BSTS Part 3 Script

Slide 1

Welcome back! We hope your journey so far has been rewarding. As we noted in the previous part, this stage of your journey is the most challenging. But worry not. For once you understand how the Bayesian philosophy can be applied to structural time series, you will be ready to apply it to data. (-> Slide 2)

Slide 2

In this part of you adventure, we will see how the Bayesian framework can be applied to structural time series. We first discuss Bayesian structural time series when we do not have a regression component. This includes:

(->) the prior distribution of our parameters (->) and how to obtain the posterior distribution.

We only briefly cover this, as our main focus is the case where we do have a regression component (->). Again, in the case where we add a regression component we will discuss

(->) the prior distribution (->) and obtaining the posterior distribution. (-> Slide 3)

Slide 3

Assuming we don't have a regression component in our model, for example, the local level model, then our parameters are the variances of the error terms.

(->) So, we would need to specify a distribution for say, σ^2_ε. To keep things simple, we won't discuss a particular prior here. Just remember the parameters in structural time series models are the error variances and we would need to specify a prior for each.

(->) The posterior distribution is obtained numerically.

(->) The Kalman filter and Kalman smoother are used,

(->) along with Markov Chain Monte Carlo sampling.

When we do have a regression component, we use what is called a *Spike and Slab Prior* (-> Slide 4).

Slide 4

We will consider our regression coefficients fixed through time because otherwise (->) they could be added as an additional state component.

(->) The *Spike* in Spike and Slab regression is a prior that specifies a positive probabilty for each regression coefficient being equal to zero.

(->) The *Slab* is the prior for each of the regression coefficients that are not zero, and also the prior for the variance of the error term.

This should become more clear as we discuss each of these in more detail. (-> Slide 5)

Slide 5

To understand the Spike component, we will need a bit of set up notation. Suppose we have a vector γ where each component γ_k is one if $\beta_k \neq 0$ and is otherwise zero. Here, β is the vector of regression coefficients.

(->) Further, we use β_γ to denote the subset of the vector β corresponding to the γ_ks that are equal to one. For example, if γ_1 and γ_2 are equal to one and all other components are zero, then β_γ will be the vector of β_1 and β_2.

The prior distribution we use for γ is an independent Bernoulli prior (->). This means that, for each component γ_k of γ, π_k is the probability that γ_k is equal to one and thus, β_k is included in the model. We will discuss how to choose the π_ks shortly.

Along with specifying our *spike* prior, we need to specify the *slab* prior. (-> Slide 6).

Slide 6

Again, we need a little notation to start with. For a symmetric matrix Ω^{-1}, (->) we let Ω_γ^{-1} denote the sub matrix whose rows and columns correspond to the indices k where $\gamma_k = 1$. This is the same idea as with β_γ.

The *slab* prior is shown here.

(->) It says that β_γ, given σ_ε^2 and γ follows a normal distribution with mean vector b_γ and covariance matrix $\sigma_\varepsilon^2 \Omega_\gamma$. As is often done in Bayesian statistics to ease calculations, we specify a distribution for $\frac{1}{\sigma_\varepsilon^2}$ instead of σ_ε^2. This distribution is Gamma with parameters $\frac{v}{2}$ and $\frac{ss}{2}$. The parameterization for the Gamma distribution here is the shape-rate parameterization.

For this prior, we need to specify $b, \sigma_\varepsilon^2, \Omega^{-1}, v$, and ss. But before we do, we want to note the full prior. (-> Slide 7)

Slide 7

The full prior is the product of the three individual priors we just discussed. We don't write it out in detail here because it might look a bit intimidating, but you are certainly welcome to do so yourself. (-> Slide 8)

Slide 8

Now that we know what the priors are, let's discuss how we choose specific priors. The hyperparameters for σ_ε^2 are v and ss. We can think of these as sort of like a prior sum of squares and a prior sample size. In the bsts package, this boils down to selecting an expected R^2 from the regression and an expected model size. The default values are $R^2 = 0.5$ and $v = 0.01$.

(->) For the π_ks, we specify the probability that β_k should be nonzero. For example, if we are very certain β_1 should be in the model, we could specify π_1 to be 0.95. If we don't have any previous knowledge as to which regression coefficients should be included, we could set π_k to 1/2 for every k, which is the default in the bsts package.

(->) Choosing the covariance matrix Ω^{-1} is based on the model matrix X.

(->) bsts sets Ω^{-1} to be $\frac{k}{n} X^T X$ for a specified value of k. This essentially reflects our prior knowledge on how close β is to the prior mean b. The default value in bsts is $k = 1$.

That takes care of the prior, so now we are ready to move on to the posterior. (-> Slide 9)

Slide 9

Again, we need a little notation to set things up. Let Z_t^* denote the observation matrix Z_t with $\beta^T x_t$ set to zero.

(->) Let y_t^* be the data y_t with ${Z_t^*}^T \alpha_t$ subtracted out.

(->) Finally, let y^* be the vector containing y_t^* for $t = 1$ to n.

(->) Then the posterior distributions of β_γ and $\frac{1}{\sigma_\varepsilon^2}$ are

β_γ given $\sigma\varepsilon$, γ, and y^*, follows a normal distribution with mean vector $\tilde{\beta}_\gamma$ and covariance matrix $\sigma_\varepsilon^2 V_\gamma$. Additionally, $\frac{1}{\sigma_\varepsilon^2}$ given γ and y^* folows a Gamma distribution with shape $\frac{N}{2}$ and rate $\frac{SS_\gamma}{2}$.

The parameters of each of these distributions are here.

(->) read this

(->) "

(->) "

(->) "

(-> Slide 10)

Slide 10

And the posterior of γ is here. (-> Slide 11)

Slide 11

To obtain a sample from the posterior distribution, we use a combination of

(->) a stochastic Kalman smoother

(->) and a Markov Chain Monte Carlo algorithm.

(->) The algorithm is called stochastic search variable selection,

(->) which is a Gibbs sampling algorithm.

Note that you don't need to worry about implementing this as the bsts package will do it for you.

Congratulations! The most strenuous part of your journey is over. Now that you understand the mysterious beast that is Bayesian structural time series, you are ready to complete your journey by learning how to implement what you have learned so far.