Developing a Bayesian method for locating breakpoints in time series data.

Kathryn Haglich [1], Jeffrey Liebner [1], Sarah Neitzel [2] and Amy Pitts [3]

[1] Department of Mathematics, Lafayette College, Easton, Pennsylvania, USA

[2] School of Biodiversity Conservation, Unity College, Unity, Maine, USA

[3] Department of Mathematics, Marist College, Poughkeepsie New York, USA

Address for correspondence: Jeffrey Liebner, Department of Mathematics, Lafayette College, Easton, Pennsylvania, USA.

E-mail: `liebnerj@lafayette.edu`.

Phone: (+420) 221 913 282.

Fax: (+420) 222 323 316.

Abstract: An abstract of up to 200 words should precede the text together with 5 or 6 keywords in alphabetical order to describe the content of the paper. Authors should take great care in preparing the abstract and not simply lift it from the main text. The abstract should describe the background and contribution of the manuscript and give a clear verbal description of the results and examples, and avoid citations whenever possible. Any acknowledgements will be printed at the end of the text.

Key words: Breakpoints; Time Series; Bayesian; AR; BARS

1 Introduction

This paper considers issues of finding number and location of breakpoints in time series data. Time series data, although a very broad category of data, has attracted ample interests in trying to describe overall trends. When modeling overarching trends it is helpful to identify significant places of change that the data may hold. This allows the data to be fit some items multiple time and combine those multiple models rather than just having one model. This way significant jump and changes data may hold is captured and accounted for. These places of significant changes serve as our breakpoints. Finding exactly where they are located and how many exist has been a sought-after goal.

In the last 15 or so years, many statisticians and others analyzing data have been creating techniques for addressing problems like our own. Techniques early on consisted on using expert opinion on locations of breakpoints. This is seen in Seidel and Lanzantes (2004) Ecology paper using expert opinion to break up global atmospheric temperature changes. This is also seen in Gamber, Liebner, and Smith's (2016) Inflation Persistence paper using expert opinion to place breakpoints in CPI time series data. Due to the data obtained from these papers, as well as other, contains compelling information, more formulaic methods using technology has been developed to address the issue of breakpoints. One of these methods is a frequentist approach developed by Bai and Perron (1998) (2003). The first paper written by Bai and Perron (1998) describes issues that structural changes in data pose for running regression.

The second paper Bai and Perron (2003) wrote discuss applications and describe a general algorithm for finding an optimal breakpoint set. From their research, an R package was developed named breakpoints located in the package strucchange (Zeileis et. al. 2007). A similar tequie to the Bai-Perron test is the Review Order Cusum (Pesaran and Timmermann, 2002). This approach flips time series and using historical data attempts to break data into groups. The diviate between groups being a breakpoint. Another method of significance was developed by Pesaran and colleges (2006) that identifies breakpoints in the United States Treasury bill rates. One application of breakpoint analysis is featured in Ruggieris paper about climatic records (2013).
In this paper, the Bayesian Adaptive Auto-Regression (BAAR) method is developed in order to create a Bayesian method to find the distribution of the number and location of breakpoints in time series data. Section two address how we approach the problem of locating number and location of breakpoints. It dives into our Metropolis-Hastings and Markov chain Monte Carlo algorithms to obtain these distributions. The technique described is inspired by Bayesian curve fitting with free knot splines that describe a method called Bayesian Adaptive Regression Splines (BARS) (DiMatteo et al., 2001) and a paper describing the implementation of BARS (Walstrom, Liebner, and Kass, 2008). In the third section, we take a look at applications and how our method finds significant breakpoints in data such as **whatever data we choose**. Finally, we will discuss significant and other applications.

2 Method

The foundation of this method is inspired on the BARS method which consists of the Metropolis Hastings algorithm containing a Markov Chain Monte Carlo (MCMC) (DiMatteo et al., 2001). The steps in the MCMC include the addition of a breakpoint, the subtraction of a breakpoint, and the moving of a breakpoint. A new breakpoint set is proposed at each step of the MCMC, and the Metropolis Hastings ratio determines the set's acceptance. From this, a distribution of possible breakpoint locations can be obtained.

2.1 Initial Breakpoint

This is going to change depending on next simulation! The BAAR function need to have an input starting breakpoint place(s). In the BARS paper we see that having a more intelligent start location, like with the logsplines starting condition, can significantly reduce burn it periods and run times (DiMatteo et al., 2001). This is opposed to starting with a single middle breakpoint or evenly placed breakpoints. With this knowledge we are taking the Bai-Perron method and breakpoint package to help obtain relatively good initial breakpoints to start out (Bai, Perron, 2003) (Zeileis et al 2007). The algorithm described by Bai and Perron is a frequentist approach that checks almost every single location for a breakpoint and returns the optimal set (Bai, Perron, 2003). The breakpoint package requires a user to specify an maximum number of breakpoint (Zeileis et al 2007). The larger the maximum number the longer the run time for the function. Because this is only our initial breakpoint set specifying a smaller maximum value closer to one will help get a intelligent starting place and increase overall speed of the breakpoint function. Having one to three intelligent

starting place, depending on the size of the dataset, helps reduce the burn in period and the overall run time to get a proper distribution.

2.2 Step Type

Changing from one state to another, the markov chain monte carlo has three different possible steps: birth, death, and move. The birth step randomly proposes a breakpoint at an available location. An available location is where a breakpoint could be placed given the following constraints. First, the location cannot have a breakpoint or an endpoint currently assigned to it. Second, for linear fits and AR(1) models, the location must be at least two data points away from the breakpoints closest to the particular location. For AR(p) models, the minimum distance away a location must be from its closest breakpoints is $2p$. If a location is in accordance with these constrains, then it is an available location.

The death step randomly chooses an existing breakpoint and proposes a set without that chosen breakpoint.

The general move step is a subtraction step followed immediately by an addition step and can be broken down to two specific types of move: jump and jiggle. Jump allows the movement of a breakpoint to any available location. Jiggle restricts the distance a breakpoint can move to a jiggle neighborhood, an interval surrounding the breakpoint's original location. To calculate the jiggle neighborhood, J_n,

$$J_n = (x_b - pn, x_b + pn)$$

where x_b is the original location of the chosen breakpoint, n is the size of the data set, and p is the user-inputed percent in decimal form. When a move step is chosen,

there is a ζ probability that a jiggle will be performed, which is determined by the user such that $0 < \zeta < 1$ and $\zeta \in \mathbb{Q}$. The probability of a jump occurring is $1 - \zeta$. Based off of data obtained by simulations on different probabilities the suggested probabilities are 75% jiggle and 25% jump. This combination increases overall speed and a combination of jiggle and jump more thoroughly explores the distribution.

2.3 Probabilities of the Steps

The combined probabilities of performing a birth step, b_p, and a death step, d_p, is equal to the user imputed value, c such that $c \in \mathbb{Q}$ and $0 < c < 1$. The ratio of birth steps to death steps is determined by c and the initial conditions of the starting number of breakpoints, K_{start}, and the starting number of available spaces, A_{start}. From this, the following equations can be derived for b_p and d_p:

$$b_p = \frac{A_{start}}{A_{start} + K_{start} + 1} \qquad d_p = \frac{K_{start} + 1}{A_{start} + K_{start} + 1}$$

Then, the probability of a specific birth step given A available locations, b, is the equation

$$b = c\ b_p \times \frac{1}{A}.$$

Thus, the probability of a specific death step given K breakpoints, d, is the equation

$$d = c\ d_p \times \frac{1}{K}.$$

The probability of a move step, m is represented by the equation $m = 1 - c$. The probability of jiggle, jg, and the probability of jump, ju, are calculated by the following equations:

$$ju_p = \zeta(1 - c(d_p + b_p)) \qquad jg_p = (1 - \zeta)(1 - c(d_p + b_p))$$

Then, the probability of a specific jump step is,

$$ju = ju_p \frac{1}{K_{old} A}.$$

For the jiggle step the probability of a specific step occurring is

$$jp = jg_p \frac{1}{K_{old} j_{free}}.$$

2.3.1 Metropolis Hastings Ratio and BIC Approximation

After a specific step is selected, the Metropolis Hastings ratio, as derived below, is used to determine the acceptance of the proposed breakpoint set. To determine the thresh hold of acceptance, r_{unif} is generated from a uniform distribution from a sample space on the interval (0,1). If the ratio is greater than r_{unif}, then the proposed breakpoint set is accepted and kept. Otherwise, the old breakpoint set is retained.

The general Metropolis Hastings ratio is the product of the Bayes factor, determined by the ratio of the posteriors, g, and the ratio of the Markov Chain Monte Carlo (MCMC) proposal densities, q, whose values depend on the current MCMC step.

$$ratio = \frac{g(\tau_n K_n | x_1, \ldots, x_t)}{g(\tau_o K_o | x_1, \ldots, x_t)} \times \frac{q(\tau_o K_o | \tau_n K_n)}{q(\tau_n K_n | \tau_o K_o)}$$

When the log likelihood of the equation is taken,

$$\begin{aligned} log(ratio) = &\Big[log\big[g(\tau_n K_n | x_1, \ldots, x_t)\big] - log\big[g(\tau_o K_o | x_1, \ldots, x_t)\big]\Big] \\ &+ \Big[log\big[q(\tau_o K_o | \tau_n K_n)\big] - log\big[q(\tau_n K_n | \tau_o K_o)\big]\Big] \end{aligned}$$

As shown by Kass and Wasserman (1995), the log of the Bayes Factor can be approximated with BIC with an error on the order of $O(n^{-1/2})$ when the data size is greater

than 25 and the prior follows a normal distribution. Therefore,

$$log\big[g(\tau_n K_n|x_1,\ldots,x_t)\big] - log\big[g(\tau_o K_o|x_1,\ldots,x_t)\big] \approx \frac{-\Delta BIC}{2}$$

which means that

$$log(ratio) \approx \frac{-\Delta BIC}{2} + \Big[log\big[q(\tau_o K_o|\tau_n K_n)\big] - log\big[q(\tau_n K_n|\tau_o K_o)\big]\Big]$$

In the case of a birth step,

$$q(\tau_o K_o|\tau_n K_n) = c \cdot d \cdot Poisson(K_{old}, \lambda), \quad q(\tau_n K_n|\tau_o K_o) = c \cdot b \cdot Poisson(K_{old}, \lambda).$$

In the case of a death step,

$$q(\tau_o K_o|\tau_n K_n) = c \cdot b \cdot Poisson(K_{new}, \lambda), \quad q(\tau_n K_n|\tau_o K_o) = c \cdot d \cdot Poisson(K_{old}, \lambda).$$

In the case of a move step, irrelevant of whether it is specifically jiggle or jump,

$$log\big[q(\tau_o K_o|\tau_n K_n)\big] - log\big[q(\tau_n K_n|\tau_o K_o)\big] = 0$$

Henceforth, for a move,

$$log(ratio) \approx \frac{-\Delta BIC}{2}$$

2.4 AR model and draws

Once a step has been completed and a new breakpoint set is proposed then the data is fit using an auto-regressive model. With this information then we can get a draw of the β coefficients and σ.

2.5 Derivations of β and σ draws

Peseran (2006), figured our the posterior draws for both β and σ when looking at linear models. the posterior for the β coefficients is

$$\beta|\sigma^2, b_0, B_0, v_0, d_0, S_t, Y_t \sim N(\overline{\beta_j}, \overline{V_j})$$

where

$$\overline{V}_j = (\sigma^{-2}X^TX + B_0^{-1})^{-1}, \quad \overline{\beta}_j = \overline{V}_j(\sigma^{-2}X^TY_t + B_0^{-1}b_0).$$

The conditions are the following: b_0 is the mean of the β coefficients, B_0 is the variance co-variance matrix of the β coefficients for the prior, v_0 and d_0 are the parameters of the inverse gamma prior of the inverse gamma squared (one being the shape the other rate), S_t is the current breakpoint set, and Y_t is the actual data values and Y_{t-p} is the lagged data values.

Pesaran (2006) derives the σ posterior such that

$$\sigma_j^{-2} \sim \Gamma(v_0, d_0) \longrightarrow \sigma_j^{-2}|\beta, b_0, B_0, v_0, d_0, S_t, Y_t \sim \Gamma(\overline{v}_0, \overline{d}_0)$$

where

$$\overline{v}_0 = v_0 + \frac{n_j}{2}, \quad \overline{d}_0 = d_0 + \frac{1}{2}(Y_t - X\beta)^T(Y_t - X\beta).$$

2.6 Simulations to evaluate

3 Results

Middle initial placement makes the function mad.

4 Discussion

5 Appendix

Acknowledgements

This research was funded by the National Science Foundation, grant number #1650222. The research was suppoerted by the Lafayette College Research Experience for Undergraduates (REU) Summer 2018.

References

Bai, J. and Perron, P., (1998). *Estimating and testing linear models with multiple structural changes*. Econometrica, pp.47-78.

Bai, J. and Perron, P., (2003). *Computation and analysis of multiple structural change models*. Journal of applied econometrics, 18(1), pp.1-22.

Denison, D.G.T., Mallick, B.K. and Smith, A.F.M., (1998). *Automatic Bayesian curve fitting*. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 60(2), pp.333-350.

DiMatteo, I., Genovese, C.R. and Kass, R.E., 2001. *Bayesian curvefitting with freeknot splines*. Biometrika, 88(4), pp.1055-1071.

Gamber, E.N., Liebner, J.P. and Smith, J.K., (2016). I*nflation persistence: revisited*. International Journal of Monetary Economics and Finance, 9(1), pp.25-44.

Kass, R.E. and Wasserman, L., (1995). *A reference Bayesian test for nested hypotheses and its relationship to the Schwarz criterion.* Journal of the american statistical association, 90(431), pp.928-934.

McLeod, A.I. and Zhang, Y., (2008). *Improved subset autoregression: With R package.* Journal of Statistical Software, 28(2), pp.1-28.

Pesaran, M.H., Pettenuzzo, D. and Timmermann, A., (2006). *Forecasting time series subject to multiple structural breaks.* The Review of Economic Studies, 73(4), pp.1057-1084.

Pesaran, M.H. and Timmermann, A., (2002). *Market timing and return prediction under model instability.* Journal of Empirical Finance, 9(5), pp.495-510.

R Core Team(2017). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing

Ruggieri, E.,(2013). *A Bayesian approach to detecting change points in climatic records.* International Journal of Climatology, 33(2), pp.520-528.

Seidel, D.J. and Lanzante, J.R., (2004). *An assessment of three alternatives to linear trends for characterizing global atmospheric temperature changes.* Journal of Geophysical Research: Atmospheres, 109(D14).

Wallstrom, G., Liebner, J. and Kass, R.E., (2008). *An implementation of Bayesian adaptive regression splines (BARS) in C with S and R wrappers.* Journal of Statistical Software, 26(1), p.1.

Zeileis, A., Leisch, F., Hansen, B., Hornik, K., Kleiber, C. and Zeileis, M.A., (2007). *The strucchange Package.* R manual.

Zhou, S. and Shen, X., (2001). *Spatially adaptive regression splines and accurate knot selection schemes*. Journal of the American Statistical Association, 96(453), pp.247-259.