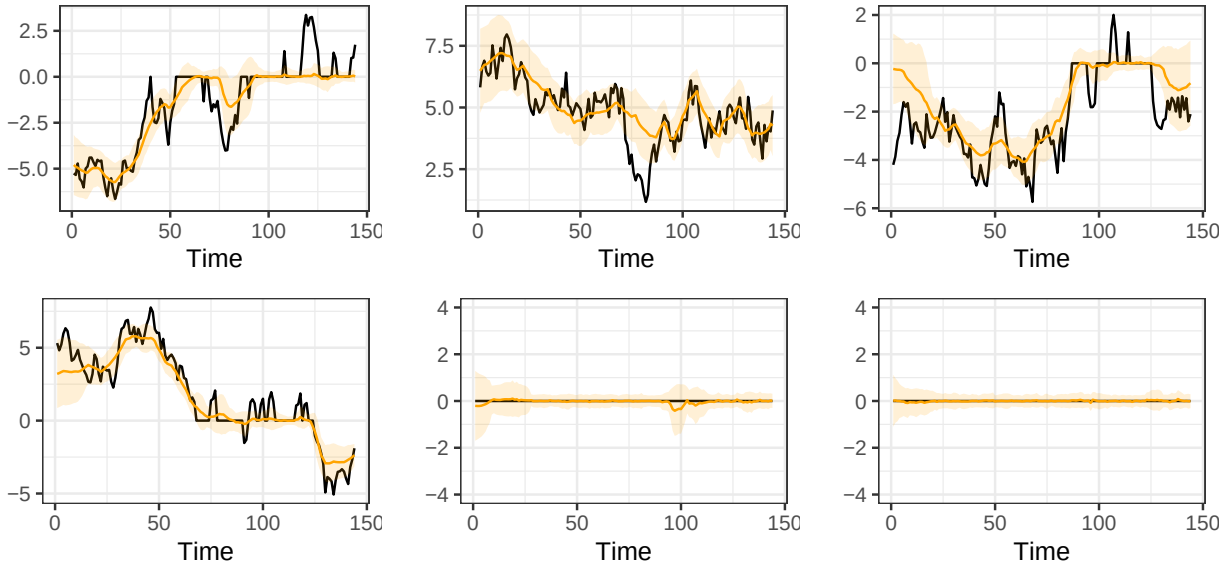


Figure 2.3: Dynamic SSVS with $\Omega = 0.2$ and stochastic volatility model for the residuals; true values of $\beta_{1:T,j}$, $j = 1, \ldots, 6$, (black) and smoothing estimates with 95 percent credible intervals (yellow) of the first six regression coefficients.

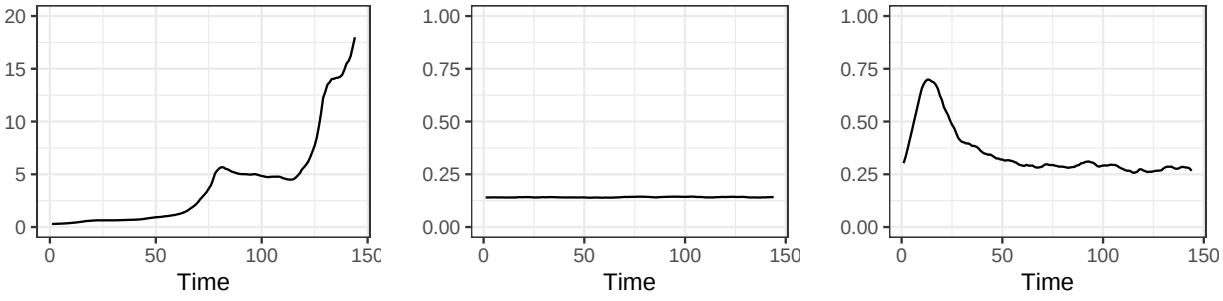


Figure 2.4: MCMC variances' estimates of the volatility process with: a discount factor model (left panel), stochastic volatility model in a FFBS scheme (central panel) and a precision sampler scheme (right panel).

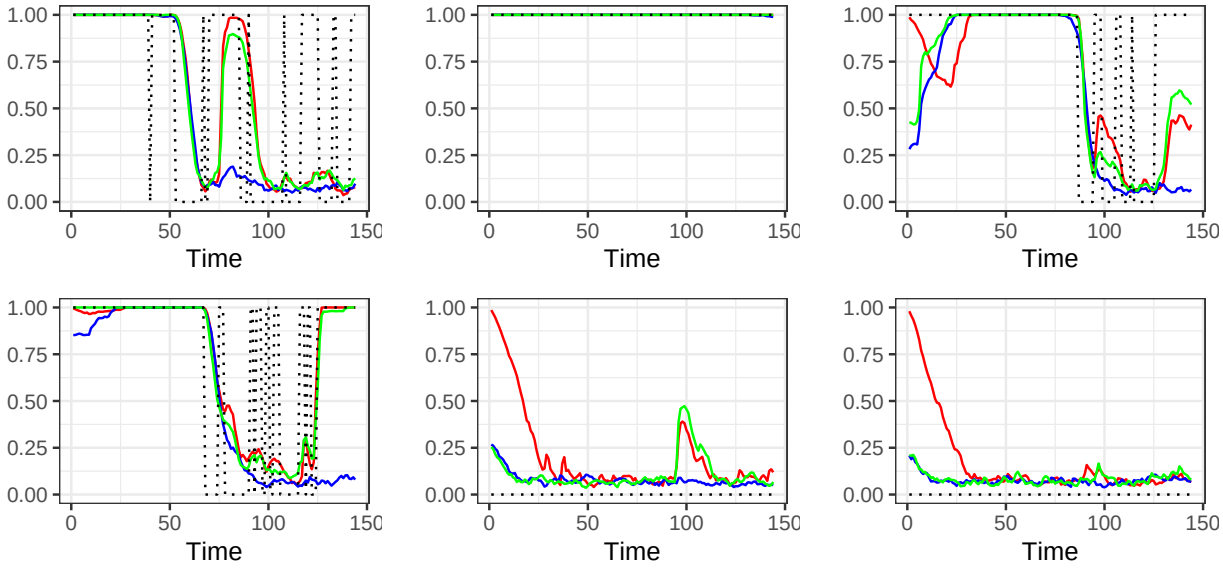


Figure 2.5: MCMC approximation of the posterior inclusion probabilities of a Dynamic SSVS with: a discount factor model (blue line), stochastic volatility model in a FFBS scheme (green line) and a precision sampler scheme (red line). Black dotted line represents the true values.

The running time of the simulation algorithms represents a limit when these strategies have to be reiterated multiple times. This happens for instance when one-step-ahead forecasts are generated for many periods or, simply, when multiple estimations are performed in order to compare results obtained with diverse specifications of the hypereparameters. By replacing the FFBS step with a precision sampler, the running time reduces while preserving performances as shown in Table 2.1. However, the greatest enhancement in terms of running time comes with the Dynamic EMVS. Therefore, here we present the results obtained by fitting the data using a Dynamic EMVS both with a discount factor model and with a stochastic volatility model. The initialization value $\boldsymbol{\beta}_t^{(0)}$ coincides with the least squares estimates. Even though the original approach of Rockova and McAlinn (2021a) does not involve least squares estimation for the initial values of the regression coefficients, we notice that this simple trick can improve convergence.

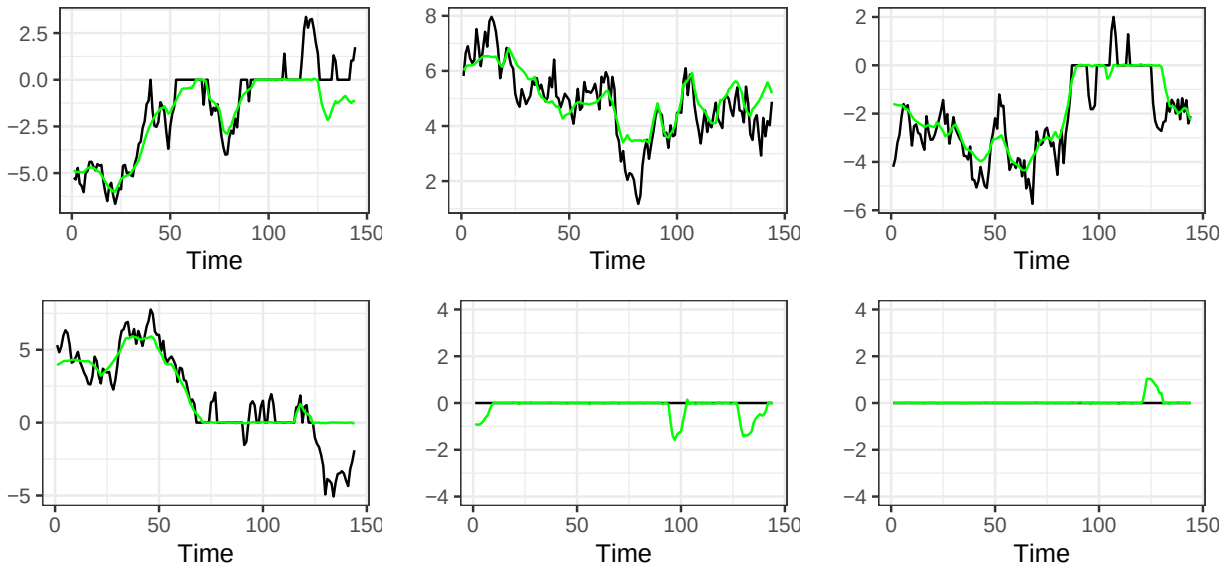


Figure 2.6: Dynamic EMVS with discount factor model for the residual; true values of $\beta_{1:T,j}$, $j = 1, \ldots, 6$, (black) and Maximum A Posteriori estimates (green) of the first six regression coefficients

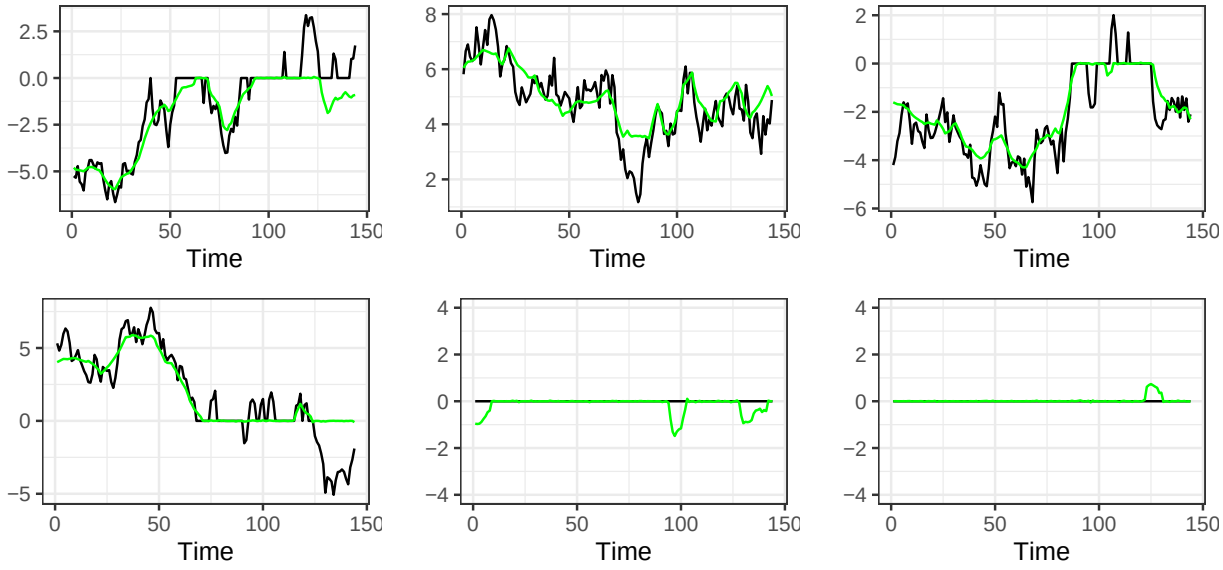


Figure 2.7: Dynamic EMVS with stochastic volatility model for the residual; true values of $\beta_{1:T,j}$, $j = 1, \ldots, 6$, (black) and Maximum A Posteriori estimates (green) of the first six regression coefficients

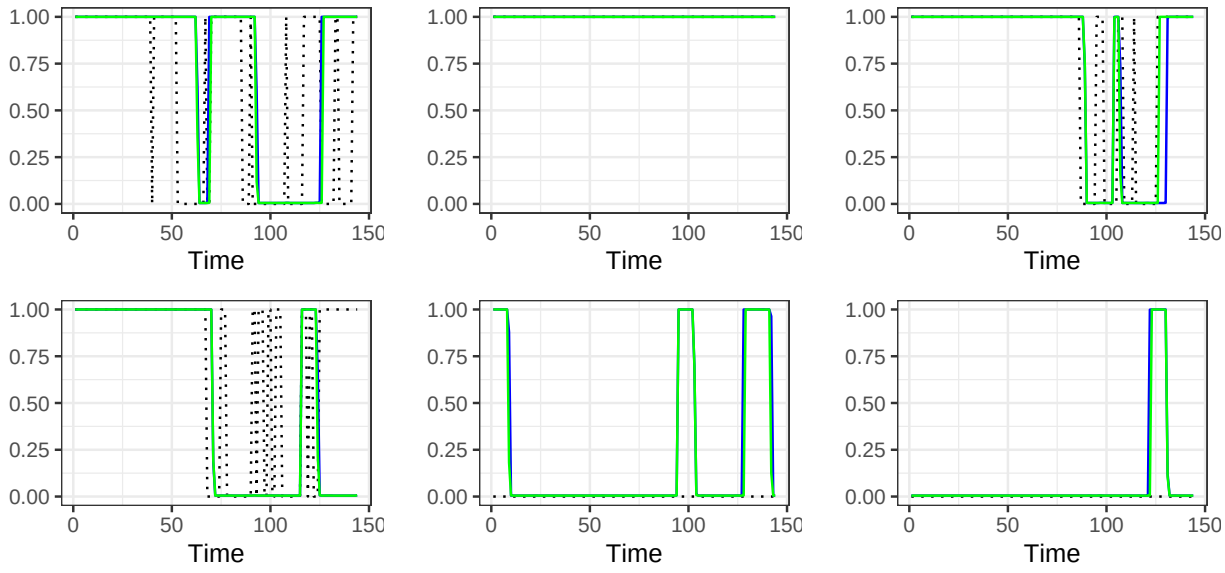


Figure 2.8: Estimates of the conditional inclusion probabilities using Dynamic EMVS with: a discount factor model (blue line) and a stochastic volatility model (green line). Black dotted line represents the true values.

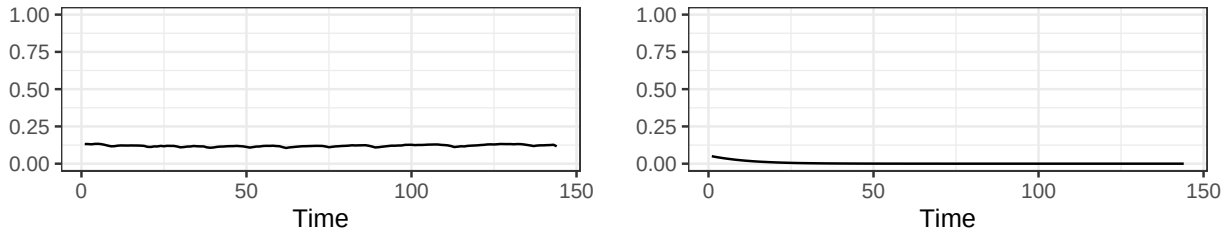


Figure 2.9: Dynamic EMVS: variances' estimates of a stochastic volatility model (left panel) and discount factor model (right panel)

Running times along with other meaningful information is presented in Tables 2.1 and 2.2. The metrics here used are: the Sum of Squared Errors (SSE) between the actual and the estimated state sequence, the Hamming distance (Ham) between the actual and the estimated indicator vectors, the amount of total false negative (FN) and false positive (FP), and the number of false detection (FD) and false non detection (FND). In details, the Hamming Distance is a metrics used in information theory to measure the number of substitutions

necessary to convert a string into another one. We use it to measure the distance between the indicator vector estimated and the true one. More formally, let $\pi_{t,j} \equiv \mathbb{P}(\gamma_{t,j} = 1|y_{1:T})$ (or $\pi_{t,j} = p^{\star}_{t,j}$ in the Dynamic EMVS case) and $\Pi = (\pi_{t,j})$, then

$$Ham(\Pi, \boldsymbol{\beta}^0_{0:T}) = \sum_{j=1}^{p}\sum_{t=1}^{T} |\mathbb{I}(\pi_{t,j} \geq 0.5) - \mathbb{I}(\beta^0_{t,j} \neq 0)|$$

False positives and false negatives stands for the number of times in which a coefficient is classified as relevant while it is not and vice versa. On the other hand, false detection indicates the number of noise covariates that are classified as relevant at least once over the whole period whereas false non detection count the number of relevant variables that have been ignored by the algorithm. The conclusion we draw from Tables 2.1 and 2.2 are the following. Firstly, without shrinkage ($\Omega = 1$) the SSE is huge and the model is clearly unable to fit the data. Therefore, the introduction of a sparsity inducing strategy is necessary. Just by lowering Ω to 0.2 leads to great improvements. The Dynamic SSVS with discount factor model provides in general the worst performances in terms of SSE and it generates many false positive as a consequence of the spike trap. On the other hand, the Dynamic SSVS with stochastic volatility shows better performances. Moreover, replacing the FFBS with a precision sampler implies minor accuracy of point estimates of the very first observations. The large share of false positive reported in Table 2.1 can be explained from the fact that the latter strategy assigns all the coefficient to the slab in the first periods (see Figure 2.8). However, estimates improve for more iterations of the Markov Chain. Overall, modelling variances as a log-Normal AR(1) process proves to return better estimates of both regression coefficients and the variance, for this reason we will consider this specification as the baseline for developing further models in the next chapters. In addition, the Dynamic EMVS represents a valid alternative to the simulation algorithms. It provides indeed precise point estimates at a reduced computational effort. However, one should consider the fact that the deterministic nature of this algorithm may result unsatisfactory whenever quantifying the uncertainty around the estimates is of interest. Moreover, it shows a tendency to produce false positives, therefore choosing the hyperparameters such that shrinkage is more severe is recommended.

Table 2.1: Performance comparison

Dynamic	Sampling Strategy	Volatility	Ω	Speed	SSE	Ham.	FP	FN	FD	FND
SSVS	FFBS	DF	1	745.56	2668.85	6754	6754	0	46	0
		DF	0.2	570.09	1031.947	127	19	108	0	0
		SV	0.2	502.03	764.934	117	37	80	1	0
	Precision Sampler	SV	0.2	251.27	1022.245	803	725	75	46	0
EMVS		DF	0.2	59.72	818.426	220	165	55	9	0
		SV	0.2	149.75	755.174	195	143	52	8	0

Table 2.2: Assessing the estimation bias with focus on the first four predictors and on the remaining forty-six.

				Predictors 1 to 4		Predictors 5 to 50	
Dynamic	Sampling Strategy	Volatility	Ω	SSE	Ham.	SSE	Ham.
SSVS	FFBS	DF	1	1658.258	130	1010.593	6624
		DF	0.2	1027.37	127	4.577	0
		SV	0.2	728.851	103	36.083	14
	Precision Sampler	SV	0.2	723.891	102	298.354	701
EMVS		DF	0.2	688.124	101	130.301	119
		SV	0.2	659.006	95	96.168	100

2.5 Macroeconomic Data

Inflation forecasting is probably the hardest challenge monetary policy authorities face. What makes it a difficult tasks is that the price growth can be explained by several factors, some of whom are not directly observable. In general this factors are divided in some macro-classes. Briefly, there are factors affecting demand which are mainly due to fiscal

and monetary policies and that are usually related to the money supply, consumption, disposable income, public expenditure, consumer expenditure, deficit financing, repayment of the public dept and exports. Then there are factors affecting supply such as shortage of important production factors, industrial disputes, natural calamities, artificial scarcities and other international factors. In addition a crucial role is played by agents' expectations which can be only measured using proxies. In the current exercise, we decide to include all the variables potentially affecting inflation in a unified Time-Varying Parameter regression model. The approach is therefore more data-driven and less theory-based. The purpose of this analysis (and the following ones) is purely illustrative and it contains several simplifications. Nonetheless, it also provides some good suggestions for further, more accurate, economic analysis. Data on the variables involved in this exercise are sourced from a large macroeconomic database maintained by M. W. McCracken and Ng (2016). The latter is primarily based on Federal Reserve Economic Data (FRED) [1] and it contains seasonally adjusted quarterly time series. Data are standardized and made stationary through log-difference, with the exception of interests rates. This transformations are in line with the instructions provided by the authors of the database. The list of the 40 explanatory variables involved in the analysis is reported in Appendix. The series we are interested to forecast is the quarterly Price Consumption Expenditure Price Index, which is commonly used as an indicator for inflation. The model we estimate is

$$
\begin{aligned}
y_{t+l} =& \mu_t + \boldsymbol{x}_t'\boldsymbol{\beta}_t + e^{\frac{h_t}{2}}\epsilon_t, \quad \epsilon_t \overset{iid}{\sim} \mathcal{N}(0,1) \\
\mu_t =& \phi_1\mu_{t-1} + \eta_t, \quad \eta_t \overset{iid}{\sim} \mathcal{N}(0,\lambda_1) \\
\boldsymbol{\beta}_t =& \Gamma_t\boldsymbol{\beta}_{t-1} + \boldsymbol{\xi}_t, \quad \boldsymbol{\xi}_t \sim \mathcal{N}(0,\Lambda_t) \\
h_t =& \alpha_0 + \alpha_1(h_{t-1} - \alpha_0) + \zeta_t, \quad \zeta_t \overset{iid}{\sim} \mathcal{N}(0,\sigma_\zeta)
\end{aligned}
$$

where l is 4 in the current analysis, $\Gamma_t = diag\{\gamma_{t,j}\phi_1\}_{j=1}^p$ and $\Lambda_t = diag\{\gamma_{t,j}\lambda_1 + (1-\gamma_{t,j})\lambda_0\}_{j=1}^p$. Moreover, $\gamma_{t,j}|\beta_{t-1,j} \overset{ind}{\sim} Bernoulli(\omega(\beta_{t-1,j}))$ for $j = 1,\dots,40$. The initial states are $\boldsymbol{\beta}_0 \sim \mathcal{N}(\boldsymbol{m}_0, \boldsymbol{C}_0)$ with $\boldsymbol{m}_0 = \mathbf{0}$ and $\boldsymbol{C}_0 = diag\{\gamma_{0,j}\lambda_1/(1-\phi_1^2) + (1-\gamma_{0,j})\lambda_0\}_{j=1}^p$ and $\mu_0 \sim \mathcal{N}(0,\lambda_1)$. The priors of the stochastic volatility model are the same of the simulation study. The model

[1] url: https://fred.stlouisfed.org/

is trained from 1968-12-01 to 2003-03-01, whereas one-step-ahead forecasts are sequentially updated from 2003-06-01 to 2015-12-01. Out-of-sample forecasts for the Dynamic SSVS are obtained by sampling $\gamma_{j,T+l}$ form a $Bernoulli(\omega(\beta_{j,T}))$, and h_{T+l} from $\mathcal{N}(\alpha_0 + \alpha_1 h_T, \sigma_\zeta^2)$ and then computing the one-step-ahead forecast mean f_{T+l} and the one-step-ahead forecast variance q_{T+l} (using the notation of Algorithm 3) via a Kalman filter recursion. This methodology for obtaining forecasts is applied by default to every algorithm discussed in this thesis. However, for Dynamic EMVS the parameters' means are replaced by their mode. In addition, predictive performances are evaluated using a combination of four different metrics. Three of them evaluate the forecast errors, i.e. $e_t = y_t - f_t$, and they are: the Root Mean Squared Error (RMSE), the Mean Absolute Scaled Error (MASE) and the Weighted Mean Absolute Percentage Error (WMAPE). For the sake of clarity, the MASE is computed in this way,

$$MASE = \frac{1}{n} \frac{\sum_{t=1}^{n} |y_t - f_t|}{\frac{1}{n-1} \sum_{t=2}^{n} |y_t - y_{t-1}|},$$

and it measures how well our model performs compared to a naive forecast. Instead, the WMAPE is given by

$$WMAPE = \frac{\sum_{t=1}^{n} |y_t - f_t|}{\sum_{i=1}^{n} |y_t|}$$

and it represents an alternative to the classical MAPE which is robust to the presence of zero or almost zero values. The forth metric instead takes into account the entire predictive distribution and it is the the sum of log predictive likelihoods (SLPL) defined as

$$SLPL = \sum_{t=1}^{n} \log \pi(y_t | f_t, q_t).$$

For this exercise we use the Dynamic SSVS strategy with precision sampler. The latter allows indeed for fast estimation without sacrificing uncertainty. The time series of inflation is plotted in the left panel of Figure 2.10. It clearly shows change points, that can be captured by a Time-Varying Parameter model. Overall, the residual standard deviations (central panel of Figure 2.10) are low, indicating that the model has succeed in selecting the right predictors, and it shows significant fluctuations during shocking events. The very first estimates however are the less reliable. The algorithm at first tends to assign all the coefficients to the slab, but after a dozen of observations the estimates became more precise

and it starts to drop unimportant covariates from the model. In every periods, about one quarter of the coefficients are active which implies a significant improvements in terms of model interpret ability. Moreover, this number usually increases during shocks or change points, for then coming back to a value of around ten.

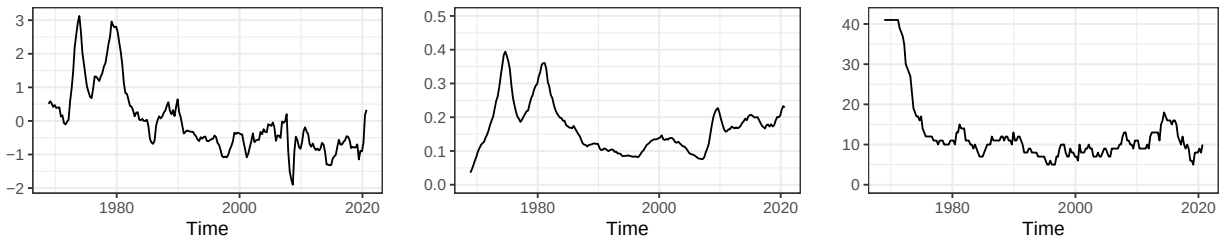


Figure 2.10: Left panel: Time series of quarterly inflation recentred and rescaled. Middle: Evolution of residual standard deviations. Right panel: Number of active covariates that contribute in forecasting inflation.

This in-sample assessment, that was useful to understand better the general features of the time series, is then followed by the forecasting exercise mentioned before. As shown in Figure 2.11 dynamic shrinkage plays an important role in reducing the forecast uncertainty. The dynamic shrinkage model here used ($\Omega = 0.1$) is able to preserve the accuracy of the accuracy of the full model ($\Omega = 1$) but it returns smaller credible intervals and therefore its point estimates are more trustworthy from a statistical standpoint.

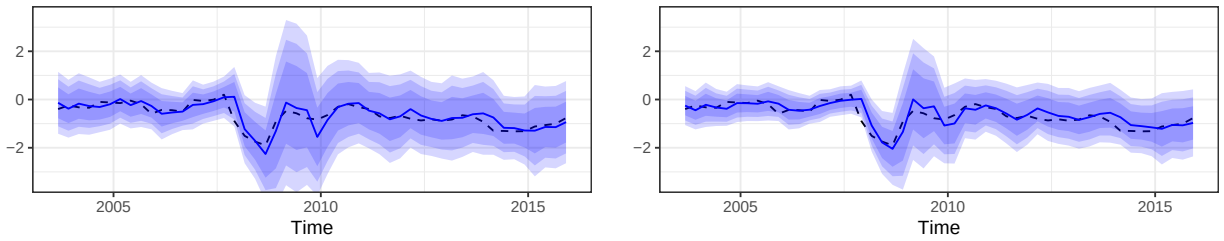


Figure 2.11: One-step-ahead forecasts (blue line) with credible intervals and realizations (black dashed line). For similar point estimates, the model withouth shrinkage ($\Omega = 1$) presents larger credible intervals than the model with actual dynamic shrinkage ($\Omega = 0.1$).

This visual insight is accompanied by a quantitative comparison in Table 2.3. Here we compared several models characterized by diverse intensity of shrinkage expressed by varying

hyperparameter Ω. In addition, we also compared models with different specification of the AR(1) latent process. In details, while the metrics based on forecast errors are almost equal, the SLPL reduces according to Ω. And for the same value of Ω, the larger is ϕ_1 (which must remain below 1 in absolute values) the better are the forecasts. This results was to be expected. We want to preserve a stationary structure of the latent process in order to maintain a stable evolution of the inclusion probabilities $\{w_t\}_{t=1}^T$, however we want to also conserve the information gained on the model's parameter from time to time. With an unjustified low value of ϕ_1 the past history of the coefficient become uninformative to determine the value of the regression coefficient in t, making the estimation task harder.

Table 2.3: Performance comparison for varying intensities of shrinkage

Ω	ϕ_1	RMSE	WMAPE	MASE	SLPL
1.0	0.98	0.264	0.287	1.021	-24.113
0.9	0.98	0.261	0.285	1.011	-23.790
0.7	0.98	0.261	0.287	1.019	-21.798
0.5	0.98	0.263	0.286	1.016	-19.450
0.3	0.98	0.264	0.289	1.027	-17.453
0.1	0.98	0.263	0.291	1.033	-14.708
	0.80	0.305	0.351	1.247	-19.209
	0.40	0.533	0.662	2.353	-40.135
	0.10	0.755	0.951	3.378	-78.036

Chapter 3

Dynamic Shrinkage in Bayesian Structural Time Series

Building on the results of the previous chapter, we want to propose a new class of models for time series analysis. It consists of an extension of the Bayesian Structural Time Series models developed by Scott and Varian (2013). Starting from this framework we develop a BSTS model with both Dynamic Spike-and-Slab process priors and stochastic volatility. This model can be also regarded as an extension of the TVP regression model defined in equation (2.1) that takes into account also structural time series components such as seasonality and trend. These features are indeed usually exhibited by many time series, especially in macroeconomics, and including them inside the model can improve forecasting performances. The model is defined by the following system of equations:

$$\begin{aligned} y_t &= \mu_t + \tau_t + \boldsymbol{x}_t'\boldsymbol{\beta}_t + \epsilon_t, \\ \mu_t &= \mu_{t-1} + \delta_{t-1} + u_t, \\ \delta_t &= \delta_{t-1} + v_t, \\ \tau_t &= -\sum_{s=1}^{S-1} \tau_{t-s} + w_t, \\ \boldsymbol{\beta}_t &= \Gamma_t \boldsymbol{\beta}_{t-1} + \boldsymbol{\xi}_t \end{aligned} \tag{3.1}$$

where the p coefficients are independent and identically distributed according to DSS priors: $\beta_{1:T,j} \overset{iid}{\sim} DSS(\Omega, \lambda_0, \lambda_1, \phi_0, \phi_1)$ for $j = 1, \ldots, p$. In addition, given the results illustrated in the previous chapter, that suggest improvements with an AR(1) stochastic volatility model, we model the residuals using following specification $\epsilon_t \sim \mathcal{N}(0, e^{h_t})$ with

$$h_t = \alpha_0 + \alpha_1(h_{t-1} - \alpha_0) + \zeta_t \tag{3.2}$$

The disturbances u_t, v_t, w_t and ζ_t are i.i.d. normally distributed with mean zero and variances respectively σ_μ^2, σ_δ^2, σ_τ^2 and σ_ζ^2. Ideally, it would be possible to estimate these variances by maximum likelihood or assigning an Inverse-Gamma prior to them and proceeding with Bayesian inference. However, there are some identifiability issues in separating the residuals of the trend, the slope and the seasonality, and the MLE or MCMC may fail at converging towards a unique solution. We do not have this problem since we set a-priori $\sigma_\mu^2 = \sigma_\delta^2 = \sigma_\tau^2 = \lambda_1$ in the applications discussed in this thesis, where λ_1 is a predetermined scalar. On the other hand, $\sigma_\zeta^2 \sim \mathcal{IG}(0.5, 0.5)$. Such choice is convenient since it is the default prior for σ_ζ^2 used in the `stochvol` R-package. Moreover, in the state equation for $\boldsymbol{\beta}_t$, $\Gamma_t = diag\{\gamma_{t,j}\phi_1\}_{j=1}^p$ and

$$\boldsymbol{\xi}_t \overset{iid}{\sim} \mathcal{N}(0, \Lambda_t) \text{ with } \Lambda_t = diag\{\gamma_{t,j}\lambda_1 + (1 - \gamma_{t,j})\lambda_0\}_{j=1}^p.$$

Let us recall that $\gamma_{t,j}|\beta_{t-1,j} \overset{ind}{\sim} Bernoulli(\omega(\beta_{t-1,j}))$ for $j = 1, \ldots, p$. The values of λ_0 and λ_1 play a crucial role in ensuring that the estimation strategy is able to effectively distinguish the signal from the noise. Usually, some preliminary estimations have to be performed to understand which values of λ_0 and λ_1 provide the best fitting of the data. Overall, it is recommended to maintain the ratio $\lambda_1/\lambda_0 = 10$.

The DLM representation of the Bayesian Structural Time Series is very useful since it allows to take advantage of the Kalman filter formulas, however some considerations are necessary. Firstly, the state vector is $\boldsymbol{\theta}_t = (\mu_t, \delta_t, \tau_t, \tau_{t-1}, ..., \tau_{t-S+1}, \boldsymbol{\beta}_t)'$. Consequently, $\boldsymbol{F}_t = (1, 0, 1, 0, ..., 0, \boldsymbol{x}_t)$ and $\boldsymbol{G}_t = blockdiag(\boldsymbol{G}_t^{(trend)}, \boldsymbol{G}_t^{(seasonal)}, \Gamma_t)$ where

$$\boldsymbol{G}_t^{(trend)} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{G}_t^{(seasonal)} = \begin{pmatrix} -1 & -1 & ... & -1 \\ 1 & 0 & ... & 0 \\ 0 & 1 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & 1 \end{pmatrix}.$$

The specification of $\boldsymbol{G}_t^{(seasonal)}$ is a direct consequence of the identifiability constraint on the seasonal factors, i.e. $\sum_{s=1}^{S} \tau_s = 0$. Similarly, $\Sigma_{\eta,t} = blockdiag(\Sigma_{\eta,t}^{(trend)}, \boldsymbol{\Sigma}_{\eta,t}^{(seasonal)}, \Lambda_t)$ where

$$\Sigma_{\eta,t}^{(trend)} = \begin{pmatrix} \sigma_\mu^2 & 0 \\ 0 & \sigma^\delta \end{pmatrix} \quad \text{and} \quad \Sigma_{\eta,t}^{(seasonal)} = \begin{pmatrix} \sigma_\tau^t & 0 & ... & 0 \\ 0 & 0 & ... & 0 \\ 0 & 0 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & 0 \end{pmatrix}.$$

In addition, nothing prevents the model to accommodate also a static regression component $\boldsymbol{z}_t'\boldsymbol{\beta}$. This can be accomplished in several ways, however the most convenient from a computational point of view is the one envisaged by Scott and Varian (2013) that consists in consists in appending a constant 1 to the state vector and $\boldsymbol{z}_t'\boldsymbol{\beta}$ to $\boldsymbol{F}_t$ in this way: $\boldsymbol{\theta}_t = (1, \mu_t, \delta_t, \tau_t, \tau_{t-1}, ..., \tau_{t-S+1}, \boldsymbol{\beta}_t)'$ and $\boldsymbol{F}_t = (\boldsymbol{z}_t'\boldsymbol{\beta}, 1, 0, 1, 0, ..., 0, \boldsymbol{x}_t)$.

Efficient posterior sampling is provided by the Dynamic SSVS we introduced in the Section 2.2 with some modifications due to the introduction of structural components. All the steps are summarized in Algorithm 6 and we refer to Section 2.2 for further details. Note that the expansion of the state vector's size due to structural components leads inevitably to a slower running time since the Kalman filter's computational complexity is linear in data length but quadratic in the state vector dimension. Considering for instance quarterly data, $\boldsymbol{\theta}_t$ becomes a column vector of p regression coefficients plus 3 seasonal components and 2 trend components. Therefore, in order to reduce computational efforts, we also propose an extension of the Dynamic EMVS seen in Section 2.3 for this novel BSTS model. The latter results to be particularly useful for exploratory analysis, when multiple attempts are performed to find satisfying values of the hyperparameters.

Algorithm 6: Dynamic SSVS in BSTS with Stochastic Volatility

Initialize $\gamma_{j,t}$ and $\sigma_{\epsilon,t}$ for $0 \leq t \leq T$ and $1 \leq j \leq p$ and set n_0 and d_0;

Step 1: Sample Time Series Structural Components

for $1 \leq t \leq T$ **do**

Compute $\boldsymbol{a}_t = \boldsymbol{G}_t \boldsymbol{m}_{t-1}$;

Compute $\boldsymbol{R}_t = \boldsymbol{G}_t \boldsymbol{C}_{t-1} \boldsymbol{G}_t' + \Sigma_{\eta,t}$;

Compute $f_t = \boldsymbol{F}_t' \boldsymbol{a}_t$ and $e_t = y_t - f_t$;

Compute $q_t = \boldsymbol{F}_t' \boldsymbol{R}_t \boldsymbol{F}_t + \sigma^2_{\epsilon,t}$;

Compute $\boldsymbol{m}_t = \boldsymbol{a}_t + \boldsymbol{A}_t e_t$ and $\boldsymbol{C}_t = \boldsymbol{R}_t - \boldsymbol{A}_t \boldsymbol{A}_t' q_t$ with $\boldsymbol{A}_t = \boldsymbol{R}_t \boldsymbol{F}_t / q_t$

Draw $\boldsymbol{\theta}_t \sim \mathcal{N}(\boldsymbol{m}_T, \boldsymbol{C}_t)$;

for $t = T-1, ..., 0$ **do**

Compute $\boldsymbol{a}_T(t-T) = \boldsymbol{m}_t + \boldsymbol{B}_t(\boldsymbol{\theta}_{t+1} - \boldsymbol{a}_{t+1})$;

Compute $\boldsymbol{R}_t(t-T) = \boldsymbol{C}_t - \boldsymbol{B}_t \boldsymbol{R}_{t+1} \boldsymbol{B}_t'$ where $\boldsymbol{B}_t = \boldsymbol{C}_t \boldsymbol{G}_{t+1}' \boldsymbol{R}_{t+1}^{-1}$;

Draw $\boldsymbol{\theta}_t \sim \mathcal{N}(\boldsymbol{a}_T(t-T), \boldsymbol{R}_t(t-T))$;

Step 2: Sample Indicators

for $j = 1, ..., p$ **do**

for $1 \leq t \leq T$ **do**

Compute $\omega_{j,t} = \omega(\beta_{j,t-1})$ from ;

Compute $p^\star_{t,j} = p^\star_{t,j}(\beta_{j,t})$ from ;

Draw $\gamma_{j,t} \sim Bernoulli(p^\star_{j,t}(\beta_{j,t}))$;

Compute $p^\star_{j,0} = \omega(\beta_{j,0})$;

Draw $\gamma_{j,0} \sim Bernoulli(p^\star_{j,0}(\beta_{j,0}))$;

Step 3: Sample Volatility

Compute $r_t = y_t - \boldsymbol{F}_t' \boldsymbol{\theta}_t$ for $t = 1, ..., T$;

Sample $(h_0, ..., h_T)$ using the efficient sampling strategy described in Section 2.2.1;

Compute $\sigma^2_{\epsilon,t} = e^{h_t}$;

3.1 Dynamic Expectation-Maximization Variable Selection

The huge computational saving in running time through the Dynamic EMVS encouraged us to develop a modified version of the strategy of Rockova and McAlinn (2021a) that takes into account also structural time series components. Here, the MAP trajectory to be approximated is $\hat{\boldsymbol{\theta}}_{1:T} = \arg\max \pi(\boldsymbol{\beta}_{1:T}, \mu_{1:T}, \delta_{1:T}, \tau_{1:T} \mid y_{1:T})$. Fortunately, the considerations done previously for the E-step in the previous chapter still hold. However, the M-step requires instead further computations. Here we describe how to implement the Dynamic EMVS for quarterly data, nonetheless the following steps can be generalized for other data frequencies. The joint density prior is factorized as it follows

$$\begin{aligned}\pi(\mu_{0:T}, \delta_{0:T}, \tau_{0:T}, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,0:T}) = \pi(\mu_0)\pi(\delta_0)\pi(\tau_0)\pi(\boldsymbol{\beta}_0|\boldsymbol{\gamma}_0)\pi(\boldsymbol{\gamma}_0)... \\ ... \prod_{t=1}^{T} \Bigg\{ \pi(\sigma^2_{\epsilon,t} \mid \sigma^2_{\epsilon,t-1})\pi(\mu_t \mid \mu_{t-1}, \delta_{t-1})\pi(\delta_t \mid \delta_{t-1})\pi(\tau_t \mid \tau_{t-1}, \tau_{t-2}, \tau_{t-3})... \\ ... \prod_{j=1}^{p} \Big[\pi(\beta_{t,j} \mid \gamma_{t,j}, \beta_{t-1,j})\pi(\gamma_{t,j} \mid \beta_{t-1,j}) \Big] \Bigg\}\end{aligned}$$

whose posterior can be decomposed as

$$\begin{aligned}&\pi(\mu_{0:T}, \delta_{0:T}, \tau_{0:T}, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,0:T} \mid y_{1:T}) \propto \\ &\pi(y_{1:T} \mid \mu_{0:T}, \delta_{0:T}, \tau_{0:T}, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,0:T})\pi(\mu_{0:T}, \delta_{0:T}, \tau_{0:T}, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,0:T}).\end{aligned}$$

Therefore, in logs, we have

$$\begin{aligned}&\log \pi(\mu_{0:T}, \delta_{0:T}, \tau_{0:T}, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,0:T} \mid y_{1:T}) = \\ &C - \frac{(\mu_0 - m_0^{(\mu)})^2}{2C_0^{(\mu)}} - \frac{(\delta_0 - m_0^{(\delta)})^2}{2C_0^{(\delta)}} - \frac{(\tau_0 - m_0^{(\tau)})^2}{2C_0^{(\tau)}} \\ &- \sum_{j=1}^{p} \left\{ \gamma_{0,j} \frac{\beta^2_{0,j}(1-\phi_1^2)}{2\lambda_1} + (1-\gamma_{0,j}) \frac{\beta^2_{0,j}}{2\lambda_0} - \gamma_{0,j} \log \Theta - (1-\gamma_{0,j}) \log(1-\Theta) \right\} \\ &+ \sum_{t=1}^{T} \sum_{j=1}^{p} [\gamma_{t,j} \log \theta_{t,j} + (1-\gamma_{t,j}) \log(1-\theta_{t,j})] \\ &- \sum_{t=1}^{T} \left\{ \frac{(y_t - \boldsymbol{x}_t' \boldsymbol{\beta}_t - \mu_t - \tau_t)^2}{2\sigma^2_{\epsilon,t}} + \sum_{j=1}^{p} \left[\gamma_{t,j} \frac{(\beta_{t,j} - \phi_1 \beta_{t-1,j})^2}{2\lambda_1} + (1-\gamma_{t,j}) \frac{\beta^2_{t,j}}{2\lambda_0} \right] + \right.\end{aligned}$$

$$\log \pi(\sigma^2_{\epsilon,t}|\sigma^2_{\epsilon,t-1}) + \frac{(\mu_t - \mu_{t-1} - \delta_{t-1})^2}{2\sigma^2_\mu} + \frac{(\delta_t - \delta_{t-1})^2}{2\sigma^2_\delta} + \frac{(\tau_t + \tau_{t-1} + \tau_{t-2} + \tau_{t-3})^2}{2\sigma^2_\tau}\Bigg\}.$$

Maximizing $\mathbb{E}(\mu_{0:T}, \delta_{0:T}, \tau_{0:T}, \boldsymbol{\beta}_{0:T}, \boldsymbol{\gamma}_{0:T}, \sigma^2_{\epsilon,0:T} \mid y_{1:T}) = Q(\boldsymbol{\Xi} \mid y_{1:T})$ with respect to $(\mu_{0:T}, \delta_{0:T}, \tau_{0:T}, \boldsymbol{\beta}_{0:T})$, one obtains the following first order conditions:

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \tau_t} = 0 \iff$$

$$\tau_t = \left(\frac{4}{\sigma^2_\tau} + \nu^\star\right)^{-1}\left\{\frac{1}{\sigma^2_\tau}(-3\tau_{t-1} - 2\tau_{t-2} - \tau_{t-3} - 3\tau_{t+1} - 2\tau_{t+2} - \tau_{t+3}) + \nu^\star(y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t - \mu_t))\right\}$$

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \tau_0} = 0 \iff \tau_0 = \left(\frac{1}{C_0^{(\tau)}} + \frac{1}{\sigma^2_\tau}\right)^{-1}\left\{\frac{m_0^{(\tau)}}{C_0^{(\tau)}} - \frac{1}{\sigma^2_\tau}(3\tau_1 + 2\tau_2 + \tau_3)\right\}$$

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \delta_t} = 0 \iff \delta_t = \left(\frac{1}{\sigma^2_\mu} + \frac{2}{\sigma^2_\delta}\right)^{-1}\left\{\frac{1}{\sigma^2_\mu}(\mu_{t+1} - \mu_t) + \frac{1}{\sigma^2_\delta}(\delta_{t-1} + \delta_{t+1})\right\}$$

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \delta_0} = 0 \iff \delta_0 = \left(\frac{1}{C_0^{(\delta)}} + \frac{1}{\sigma^2_\mu} + \frac{1}{\sigma^2_\delta}\right)^{-1}\left\{\frac{m_0^{(\delta)}}{C_0^{(\delta)}} + \frac{1}{\sigma^2_\mu}(\mu_1 - \mu_0) + \frac{1}{\sigma^2_\delta}\delta_1\right\}$$

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \mu_t} = 0 \iff \mu_t = \left(\frac{2}{\sigma^2_\mu} + \nu^\star\right)^{-1}\left\{\frac{1}{\sigma^2_\mu}(\mu_{t+1} - \mu_{t-1} + \delta_{t-1} - \delta_t) + \nu^\star(y_t - \boldsymbol{x}_t'\boldsymbol{\beta}_t - \tau_t)\right\}$$

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \mu_0} = 0 \iff \mu_0 = \left(\frac{1}{\sigma^2_\mu} + \frac{1}{C_0^{(\mu)}}\right)^{-1}\left\{\frac{m_0^{(\mu)}}{C_0^{(\mu)}} + \frac{\mu_1}{\sigma^2_\mu} - \frac{\delta_0}{\sigma^2_\mu}\right\}$$

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \boldsymbol{\beta}_t} = 0 \iff \boldsymbol{\beta}_t = \boldsymbol{D}_t^{-1}\left\{\nu^\star(y_t - \mu_t - \tau_t)\boldsymbol{x}_t + \frac{\phi_1}{\lambda_1}\boldsymbol{\beta}_{t-1} \odot \boldsymbol{p}_t^\star + \frac{\phi_1}{\lambda_1}\boldsymbol{\beta}_{t+1} \odot \boldsymbol{p}_{t+1}^\star\right\}$$

$$\frac{\partial Q(\boldsymbol{\Xi} \mid y_{1:T})}{\partial \boldsymbol{\beta}_0} = 0 \iff \boldsymbol{\beta}_0 = \boldsymbol{D}_0^{-1}\frac{\phi_1}{\lambda_1}\boldsymbol{\beta_1} \odot \boldsymbol{p_1^\star}$$

where

$$\boldsymbol{D}_t = \frac{\boldsymbol{x}_t\boldsymbol{x}_t'}{\sigma^2_{\epsilon,t}} + diag\left\{\frac{p^\star_{t,j}}{\lambda_1} + \frac{1 - p^\star_{t,j}}{\lambda_0} + \frac{\phi_1^2 p^\star_{t+1,j}}{\lambda_1}\right\}_{j=1}^p$$

and

$$\boldsymbol{D}_0 = diag\left\{\frac{(1-\phi_1^2)p^\star_{0,j}}{\lambda_1} + \frac{1 - p^\star_{0,j}}{\lambda_0} + \frac{\phi_1^2 p^\star_{1,j}}{\lambda_1}\right\}_{j=1}^p$$

Again, the Woodburry formula simplifies calculations and reduces running time. The first order conditions listed above vary for $t = \{0, T\}$. For instance, the argmax with respect to τ_T is

$$\tau_T = \left(\frac{1}{\sigma^2_\tau} + \nu^\star\right)^{-1}\left\{\frac{1}{\sigma^2_\tau}(-\tau_{T-1} - \tau_{T-2} - \tau_{T-3}) + \nu^\star(y_T - \boldsymbol{x}_T'\boldsymbol{\beta}_T - \mu_T))\right\}$$

and so on. In the next section we explore the potential of the proposed model for structural time series in a simulation study.

3.2 Simulation Study

In order to illustrate of the validity of this novel approach, we propose a toy example on quasi-synthetic data. It is possible to generate a Bayesian Structural Time Series easily by summing up two or more processes. Therefore we generate synthetic data from a TVP regression model with $p = 20$ explanatory variables of which the first four predictors affect the outcome whereas the remaining 16 predictors are just disturbing factors following the same steps illustrated in Section 2.4. For this simulation study, we want to also include a trend and a seasonal component. To this aim, rather than generating synthetic data, I used a time series already available in R, namely the series "AirPassengers" (the classic Box and Jenkins airline data consisting of monthly totals of international airline passengers from 1949 to 1960), which shows a trend and a multiplicative seasonality. The data used in the simulation study is obtained as the composition of the synthetis data plus the log of the series AirPassengers, as shown in Figure 3.1. Note, we rescale the data to emphasize its structural features.

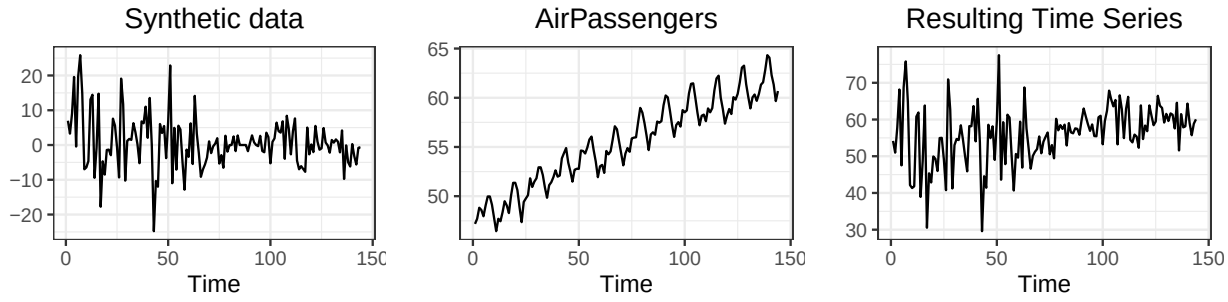


Figure 3.1: Left panel: Synthetic data generated from a TVP regression model. Middle: Rescaled Logarithmic AirPassengers. Right panel: Resulting time series used in the simulation.

Clearly, fitting the resulting time series is anything but easy and this is also the reason why we restricted the number of (potential) regressors to 20 (rather than 50 as in Section 2.4). Moreover, another reason fro limiting the simulation study to a moderate number of regressors is the size of the state vector, that here includes the structural components. For instance, using (dynamic) seasonal factors, 13 additional latent states have to be estimated.

Two of them are the time-varying trend with stochastic slope, whereas the remaining eleven represent the seasonality. Nevertheless, Dynamic SSVS and Dynamic EMVS prove to be able to capture the source of the signal. This is shown in Figure 3.2 and 3.3 and in particular in the table below.

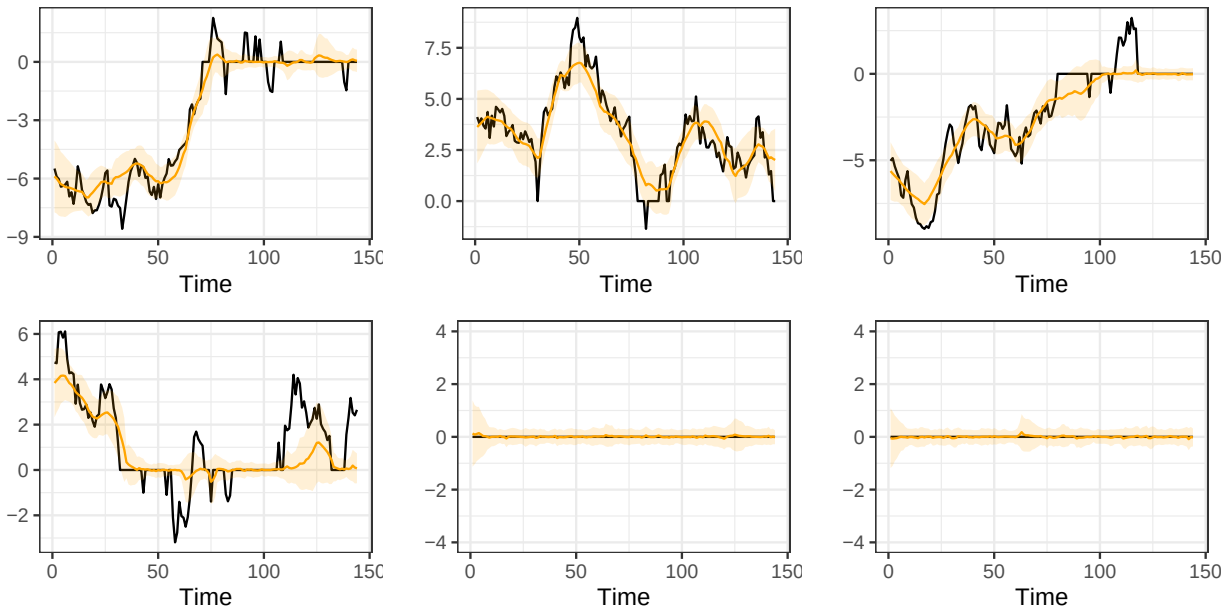


Figure 3.2: Dynamic SSVS for BSTS model; true values of $\beta_{1:T,j}$, $j = 1, \ldots, 6$, (black) and smoothed estimates with 95 percent credible intervals (yellow) of the first six regression coefficients.

In details, Table 3.1 shows that a BSTS with trend and seasonality without dynamic shrinkage ($\Omega = 1$) is not able to individuate the source of signal, therefore the estimates are biased and definitely not reliable. On the other hand, both the Dynamic SSVS and Dynamic EMVS performed well. We fix $\sigma_\tau^2 = \sigma_\mu^2 = \sigma_\delta^2 = 0.1$. We acknowledge that the latter is a strong assumption, however we find it to work well for the estimation of the time series structural components. Figure 3.4 shows smoothing estimates for the seasonal, the trend and the regression components.

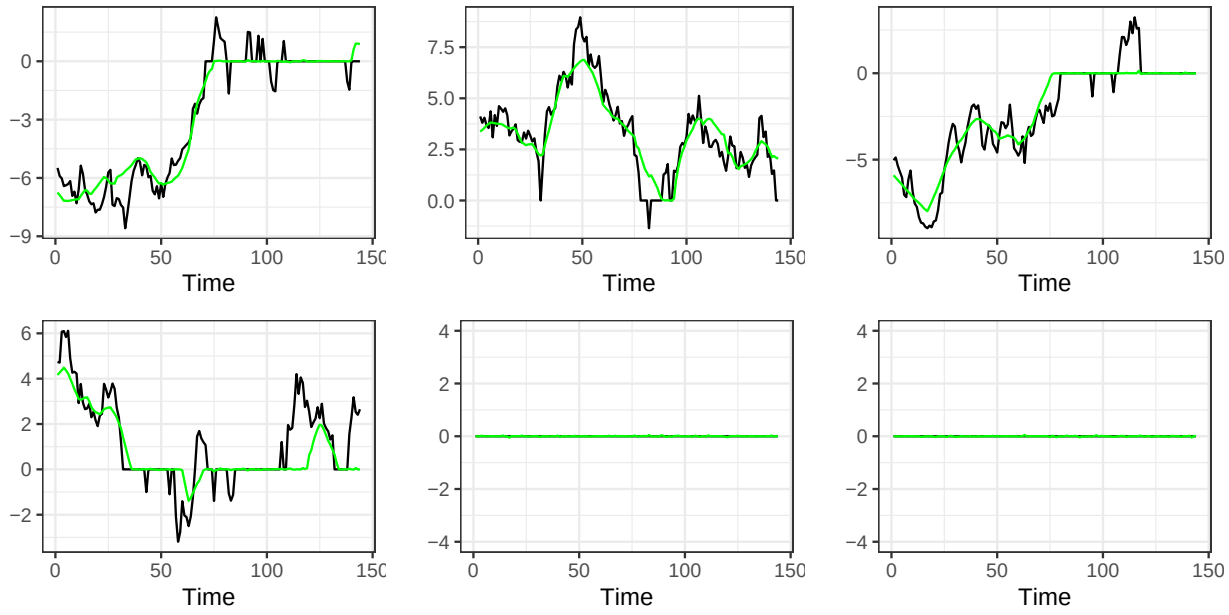


Figure 3.3: Dynamic EMVS for BSTS model; true values of $\beta_{1:T,j}$, $j = 1, \ldots, 6$, (black) and MAP trajectory (green) of the first six regression coefficients.

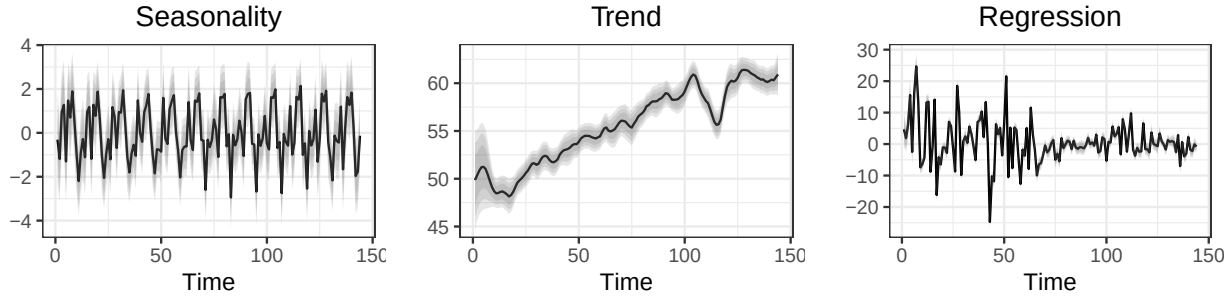


Figure 3.4: Structural time series components. Point estimates (black) with credible intervals (gray).

For the sake of completeness, the analysis is carried out for $p = 40$. Results are promising and they are shown in the last rows of Table 3.1. Overall, the Dynamic SSVS with $\Omega = 0.2$ shows the best performance. In particular it is better than the Dynamic EMVS in terms of SSE and Hamming distance. In addition, the simulation algorithm did not produce any false detection or false non detection.

Table 3.1: Performance comparison

Dynamic	Ω	p	SSE	Ham.	FP	FN	FD	FND
SSVS	1	20	1796.832	2493	2493	0	16	0
SSVS	0.2	20	569.214	106	44	62	0	0
EMVS	0.2	20	590.676	149	85	64	3	0
SSVS	0.2	40	610.903	113	44	69	0	0
EMVS	0.2	40	702.192	190	114	76	8	0

3.3 Macroeconomic Data

Given the encouraging results obtained with synthetic data, here we provide an empirical macroeconomic application concerning unemployment forecasting. The unemployment time series is usually characterized by seasonal and trend-cycle components due to cyclical phenomena such as seasonal work and long-run economic fluctuations. However, in some specific circumstances, macroeconomic shocks may hit abruptly the series causing unexpected deviations from its natural path. For these reasons, unemployment forecasting represents a good exercise to test the strength of our model in capturing all these features. Therefore, the objective of this analysis is to fit the time series of unemployment rate not transformed and not seasonally adjusted.

The BSTS model used in this exercise incorporates monthly seasonality, a stochastic trend with time-varying slope, and a large set of predictors whose role is to explain possible shocks. Predictors include: lagged values of the dependent variable, interest rates, financial and economics indicators, unemployment claims, and Google Trends data. The sources of the data are the Federal Reserve Economic Data (FRED)[1], Yahoo Finance[2] and Google Trends[3]. The third one is a less conventional data source. In a nutshell, Google Trends is a free

[1]url: https://fred.stlouisfed.org/

[2]url: https://it.finance.yahoo.com/

[3]url: https://trends.google.com/

service provided by Google LLC which is meant to inform the user about how frequently a given search term is entered into Google's research engine. However, Google Trends provides only relative frequencies of the data, such that a value equal to 100 identifies the highest frequency of the searched term over a certain period, while the value 0 indicates the lowest. The predictors are divided into two classes according to their frequency. For variables that present a monthly frequency we considered the lagged values, while for those having a daily frequency we considered the contemporaneous values. The list of predictors used in this study is provided in Appendix. More formally, the observation equation of the model here proposed is

$$y_t = \mu_t + \tau_t + \boldsymbol{w}_t'\boldsymbol{\beta}_{1,t} + \boldsymbol{z}_{t-1}'\boldsymbol{\beta}_{2,t-1} + \epsilon_t \tag{3.3}$$

which is the same as for model (3.1) (3.1), but with $\boldsymbol{x}_t = (\boldsymbol{w}_t, \boldsymbol{z}_{t-1})$ and $\boldsymbol{\beta}_t = (\boldsymbol{\beta}_{1,t}, \boldsymbol{\beta}_{2,t-1})$. The problem of forecasting using contemporaneous data is often referred to as "nowcasting." Such a term was originally coined in metereology to indicate the prediction of the present or the very near future of an economic or business indicator. What makes nowcasting appealing is the fact that official statistics are usually published with a time lag, whereas other type of data, such as financial data or web data, are available at lower frequencies. For example, Varian and Choi (2009b) had the idea of using Google Trends data for anticipating the release of official statistics. Official statistics are indeed released one or even two weeks after the end of the month while web data related to them are updated every day. This allows to gather meaningful information about the statistics of interest before the release. For example, observing Figure 3.5, where the time series of unemployment rate is presented together with some correlated Google searches, it would be evident that a certain correlation exists among them. Indeed, Google searches show a similar seasonality and synchronous spikes in proximity of shocking events. For instance, searches for unemployment depression, unemployment insurance and unemployment agencies drastically increased at the beginning of pandemic crisis. And while the official statistic for unemployment rate in April 2020 was released only on 8th May 2020 by the US Bureau of Labor Statistics, Google Trends data were already available in April. Therefore, we could say that these Google Trends anticipated

the observed peak in unemployment rate.

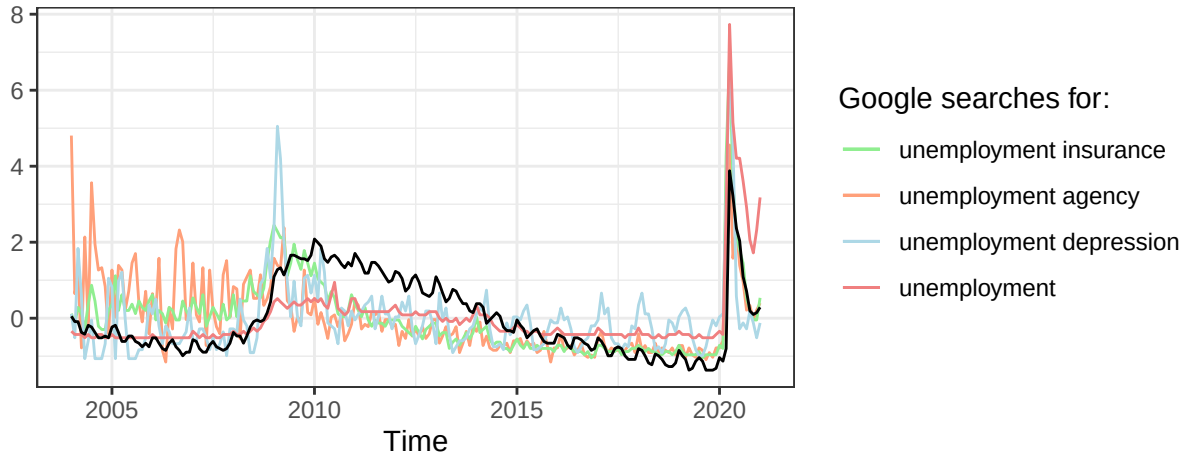


Figure 3.5: Scaled time series of unemployment rate (black) and Google searches in different colors.

Before proceeding to forecasting, we carry out a analysis we carry out an explorative analysis. The hyperparameters are set in this way: $\Omega = 0.2$, $\phi_0 = 0$, $\phi_1 = 0.98$, $\lambda_0 = 0.01$ and $\lambda_1 = 0.1$. This setting prove good results in the quasi-synthetic study and we maintain these choices here. After some preliminary analysis performed with a large set of 61 predictors (40 of which Google Trends), 12 seasonal factors, and a stochastic trend with a time-varying slope we decide to manually select only 15 Google Trends since we notice some redundancy in the information provided by some of them. In particular, we remove the US states-specific ones, e.g. "ny unemployment" and "florida unemployment," and those not showing any sort of correlation with the dependent variable. In addition, removing collinear predictors allows for better identification and it reduces the running time of the algorithms too.

As shown in Figure 3.6, the BSTS model succeed in identifying the trend and seasonal components of the time series. Instead, Google Trends data intervene principally in proximity of the pandemic crisis, as reported also in Figure 3.7 which shows that six predictors are particularly important for anticipating the event. The latter are: "searches for unemployment," "federal unemployment," "unemployment check," "unemployment depression," "unemployment office" and "unemployment pa."

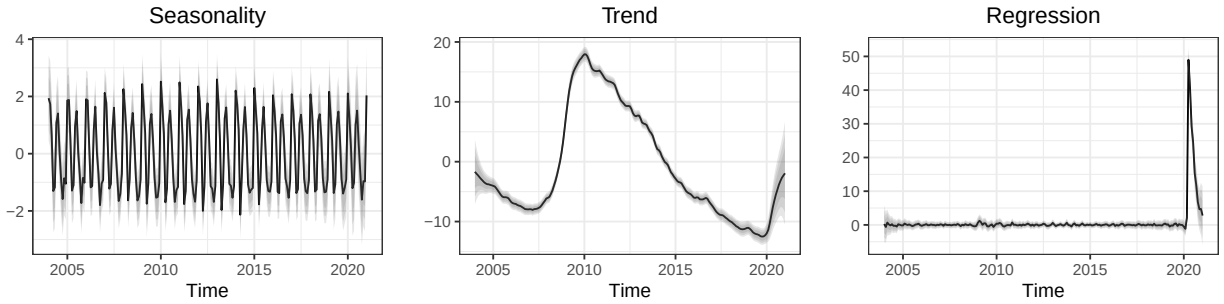


Figure 3.6: Structural time series components. MCMC approximation of the expected values and their credible intervals. Data are rescaled to better visualize the main features.

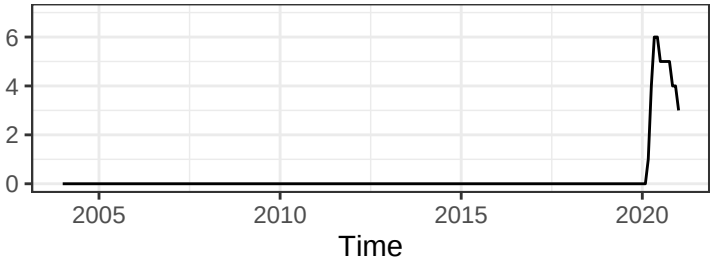


Figure 3.7: Number of active TVP regression coefficients that contribute to unemployment rate nowcasting.

We wondered why many Google Trends are insignificant and, in particular, why some of them can anticipate the 2020 pandemic crises but none of them can anticipate the 2008 financial crises. The answers we found are the following. Firstly, Google economists used to rely on another and maybe more powerful tool called Google Correlate. The difference between the two services is that in Google Correlate the user uploaded a time series and the system returned directly the most correlated queries. On the other hand, in Google Trends correlated series have to be selected manually. This makes it more difficult to include interesting predictors inside the model. Secondly, Google Trends data can assume only discrete values that depend on the time frame considered since they are rescaled according to the rule mentioned before. This fact may explain why Google Trends seem to have an important role in anticipating the pandemic crisis while they are insignificant for predicting

the financial crises of 2008. We notice indeed that the enormous increase in Google searches due to the pandemic flatten the series to zero at every other time.

Once we admit that there is a difference between the two crisis also under the aspect of Google searches, then it is then natural to wonder why this occurs. In our opinion, the huge increase in Google searches during the pandemic crisis can be due to two possible causes. The lockdown, which forced people to stay at home, and the recent possibility to use web tools to search a job and carry out administrative procedure such as unemployment benefits applications. This can explain why the use of Google has been more intense during the pandemic crisis compared to the financial crisis.

In any case, the model we developed is enough flexible to perform well even in the absence of relevant predictors. This is shown in Figure 3.8 in which we compare the volatility process of two BSTS models with dynamic shrinkage, one including all the predictors we mentioned, i.e. economic variables and Google Trends, and the other including only one predictor, i.e. the one period lag of the dependent variable. In the first case (left panel) the model is able to capture the predictable dependence through the (flexible) conditional mean. In the second case (right panel), instead, the shock is captured by the residuals' variance, which increase during the pandemic crisis.

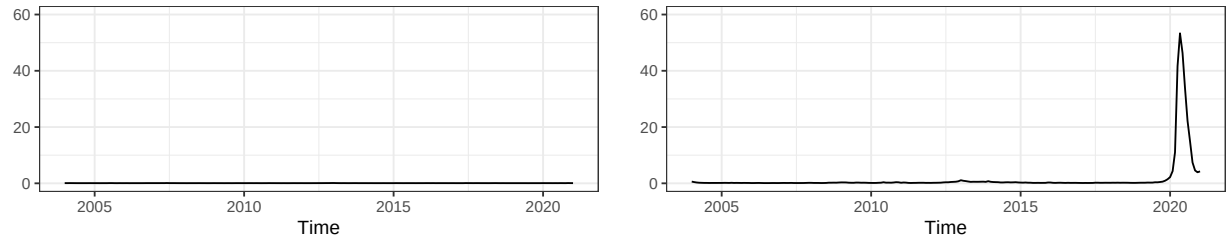


Figure 3.8: Left panel: MCMC approximation of the expected values of $\sigma_{\epsilon,t} = \exp(h_t/2)$ of a BSTS model with 36 predictors, including economic indicators and Google Trends data. Right panel: MCMC approximation of the expected values of $\sigma_{\epsilon,t} = \exp(h_t/2)$ of a BSTS model with one predictors, i.e. the one period lag of unemployment rate.

Let us focus on the forecasting performances. Because of the shock due to the pandemic crisis, that induces structural changes, we decided to split the time series in two periods:

from 2004-01-01, day in which Google Trends was released by Google Inc., to 2013-08-01 and from 2011-06-01 to 2020-12-01. The last 20 observations of each period form the testing set. The idea is to evaluate the model's performances both under standard conditions and when shocks occur. The objective is to show that BSTS with dynamic shrinkage and stochastic volatility represents a very flexible approach to time series modeling and forecasting. In Tables 3.2 and 3.3, forecasting performances of diverse specifications of Bayesian Structural Time Series models are compared. Under standard conditions, the model endowed with a seasonal component, which we call "seasonal model," provides better results. On the other hand, contrary to expectations, the full ($\Omega = 1$) non-seasonal model performs curiously better than the shrunk ($\Omega = 0.2$) non-seasonal model. This effect is due to the seasonality shown by some predictors that may help the model with more predictors to approximate somehow the seasonal component of the dependent variable. Note, however, that seasonal fluctuations observed in the explanatory variables are usually caused by weather effects, administrative measures or other events that may not affect significantly the dependent variable. This explains the better accuracy obtained using a seasonal component rather than extracting the seasonality from the predictors. On the other hand, regarding the quantification of the uncertainty, the similar SLPL between the models with and without seasonality when $\Omega = 1$ (as shown in Table 3.2) can be explained by the larger credible intervals due to the bigger size of the state vector in the seasonal model which tends to inflate the state variance and, consequently, the forecast variance.

On the other hand, by seasonally adjusting the predictors the resulting picture becomes much more consistent with the expectations. Indeed non-seasonal models are now completely unable to capture seasonal fluctuations. The adjustment of the series has been carried out using the X-13ARIMA-SEATS Seasonal Adjustment strategy of the Census Bureau.

So far we explained why a seasonal model should be preferred over a non-seasonal one. Regarding the comparison dynamic shrinkage versus no shrinkage, the results of this forecasting exercise are the following. Even though the predictive means are rather similar in both models, the reduction of the noise induced by shrinkage priors translates to narrower credible intervals and hence into an higher reliability of the point estimates. This peculiarity

makes the penalized model preferable under standard conditions. On the other hand, when big shocks occur, the larger credible intervals produced by the full model allow to cover most of the fluctuation observed in the time series of unemployment rate, whereas the penalized model seems unable to properly represent the uncertainty around the point forecasts. This happens because the conditional inclusion probabilities evolve smoothly over time and they are not adequate for sudden changes. Therefore, under regular conditions, BSTS models with dynamic shrinkage are very promising for time series analysis of high-dimensional non-stationary and not seasonally adjusted time series. In order to make them more suitable also for predicting shocking events, Ω could be modeled as a random variable with support $(0, 1]$, whose hyperparameters are set as a function of $\sigma^2_{\epsilon,t}$. Developments in this direction are not dealt with here.

Table 3.2: Not seasonally adjusted predictors

			From 01/12/2011 to 01/08/2013				From 01/04/2019 to 01/12/2020			
Dynamic	Seasonality	Ω	RMSE	WMAPE	MASE	SLPL	RMSE	WMAPE	MASE	SLPL
SSVS	Yes	0.2	0.991	0.103	0.483	-31.666	12.024	0.355	0.957	-132.723
	Yes	1	0.905	0.096	0.45	-40.387	10.428	0.325	0.875	-58.027
		0.2	2.095	0.207	0.973	-82.824	11.232	0.383	1.032	-121.247
		1	1.772	0.17	0.795	-41.912	10.53	0.333	0.898	-56.185
EMVS	Yes	0.2	1.312	0.141	0.602	-	8.365	0.352	0.949	-
	Yes	1	1.36	0.137	0.585	-	8.422	0.362	0.977	-
		0.2	1.8	0.181	0.847	-	9.61	0.373	1.006	-
		1	1.807	0.165	0.776	-	9.231	0.368	0.992	-

Table 3.3: Seasonally adjusted predictors

			From 01/12/2011 to 01/08/2013				From 01/04/2019 to 01/12/2020			
Dynamic	Seasonality	Ω	RMSE	WMAPE	MASE	SLPL	RMSE	WMAPE	MASE	SLPL
SSVS	Yes	0.2	1.019	0.107	0.504	-31.389	11.448	0.298	0.804	-174.256
	Yes	1	0.943	0.093	0.437	-39.315	10.541	0.293	0.791	-61.855
		0.2	2.083	0.212	0.994	-46.198	13.502	0.514	1.386	-116.424
		1	2.267	0.232	1.091	-50.392	9.901	0.32	0.864	-60.674
EMVS	Yes	0.2	1.042	0.148	0.628	-	10.693	0.305	0.822	-
	Yes	1	1.042	0.16	0.683	-	9.796	0.301	0.812	-
		0.2	2.076	0.198	0.931	-	12.781	0.429	1.158	-
		1	2.258	0.23	1.078	-	13.17	0.437	1.179	-

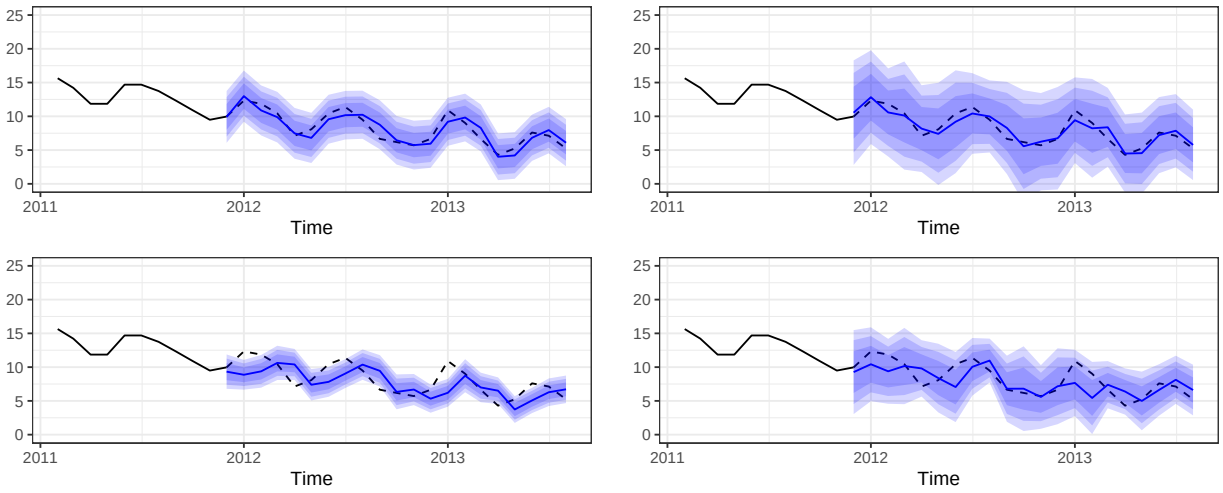


Figure 3.9: Not seasonally adjusted predictors. One-step-ahead forecasts (blue) with their credible intervals and the true series (black). Upper-left panel: Seasonal BSTS model with $\Omega = 0.2$. Bottom-left panel: Not seasonal BSTS model with $\Omega = 0.2$. Upper-right panel: Seasonal BSTS model with $\Omega = 1$. Bottom-right panel: Not seasonal BSTS model with $\Omega = 1$. These plots shows how the model behaviour under regular conditions.

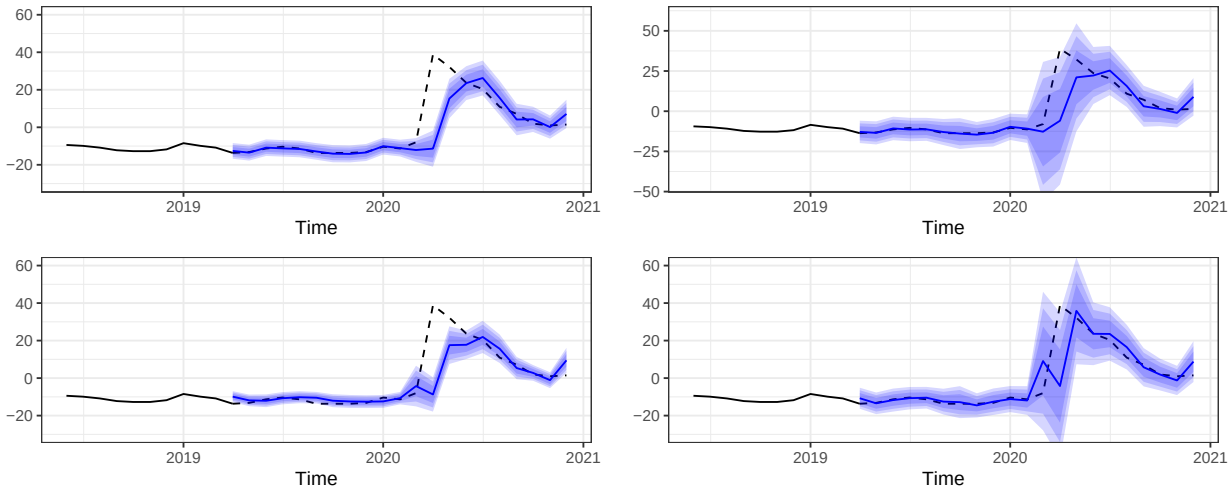


Figure 3.10: Not seasonally adjusted predictors. One-step-ahead forecasts (blue) with their credible intervals and the true series (black). Upper-left panel: Seasonal BSTS model with $\Omega = 0.2$. Bottom-left panel: Not seasonal BSTS model with $\Omega = 0.2$. Upper-right panel: Seasonal BSTS model with $\Omega = 1$. Bottom-right panel: Not seasonal BSTS model with $\Omega = 1$. These plots shows how the model behaviour when shocks occur.

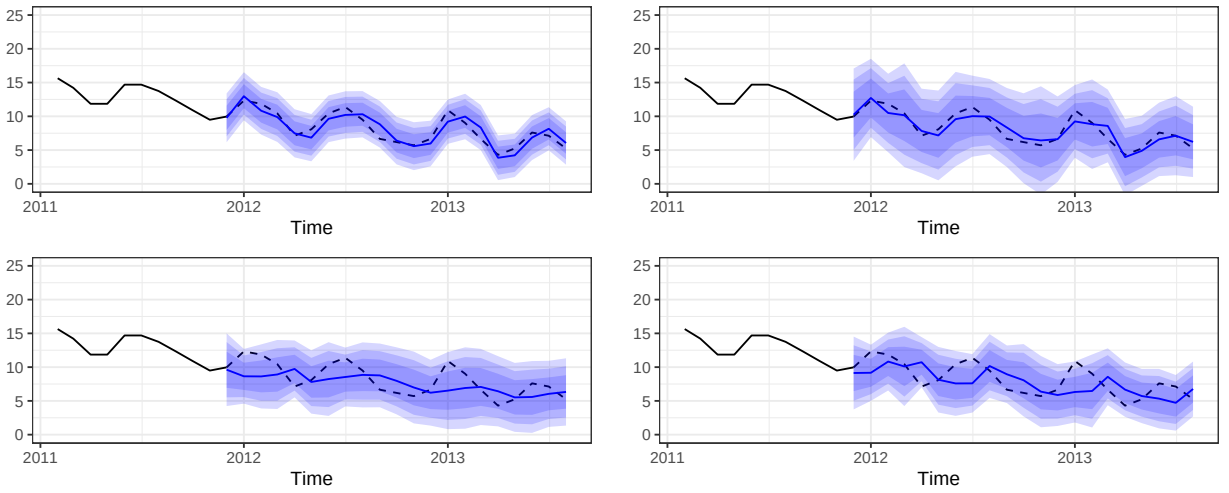


Figure 3.11: Seasonally adjusted predictors. One-step-ahead forecasts (blue) with their credible intervals and the true series (black). Upper-left panel: Seasonal BSTS model with $\Omega = 0.2$. Bottom-left panel: Not seasonal BSTS model with $\Omega = 0.2$. Upper-right panel: Seasonal BSTS model with $\Omega = 1$. Bottom-right panel: Not seasonal BSTS model with $\Omega = 1$. These plots shows how the model behaviour under regular conditions.

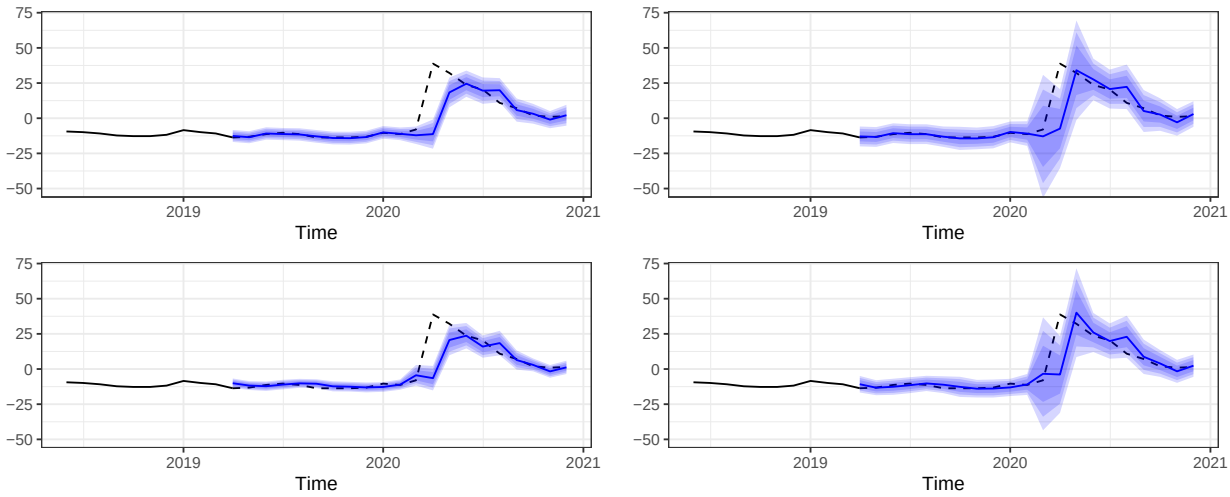


Figure 3.12: Seasonally adjusted predictors. One-step-ahead forecasts (blue) with their credible intervals and the true series (black). Upper-left panel: Seasonal BSTS model with $\Omega = 0.2$. Bottom-left panel: Not seasonal BSTS model with $\Omega = 0.2$. Upper-right panel: Seasonal BSTS model with $\Omega = 1$. Bottom-right panel: Not seasonal BSTS model with $\Omega = 1$. These plots shows how the model behaviour when shocks occur.

3.4 Discussion

Bayesian Structural Time Series models with Dynamic Spike-and-Slab regression and stochastic volatility represent the natural evolution of the model proposed by Scott and Varian (2013). For the same reasons we allow for stochastic fluctuations in the trend, slope, and seasonality, it is logical to model the remaining parameters as time-varying too. The assumption of constant regression coefficients may hold when it found justification on physical models or when the time series evolves over few years. On the other hand, this assumption is likely to be too limiting for most applications, particularly in the economic and social sciences. The analysis presented in the preceding paragraph demonstrates this fact by proving that some predictors, in that case Google Trends data, play an important role when the time series exhibits shocks that cause it to deviate from its natural pattern. Otherwise, the time series can be satisfactorily modeled using a simple BSTS model with trend and

seasonality. This raise two considerations. To begin with, simple models can occasionally operate admirably. In this scenario, adding predictors is not only pointless, but it also diminishes the model's interpretability, which is critical for economists. As a result, our novel class of BSTS models fits for the purpose since it is particularly parsimony oriented, permitting the intervention of additional predictors only when absolutely essential. Second, while time-varying coefficients can capture both permanent and transient phenomena like the pandemic crisis, time-varying residual variances hedge the researcher against biases that may arise as a consequence of the exclusion of some important predictors. The model indeed would still work without the need to adjust parameter estimates also in the latter case. Overall, the results obtained using both simulated and real data are consistent with our goal of developing a highly flexible model that can be used to analyze long macroeconomic time series under a variety of scenarios. On the other hand, the analysis brings out some limitations of our approach too. The most crucial ones have concern how to deal with the model's hyperparameters. In particular, Ω, which drives shrinkage globally, may be recalibrated in proximity of structural breaks. This would make the model even more flexible. Therefore, further developments on this issue will follow. A possible solution we envisage is to consider Ω as a random variable and letting its values changing according to the magnitude of shocking events such as the one exhibited by the unemployment rate in April 2020.

Chapter 4

Dynamic Shrinkage in Multivariate Time Series Models

Introducing dynamic shrinkage methods in multivariate time series models is undoubtedly interesting and challenging at the same time. Multivariate time series models are widely used and they are often necessary to study macroeconomic phenomena phenomena, that are characterized by the interactions of many economic variables so that univariate models can hardly capture all the dynamics of interest. A multivariate model allows to better describe past and contemporaneous relationships between the key variables. This however comes with some costs. When the number of dependent variables increases, the curse of dimensionality becomes more severe and it may eventually be unsustainable. For this reason, many authors have addressed their researches to the development of variable selection methods in this area. Nevertheless, dynamic shrinkage in multivariate time series analysis is still an open problem. With this chapter we want to provide a small contribution in this direction by embedding the Dynamic Spike-and-Slab process priors within the framework of Time-Varying Parameter Vector Autoregressive (TVP-VAR) models.

4.1 Time-Varying Parameter VAR Models

The introduction of TVP-VAR models into the literature was prompted by a wide range of phenomena observed in time series analysis between the 1960s and the 1990s. For instance, strong evidence suggests that unemployment and inflation in the United States were greater and more volatile in the 1960s and 1970s than in the 1980s and 1990s. Such changes in the model parameters are difficult to be modeled within the classical frequentist framework of VAR models and this encouraged researcher to adopt a Bayesian approach. In the last decades, many authors developed even more flexible and sophisticated models in which the regression parameters or the residual variances are subject to stochastic variations. Some interesting contributions in this direction come from Canova and Dellas (1993), Stock (2001) and Cogley and Sargent (2005), to mention a few.

The assumption of time-varying parameters should be natural. For example recent structural macroeconomic shocks such as the 2008 financial crisis and the pandemic crisis, would suggest time varying models, that are likely to become the standard in macroeconomics. However, the lack of a variable selection mechanism in these models has limited their potential for many interesting applications. This limitation has however stimulated a growing research in this topic. Here we decided to focus on the Time-Varying Parameter VAR model of Primiceri (2005) that had a certain impact in the econometric literature. Our contribution is to extend this model by allowing for dynamic variable selection through a Dynamic Spike-and-Slab Process Priors. With respect to previous approaches which focused on heteroschedasticity or time-varying regression coefficients, the model of Primiceri (2005) includes both of them. This model is particularly flexible since it provides a data-driven guidance in establishing whether the overall time variation of the linear structure is explained by a change in the size of the shocks, and hence in the volatility process, or a change in the propagation mechanism, and hence the coefficients. Formally, let $\boldsymbol{y}_t$ be an $n \times 1$ vector of observations, for $t = 1, ..., T$, the TVP-VAR is

$$\boldsymbol{y}_t = \boldsymbol{c}_t + \sum_{h=1}^{H} \boldsymbol{B}_{h,t} \boldsymbol{y}_{t-h} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \Omega_t) \tag{4.1}$$

where $\boldsymbol{c}_t$ is a $n \times 1$ vector of time-varying coefficients associated to a constant term, $\boldsymbol{B}_{h,t}$ (for $h = 1, ..., H$ number of lags) is a $n \times n$ matrix of time varying coefficients and ϵ_t is a vector of $n \times 1$ residuals. For identificability reasons we follow the proposal of Sims (1980) and we assume a lower triangular matrix $\boldsymbol{A}_t$ of contemporaneous relationship such that

$$\boldsymbol{A}_t \Omega_t \boldsymbol{A}_t' = \Sigma_t^{\frac{1}{2}} \Sigma_t'^{\frac{1}{2}}$$

and

$$\boldsymbol{A}_t = \begin{pmatrix} 1 & 0 & ... & 0 \\ \alpha_{21,t} & 1 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n1,t} & ... & \alpha_{nn-1,t} & 1 \end{pmatrix}$$

while

$$\Sigma_t = \begin{pmatrix} \sigma_{1,t}^2 & 0 & ... & 0 \\ 0 & \sigma_{2,t}^2 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & ... & 0 & \sigma_{n,t}^2 \end{pmatrix}$$

A more convenient way to write equation (4.1) is

$$\boldsymbol{y}_t = \boldsymbol{c}_t + \sum_{h=1}^{H} \boldsymbol{B}_{h,t} \boldsymbol{y}_{t-h} + \boldsymbol{A}_t^{-1} \Sigma_t^{\frac{1}{2}} \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$$

This equation can be re-written more compactly in state space form. Let $\boldsymbol{X}_t' \equiv \boldsymbol{I}_n \otimes (1, \boldsymbol{y}_{t-1}', ..., \boldsymbol{y}_{t-H}')$ and $\boldsymbol{\beta}_t \equiv vec([\boldsymbol{c}_t, \boldsymbol{B}_{1,t}, ..., \boldsymbol{B}_{H,t}]')$, then

$$\boldsymbol{y}_t = \boldsymbol{X}_t' \boldsymbol{\beta}_t + \boldsymbol{A}_t^{-1} \Sigma_t^{\frac{1}{2}} \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n) \tag{4.2}$$

Primiceri (2005) considers the regression coefficients $\boldsymbol{\beta}_t$, the elements $\alpha_{ij,t}$ of the matrix $\boldsymbol{A}_t$, and the logarithm of the diagonal elements of Σ_t to behave like random walks. Even though this specification has the advantage of reducing the number of unknown parameters which is already large, it also entails undesirable implications from a theoretical point of view since random walk processes are intrinsically unstable. Moreover, although Rockova and McAlinn (2021a) observe that it would be possible to extend the DSS approach to a random walk slab process by reformulating opportunely the transition weights, one must be careful to avoid the series $\omega_{1:T}$ being too unstable. This would indeed lead to erratic transitions from the

spike to slab and vice versa. Therefore, we will stick to the considerations made in Chapter 2 and assume the regression coefficients to follow independent stationary Gaussian AR(1) processes. The dynamics of the model is described by the following equations:

$$\begin{aligned} \boldsymbol{y}_t &= \boldsymbol{X}_t'\boldsymbol{\beta}_t + \boldsymbol{A}_t\Sigma^{\frac{1}{2}}\boldsymbol{\epsilon}_t, \\ \boldsymbol{\beta}_t &= \boldsymbol{G}_t^{(\beta)}\boldsymbol{\beta}_{t-1} + \boldsymbol{\xi}_t, \\ \boldsymbol{\alpha}_t &= \boldsymbol{G}_t^{(\alpha)}\boldsymbol{\alpha}_{t-1} + \boldsymbol{\eta}_t, \\ \boldsymbol{h}_t &= \boldsymbol{\mu} + \boldsymbol{R}(\boldsymbol{h}_{t-1} - \boldsymbol{\mu}) + \boldsymbol{\zeta}_t \end{aligned} \tag{4.3}$$

where $\boldsymbol{G}_t^{(\beta)} = diag\{\gamma_{t,j}^{(\beta)}\phi_1\}_{j=1}^{p_\beta}$, $\boldsymbol{G}_t^{(\alpha)} = diag\{\gamma_{t,j}^{(\alpha)}\phi_1\}_{j=1}^{p_\alpha}$, $\boldsymbol{\mu} = (\mu_1, ..., \mu_n)'$ and $\boldsymbol{R} = diag(\rho_1, ..., \rho_n)$. The parameter ϕ_1 will be set equal to 0.98 in the applications that will follow. We acknowledge that such assumption may seem too strong, however empirical analysis proved that fixing ϕ_1 close to one leads to satisfying results while preserving stationary. On the other hand, it would be possible to assign a Beta distribution to this parameter and update its posterior; however, this would inevitably increase the amount of unknown parameters and add further steps into the MCMC scheme with the consequent increase in the running time (which is already long). Regarding the choice of $\boldsymbol{\beta}_0$ and $\boldsymbol{\alpha}_0$, one may assume a Gaussian distribution centered on the least squares estimates computed on a subsample of the data or on the whole sample with large variance. Alternatively, another strategy is to model $\boldsymbol{\beta}_0 \sim \mathcal{N}(\boldsymbol{m}_0, \boldsymbol{C}_0)$ where $\boldsymbol{m}_0 = \phi_1\boldsymbol{\gamma}_0^{(\beta)}$ and $\boldsymbol{C}_0 = diag\{\gamma_{0,j}^{(\beta)}\lambda_1/(1+\phi_1) + (1-\gamma_{0,j}^{(\beta)})\lambda_0\}_{j=1}^{p_\beta}$ and same for $\boldsymbol{\alpha}_0$.

In this model we distinguish between two types of auxiliary variables: $\gamma_{t,j}^{(\beta)}$ and $\gamma_{t,j}^{(\alpha)}$. The reason behind this separation is that we consider relationships between contemporaneous variables more meaningful with respect to the ones with lagged values of the variables. Therefore we would set Ω_α close to one and Ω_β commensurate to the number of lags of our model such that models with more lags are shrunk with greater severity. Our proposal is to assume that both $\beta_{1:T,j}$ and $\alpha_{1:T,k}$ for $j = 1, ..., p_\beta$ and $k = 1, ..., p_\alpha$ are i.i.d. distributed with DSS priors. That is, we assume that the generic β_t has a mixture density prior of the kind

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

where

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

and

$$P(\gamma_t = 1|\beta_{t-1}) = \omega_t^{(\beta)}$$

The inclusion probabilities $\omega_t^{(\beta)}$ are definite exactly as in Chapter 2 and they depend on Ω_β. Analogously we assign DSS priors on α_t. Finally, we assume that

$$(\boldsymbol{\xi}_t, \boldsymbol{\eta}_t, \boldsymbol{\zeta}_t)' \sim \mathcal{N}(\mathbf{0}, \boldsymbol{W}_t)$$

where

$$\boldsymbol{W}_t = \begin{pmatrix} \Lambda_\beta & 0 & 0 \\ 0 & \Lambda_\alpha & 0 \\ 0 & 0 & \Sigma_\zeta \end{pmatrix}$$

with $\Lambda_\beta = \{\gamma_{t,j}^{(\beta)}\lambda_1 + (1-\gamma_{t,j}^{(\beta)})\lambda_0\}_{j=1}^{p_\beta}$, $\Lambda_\alpha = \{\gamma_{t,j}^{(\alpha)}\lambda_1 + (1-\gamma_{t,j}^{(\alpha)})\lambda_0\}_{j=1}^{p_\alpha}$ and $\Sigma_\zeta = diag(\sigma^2_{\zeta,1}, ..., \sigma^2_{\zeta,n})$. The hyperparameters λ_1 and λ_0 must be fixed in such a way that allows a sufficiently large ratio between spike and slab variances (George and McCulloch 1993). For example, in the empirical study of Section 4.2, we notice that fixing $\lambda_1 = 0.1$ and $\lambda_0 = 0.01$ provides satisfying results. In general, we recommend to repeat the study for different values of λ_0 and λ_1 and pick the ones that produce the best predictive performances. The parameters $\sigma^2_{\zeta,i}$ for $i = 1, ..., n$ are independently estimated using the sampling strategy described in Section 2.2.1.

4.1.1 Dynamic Stochastic Search Variable Selection

Efficient posterior sampling for the TVP-VAR model with dynamic shrinkage can be performed using a Dynamic SSVS strategy. A possible sampling scheme is the one proposed by Primiceri (2005) that we extend in order to deal with the dynamic shrinkage components, as reported below. The overall strategy, resumed in Algorithm 7. We here detail the main steps. Let $\boldsymbol{\gamma} = (\boldsymbol{\gamma}^{(\beta)}, \boldsymbol{\gamma}^{(\alpha)})$ then

- Step 1; Conditionally on $(\boldsymbol{y}, \boldsymbol{\Sigma}, \boldsymbol{A}, \boldsymbol{\gamma}, \boldsymbol{W})$, the space state model is linear and Gaussian

Algorithm 7: Dynamic Shrinkage in Time-Varying Parameter VAR models

1 Draw $\boldsymbol{\beta}$ from $\pi(\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{\Sigma}, \boldsymbol{A}, \boldsymbol{\gamma}, \boldsymbol{W})$;

2 Draw $\boldsymbol{A}$ from $\pi(\boldsymbol{A}|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\Sigma}, \boldsymbol{W})$;

3 Draw $\gamma_{j,t}^{(\beta)}$ individually and independently from $\pi(\gamma_{j,t}^{(\beta)}|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{A}, \boldsymbol{\Sigma}, \boldsymbol{W}, \boldsymbol{\gamma}_{-j,t}^{(\beta)})$ and $\gamma_{j,t}^{(\alpha)}$ individually and independently from $\pi(\gamma_{j,t}^{(\alpha)}|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{A}, \boldsymbol{\Sigma}, \boldsymbol{W}, \boldsymbol{\gamma}_{-j,t}^{(\alpha)})$;

4 Compute Λ_β and Λ_α;

5 Draw $\boldsymbol{\Sigma}$ from $\pi(\boldsymbol{\Sigma}|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{A}, \boldsymbol{\gamma}, \boldsymbol{W}, \boldsymbol{\mu}, \boldsymbol{R}, \Sigma_\zeta)$;

6 Draw $\boldsymbol{\mu}, \boldsymbol{R}, \Sigma_\zeta$ independently from each time series using the passages illustrated in section 2.2.1 ;

with known variance, therefore draws from $\pi(\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{\Sigma}, \boldsymbol{A}, \boldsymbol{\gamma}, \boldsymbol{W})$ can be obtained through FFBS algorithm.

- Step 2; Let $\boldsymbol{y}_t - \boldsymbol{X}_t'\boldsymbol{\beta}_t = \hat{\boldsymbol{y}}_t$, which is observable once $\boldsymbol{\beta}_t$ is sampled, and note that equation (4.2) can be rewritten as

$$\boldsymbol{A}_t\hat{\boldsymbol{y}}_t = \Sigma_t^{\frac{1}{2}}\boldsymbol{\epsilon}_t \tag{4.4}$$

Given the lower triangular shape of $\boldsymbol{A}_t$ with ones on its diagonal, then equation (4.4) can be seen as a State-Space Model

$$\hat{\boldsymbol{y}}_t = \boldsymbol{Z}_t\boldsymbol{\alpha}_t + \Sigma_t^{\frac{1}{2}}\boldsymbol{\epsilon}_t \tag{4.5}$$

where

$$\boldsymbol{Z}_t = \begin{pmatrix} 0 & \dots & \dots & 0 \\ -\hat{y}_{1,t} & 0 & \dots & \\ 0 & -\hat{y}_{[1,2],t} & \dots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & -\hat{y}_{[1,\dots,n-1],t} \end{pmatrix}$$

where $\hat{y}_{[1,\dots,n-1],t} = (\hat{y}_{1,t}, \dots, \hat{y}_{n-1,t})$. It is evident that, even in Step 2, FFBS algorithm can be used to sample from the full conditional distribution of $\boldsymbol{A}_t$.

- Step 3; Samples of the indicators $\gamma_{t,j}$ are obtained individually and independently from

their full conditional distribution as in Step 2 of Algorithm 3.

- Step 4; Compute $\Lambda_\beta = \{\gamma_{t,j}^{(\beta)}\lambda_1 + (1-\gamma_{t,j}^{(\beta)})\lambda_0\}_{j=1}^{p_\beta}$ and $\Lambda_\alpha = \{\gamma_{t,j}^{(\alpha)}\lambda_1 + (1-\gamma_{t,j}^{(\alpha)})\lambda_0\}_{j=1}^{p_\alpha}$.
- Step 5; Let $\boldsymbol{A}_t(\boldsymbol{y}_t - \boldsymbol{X}_t'\boldsymbol{\beta}_t) = \boldsymbol{y}_t^*$ then

$$\boldsymbol{y}_t^* = \Sigma_t^{\frac{1}{2}}\boldsymbol{\epsilon}_t,$$

with $\Sigma_t = diag(\exp(h_{1,t}), ..., \exp(h_{n,t}))$, where

$$h_{i,t} = \mu_i + \rho_i(h_{i,t-1} - \mu_i) + \zeta_{i,t}, \quad \zeta_{i,t} \overset{iid}{\sim} \mathcal{N}(0, \sigma_{i,\zeta}^2)$$

for $i \in \{1, ..., n\}$ and $t \in \{1, ..., T\}$. The latter represents a stochastic volatility model from which samples can be drawn using the sampling strategy of Kastner (2016) independently for each univariate time series, which is possible thanks to the diagonal structure of Σ_t.

- Step 6; Draw $\boldsymbol{\mu}, \boldsymbol{R}, \Sigma_\zeta$ independently from each time series using the passages illustrated in section 2.2.1

The sampling strategy described above is very intuitive, however it has an important limit: the computational cost. Given that the Kalman filter's computational complexity is linear in data length but quadratic in the state vector dimension, it's easy to see how the FFBS strategy would be incredibly slow in large TVP-VAR models. Therefore, following Eisenstat, Chan, and Strachan (2014) and Chan and Eisenstat (2018), we replace the FFBS steps with a precision sampler. Moreover, the author propose the following model specification which is shown to greatly increase the sampler's efficiency in Structural VAR models:

$$\boldsymbol{y}_t = \boldsymbol{X}\boldsymbol{\beta}_t + \boldsymbol{W}\boldsymbol{\alpha}_t + \Sigma_t^{\frac{1}{2}}\boldsymbol{\epsilon}_t$$

where $\boldsymbol{X}_t \equiv \boldsymbol{I}_n \otimes (1, \boldsymbol{y}_{t-1}', ..., \boldsymbol{y}_{t-H}')$ and $\boldsymbol{W}$ is a $n \times n(n-1)/2$ matrix such that

$$\boldsymbol{W} = \begin{pmatrix} 0 & ... & ... & 0 \\ -y_{1,t} & 0 & ... & \\ 0 & -y_{[1,2],t} & ... & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & -y_{[1,...,n-1],t} \end{pmatrix},$$

that allows to simultaneously draw $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$. Clearly, its State Space Model form is

$$\boldsymbol{y}_t = \tilde{\boldsymbol{X}}_t \boldsymbol{\theta}_t + \boldsymbol{\epsilon}_t$$

with $\tilde{\boldsymbol{X}}_t = (\boldsymbol{X}_t, \boldsymbol{W}_t)$ and $\boldsymbol{\theta}_t = (\boldsymbol{\beta}_t', \boldsymbol{\alpha}_t')'$. Details on the precision sampler have been already provided in section 2.2.2, here we provide a brief recap.

- Step 1; Let us recall the State Space representation of a TVP-VAR model

$$\underset{(T\times n)\times 1}{\boldsymbol{y}} = \underset{(T\times n)\times p}{\tilde{\boldsymbol{X}}} \; \underset{p\times 1}{\boldsymbol{\theta}} + \underset{(T\times n)\times 1}{\boldsymbol{\epsilon}}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\left(\underset{(T\times n)\times 1}{\mathbf{0}}, \underset{(T\times n)\times (T\times n)}{\boldsymbol{\Sigma}_\epsilon} \right)$$

  where $\boldsymbol{\epsilon} = (\boldsymbol{\epsilon}_1', ..., \boldsymbol{\epsilon}_T')'$, $\boldsymbol{\Sigma}_\epsilon = diag(\Sigma_{\epsilon,1}, ..., \Sigma_{\epsilon,T})$ and $\tilde{\boldsymbol{X}} = diag(\tilde{\boldsymbol{X}}_1, ..., \tilde{\boldsymbol{X}}_T)$. Remember that the latent process of the stochastic coefficients evolves as

$$\boldsymbol{\theta}_t = \boldsymbol{G}_t \boldsymbol{\theta}_{t-1} + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \Lambda_t)$$

  where in the DSS scheme $\boldsymbol{G}_t = diag\{\gamma_{j,t}\phi_1\}_{j=1}^p$ and $\Lambda_t = diag\{\gamma_{j,t}\lambda_1 + (1-\gamma_{j,t})\lambda_0\}_{j=1}^p$. Define the matrix

$$\boldsymbol{D} = \begin{pmatrix} \boldsymbol{I}_k & 0 & ... & 0 \\ -\boldsymbol{G}_1 & \boldsymbol{I}_k & ... & 0 \\ ... & ... & ... & 0 \\ 0 & ... & -\boldsymbol{G}_T & \boldsymbol{I}_k \end{pmatrix} \tag{4.6}$$

  Therefore we can write

$$\boldsymbol{D}\boldsymbol{\theta} = \tilde{\boldsymbol{\alpha}}_0 + \boldsymbol{\xi}, \quad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{S}_{\boldsymbol{\theta}})$$

  where $\tilde{\boldsymbol{\alpha}}_0 = (\boldsymbol{\theta}_0', \mathbf{0}, ...\mathbf{0})'$ and $\boldsymbol{S}_{\boldsymbol{\theta}} = diag(\Lambda_1, ..., \Lambda_T)$. Equivalently we write

$$\boldsymbol{\theta}|\boldsymbol{\theta}_0, \boldsymbol{\gamma} \sim \mathcal{N}(\boldsymbol{D}^{-1}\tilde{\boldsymbol{\alpha}}_0, (\boldsymbol{D}'\boldsymbol{S}_{\boldsymbol{\theta}}{}^{-1}\boldsymbol{D})^{-1})$$

  and we label $\boldsymbol{\alpha}_0 = \boldsymbol{D}^{-1}\tilde{\boldsymbol{\alpha}}_0$. Thanks to Corollary 8.1 of Theorem 8.1 of Kroese and Chan. (2014), that we mentioned in Section 1.1, we can sample from

$$\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{h}, \boldsymbol{\gamma}, \boldsymbol{\theta}_0, \boldsymbol{h}_0 \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \boldsymbol{K}_{\boldsymbol{\theta}}^{-1})$$

  where $\hat{\boldsymbol{\theta}} = \boldsymbol{K}_{\boldsymbol{\theta}}^{-1}\boldsymbol{d}_{\boldsymbol{\theta}}$, $\boldsymbol{K}_{\boldsymbol{\theta}} = \boldsymbol{D}'\boldsymbol{S}_{\boldsymbol{\theta}}{}^{-1}\boldsymbol{D} + \tilde{\boldsymbol{X}}'\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-1}\tilde{\boldsymbol{X}}$ and $\boldsymbol{d}_{\boldsymbol{\theta}} = \boldsymbol{D}'\boldsymbol{S}_{\boldsymbol{\theta}}{}^{-1}\boldsymbol{D}\boldsymbol{\alpha}_0 + \tilde{\boldsymbol{X}}'\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-1}\boldsymbol{y}$.

- Step 2; Sample $\boldsymbol{\theta}_0$ from the full conditional distribution

$$(\boldsymbol{\theta}_0|\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{h}, \boldsymbol{\Lambda}_0) \sim \mathcal{N}(\hat{\boldsymbol{\theta}}_0, \boldsymbol{K}_{\boldsymbol{\theta}_0}{}^{-1})$$

where $\boldsymbol{K}_{\boldsymbol{\theta_0}} = \boldsymbol{C}_0^{-1} + \Lambda_0^{-1}$ and $\hat{\boldsymbol{\theta}}_0 = \boldsymbol{K}_{\boldsymbol{\theta_0}}{}^{-1}(\boldsymbol{C}_0^{-1}\boldsymbol{m}_0 + \Lambda_0^{-1}\boldsymbol{\theta}_1)$, with $\boldsymbol{m}_0$ and C_0 respectively the prior mean and the prior variance of the state vector.

- Step 3; Sample individually and independently the indicators $\gamma_{j,t}$ from their full conditional distribution as in Step 2 of Algorithm 3.
- Step 4; Compute $\Lambda_t = diag\{\gamma_{j,t}\lambda_1 + (1-\gamma_{j,t})\lambda_0\}_{j=1}^{p}$.
- Step 5; Compute $\boldsymbol{D}$ as in equation (2.17).
- Step 6; Compute $\boldsymbol{r} = \boldsymbol{y} - \tilde{\boldsymbol{X}}\boldsymbol{\theta}$ and use the residuals $\boldsymbol{r}_i = (r_{1,i}, ..., r_{T,i})'$ (for $i = 1, ..., n$) to perform the AWOL-ASIS strategy of Kastner (2016).

In the following section we provide an empirical example on macroeconomic data that will be useful to illustrate the performance of dynamic shrinkage in VAR models and to compare these two alternative estimation strategies.

4.2 Macroeconomic Data

The dataset used for this empirical exercise is included in the package `bvarsv` and it contains quarterly time series data of inflation rate, unemployment rate and treasury bill interest rate for the US. Data are standardized and made stationary using log-differences. Overall, the data set covers the period from March 1953 to June 2015. In the current exercise the focus is on forecasting, therefore identification plays a secondary role in this analysis. However, the Structural VAR is built in such a way to allow for identification and, possibly, to assess Granger causality or to compute impulse response functions. The TVP-VAR we consider for this analysis is characterized by the three variables just mentioned, i.e. inflation, unemployment and interest rate, with eight lags for each of them. In other words, the model comprises a total of 75 regression coefficients to be estimated, plus the time-varying parameters included in the covariance matrix. As mentioned before, identification is made possible by a Cholesky scheme (or triangular scheme). Therefore, the order of the variables

entering the vector $\boldsymbol{Y}_t$ must be chosen accurately. This issue is actually less important when we are interested in forecasting, however we decided to follow the strategy used by Primiceri (2005) for monetary policy shock identification. Clearly, the interest rate is ordered last since we can assume that the monetary policy authority responds to a monetary policy shock according to the Taylor rule

$$i_t = \pi_t + i_t^* + a_\pi(\pi_t - \pi_t^*) + a_y(y_t - \bar{y}_t),$$

where π_t is the current inflation rate, i_t^* is the natural interest rate, and $(\pi_t - \pi_t^*)$ and $(y_t - y_t^*)$ are respectively the deviations of inflation and output from their natural level. Unemployment is ordered second and finally inflation is ordered first. According to the author, this is more a normalization rather than an identification condition. This scheme implies that unemployment and inflation do not respond at a time t to a contemporaneous monetary shock, represented by the innovation term of the interest rate, but the response occurs only after some periods. Figures 4.1 – 4.3, show heat maps comparing a model without shrinkage ($\Omega_\beta = 1$) to a model with a quite severe degree of shrinkage ($\Omega_\beta = 0.2$). The remaining model's hyperparameters are set accordingly to previous applications: $\lambda_0 = 0.01$, $\lambda_1 = 0.1$ and $\phi_1 = 0.98$. The stochastic volatility process has parameters' priors: $\alpha_0 \sim \mathcal{N}(-10, 100)$, $\alpha_2 \sim \mathcal{B}(20, 1.5)$ and $\sigma_\zeta^2 \sim \mathcal{IG}(0.5, 0.5)$. As the heat maps show, the introduction of dynamic shrinkage priors produces a drastic change in the coefficients values. Before shrinkage, the signs and the magnitude of the regression coefficients estimated is consistent with economic theory. The interest rate for example is negatively correlated with unemployment rate and its one period lag and positively correlated with inflation rate and its one period lag, while this relationships became less clear for further lags. The relationship between unemployment and inflation is less stable and it depends on the lag considered. Overall, we would expect an overall negative relationship which is more evident in the equation of inflation than in the one of unemployment. In any case, what we want to highlight is the drastic change of these relationships once dynamic shrinkage occurs. Indeed, when $\Omega_\beta = 0.2$ the heat maps show that for each response variable the most significant predictor is its lagged values. These results are surprising, but not too much. In fact, it is reasonable to think that the most important

coefficients are the most recent ones and that the signal dissipates as lags increase. In addition, because of the instability of the relationships among lagged variables, a dynamic shrinkage priors is likely to make a safe choice, entrusting the highest predictive power to those predictors that show a consistent behavior over time.

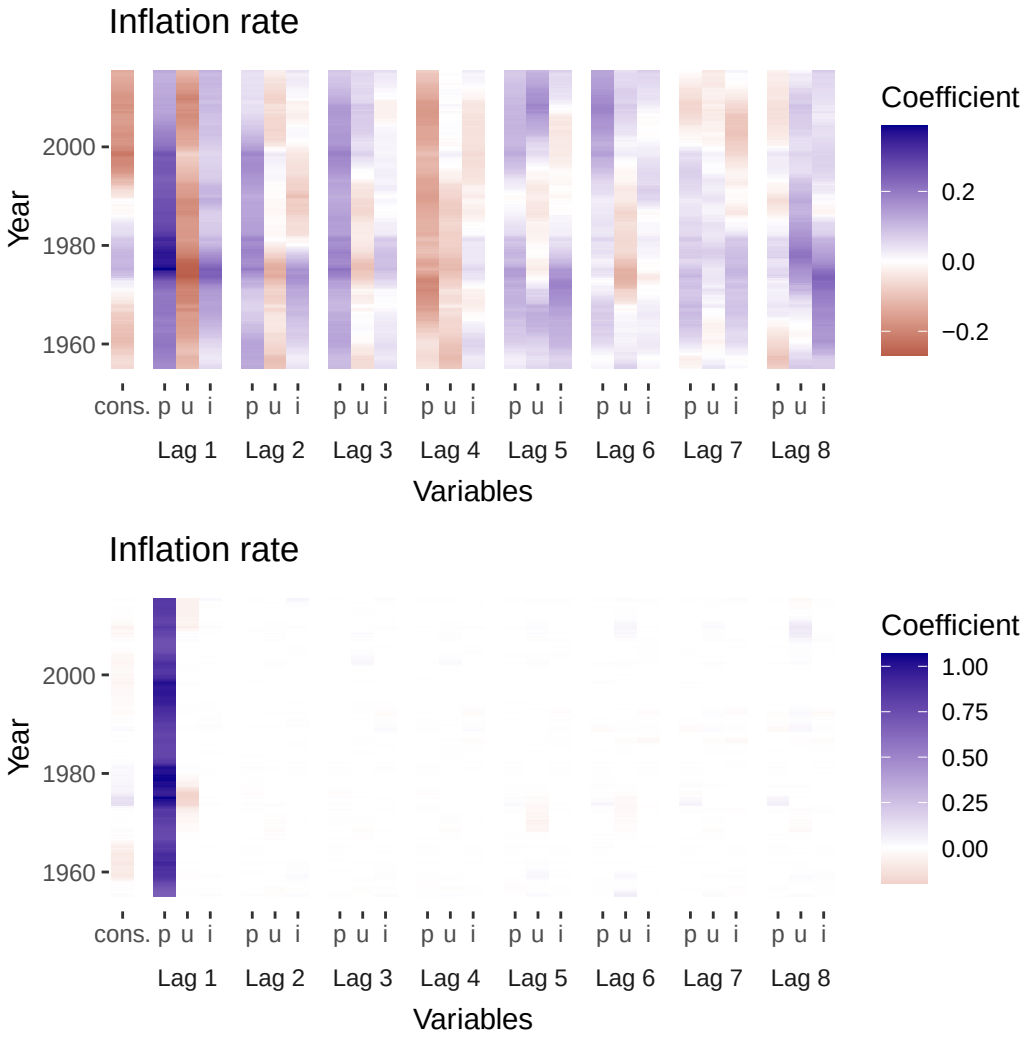


Figure 4.1: Heat map representing estimated time-varying coefficients of the lagged VAR variables affecting inflation. In the upper panel, a VAR(8) model without shrinkage ($\Omega_\beta = 1$) is presented. In this case, the signal is redistributed across all the predictors. In the lower panel, the same VAR(8) model is estimated when a severe shrinkage ($\Omega_\beta = 0.2$) applies. Here, the signal is concentrated on the first lag of the dependent variable.

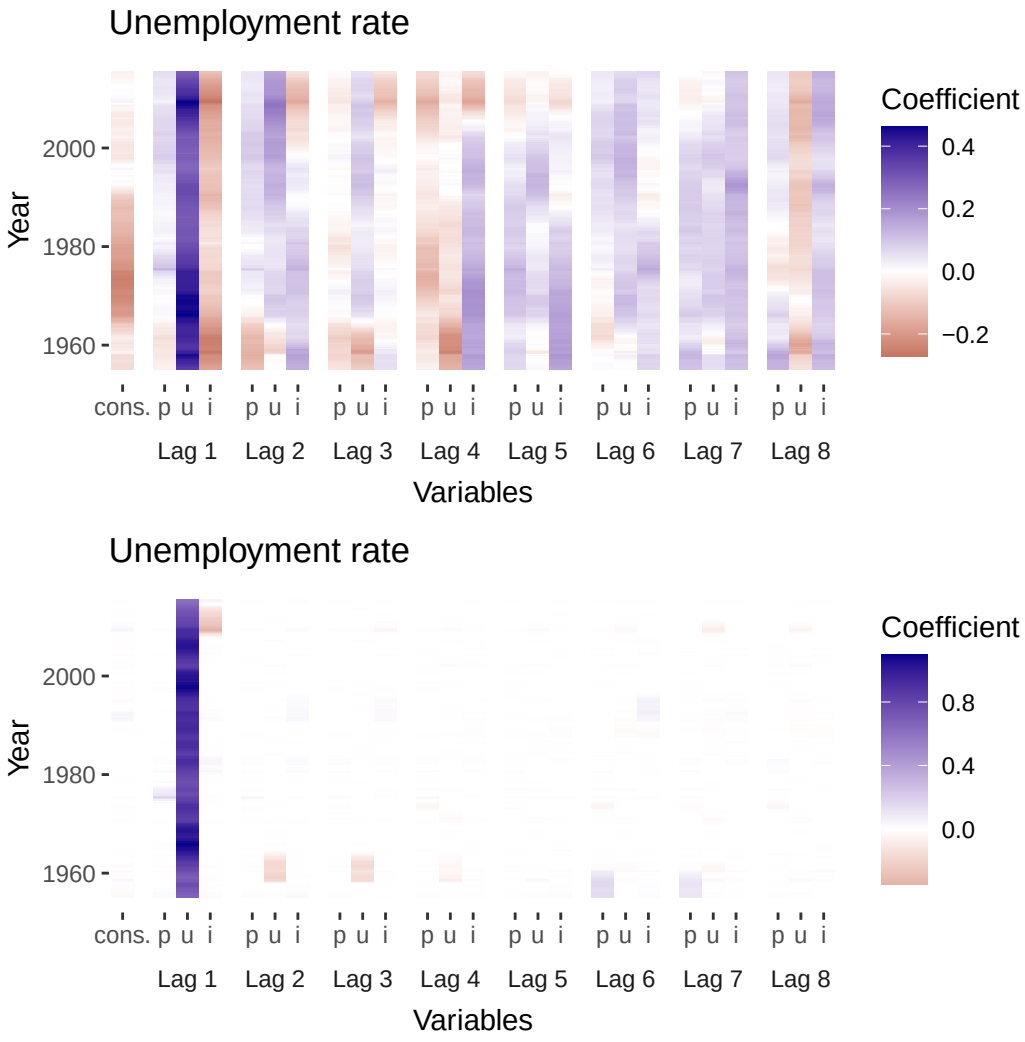


Figure 4.2: Heat map representing estimated time-varying coefficients of the lagged VAR variables affecting unemployment rate. In the upper panel, a VAR(8) model without shrinkage ($\Omega_\beta = 1$) is presented. In this case, the signal is redistributed across all the predictors. In the lower panel, the same VAR(8) model is estimated when a severe shrinkage ($\Omega_\beta = 0.2$) applies. Here, the signal is concentrated on the first lag of the dependent variable.

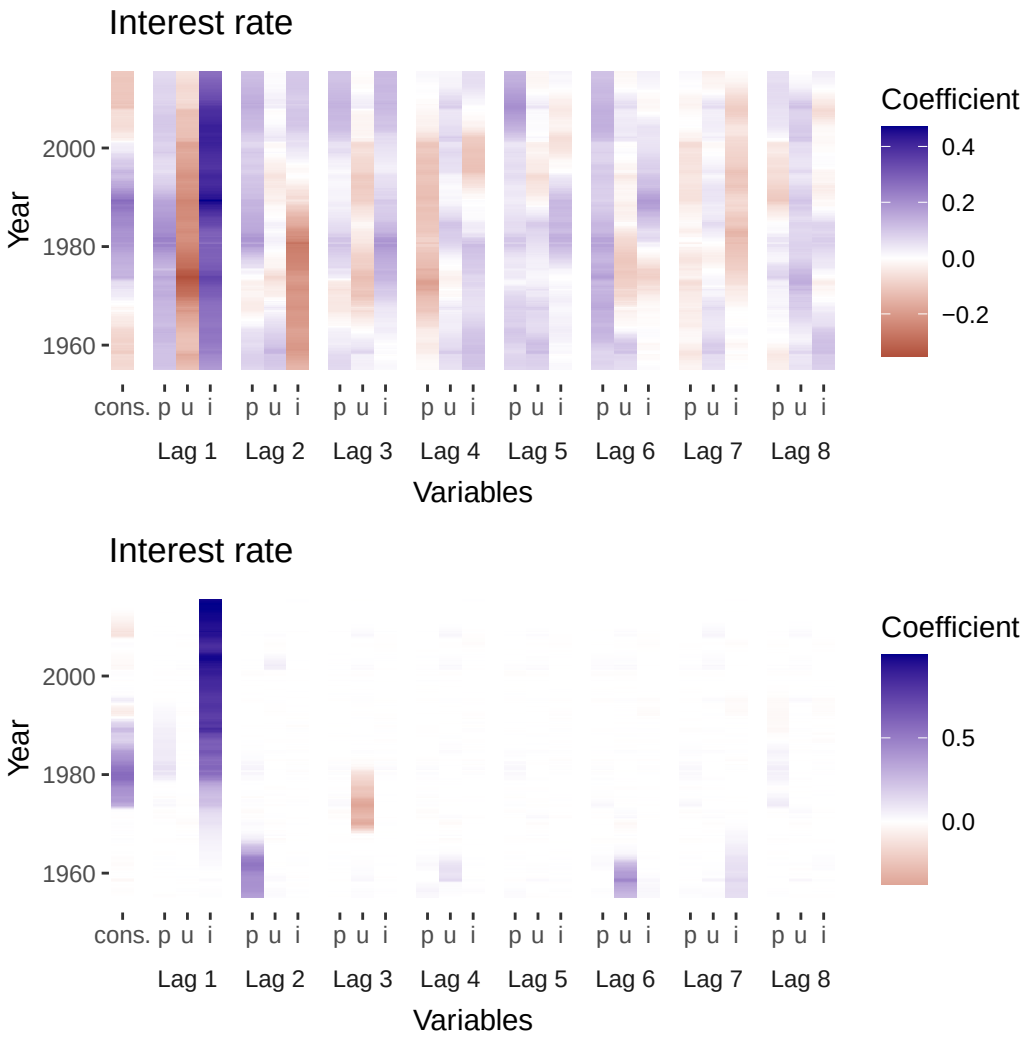


Figure 4.3: Heat map representing estimated time-varying coefficients of the lagged VAR variables affecting interest rate. In the upper panel, a VAR(8) model without shrinkage ($\Omega_\beta = 1$) is presented. In this case, the signal is redistributed across all the predictors. In the lower panel, the same VAR(8) model is estimated when a severe shrinkage ($\Omega_\beta = 0.2$) applies. Here, the signal is concentrated on the first lag of the dependent variable.

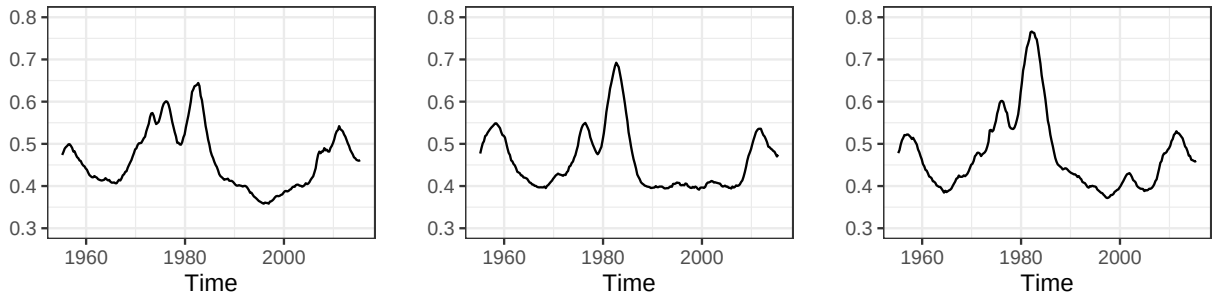


Figure 4.4: MCMC approximation of the expected value of the residual standard deviations using precision sampler of inflation rate (left panel), unemployment rate (central panel), interest rate (right panel).

We now test the out-of-sample performances of the models. Therefore, one-step-ahead forecast distributions are approximated via MCMC for the last ten periods of the series. The results are reported in Table 4.1 and they show that point forecasts improve for some variables (interest rate and unemployment rate) when shrinkage applies. Gains can be rather minor as in the case of inflation and unemployment or very large as for the interest rate. Values of WMAPE and MASE are in general acceptable. The huge values of MASE for interest rate are motivated by the fact that interest rates have been kept close to the zero lower bound in the effort of stimulating the economy after the 2008 financial crisis. Therefore the denominator of MASE is almost near zero and the metric tends to be large by construction. The most impressive gain lies in the SLPL. Indeed, noise reduction due to shrinkage of unimportant coefficients toward zero lead to smaller credible intervals and, thus, more precise estimates. From a visual standpoint, this effect is shown in Figure 4.5.

Using the precision sampler we obtain very similar results but with a significant gain in running time. For 1000 iterations the Dynamic SSVS with a precision sampler scheme takes 170.76 seconds against the 1835.97 seconds of the Dynamic SSVS with double FFBS scheme. This fact leads to new opportunities in the field of large TVP-VAR since it allows to deal with a vast amount of coefficients in a reasonable amount of time and, at the same time, it avoids overfitting thanks to dynamic shriankage.

Table 4.1: Dynamic SSVS with FFBS: performance comparison.

	Ω	RMSE	WMAPE	MASE	SLPL
Inflation rate	0.2	0.121	0.159	0.998	-18.665
	1	0.127	0.135	0.846	-25.84
Unemployment rate	0.2	0.116	0.055	0.813	-17.96
	1	0.16	0.065	0.966	-25.821
Interest rate	0.2	0.097	0.063	10.783	-16.722
	1	0.235	0.155	26.465	-27.216
Total	0.2				-53.343
	1				-78.879

Table 4.2: Dynamic SSVS with precision sampler: performance comparison.

	Ω	RMSE	WMAPE	MASE	SLPL
Inflation rate	0.2	0.114	0.132	0.827	-17.604
	1	0.111	0.128	0.805	-21.414
Unemployment rate	0.2	0.11	0.053	0.791	-17.219
	1	0.137	0.065	0.966	-21.861
Interest rate	0.2	0.062	0.042	7.119	-14.875
	1	0.147	0.095	16.116	-22.13
Total	0.2				-49.697
	1				-65.405

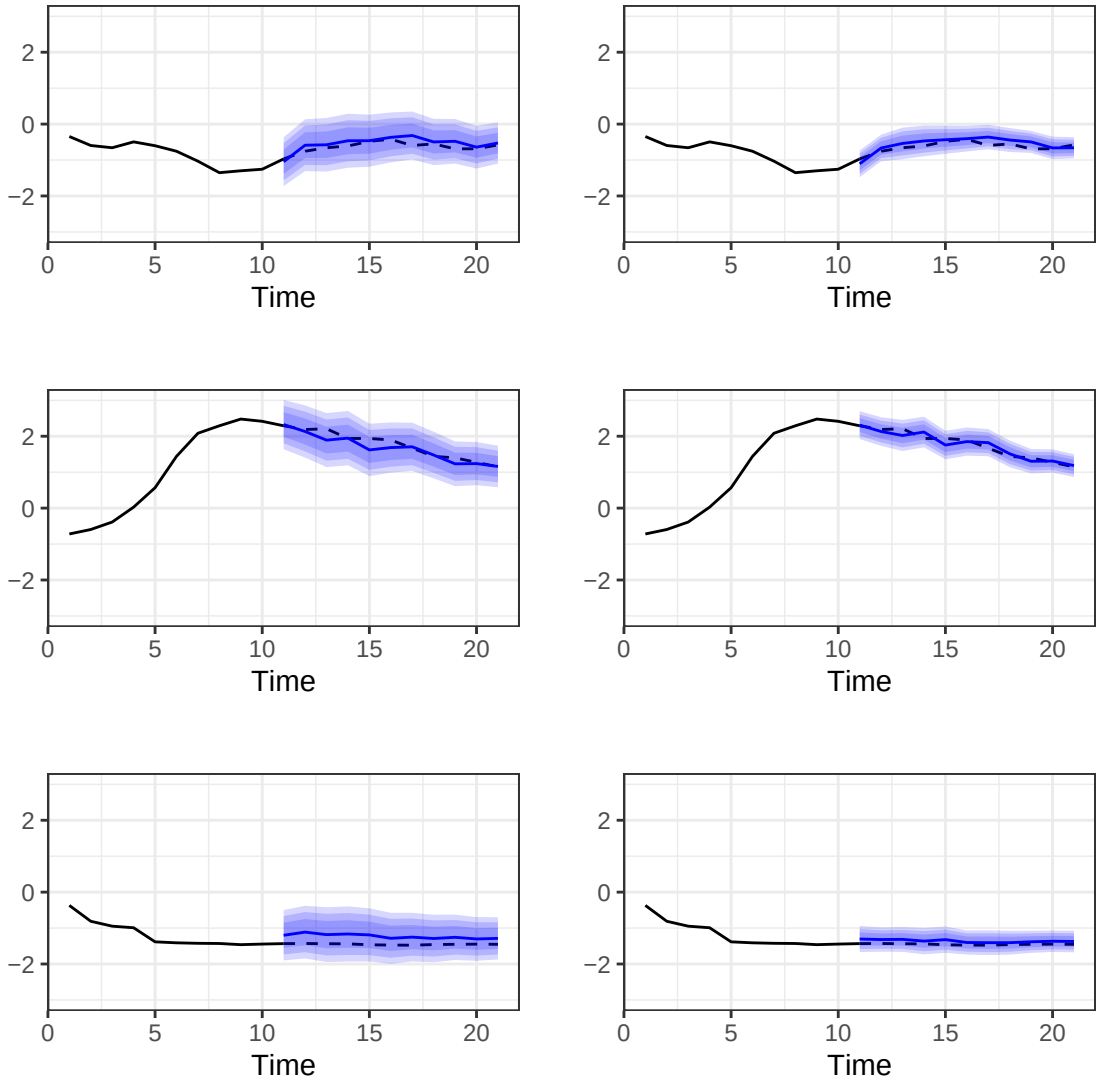


Figure 4.5: TVP-VAR model comprising US inflation rate, unemployment rate and federal funds rate with 8 lags. One-step-ahead forecasts with credible intervals (blue) estimated using the precision sampler strategy and the true time series (black). On the left column, the graphs show forecasts generated by a TVP-VAR model with no shrinkage ($\Omega_\beta = 1$), while on the right column forecasts are generated by a TVP-VAR model with severe shrinkage ($\Omega_\beta = 0.2$). The latter presents smaller credible intervals.

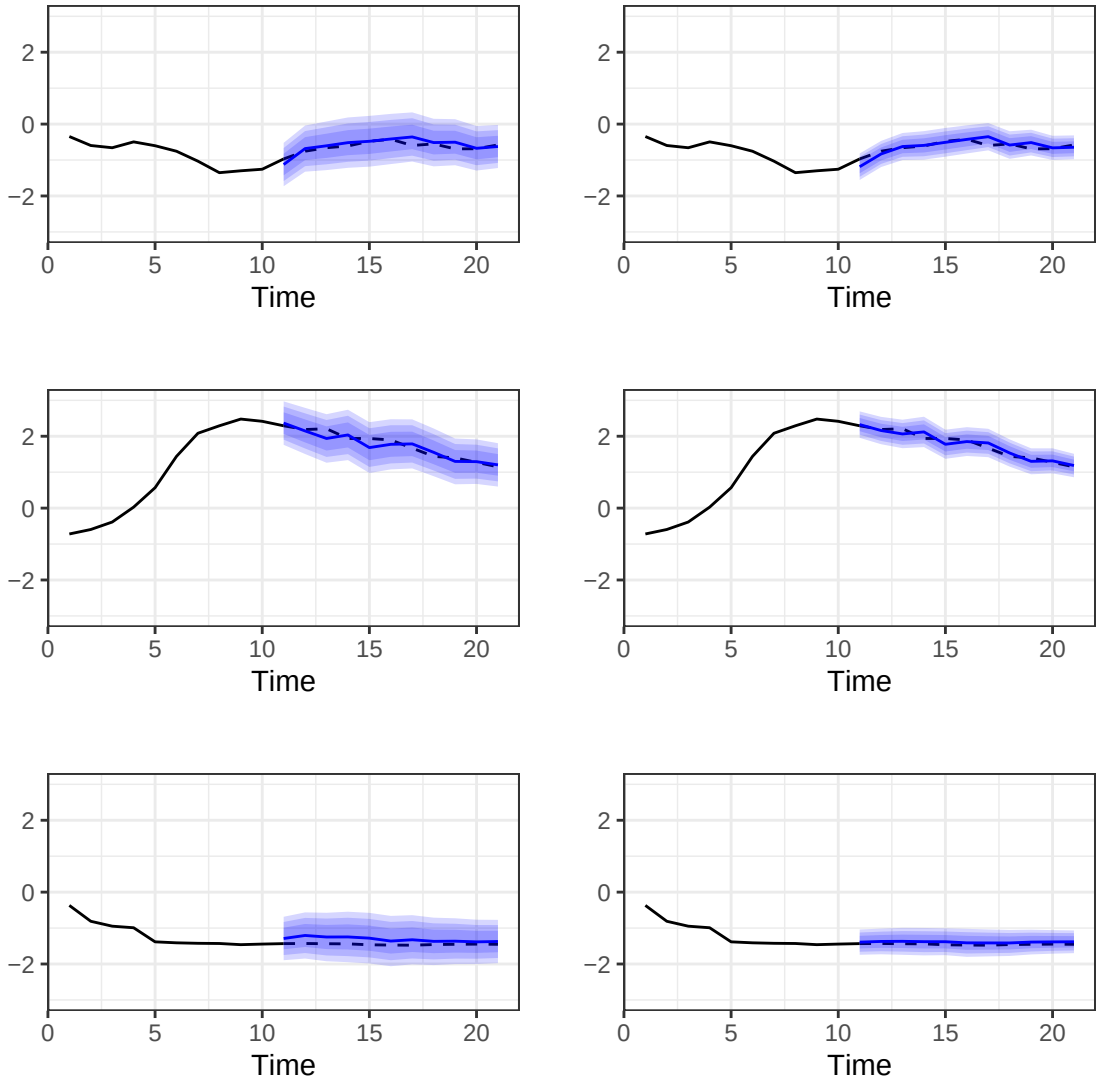


Figure 4.6: TVP-VAR model comprising US inflation rate, unemployment rate and federal funds rate with 8 lags. One-step-ahead forecasts with credible intervals (blue) estimated using the precision sampler strategy and the true time series (black). On the left column, the graphs show forecasts generated by a TVP-VAR model with no shrinkage ($\Omega_\beta = 1$), while on the right column forecasts are generated by a TVP-VAR model with severe shrinkage ($\Omega_\beta = 0.2$). The latter presents smaller credible intervals.

4.3 Discussion

With time-varying parameters, the risk of overfitting in VAR models is round the corner. This prevents TVP-VAR model from being used in high dimensions, limiting interesting modelling opportunities for applied research. With the example described in the previous section, we show the potentiality of Dynamic Spike-and-Slab process priors in mitigating this problem. Our aim is to provide an approach to estimate large TVP-VAR and Structural TVP-VAR that can be simultaneously fast and precise. The combination of the precision sampler with the Dynamic SSVS proved to be a valid strategy to achieve both goals. Thanks to dynamic shrinkage priors it is possible to obtain a simpler model and achieve important benefits in terms of interpretability. For what concern forecasting, we shown that by removing noisy variables it is still possible to preserve accuracy while reducing uncertainty. In addition, here we decided to focus on forecasting as a way to validate model performances, we do not exclude to expand this approach on causality and impulse responses assessment features in further researches. Nevertheless, the output of the `DSSTVPVAR_SV` function in our `dynamicshrink` package can be easily converted and used to compute impulse response functions using the existing `bvarsv` package. Another interesting fact we notice through the previous analysis, which is true in principle, is that $\boldsymbol{Y}_t$ is less correlated with $\boldsymbol{Y}_{t-h}$ when h increases. In order to emphasize this phenomenon, dynamic shrinkage priors that induce a greater penalization for h large could be developed. The work of Legramanti, Durante, and Dunson (2020) presents a proposal in this direction for static problems. It would interesting to translate it into a dynamic environment.

Chapter 5

My dynamicshrink R-package

The absence in literature of a unified `R`-package for dynamic variable selection constitutes an important limit for empirical research. With this thesis, we fill this gap by making available a set of `R` functions that implement the estimation strategies described in the previous chapters. Therefore, here we briefly introduce our `dynamicshrink` package. This sort of experimental package allows firstly to replicate the results obtained in previous sections and, eventually, to perform further analysis on high dimensional time series. The adjective "experimental" used here refers to the fact that, even though the great efforts to make these functions high performing, we acknowledge that there is room for improvements in terms of computational time. The code, which is entirely written in `R`, recalls for some specific tasks high-performing functions provided by other popular packages. Reference `R` packages are: `dlm`, `stochvol` and `matlib`, `MASS`, `Rcpp` and the `tidyverse`. Further developments in `C` will eventually follow. Below we present the list of the most useful functions complemented by a brief description of them. In addition, a little helper is provided in Appendix. Further details on these functions (and on the other functions) can be found at the personal Github page of the author, along with the rest of the code.

`DSSBSTS_SV`	Dynamic Spike-and-Slab Bayesian Structural Time Series with log-normal AR(1) stochastic volatility
`DSSBSTS_DF`	Dynamic Spike-and-Slab Bayesian Structural Time Series with discount factor model for time-varying variances
`MCMC.out`	Compute meaningful statistics from the output of `DSSBSTS_SV` and `DSSBSTS_DF`
`DEMVS`	Dynamic Expectation-Maximization Variable Selection with discount factor model for time-varying variances
`DEMVS_PS`	Dynamic Expectation-Maximization Variable Selection with log-normal AR(1) stochastic volatility estimated via Particle Smoothing (PS)
`DEMVS.Quarterly`	
`DEMVS.Monthly`	Dynamic Expectation-Maximization Variable Selection for Bayesian Structural Time Series with log-normal AR(1) stochastic volatility estimated via Particle Smoothing (PF). The functions are respectively built for quarterly and monthly data.
`DSSTVPVAR_SV`	Dynamic Stochastic Search Variable Selection for Time-Varying Parameter Vector Autoregressive models with Stochastic Volatility.
`MCMC.TVPVAR.out`	Compute meaningful statistics from the output of `DSSTVPVAR_SV`

`sparse.data.sim`	Generates dynamic sparse time series data.
`varplot`	Generates the plot of the volatility process
`Indicatorplot`	Generates the plot of the indicator variables
`Coef.compare.plot`	Generates the plot the regression coefficients. It can be used for comparisons
`SeasTrendSlope.plot`	Generates three plots concerning the time series seasonality, trend and trend's slope.
`SeasTrendReg.plot`	Generates three plots concerning the time series regression component, seasonal component and trend component estimated using Dynamic SSVS.
`SeasTrendReg.plot.1`	Generates three plots concerning the time series regression component, seasonal component and trend component estimated using Dynamic EMVS.
`Active.coef.plot`	Generates a plot indicating how many coefficients result active at each period in time.
`Forecastplot`	Compares forecasts with the actual series. Developed for BSTS models.
`Forecastplot.TVP.VAR`	Compares forecasts with the actual series. Developed for TVP-VAR models.
`DSS_PRECISIONSAMPLER`	Dynamic Stochastic Search Variable Selection for Time-Varying Parameter Vector Autoregressive models with Stochastic Volatility. The FFBS step is replaced by a precision sampler.
`DSSTVP_PRECISIONSAMPLER`	Dynamic Stochastic Search Variable Selection for Time-Varying Parameter Vector Autoregressive models with Stochastic Volatility. The FFBS step is replaced by a precision sampler.

Here we discuss briefly how the package works presenting the code used for the simulations study on quasi-synthetic data. Let's thus simulate a TVP regression model with 20 predictors of which only the first four are truly relevant using the function `sparse.data.sim`. We label data generated through this function as `synthetic.y` and then we add to then Airpassengers data in lags and rescaled by ten. We rescaled real data to adapt them to the scale of simulated data, note that results would not be affected if data are not rescaled. The true model's parameter are labeled `true_b` and `true_ind`.

```
sim = sparse.data.sim(TIME=144,P=4,FP=16,
                      phi0=0,phi1=0.98,lambda1=0.1,v=0,seed=100)
synthetic.y = sim$y
X = sim$X
real.y = log(as.numeric(AirPassengers))
y = real.y*10+synthetic.y
true_b = sim$true_b
true_ind = sim$true_ind
```

The estimation of the models parameters can be carried out using one of the following functions: `DSSBSTS_SV`, `DSSBSTS_DF` or `DEMVS.Monthly` whose general functionality is introduced in the tables above. The positive note here is that, even though they implements different strategies their usage is very similar, therefore we focus on the first function mentioned before to provide an overall guidance. The arguments of the function are self-explanatory. The model's hyperparameters are `phi1`, `phi0`, `lambda1`, `lambda0` and `OMEGA`, the starting values of the indicators, `gamma0`, and the volatility, `v0`, and the structural components, `S` and `U`. Moreover, `sig2u`, `sig2d`, `sig2tau` are the variances of, respectively, the trend, the slope and the seasonal factor. Note that the value of `S` is equal to the number of seasonal factors to be estimated i.e. $S-1$ where S is equal for example to 12 in monthly time series. On the other hand, `U` may be equal to 2 if our base model is a linear growth model or equal to 1 if it is a local level model. Alternatively to `U`=1, one can add a constant to the matrix of predictors and set `ACTIVATE=TRUE` which means that the indicator of the

first predictor will be always set equal to one. This latter option is useful also when we do not want to shrinkage an important predictor. In the latter case that predictors has to stay in the first column of `X`. If present, `new.X` contains future values of the predictors, $\boldsymbol{x}_{t+1}$, which are used for one-step-ahead forecasting. Finally `start_params` and `start_latent` are two objects defined in the `stochvol` package and they refer to the starting parameters and starting latent values of a stochastic volatility model.

```
set.seed(1)
ssvs.bsts = DSSBSTS_SV(y=y,X=X,N=1000,S=11,U=2,
                       phi0=0,phi1=0.98,gamma0=0.5,v0=0.25,
                       sig2u=0.1,sig2d=0.1,sig2tau=0.1,
                       lambda0=0.01,lambda1=0.1,OMEGA=0.2,
                       start_params,start_latent,new.X,activate=F)
```

This function returns a huge list of objects of which the most important are the samples generate by the Markov Chain. These samples are indeed stored in arrays whose interpretation is not immediate, therefore we need to convert them into some meaningful statistics. This is done by the function `MCMC.out` which returns the sample means and the sample variances. Here we just plug the name of the function used for model estimation and the number of samples to burn-in. Note that this function is already endowed in many other functions such as plot functions. Moreover, the functions carrying out the precision sampler do not need this passage, but they directly return the mean and the variance of the MCMC estimates.

```
mcmc = MCMC.out(fun=ssvs.bsts,burn=200)
```

Once the model parameters are estimated, there are several useful tool for visualization. For example, `Coef.compare.plot` provides the plot of the estimated regression coefficients and, if `trueb` is not missing, it compares them to their true values or, if `trueb` is equal to coefficients estimated with other model's specifications, it compares the two. `column`

indicates the column of interest of the regression matrix and if for example `column=1` the plot of $\beta_{1:T,1}$ will be generated. Furthermore, when we use the Dynamic EMVS instead of the Dynamic SSVS, then `EMVS` should be explicated. If we want dates on the x-axis then `time` must be equal to a date vector of the same length of `y`, otherwise if `time` is missing the x-axis will be a vector of natural numbers. Figure 3.2 is generated from the following command for $i = 1, ..., 6$.

```
Coef.compare.plot(trueb=true_b,column=i,fun=ssvs.bsts,burn=200,time,EMVS)
```

Similarly, `Indicatorplot`, `varplot` provides plots of the MCMC approximation of the expected values of the indicators and the variance. `Active.coef.plot` instead shows how many variables are relevant at any moment in time. If structural time series components are estimated, they can be represented by the two following function which plots respectively the trend, the seasonality and the regression component. The following generates Figure 3.4

```
SeasTrendReg.plot(ssvs.bsts,S=11,U=2,burn=200)
```

Forecasts one-step-ahead, two-step-ahead and so on are visualized via the function `Forecastplot`, for univariate time series the latter provides a comparison between realizations and point forecasts along with their credible intervals. Below we present an example. The time series `y` is plotted from `from` to `to`, whereas the forecasts are plotted from `start_fc` to `to`. This allows to show the path of the dependent variable before forecasting starts.

```
Forecastplot(y,from,to,start_fc,fc.f,fc.Q,ub,lb,timeframe)
```

In addition, the package provides all the metrics relevant for comparisons such as SSE, Hamming Distance, MASE, MAFE, RMSE, LPDS, FP, FN etc. The case in which the time series does not show evident time series structural component can be estimated by the same functions with missing `S` and `U` or, according to need, by `DSS_PRECISIONSAMPLER`, `DEMVS`

and DEMVS_PS. In general, as already mentioned, the usage is similar among these functions. For the multivariate case, instead, another set of functions is envisaged. The relevant functions here are: DSSTVPVAR_SV which estimates the Time-Varying Parameter Structural VAR model of Primiceri (2005) whit shrinkage, DSSTVP_PRECISIONSAMPLER_VAR and DSSTVP_PRECISIONSAMPLER_SVAR that estimates respectively the Time-Varying Parameter VAR and Structural VAR models with the precision sampler. As we can see from the code below, the overall structure of the command is similar to the one mentioned before. X is the regression matrix with the variables of the VAR model. If constant=TRUE then a constant will be added to the model. Now we have two hyperparameters OMEGA which are respectively Ω_α and Ω_β. In this experimental version of the package the function are built in order to provide just one-step-ahead forecasts, thus step_ahead=1, however we do not exclude further developments to extend the number of steps ahead. Finally, together with the starting value of the variances, v0, the starting value of the covariances, cov0, should be set. We decided to keep it simple and let all the covariances to have the same starting values.

```
DSSTVPVAR_SV(X,constant,N,lags,
             step_ahead=1,gamma0,phi0,v0,cov0,phi1,
             OMEGA.A,OMEGA.B,lambda0,lambda1,
             start_params,start_latent)
```

Ad-hoc functions are created to deal with the output of DSSTVPVAR_SV, these are very similar to the ones mentioned above and they are MCMC.TVP.out for extracting the meaningful statistics, Forecastplot.TVP.VAR to plot forecasts and lpds_multi which is the multivariate version of the lpds command for Log-Predicitive Density Sum comparison. Again DSSTVP_PRECISIONSAMPLER_VAR and DSSTVP_PRECISIONSAMPLER_SVAR return directly the mean and the variances of the estimates.

Bibliography

Andrews, D. F., and C. L. Mallows. 1974. "Scale Mixtures of Normal Distributions." *Journal of the Royal Statistical Society. Series B (Methodological)* 36 (1): 99–102.

Belmonte, M. A. G., G. Koop, and D. Korobilis. 2014. "Hierarchical Shrinkage in Time-Varying Parameter Models." *Journal of Forecasting* 33 (1): 80–94.

Bitto, A., and S. Fruhwirth-Schnatter. 2019. "Achieving Shrinkage in a Time-Varying Parameter Model Framework." *Journal of Econometrics* 210: 75–97.

Bitto, A., and S. Frühwirth-Schnatter. 2018. "Achieving Shrinkage in a Time-Varying Parameter Model Framework."

Canova, Fabio, and Harris Dellas. 1993. "Trade Interdependence and the International Business Cycle." *Journal of International Economics* 34 (1-2): 23–47.

Carter, C. K., and R. Kohn. 1994. "On Gibbs Sampling for State Space Models." *Biometrika* 81 (3): 541–53.

Chan, J., and E. Eisenstat. 2018. "Bayesian Model Comparison for Time-Varying Parameter VARs with Stochastic Volatility." *Journal of Applied Econometrics* 33 (4): 509–32.

Chan, J., and I. Jeliazkov. 2009. "Efficient Simulation and Integrated Likelihood Estimation in State Space Models." *International Journal of Mathematical Modelling and Numerical Optimisation* 1 (1/2): 101.

Chopin, N., and O. Papaspiliopoulos. 2020. "An Introduction to Sequential Monte Carlo," January.

Clark, T. E., and F. Ravazzolo. 2015. "Macroeconomic Forecasting Performance Under Alternative Specifications of Time-Varying Volatility." *Journal of Applied Econometrics*

30 (4): 551–75.

Cogley, T., and T. J. Sargent. 2005. "Drifts and Volatilities: Monetary Policies and Outcomes in the Post WWII US." *Review of Economic Dynamics* 8 (2): 262–302.

Durbin, J., and S. J. Koopman. 2002. "A Simple and Efficient Simulation Smoother for State Space Time Series Analysis." *Biometrika* 89 (3): 603–15.

Eisenstat, E., J. Chan, and R. Strachan. 2014. "Stochastic Model Specification Search for Time-Varying Parameter VARs." *SSRN Electronic Journal*, January. https://doi.org/10.2139/ssrn.2403560.

Fan, Y., and S. A. Sisson. 2010. "Reversible Jump Markov Chain Monte Carlo."

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

George, E. I. 1986a. "Combining Minimax Shrinkage Estimators." *Journal of the American Statistical Association* 81 (394).

———. 1986b. "Minimax Multiple Shrinkage Estimation." *The Annals of Statistics* 14 (1).

George, E. I., and R. E. McCulloch. 1993. "Variable Selection via Gibbs Sampling." *Journal of the American Statistical Association* 88 (423): 881–89.

———. 1997. "Approaches for Bayesian Variable Selection." *Statistica Sinica* 7 (2): 339–73.

Godsill, S. J., A. Doucet, and M. West. 2004. "Monte Carlo Smoothing for Nonlinear Time Series." *Journal of the American Statistical Association* 99: 156–68.

Green, P. J. 1995. "Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination." *Biometrika* 82 (4): 711–32.

Kalli, M., and J. E. Griffin. 2014. "Time-Varying Sparsity in Dynamic Regression Models." *Journal of Econometrics* 178 (2): 779–93.

Kastner, G. 2016. "Dealing with Stochastic Volatility in Time Series Using the r Package Stochvol." *Journal of Statistical Software* 69 (5): 1–30.

Kastner, G., and S. Frühwirth-Schnatter. 2014. "Ancillarity-Sufficiency Interweaving Strategy (ASIS) for Boosting MCMC Estimation of Stochastic Volatility Models." *Computational Statistics and Data Analysis* 76: 408–23.

Kim, S., N. Shephard, and S. Chib. 1998. "Stochastic Volatility: Likelihood Inference and

Comparison with ARCH Models." *The Review of Economic Studies* 65 (3): 361–93.

Koop, G., and L. Onorante. 2019. "Macroeconomic Nowcasting Using Google Probabilities." In *Topics in Identification, Limited Dependent Variables, Partial Observability, Experimentation, and Flexible Modeling: Part a*, 40A:17–40. Emerald Publishing Ltd.

Korobilis, D. 2013. "VAR Forecasting Using Bayesian Variable Selection." *Journal of Applied Econometrics* 28 (2): 204–30.

Kroese, D. P., and J. C. Chan. 2014. *Statistical Modeling and Computation*. Springer, New York, 2014.

Legramanti, S., D. Durante, and D. B Dunson. 2020. "Bayesian Cumulative Shrinkage for Infinite Factorizations."

Liu, J., and M. West. 2001. "Combined Parameter and State Estimation in Simulation-Based Filtering." In *Sequential Monte Carlo Methods in Practice*.

McCausland, W. J., S. Miller, and D. Pelletier. 2011. "Simulation Smoothing for State–Space Models: A Computational Efficiency Analysis." *Computational Statistics and Data Analysis* 55 (1): 199–212.

McCracken, M. W., and S. Ng. 2016. "FRED-MD: A Monthly Database for Macroeconomic Research." *Journal of Business & Economic Statistics* 34 (4): 574–89.

McCracken, Michael W., and Serena Ng. 2021. "FRED-QD: A Quarterly Database for Macroeconomic Research." *Review* 103 (1): 1–44.

Mitchell, T. J., and J. J. Beauchamp. 1988. "Bayesian Variable Selection in Linear Regression." *Journal of the American Statistical Association* 83 (404): 1023–32.

Nakajima, J., and M. West. 2013. "Bayesian Analysis of Latent Threshold Dynamic Models." *Journal of Business & Economic Statistics* 31 (2): 151–64.

Ning, N., and J. Qiu. 2021. "The Mbsts Package: Multivariate Bayesian Structural Time Series Models in r."

O'Hara, R. B., and M. J. Sillanpää. 2009. "A review of Bayesian variable selection methods: what, how and which." *Bayesian Analysis* 4 (1): 85–117.

Park, T., and G. Casella. 2008. "The Bayesian Lasso." *Journal of the American Statistical Association* 103 (482): 681–86.

Petris, G. 2010. "An R Package for Dynamic Linear Models." *Journal of Statistical Software* 36 (12): 1–16.

Petris, G., S. Petrone, and P. Campagnoli. 2009. "Dynamic Linear Models." *Dynamic Linear Models with R*, 31–84.

Primiceri, G. E. 2005. "Time Varying Structural Vector Autoregressions and Monetary Policy." *The Review of Economic Studies* 72 (3): 821–52.

Rockova, V. 2013. "Bayesian Variable Selection in High-dimensional Applications."

Rockova, V., and E. I. George. 2014. "EMVS: The EM Approach to Bayesian Variable Selection." *Journal of the American Statistical Association* 109 (506): 828–46.

Rockova, V., and K. McAlinn. 2021a. "Dynamic Variable Selection with Spike-and-Slab Process Priors." *Bayesian Analysis* 16 (1).

———. 2021b. "Dynamic Variable Selection with Spike-and-Slab Process Priors - Supplementary Material." *Bayesian Analysis*.

Scott, S. L., and H. R. Varian. 2013. "Predicting the Present with Bayesian Structural Time Series." *SSRN Electronic Journal*.

Shephard, N. 1994. "Partial Non-Gaussian State Space." *Biometrika* 81 (1): 115–31.

Sims, C. A. 1980. "Macroeconomics and Reality." *Econometrica* 48 (1): 1–48.

Stock, J. H. 2001. "Evolving Post-World War II u.s. Inflation Dynamics." *NBER Macroeconomics Annual* 16: 379–87.

Tibshirani, R. 1996. "Regression Shrinkage and Selection via the Lasso." *Journal of the Royal Statistical Society. Series B (Methodological)* 58 (1): 267–88.

Varian, H., and H. Choi. 2009a. "Predicting the Present with Google Trends." *SSRN Electronic Journal*.

———. 2009b. "Predicting the Present with Google Trends." *Economic Record* 88 (April).

Venables, W. N., and B. D. Ripley. 2002. *Modern Applied Statistics with s*. Fourth. New York: Springer.

West, M., and J. Harrison. 1997. *Bayesian Forecasting and Dynamic Models (2nd Ed.)*. Berlin, Heidelberg: Springer-Verlag.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino

McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686.

William, W. H. 1989. "Updating the Inverse of a Matrix." *SIAM Review* 31 (2): 221–39.

Wong, C. S., and W. K. Li. 2000. "On a Mixture Autoregressive Model." *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 62 (1): 95–115.

Yu, Y., and X. Meng. 2011. "To Center or Not to Center: That Is Not the Question—an Ancillarity–Sufficiency Interweaving Strategy (ASIS) for Boosting MCMC Efficiency." *Journal of Computational and Graphical Statistics* 20 (3): 531–70.

Zellner, A. 1986. "On Assessing Prior Distributions and Bayesian Regression Analysis with g-Prior Distributions." *Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti*.

Acknowledgement

A conclusione di questo elaborato, desidero ringraziare tutte le persone che mi hanno supportato nella stesura di questa tesi e, più in generale, nel mio percorso accademico.

In particolar modo, voglio ringraziare la Professoressa Sonia Petrone che ha saputo guidarmi, con suggerimenti pratici, nelle ricerche e nella stesura dell'elaborato. Il suo lavoro meticoloso di revisione di questa tesi ha contribuito immensamente a migliorarne l'esposizione.

Ringrazio inoltre il Professor Marco Bonetti che è stato mio mentore presso il BIDSA per avermi trasmesso la sua passione per la statistica e la ricerca.

Ringrazio infinitamente mia madre e mio padre, senza il loro supporto, questo lavoro di tesi non esisterebbe nemmeno.

Infine voglio ringraziare i miei amici storici e quelli conosciuti in Bocconi per tutti i momenti passati insieme. Un riconoscimento speciale va al mio amico e coinquilino Yuri, per essere riuscito a strapparmi una risata anche nei momenti più ardui vissuti durante la magistrale, ed alla mia ragazza Lilia, per essermi stata a fianco in questi mesi.

Appendix

.1 Dynamic Spike-and-Slab with Laplace priors

As already mentioned in the text, using a Laplace spike instead of a Gaussian one does not involves new implementation challenges. Moreover, under Laplace spike, the series $\{\beta_t\}$ can be shown to be stationary and identically and independently distributed. On the other hand, the conditional Laplace density does not imply marginal Laplace distribution. However, building an autoregressive path with Laplace marginals is still possible. For example, Rockova and McAlinn (2021a) define the so called Laplace autoregressive (LAR) process:

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

with $\{\psi_t\}_{t=1}^T$ being an exponential autoregressive process specified by $\psi_t|\kappa_{t-1} \sim \mathcal{G}(1 + \kappa_{t-1}, \lambda_1^2/[2(1-\rho)])$ and $\kappa_{t-1}|\psi_{t-1} \sim Poisson(\lambda_1^2\psi_{t-1}\rho/[2(1-\rho)])$ with marginal distribution $Exp(\lambda_1^2/2)$. The Laplace process implies Laplace marginals $\beta_t \sim \psi_1^{ST}(\beta_t|\lambda_1) \equiv Laplace(\lambda_1)$.

.2 More details about the dynamicshrink package

`DSSBSTS_SV` Dynamic Spike-and-Slab Bayesian Structural Time Series with log-normal AR(1) stochastic volatility

---

Usage

`DSSBSTS_SV(y,X,N,S,U,gamma0,phi0,v0,phi1,OMEGA,lambda0,lambda1,`
`sig2u=0.1,sig2d=0.1,sig2tau=0.1,start_params, start_latent,new.X)`

Arguments

`y`	the response variable. It can be a vector or a $T \times 1$ matrix object.
`X`	$T \times p$ matrix of explanatory variables.
`N`	number of MCMC iterations.
`S`	number of seasonal factors.
	Note if data is quarterly $S = 3$, whereas if data is annually $S = 11$.
	If S is missing then no seasonal component
	is added to the the regression ($S = 0$).
`U`	the trend component.
	If $U = 2$ then a linear growth model is implemented.
	If $U = 1$ then the δ_t component is missing.
	If U is missing, no trend is considered.
`gamma0`	initial matrix of indicators $\boldsymbol{\gamma}_{0:T}^{(0)}$. Recommended choice $\boldsymbol{\gamma}_{0:T} = 0.5$.
`phi0`	ϕ_0 in equation (3.1)
`phi1`	ϕ_1 in equation (3.1)
`v0`	initial vector of volatilities $v_{0:T}^{(0)}$.
	It is recommended to assign an initial small value for them.
`OMEGA`	Ω in equation (3.1)
`lambda0`	λ_0 in equation (3.1)
`lambda1`	λ_1 in equation (3.1)
`sig2u`	σ_μ^2 in equation (3.1)
`sig2d`	σ_δ^2 in equation (3.1)
`sig2tau`	σ_τ^2 in equation (3.1)
`start_params`	list of starting values for α_0, α_1, σ_ζ in equation (3.2) and h_0.
`start_latent`	vector of starting values for the latent log-Normal volatility process $\boldsymbol{h}_{0:T}^{(0)}$.
`new.X`	vector of explanatory variables at time $t+1$ used for Kalman forecasting.

Value

`theta`	MCMC draws from the posterior distribution of the latent process $\boldsymbol{\theta}_{0:T}$
`gamma`	MCMC draws from the posterior distribution of the auxiliary variable $\boldsymbol{\gamma}_{0:T}$
`v`	MCMC draws from the posterior distribution of the volatility process $\sigma^2_{\epsilon,0:T}$
`fc.f`	mean of the one-step-ahead posterior predictive distribution.
`fc.Q`	variance of the one-step-ahead posterior predictive distribution.
`...`	

`DSSBSTS_DF` Dynamic Spike-and-Slab Bayesian Structural Time Series with discount factor model for time-varying variances

---

Usage

```
DSSBSTS_DF(y,X,N,S,U,n0,d0,v0,gamma0,phi0,phi1,OMEGA,lambda0,lambda1,
delta,sig2u=0.1,sig2d=0.1,sig2tau=0.1,new.X)
```

Arguments

`y`	the response variable. It can be a vector or a $T \times 1$ matrix object.
`X`	$T \times p$ matrix of explanatory variables.
`N`	number of MCMC iterations.
`S`	number of seasonal factors.
	Note if data is quarterly $S = 3$, whereas if data is annually $S = 11$.
	If S is missing then no seasonal component
	is added to the the regression ($S = 0$).
`U`	the trend component.
	If $U = 2$ then a linear growth model is implemented.
	If $U = 1$ then the δ_t component is missing.
	If U is missing, no trend is considered.
`n0,d0`	hyperparameters of equation (2.13) at time $t = 0$.
`v0`	initial vector of volatilities $v^{(0)}_{0:T}$.
	It is recommended to assign an initial small value for them.
`gamma0`	initial matrix of indicators $\boldsymbol{\gamma}^{(0)}_{0:T}$. Recommended choice $\boldsymbol{\gamma}_{0:T} = 0.5$.

`phi0`	ϕ_0 in equation (3.1)
`phi1`	ϕ_1 in equation (3.1)
`OMEGA`	Ω in equation (3.1)
`lambda0`	λ_0 in equation (3.1)
`lambda1`	λ_1 in equation (3.1)
`delta`	the discount factor of the variance.
`sig2u`	σ_μ^2 in equation (3.1)
`sig2d`	σ_δ^2 in equation (3.1)
`sig2tau`	σ_τ^2 in equation (3.1)
`new.X`	vector of explanatory variables at time $t+1$ used for Kalman forecasting.

Value

`theta`	MCMC draws from the posterior distribution of the latent process $\boldsymbol{\theta}_{0:T}$
`gamma`	MCMC draws from the posterior distribution of the auxiliary variables $\boldsymbol{\gamma}_{0:T}$
`v`	MCMC draws from the posterior distribution of the volatility process $\sigma^2_{\epsilon,0:T}$
`fc.f`	mean of the one-step-ahead posterior predictive distribution.
`fc.Q`	variance of the one-step-ahead posterior predictive distribution.
`...`	

`MCMC.out` Compute meaningful statistics from the output of `DSSBSTS_SV` and `DSSBSTS_DF`

Usage

```
MCMC.out(fun,burn)
```

Arguments

`fun`	an objects of the type `DSSBSTS_SV` or `DSSBSTS_DF`.
`burn`	size of burnin sample.

Value

`mean`	mean of the posterior distribution of the latent process.
`var`	var of the posterior distribution of the latent process.
`ind`	mean of the posterior distribution of the auxiliary variables.
`v`	mean of the posterior distribution of the volatility process.
`var.v`	var of the posterior distribution of the volatility process.
`fit.reg`	mean of the time series regression component.
`var.fit.reg`	var of the time series regression component.

`DEMVS` Dynamic Expectation-Maximization Variable Selection with discount factor model for time-varying variances

---

Usage

```
DEMVS(y,X,N,n0,d0,phi0,phi1,OMEGA,lambda0,lambda1,delta,new.X)
```

Arguments

`y`	the response variable. It can be a vector or a $T \times 1$ matrix object.
`X`	$T \times p$ matrix of explanatory variables.
`N`	number of iterations of the EM algorithm.
`n0,d0`	hyperparameters of equation (2.13) at time $t = 0$.
`phi0`	ϕ_0 in equation (3.1)
`phi1`	ϕ_1 in equation (3.1)
`OMEGA`	Ω in equation (3.1)
`lambda0`	λ_0 in equation (3.1)
`lambda1`	λ_1 in equation (3.1)
`delta`	the discount factor of the variance.
`new.X`	vector of explanatory variables at time $t + 1$ used for forecasting.

Value

`beta`	Maximum a Posteriori estimates of the latent state process.
`gamma`	conditional inclusion probabilities.

`v`	residual variances.
`fc.f`	one-step-ahead forecast.

`DEMVS_PS` Dynamic Expectation-Maximization Variable Selection with log-normal AR(1) stochastic volatility estimated via Particle Smoothing (PS)

Usage

```
DEMVS_PS(y,X,N,phi0,phi1,OMEGA,lambda0,lambda1,new.X)
```

Arguments

`y`	the response variable. It can be a vector or a $T \times 1$ matrix object.
`X`	$T \times p$ matrix of explanatory variables.
`N`	number of iterations of the EM algorithm.
`n0,d0`	hyperparameters of equation (2.13) at time $t = 0$.
`phi0`	ϕ_0 in equation (3.1)
`phi1`	ϕ_1 in equation (3.1)
`OMEGA`	Ω in equation (3.1)
`lambda0`	λ_0 in equation (3.1)
`lambda1`	λ_1 in equation (3.1)
`new.X`	vector of explanatory variables at time $t + 1$ used for forecasting.

Value

`beta`	Maximum a Posteriori estimates of the latent state process.
`gamma`	conditional inclusion probabilities.
`v`	residual variances.
`fc.f`	one-step-ahead forecast.

`DEMVS.Quarterly` & `DEMVS.Monthly` Dynamic Expectation-Maximization Variable Selection for Bayesian Structural Time Series with log-normal AR(1) stochastic volatility estimated via Particle Smoothing (PF). The functions are respectively built for quarterly and monthly data.

Usage

```
DEMVS.Monthly(y,X,N,phi0,phi1,OMEGA,lambda0,lambda1,
sig2u=0.1,sig2d=0.1,sig2tau=0.1,new.X)
```

Arguments

`y`	the response variable. It can be a vector or a $T \times 1$ matrix object.
`X`	$T \times p$ matrix of explanatory variables.
`N`	number of iterations of the EM algorithm.
`phi0`	ϕ_0 in equation (3.1)
`phi1`	ϕ_1 in equation (3.1)
`OMEGA`	Ω in equation (3.1)
`lambda0`	λ_0 in equation (3.1)
`lambda1`	λ_1 in equation (3.1)
`sig2u`	σ_μ^2 in equation (3.1)
`sig2d`	σ_δ^2 in equation (3.1)
`sig2tau`	σ_τ^2 in equation (3.1)
`new.X`	vector of explanatory variables at time $t+1$ used for forecasting.

Value

`beta`	Maximum a Posteriori estimates of the regression coefficients.
`u`	Maximum a Posteriori estimates of the stochastic trend.
`d`	Maximum a Posteriori estimates of the stochastic trend's slope.
`tau`	Maximum a Posteriori estimates of the seasonality.
`gamma`	conditional inclusion probabilities.
`v`	residual variances.
`fc.f`	one-step-ahead forecast.
`...`	

`DSSTVPVAR_SV` Dynamic Stochastic Search Variable Selection for Time-Varying Parameter Vector Autoregressive models with Stochastic Volatility.

Usage

```
DSSTVPVAR_SV(X,constant,N,lags,step_ahead,gamma0,phi0,
v0,cov0,phi1,OMEGA.A,OMEGA.B,lambda0,lambda1,start_params,start_latent)
```

Arguments

`X`	$T \times n$ matrix of regressors.
`constant`	if TRUE then a constant is added to the model.
`N`	number of iterations of the MCMC.
`lags`	number of lags.
`step_ahead`	number of step ahead for forecasting.
`gamma0`	initial matrix of indicators $\boldsymbol{\gamma}_{0:T}^{(0)}$. Recommended choice $\boldsymbol{\gamma}_{0:T} = 0.5$
`phi0`	ϕ_0 in equation (3.1)
`phi1`	ϕ_1 in equation (3.1)
`v0`	diagonal elements of the starting covariance matrix.
`cov0`	non-diagonal elements of the starting covariance matrix.
`OMEGA`	Ω in equation (3.1)
`lambda0`	λ_0 in equation (3.1)
`lambda1`	λ_1 in equation (3.1)
`start_params`	list of starting values for α_0, α_1, σ_ζ in equation (3.2) and h_0.
`start_latent`	vector of starting values for the latent log-Normal volatility process $\boldsymbol{h}_{0:T}^{(0)}$.

Value

`beta`	MCMC draws from the posterior distribution of $\boldsymbol{\beta}_{0:T}$.
`alpha`	MCMC draws from the posterior distribution of $\boldsymbol{\alpha}_{0:T}$.
`gamma_beta`	MCMC draws from the posterior distribution of $\boldsymbol{\gamma}_{0:T}$ related to $\boldsymbol{\beta}_{0:T}$.
`gamma_alpha`	MCMC draws from the posterior distribution of $\boldsymbol{\gamma}_{0:T}$ related to $\boldsymbol{\alpha}_{0:T}$.
`v`	MCMC draws from the posterior distribution of the volatility process.
`fc.m`	one-step-ahead forecast mean generated at each MCMC iteration.
`fc.v`	one-step-ahead forecast variance generated at each MCMC iteration.

`fc.y`	MCMC draws from the one-step-ahead forecast density.
...	

`DSS_PRECISIONSAMPLER` Dynamic SSVS for Time-Varying Parameter regression models with stochastic volatility. Posterior sampling is performed using a precision sampler

---

Usage

```
DSSBSTS_DF(y,X,N,burn,OMEGA,lambda1,lambda0,
phi1,phi0,gamma0,v0,start_params,start_latent,new.X,activate=F)
```

Arguments

`y`	the response variable. It can be a vector or a $T \times 1$ matrix object.
`X`	$T \times p$ matrix of explanatory variables.
`N`	number of MCMC iterations.
`v0`	initial vector of volatilities $v_{0:T}^{(0)}$. It is recommended to assign an initial small value for them.
`gamma0`	initial matrix of indicators $\boldsymbol{\gamma}_{0:T}^{(0)}$. Recommended choice $\boldsymbol{\gamma}_{0:T} = 0.5$.
`phi0`	ϕ_0 in equation (3.1)
`phi1`	ϕ_1 in equation (3.1)
`OMEGA`	Ω in equation (3.1)
`lambda0`	λ_0 in equation (3.1)
`lambda1`	λ_1 in equation (3.1)
`start_params`	list of starting values for α_0, α_1, σ_ζ in equation (3.2) and h_0.
`start_latent`	vector of starting values for the latent log-Normal volatility process $\boldsymbol{h}_{0:T}^{(0)}$.
`new.X`	vector of explanatory variables at time $t+1$ used for Kalman forecasting.
`activate`	If `TRUE` no shrinkage applies to the first variable of the regression matrix. Useful for example if

Value

`beta`	Expected value of the posterior distribution of the latent process $\boldsymbol{\beta}_{1:T}$

`beta0`	Expected value of the posterior distribution of the latent process $\boldsymbol{\beta}_0$
`ind`	Expected value of the posterior distribution of the auxiliary variables $\boldsymbol{\gamma}_{1:T}$
`ind0`	Expected value of the posterior distribution of the auxiliary variables $\boldsymbol{\gamma}_0$
`h`	Median of the posterior distribution of the volatility process $h_{0:T}$
`beta.sample`	MCMC draws from the posterior distribution of the latent process $\boldsymbol{\beta}_{1:T}$
`gamma.sample`	MCMC draws from the posterior distribution of the auxiliary variables $\boldsymbol{\gamma}_{1:T}$
`beta0.sample`	MCMC draws from the posterior distribution of the latent process $\boldsymbol{\beta}_0$
`gamma0.sample`	MCMC draws from the posterior distribution of the auxiliary variables $\boldsymbol{\gamma}_0$
`h.sample`	MCMC draws from the posterior distribution of the volatility process $h_{0:T}$
`fc.f`	mean of the one-step-ahead posterior predictive distribution.
`fc.Q`	variance of the one-step-ahead posterior predictive distribution.
`...`	

.3 Inflation forecasting: list of predictors

Table 17: List of Predictors

GDPC1	INDPRO	DPIC96	DIFSRG3Q086SBEA	TB6MS
DGOERG3Q086SBEA	PCECC96	AHETPIx	PCECTPI	GS1
SP500	GPDIC1	DHUTRG3Q086SBEA	CPIAUCSL	GS5
M2REAL	GCEC1	PAYEMS	GPDICTPI	GS10
M1REAL	EXPGSC1	DHLCRG3Q086SBEA	FEDFUNDS	UNRATE
INVEST	IMPGSC1	DTRSRG3Q086SBEA	HOUST	
PCDGx	DFXARG3Q086SBEA	WPU0561	CONSUMERx	
PCESVx	DREQRG3Q086SBEA	WPSFD49207	UMCSENTx	
PCNDx	PPIDC	DFSARG3Q086SBEA	TB3MS	

For details on the labels please see Michael W. McCracken and Ng (2021)

.4 Inflation forecasting: plots of some regression coefficients

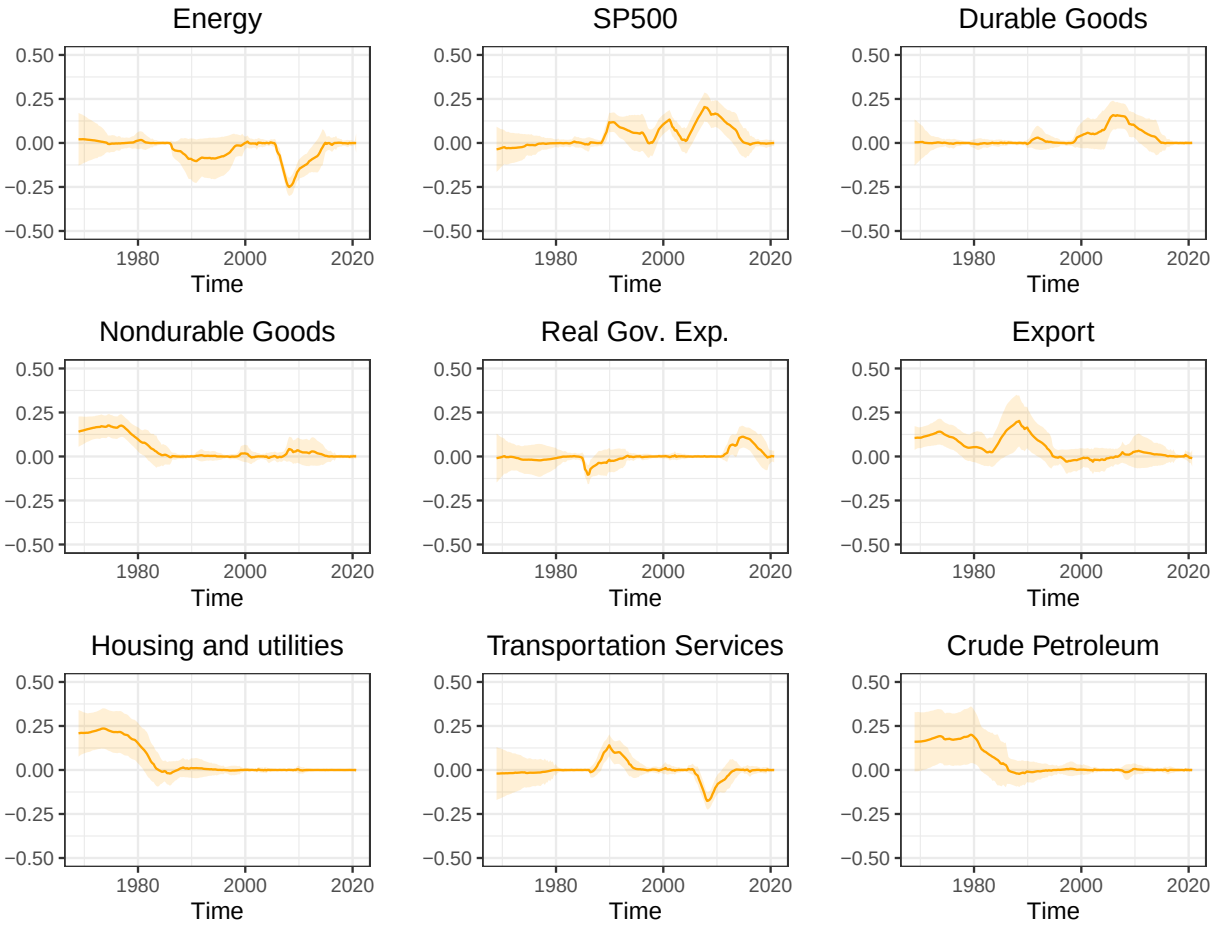


Figure 1: Dynamic SSVS for BSTS model; smoothed estimates (yellow) with 95 percent credible intervals of some interesting regression coefficients.

.5 Unemployment rate nowcasting: list of predictors

Table 18: List of Predictors

Source	Type	Label
Yahoo	Contemporary	Dow Jones
FRED	Contemporary	NIKKEI225
FRED	Contemporary	NASDAQ100
FRED	Contemporary	SP500
FRED	One period lag	OILPRICEx
FRED	One period lag	CPIAUSL
FRED	One period lag	INDPRO
FRED	One period lag	TWEXAFEGSMTHx
FRED	One period lag	RETAILx
FRED	One period lag	DPCERA3M086SBEA
FRED	One period lag	CMRMTSPLx
FRED	One period lag	HOUST
FRED	One period lag	PAYEMS
FRED	One period lag	RPI
Google Trends	Contemporary	unemployment
Google Trends	Contemporary	federal unemployment
Google Trends	Contemporary	compensation package
Google Trends	Contemporary	Unemployment compensation
Google Trends	Contemporary	Unemployment agency
Google Trends	Contemporary	employee benefits
Google Trends	Contemporary	unemployment check
Google Trends	Contemporary	unemployment statistics
Google Trends	Contemporary	unemployment pa
Google Trends	Contemporary	unemployment office
Google Trends	Contemporary	unemployment insurance
Google Trends	Contemporary	unemployment depression
Google Trends	Contemporary	unemployment benefits
Google Trends	Contemporary	subsidies

.6 BSTS model for Airpassengers data

Here, a BSTS model with DSS priors is used to fit AirPassengers data. The coefficients are correctly shrunk to zero and the model succeed in capturing trend and seasonality.

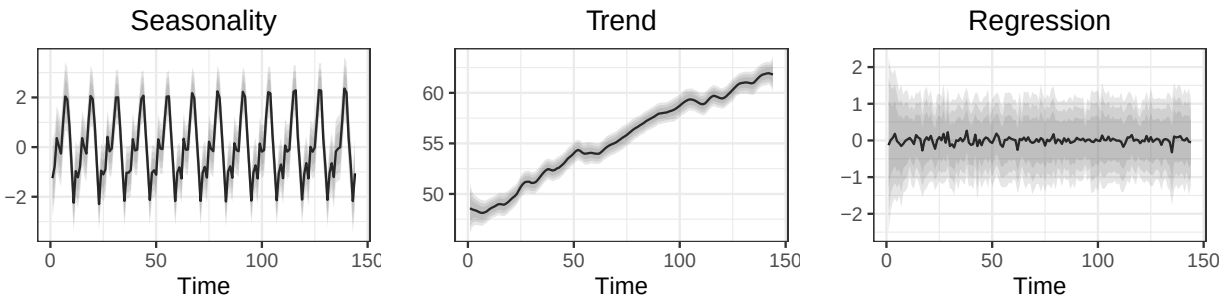


Figure 2: Structural time series components. Point estimates (black) with credible intervals (gray).

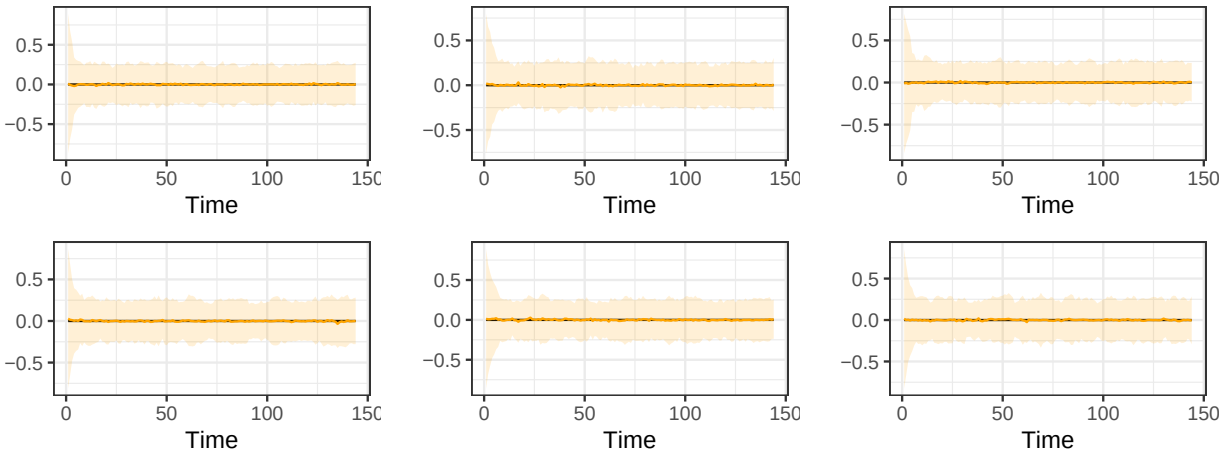


Figure 3: Dynamic SSVS for BSTS model; true values of $\beta_{1:T,j}$, $j = 1, \ldots, 6$, (black) and smoothed estimates with 95 percent credible intervals (yellow) of the first six regression coefficients. The coefficients are correctly shrunk to zero at every time.

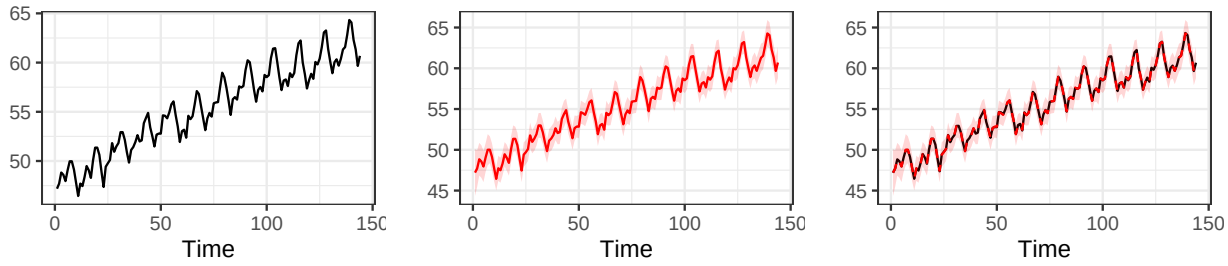


Figure 4: Left panel: Airpassengers time series. Middle: fitted values with 95 credible intervals. Right panel: Airpassengers time series (black) and fitted values (red) with 95 credible intervals.