An Exact Approach to Bayesian Sequential Change Point Detection

Eric Ruggieri^{a,*} and Marcus Antonellis^a

^a Department of Mathematics and Computer Science, College of the Holy Cross, 1 College Street, Worcester, MA 01610, USA

* Corresponding author. Tel: (508) 793-2400

E-mail addresses: eruggier@holycross.edu (E. Ruggieri), mjanto15@g.holycross.edu (M. Antonellis)

Abstract:

Change point models seek to fit a piecewise regression model with unknown breakpoints to a data set whose parameters are suspected to change through time. However, the exponential number of possible solutions to a multiple change point problem requires an efficient algorithm if long time series are to be analyzed. A sequential Bayesian change point algorithm is introduced that provides uncertainty bounds on both the number and location of change points. The algorithm is able to quickly update itself in linear time as each new data point is recorded and uses the exact posterior distribution to infer whether or not a change point has been observed. Simulation studies illustrate how the algorithm performs under various parameter settings, including detection speeds and error rates, and allow for comparison with several existing multiple change point algorithms. The algorithm is then used to analyze two real data sets, including global surface temperature anomalies over the last 130 years.

Keywords: Dynamic Programming; Exact Bayesian Inference; Global Temperature Anomalies; Multiple Change Point; Piecewise Regression

Availability: Matlab code for the Sequential Bayesian Change Point algorithm and the associated data sets described in this manuscript can be obtained by contacting the first author, Eric Ruggieri, at eruggier@holycross.edu

1. Introduction

Long time series are often heterogeneous in nature. As Chopin (2007, p. 349) notes, ‘the assumption that an observed time series follows the same fixed stationary model over a very long period is rarely realistic.’ Since the inability to recognize a regime change in a data set can have a detrimental effect on an algorithm’s predictive performance, the ultimate goal of change point analysis is to fit a piecewise regression model to a data set where the exact timing of these regime changes is unknown. A ‘change point’ is defined as an abrupt shift in the parameters of a model. Common examples include detecting a change in the mean, variance or trend of the response variable. Ideally, one would want to identify exactly when this ‘change point’ occurs so that each regime can be separately fit by an appropriate model.

Change point models have been applied in a wide range of settings including finance (Chopin, 2007), climate (Ruggieri et al., 2009; Gallagher et al., 2012; Ruggieri, 2013), biology (Fearhead and Liu, 2007), earthquake data (Grigg and Spiegelhalter 2008), DNA segmentation (Liu and Lawrence, 1999), historical time series such as average annual wage growth (Western and Kleykamp, 2004), and in other areas where long sequences of data are available. A Bayesian approach to the change point problem is particularly appealing for two reasons. First, a Bayesian approach does not rely on the asymptotic assumptions about test statistics that are present in frequentist algorithms, which can be problematic in situations where the parametric models considered are restricted to a finite, possibly small interval of time (Chopin, 2007). Second, a Bayesian approach will allow one to quantify uncertainty both in the number and the positioning of the change points.

Here, we seek to describe an efficient and exact Bayesian change point model that can quickly update itself as each new observations is recorded [known as sequential change point detection]. Each data point will require an update that has linear time complexity, giving the algorithm an overall complexity that is quadratic in the length of the data set. This stands in stark contrast to the exponential time required by a brute force approach. Once a new change point has been detected, the algorithm samples directly from the exact posterior distribution on the number of change points, their locations, and the parameters of the regression model, yielding uncertainty estimates for each of these quantities.

The rest of the article is organized as follows. Section 2 describes some of the existing change point techniques, with a special focus on sequential Bayesian methods. Following the literature review, Section 3 presents an overview of the Bayesian Change Point algorithm of Ruggieri (2013) that will be used as the foundation for the sequential method presented in this paper. In Section 4, we describe how the Bayesian Change Point algorithm of Ruggieri (2013) can be modified to handle sequential observations. In Section 5, the new sequential change point detector tested on simulated data sets as well as two real data sets and compare the results to several existing multiple change point algorithms. Section 6 provides discussion and conclusions.

2. Background

Due to the vast array of change point algorithms that have been developed, a full review of existing change point models is beyond the scope of this paper. Instead, the focus will be on giving a brief overview of some of the prominent categories of change point algorithms, emphasizing those which are most similar to what is proposed in this paper. Specifically, attention will focus on highlighting the similarities and differences among the most popular Bayesian sequential change point algorithms.

2.1 Batch vs. Sequential Change Point Algorithms

There are two main branches of change point detection algorithms, batch and sequential. In the batch setting, the data set is fixed and an algorithm will retrospectively look for change points in a time series. Given a time series with N observations, there are approximately $\binom{N}{k}$ ways to place k change points in the time series, partitioning the data set into $k+1$ segments. Because the number of solutions grows exponentially in the number of change points, an exact solution will require an efficient algorithm in order to be practical. On the frequentist side, solutions to the change point problem in regression often focus on minimizing squared error. For example, dynamic programming algorithms (e.g. Auger and Lawrence, 1989; Bai and Perron, 2003; Ruggieri et al., 2009) reduce the complexity to quadratic in the number of observations and are guaranteed to find the optimal solution. Another popular approach is binary segmentation (Scott and Knott 1974), which is a greedy approach that recursively splits a data set at each identified change point until only homogeneous segments remain. A recent adaptation is that of Fryzlewicz (2013), whose wild binary segmentation algorithm uses a localized [rather than global] cumulative sum statistic to recursively split the data set. One shortcoming of these frequentist algorithms is that they are unable to quantify the uncertainty associated with their solutions, both in the number and locations of the change points. On the Bayesian side, Gibbs sampling (e.g. Carlin et al., 1993; Stephens, 1994; Western and Kleykamp, 2004) and MCMC (e.g. Barry and Hartigan, 1993; Green, 1995; Chib, 1998; Lavielle and Lebarbier, 2001) approximations have dominated the probabilistic solutions to the change point problem, although convergence issues exist (Fearnhead, 2006; Whiteley et al., 2011). Specifically, MCMC procedures (e.g. Stephens (1994); Lavielle and Lebarbier (2001)) which update change point locations one at a time or condition on latent variables associated with each segment (Chib, 1998) can be slow mixing due to the strong correlations in the target distribution (Whiteley et al., 2011). Alternatively, Fearnhead (2006) and Ruggieri (2013) both avoid these approximations by directly sampling from the posterior distribution through the use of dynamic programming-like recursions.

Conversely, sequential change point algorithms need to be able to handle a constant stream of new observations and make a decision about whether or not a change has occurred based

only on the data observed to that point. Essentially, a sequential change point algorithm strives to make a ‘quick’ update of the inference as each new observation becomes available, rather than having to re-analyze the entire data set. The ultimate goal is to be able to quickly detect a change in the system, subject to some tolerable limit on the risk of a false positive [i.e. false detection of a change point]. An appropriate balance has to be struck since the desire to detect a change point quickly can lean to a high risk of false positives, while avoiding false alarms too strenuously can cause a longer delay between the actual occurrence of the change and its detection by the algorithm. If a change has been detected, then the algorithm can then stop to make an inference about its location before moving on to the next observation; however, if no change is detected, then the algorithm simply moves on to process the next observation in the sequence.

2.2 Sequential Change Point Detection

As with the batch change point methods, there are both frequentist and Bayesian algorithms that deal with sequential change point detection. On the frequentist side, methods involving variations of the cumulative sum [CUSUM], exponentially weighted moving average [EWMA], and likelihood ratio test statistics have been the most popular. For the likelihood ratio test, the null hypothesis of no change point is tested against the alternative of a single change point at each data point (see Siegmund and Venkatraman (1995), Hawkins et al. (2003), Ross (2013), and references therein). A change point is detected if the test statistic exceeds some threshold value which takes into account the multiple testing that has occurred. The EWMA control scheme monitors small changes in the mean of a system by using a weighting factor which gives less and less weight to data that are further removed in time (Lucas and Saccucci, 1990). Instead of using just the most recent data points, CUSUM statistics use all past data points for inference [Page 1954]. The CUSUM statistic monitors either the mean or the variance of the residuals to raise a warning when it appears that the parameters of the model have changed. Generally, the cumulative sum of scaled residuals is updated in a recursive fashion as new observations arrive. The idea behind CUSUM is that as long as the time series remains centered at the mean, μ , then a plot of the cumulative sum of the residuals will show a random pattern centered at zero. However, if the mean increases at a change point, then the CUSUM statistic will drift upward [and vice versa if the mean decreases at a change point].

2.3 Bayesian Sequential Change Point Detection

However, the main focus of this paper is on Bayesian sequential change point algorithms. The Bayesian algorithms can predominantly be characterized as MCMC, HMM, and particle filter algorithms, or combinations thereof. If the number of change points is known in advance, then the MCMC approach of Chib (1998) can be used to approximate the posterior distribution of the change point locations. Given k change points, Chib reparameterizes the change point

model in terms of a unidirectional HMM that is required to begin in state 1 and terminate in state $k+1$. A prior that imposes a specified number of change points can potentially lead to undesirable behavior at the end of the sample (Koop and Potter 2009). Koop and Potter (2007, 2009) have developed extensions of Chib (1998) that allow for an unknown number of breaks, most recently by developing a more non-informative uniform prior. A second example is Chopin (2007), who presented a hybrid algorithm that relies on both particle filtering and MCMC ideas to search for an unknown number of change points. Chopin utilizes MCMC sampling to make the particle filter more computationally efficient. One downside is that Chopin must set a prior distribution over the duration of a regime [i.e. the distance between successive change points]. This implicitly forces a prior distribution over the number of change points. A prior of long duration will make it hard to detect change points, while one of short duration may produce spurious change points.

One of the more popular sequential Bayesian change point algorithms is called Bayesian Online Change Point Detection [BOCPD] (Adams and MacKay, 2007). BOCPD makes the common assumption of a product partition model (Barry and Hartigan, 1993) where each data segment is assumed independent of all others, and uses a recursive message passing algorithm to calculate the probability distribution of the current 'run length', or distance since the previous change point. As opposed to making inferences about the joint distribution of all change points in a data set, exact inference about the run length can be made after specifying both an underlying predictive model [ex. iid Gaussian] and a hazard function which describes the likelihood of a change point at a given run length. Each step of the algorithm has linear time complexity, since the posterior distribution over the run length at time t is a vector of length t . One suggestion for reducing the complexity further is to place a threshold over which run lengths can be considered, which yields a constant average complexity per iteration on the order of the expected run length.

Because BOCPD can be sensitive to hand-selected hyper-parameters of the model, Saatci et al. (2010) developed a generalization of BOCPD that learns the hyper parameters of the model from the data and also removes the assumption of independence between data segments. Adams and MacKay (2007) treat the hyper-parameters for both the predictive model and the hazard function as fixed and known. This is especially problematic for change point detection as the hazard rate implicitly places a prior distribution over the distance between adjacent change points, which is undesirable. Additionally, Saatci et al. (2010) note that the posterior distribution on the run length forms a vector of length t , which is a drawback for long time series since it requires $t + 1$ updates to propagate it to the next time step. Thus, Saatci et al. (2010) make two suggestions to reduce the computational complexity: i) consider only the K most probable run lengths or ii) eliminate run lengths which have probability below some threshold value. One problem with pruning nodes is that nodes with small weights can become

important in the future. However, once the nodes are pruned they cannot be brought back and the algorithm would no longer provide exact inferences.

Standard MCMC methods are poorly equipped to handle online analysis of growing data sets (Fearnhead and Clifford, 2003), so sequential Monte Carlo methods, called particle filters, are often used instead. As with BOCPD, both an underlying predictive model and a hazard function must be specified in order to make exact online inferences about the ‘run length’, which again implicitly forces a prior distribution over the number of change points. Fearnhead and Clifford (2003) propose a particle filter structured as an HMM that uses resampling to limit the computational burden at each time step. The approximate distribution can be described by a set of support points, called particles, each of which represent one possible realization of the history of both the hidden and observed states of the system. These particles are propagated through time and their weights are updated to take into account the information contained in each new observation. Resampling the particles at each step is necessary to avoid an exponentially increasing number of particles. Alternatively, Fearnhead and Liu (2007) propose an online algorithm for the exact filtering of multiple change point problems which enables direct simulation from the true joint posterior of the number and position of change points. Here, each particle represents the time since the most recent change point [rather than the entire state space in the HMM], so the goal is to calculate the posterior distribution of this quantity, which is a calculation of linear complexity. Thus, the computational cost of Fearnhead and Liu’s (2007) exact algorithm is quadratic in the number of observations. A resampling procedure can reduce the complexity to constant [the number of particles] at each time step, but introduces small errors that can accumulate over time because once a particle has been removed, it cannot be resurrected.

Finally, Whitely et al. (2011) introduced a Particle MCMC algorithm to perform multiple change point analysis, which is a class of MCMC algorithms that allow sequential Monte Carlo methods to be used in building high-dimensional proposals within an MCMC scheme. Specifically, Whiteley et al. (2011) proposes a MCMC algorithm which uses the sequential Monte Carlo procedure of Fearnhead and Liu (2007) to efficiently produce full Bayesian inference in the presence of parameter uncertainty. Each MCMC iteration is $O(MT)$, where M is the number of particles used in the sequential Monte Carlo approximation of the filtering distribution and T is the length of the data segment analyzed to that point. This is as opposed to exact MCMC algorithms which have quadratic complexity as they rely on calculating the exact filtering distributions and likelihood functions. However, like Fearnhead and Liu (2007) [among others], this algorithm employs a hazard function which conditions on the location of the most recent change point.

2.4 Limitations of Existing Bayesian Sequential Change Point Approaches

Because the hazard rate can have a considerable impact on the inference, Wilson et al. (2010) states that one practical limitation of existing online algorithms is the unrealistic assumption that the frequency which change points occur is fixed and known in advance. Since the ‘best’ value will not always be obvious in advance, Wilson et al. (2010) develop a hierarchical model that allows that hazard rate to be inferred from the data. Wilson et al. (2010) also describe a pruning strategy of merging similar nodes in the particle filter to reduce the computational complexity.

Sequential Monte Carlo methods also suffer from the well-known particle path degeneracy problem (Yildirim et al., 2013) related to estimating the static parameters of the underlying predictive model. In short, the sequential Monte Carlo estimates of the sufficient statistics necessary to perform the MCMC update of the particles degrade as n increases because they are based on successive approximations of a joint distribution that cannot be consistently estimated over time. The particles tend to merge as the algorithm progresses through the data set due to the resampling that takes place at each time step. To avoid this problem, Yildirim et al. (2013) present an online EM algorithm for estimating the static parameters of change point models whose cost is linear in the number of particles, as opposed to quadratic for the exact filtering distribution.

2.5 How this Approach Differs

In summary, many of the existing sequential Bayesian change point algorithms only approximate the posterior distribution over the location of the change points and impose a prior distribution on ‘run length’, which can strongly affect their inference. The Sequential Bayesian Change Point algorithm presented below allows direct simulation from the exact posterior distribution on the change point locations and the parameters of the regression model without imposing any assumptions on the distance between adjacent change points, all while maintaining a linear time complexity per time step. The description of the [batch] Bayesian Change Point algorithm (Ruggieri, 2013) given in the next section provides the foundation upon which the Sequential Bayesian Change Point algorithm proposed in Section 4 will be constructed. In terms of the recursions utilized by the algorithm, they are fundamentally different from those seen in MCMC and HMM algorithms that typically dominate Bayesian change point analysis.

3. The Recursive Structure of the Bayesian Change Point Algorithm

The Bayesian Change Point algorithm introduced by Ruggieri (2013) assumes a linear regression model. However, the algorithm is applicable to a wide range of underlying predictive models, limited only by one’s ability to specify an appropriate model. For example, Liu and Lawrence (1999) use a multinomial function to model DNA sequences. In what follows, it is assumed that

a linear regression model is being used to fit a time series, although that is by no means the only potential application of this algorithm.

Given the dependent variable, Y , and m known predictor variables $X_1, \dots, X_m$, a linear regression model takes the form:

$$Y = \sum_{l=1}^m \beta_l X_l + \varepsilon \quad (1)$$

where β_l is the l^{th} regression coefficient and ε is a random error term. The predictors, X_l , are functions of time for a time series, but in general can take on any form. For example, this model is sufficiently general to include periodic phenomenon and many common time series representations such as AR and ARMA.

Assume that the regression model given by Equation 1 applies between each pair of consecutive change points. To solve the multiple change point problem, the probability of the data given the regression model $[f(Y_{i:j}) = f(Y_{i:j} | X)]$ is calculated for every possible sub-string of the data, $Y_{i:j} = \{Y_i, Y_{i+1}, \dots, Y_{j-1}, Y_j\}$, $1 \leq i < j \leq N$. For example, $f(Y_{i:j})$ is multivariate Normal if the error terms, ε , are assumed to be independent, normally distributed random variables. The values are then stored in memory for later use. The ability to reuse these calculations is what reduces the algorithm's complexity from exponential to quadratic.

Dynamic programming works by taking a complex problem [i.e. the multiple change point problem] and breaking it down into a series of simpler problems, the smallest of which [i.e. the single change point problem] can easily be solved. Let $P_k(Y_{1:j}) = P_k(Y_{1:j} | X)$ be the probability density of the first j observations containing k change points given the regression model. To place the first change point, start at one end of the time series and piece together two non-overlapping substrings, which are assumed to be independent according to the product partition model (Barry and Hartigan 1993) that underlies this approach. Therefore, probability of any prefix of the data $[Y_{1:j}]$, the first j data points in a time series] can be calculated by multiplying together the probabilities of these two substrings [calculated above] and summing [marginalizing] over all possible placements of this first change point. In other words, when $k = 1$, $P_1(Y_{1:j}) = \sum_{v=1}^{j-1} f(Y_{1:v}) * f(Y_{v+1:j})$. To be consistent with the notation used below, the homogeneous segment $f(Y_{1:v})$ can be viewed as equivalent to $P_0(Y_{1:v})$.

To place the second change point, two non-overlapping segments are again pieced together, but this time the first segment contains a single change point [previously calculated] while the second does not. As before, the probability of any prefix of the data containing two change points can be calculated by multiplying together the probabilities of these two substrings and then marginalizing over all possible placements of this second change point, $P_2(Y_{1:j}) = \sum_{v=1}^{j-1} P_1(Y_{1:v}) * f(Y_{v+1:j})$. Additional change points are added in a similar manner,

296 $P_k(Y_{1:j}) = \sum_{v=1}^{j-1} P_{k-1}(Y_{1:v}) * f(Y_{v+1:j})$, until the maximum number of allowed change points,
 297 k_{max} , has been reached.

298 Once complete, Bayes Rule can be used to make inferences about the parameters of
 299 interest by drawing samples directly from their exact posterior distribution. The sampling
 300 procedure is simple and efficient. Each sampled solution begins by selecting a number of
 301 change points from the posterior distribution. After the number of change points has been
 302 established, their locations are recursively sampled. Finally, the parameters of the regression
 303 model are sampled for each regime bounded by a pair of change points. More importantly, by
 304 looking holistically at the set of sampled solutions, uncertainty bounds for both the number and
 305 locations of the change points can be established. The three steps of the Bayesian Change
 306 Point algorithm are summarized below.

307

308 **[1] Calculating the Probability Density of the Data $f(Y_{ij} / X)$:**

309 The exact form of this calculation depends on the nature of the underlying predictive
 310 model. For example, Ruggieri (2013) assumes iid Gaussian errors along with conjugate
 311 priors for the parameters of the regression model. This quantity [$f(Y_{ij}) = f(Y_{ij} / X)$] is
 312 calculated and then stored in memory for all possible substrings of the data, Y_{ij} , with $1 \leq i < j$
 313 $\leq N$.

314 **[2] Forward Recursion [Dynamic Programming]:**

315 Let $P_k(Y_{1:j})$ be the density of the data $[Y_{1:} \dots Y_j]$ with k change points. For $k > 0$, define

$$P_k(Y_{1:j}) = \sum_{v < j} P_{k-1}(Y_{1:v}) f(Y_{v+1:j}) \quad (2)$$

316 for $j = (k+1):N$, where $P_0(Y_{1:v}) = f(Y_{1:v})$ as calculated in step 1 of the algorithm.

317 **[3] Stochastic Backtrace via Bayes Rule:**

318 Two additional quantities need to be specified in order to have a completely defined
 319 partition function / normalization constant. First, the prior distribution on the number of
 320 change points is assumed to have a uniform distribution [i.e. $f(K=k) = 1/k_{max}$]. Next,
 321 suppose that the data set contains k change points and that the location of the change
 322 points are given by $C = \{c_1, c_2, \dots, c_k\}$. An uninformative prior distribution is placed on the
 323 location of the change points which assumes that all change point solutions with exactly k
 324 change points are equally likely, i.e. $f(c_1, \dots, c_k | K = k) = 1/N_k$, where N_k is the number of
 325 possible solutions containing k change points. This combinatorial prior accounts for the
 326 growing number of potential solutions as the number of change points increases. The
 327 uniform nature of this prior avoids the use of a hazard function that would place an implicit
 328 prior on the number of change points akin to several of the existing sequential change point
 329 methods (see for example, Adams and Mackay (2007) and Fearnhead and Liu (2007),
 330 among others). Taken together, the normalization constant can now be specified:

$$f(Y_{1:N}) = \sum_{k=0}^{k_{max}} \sum_{c_1 \dots c_k} f(Y_{1:N} | K = k, c_1 \dots c_k) * f(K = k, c_1 \dots c_k) \quad (3)$$

The use of a non-uniform prior would have to be integrated into the Forward Recursion step. However, because the prior on the distribution of the change points is uniform, it can be factored out and incorporated once the Forward Recursion step [Equation 2, the inner summation] is complete. With a fully specified normalization constant, the parameters of interest can now be sampled directly from their respective posterior distributions.

[3.1] Sample a Number of Change Points, k :

Using Bayes Rule:

$$f(K = k | Y_{1:N}) = \frac{P_k(Y_{1:N})f(K = k, c_1, \dots, c_k)}{f(Y_{1:N})} \quad (4)$$

with $P_k(Y_{1:N}) = f(Y_{1:N} | K = k)$ calculated on the Forward Recursion step and $f(Y_{1:N})$ defined in Equation 3 above.

[3.2] Sample the Locations of the Change Points, c_k :

Define $c_{K+1} = N$. Then, for $k=K, K-1, \dots, 1$

$$f(c_k | c_{k+1}) = \frac{P_{k-1}(Y_{1:v})f(Y_{v+1:c_{k+1}})}{\sum_{v \in [k-1, c_{k+1})} P_{k-1}(Y_{1:v})f(Y_{v+1:c_{k+1}})} \quad (5)$$

When $k = 1$, $P_0(Y_{1:v}) = f(Y_{1:v})$ is given by the calculation in Step 1 of the algorithm.

[3.3] Sample the Regression Parameters for the Interval Between Adjacent Change Points c_k and c_{k+1} :

The exact form of this sampling depends on the nature of the underlying predictive model being used. Assuming iid Gaussian errors and conjugate priors for the regression parameters, sampling the regression parameters requires a random draw from the multivariate normal distribution.

Calculating the probability of the data is $O(N^2)$, the Forward Recursion is $O(k_{max}N^2)$, and Stochastic Backtrace steps are $O(kN)$. Therefore, the algorithm has a total time complexity of $O(N^2)$. See Ruggieri (2013) for full implementation details using a regression model with uncorrelated error terms.

4. Adaptation to a Sequential Algorithm

4.1 Altering the Order of Recursion

The Bayesian Change Point algorithm described in Section 3 takes a single data set of length N and looks for changes in that data set. However, if a new observation is added to the time series [such as an observation from the current time] then the entire algorithm has to be rerun, a process that becomes increasingly time consuming as the size of the data set increases. Here, we describe how to adapt the retrospective Bayesian Change Point algorithm to handle

sequential data. The main modification to the algorithm consists of changing the order of the dynamic programming recursions. The resulting posterior distributions of the parameters of interest for this new sequential algorithm are identical to what would be obtained by a retrospective batch analysis. In what follows, assume that t data points have already observed / analyzed. If the data set is dynamic, then $t = N$; however, if a static data set is being analyzed sequentially, then $t < N$. In either case, suppose that observation $t+1$ is next to be evaluated.

Two matrices are created to store values for use by the Bayesian Change Point algorithm. The first is a $t \times t$ matrix created in step 1 of the algorithm that holds the probability of each and every possible substring of the data. Suppose that the row index corresponds to the starting point of a segment and the column index corresponds to the ending point of that segment. For example, given two indices, $i < j$, row i , column j of this matrix will contain $f(Y_{i:j})$, the probability density of $Y_{i:j}$ given the regression model. As each new observation arrives, this matrix needs to expand by one row and one column. Row $t+1$ consists of the probability density of data segments that begin at time $t+1$ and end at the column index. Since a data segment cannot end before it begins, the entire row will be set to zero. On the other hand, column $t+1$ consists of the probability density of the data segments that begin at the row index and end at time $t+1$. To fill out this column, $f(Y_{i:t+1})$ is calculated for $i=1$ to $t+1$. Thus, when each new observation arrives, an $O(T)$ operation is needed to update the matrix containing the probability densities of the data.

$f(Y_{ij})$	1	2	...	t-1	t	t+1
1						$f(Y_{1:t+1})$
2						$f(Y_{2:t+1})$
$\vdots$						$\vdots$
t-1						$f(Y_{t-1:t+1})$
t						$f(Y_{t:t+1})$
t+1	0	0	...	0	0	$f(Y_{t+1:t+1})$

The second matrix is created on the Forward Recursion step of the Bayesian Change Point algorithm. Let k_{max} be the maximal number of change points that the user is interested in finding. Then row k , column j of this $k_{max} \times t$ matrix contains the value $P_k(Y_{1:j})$. As each new observation is recorded, this matrix has to expand by one column. To fill out this matrix, one must calculate

$$P_k(Y_{1:(t+1)}) = \sum_{v < t+1} P_{k-1}(Y_{1:v}) f(Y_{v+1:(t+1)}) \quad (6)$$

for $k = 1$ to k_{max} , where $P_0(Y_{1:v}) = f(Y_{1:v})$ as in Equation 2. Thus, each new observation will result in an $O(k_{max}T)$ operation to update the matrix used in the Forward Recursion step of the Bayesian Change Point algorithm.

$P_k(Y_{ij})$	1	2	...	t-1	t	t+1

1						$P_1(Y_{i:t+1})$
2						$P_2(Y_{i:t+1})$
$\vdots$						$\vdots$
k_{max}						$P_{k_{max}}(Y_{i:t+1})$

4.2 Detecting a Change Point

The posterior distribution on the number of change points can be monitored in order to determine when a new change point has occurred. At each time step, an $O(k_{max})$ calculation is performed:

$$f(K = k | Y_{1:(t+1)}) = \frac{P_k(Y_{1:(t+1)})f(K = k, c_1, \dots, c_k)}{f(Y_{1:(t+1)})} \quad (7)$$

which uses the quantity, $P_k(Y_{1:(t+1)})$, given by Equation 6 in the preceding paragraph. Note that if the posterior distribution on the number of change points begins to approach the value of k_{max} as the data set grows, then the value of k_{max} can be increased accordingly and additional rows can be added to the P_k matrix using the forward recursion step outline in the batch algorithm [Equation 2]. A shift in the posterior distribution towards a larger number of change points would indicate that a new change point has been detected. Three potential criteria for detecting a change point include monitoring:

- a) the average [expected value] number of change points in the data set
- b) the smallest value of k such that $P(K > k) > 0.50$ [or any other percentile]
- c) the largest value of $P(K = k)$

These three criteria amount to studying the mean, median, and mode of the posterior distribution on the number of change points, respectively.

Once a change point has been ‘detected’, determining the exact posterior distribution of the location of the most recent change point is a straightforward $O(T)$ calculation. Let c_k be the location of the k^{th} (most recent) change point and $c_{k+1} = t+1$ be the most recent observation. Then

$$f(c_k | c_{k+1} = t + 1) = \frac{P_{k-1}(Y_{1:v})f(Y_{v+1:(t+1)})}{\sum_{v \in [k-1, (t+1))} P_{k-1}(Y_{1:v})f(Y_{v+1:(t+1)})} \quad (8)$$

Additional change points can be sampled from their respective posterior distributions in a recursive fashion [Equation 5]. One caution is that the posterior distribution of the location of the most recent change point may change as more data is collected. Thus, the initial location should be confirmed once more observations are recorded.

In total, each new observation requires an $O(T)$ operation for the Sequential Bayesian Change Point algorithm to update itself.

5. Simulations and Analysis of Real Data Sets

Because data sets of finite length are generally of interest, studying the asymptotic properties of these estimators [as is often done in the frequentist setting] would not be meaningful (see Chopin (2007) for a more complete discussion). Instead, the relative merits of these three detection criteria in terms of speed and accuracy of detection will be studied through simulated and real data sets by comparing them amongst themselves and to other popular sequential change point algorithms such as CUSUM, wild binary segmentation (Fryzlewicz 2014) and BOCPD (Adams and MacKay, 2007), as well as the algorithms developed by Fearnhead and Clifford (2003) and Whiteley et al. (2011).

As was done in Ruggieri (2013), the error terms of the regression model [Equation 1] are assumed to be independent, mean zero, normally distributed random variables, and conjugate priors have been chosen for the regression parameters, β , and the error variance, σ^2 . Specifically, β is multivariate Normal, $\beta/\sigma^2 \sim N(\beta_0, \sigma^2/k_0)$, where k_0 is a scale parameter relating the variance of the regression coefficients to the residual variance, and $\sigma^2 \sim \text{Scaled-Inverse } \chi^2(v_0, \sigma_0^2)$, where v_0 and σ_0^2 act as pseudo data points [essentially, unspecified training data] - v_0 pseudo data points of variance σ_0^2 . The value of v_0 is fixed at 1. Thus, for each simulation, there are four parameters that can be tuned:

- a) k_0 : The value of k_0 has two effects on the algorithm. First, the smaller the value of k_0 , the more the regression parameters are allowed to deviate from their prior values. This becomes especially important as the length of the data set grows. For example, suppose that a change in the trend of a data set exists. Then a logical choice for β_0 might be $\beta_0 = 0$. If the trend changes from positive to negative, then the constant will also shift dramatically. Too large of a value for k_0 might hinder this shift and bias the regression coefficients inferred for the model towards the prior mean. Second, since the probability of each segment has a $(k_0)^{m/2}$ term in the numerator, small values of k_0 will also introduce a penalty against additional change points in terms of a proportionally smaller probability for the multiple change point solution when compared to the no change point solution.
- b) σ_0^2 : This quantity represents a researcher's best guess at the error variance. Larger values of σ_0^2 act as a penalty against introducing a change point into the model by indirectly serving as a sort of threshold for the amount that the squared error $[(Y - X\beta)^T(Y - X\beta)]$, which is part of one of the parameters for the posterior distribution on the error variance σ^2 must be reduced in order for a change point to be significant. In essence, the data is asked to provide additional evidence that the change point exists.
- c) d_{min} : d_{min} represents the minimum distance between consecutive change points, or alternatively, the minimum segment length. Larger values of d_{min} will slow down the detection of a change point, but counteracts false positives that are due to random

variation. In general, there should be at least as many data points as parameters being estimated.

- d) *NOISE*: This quantity represents the magnitude of the random variation added to each simulated data sets. In general, change point will be more difficult to detect if the amount of noise is large relative to the underlying signal.

5.1 No Change Points – A Homogeneous Segment

The purpose of this section's simulations is to study the error rate of the Sequential Bayesian Change Point algorithm. In other words, how often does the algorithm detect a change point when none actually exist? One thousand data sets of length $N = 100$ were randomly generated for each combination of parameter values. The number of errors for each of the three detection criteria described in Section 4.2 [i.e. mean, median, and mode of the posterior distribution of the number of change points] is given below. Since the goal of these simulations is to show how the error rate changes in response to changes in the parameter values, the values of the parameters are not chosen optimally. Subtle differences in the detection criteria would be extremely difficult to identify if the parameter values were chosen to push the error rates to zero. However, from these simulations it should become clear how to choose the parameter values in order to produce a low error rate.

Two types of models were used, a constant mean, and a constant trend. Figure 1 illustrates representative data set for each situation.

Figure 1: Examples of Simulated Data Sets – No Change Points

Figure 1: Sample data sets for both the constant mean and the constant linear trend simulations. Neither data set contains a change point.

For the first simulation, the data sets are generated according to $Y = \beta$. Gaussian white noise of a specified magnitude was then added to each data set. The default parameter values are: $v_0 = 1$, $\sigma_0^2 = 1$, $k_0 = 0.01$, and $d_{min} = 5$, and the added Gaussian noise was set to have variance 1. Table 1 shows the error rates for the constant model for each of the three detection criteria under a variety of parameter settings as well as error rates for several other change point methods.

Table 1: Number of Times a Change Point was Detected, $Y = \beta$

	Mean	Median	Mode		Mean	Median	Mode
$k_0 = 0.1$	222	153	124	Noise= 0.1, $\sigma_0^2 = 1$	0	0	0
$k_0 = 0.01$	51	41	36	Noise = 1, $\sigma_0^2 = 1$	51	41	36
$k_0 = 0.001$	12	12	12	Noise = 10, $\sigma_0^2 = 1$	182	171	154
$k_0 = 0.0001$	4	4	4				
$d_{min}=2$	82	59	49	Noise = 0.1, $\sigma_0^2 = 0.1$	48	36	34
$d_{min}=5$	51	41	36	Noise = 1, $\sigma_0^2 = 1$	51	41	36
$d_{min}=10$	27	23	24	Noise = 10, $\sigma_0^2 = 10$	49	33	31
Method	Noise = 0.1	Noise = 1	Noise = 10	Method	Noise = 0.1	Noise = 1	Noise = 10
BOCPD	96	83	88	CUSUM	21	15	13
WBS	114	91	121	BCP	2	1	0
BS	13	8	9				

Table 1: The number of times that a change point was erroneously detected in the 1000 simulated data sets. The default parameter values for the sequential Bayesian change point algorithm are Noise=1, $\sigma_0^2 = 1$, $k_0 = 0.01$, and $d_{min} = 5$. For each simulation, any parameters not listed are assumed to take on their default value. Error rates for five other multiple change point algorithms are given in the bottom three rows of the table.

Several conclusions can be drawn from Table 1 about the effect each parameter value has on the error rate of the Sequential Bayesian Change Point algorithm. In general, the error rate decreases as k_0 decreases, as d_{min} increases, and as the added noise decreases. As indicated above, the value of k_0 can act as a penalty against additional change points because of its inclusion in the probability function. Larger values of d_{min} require more data before coming to a conclusion about a new change point. Since false positives are often a result of a series of random errors that are either above or below the baseline signal, requiring a larger amount of data before drawing a conclusion mitigates this random effect. As might be expected, the greater the random variation in the data set, the more difficult change point detection becomes. Interestingly, the value of σ_0^2 [a prior parameter for the variance] can offset the value of the noise parameter. If the two quantities are of the same magnitude, then the error rate stays relatively constant. On the other hand, if σ_0^2 exceeds the value of the noise, then the error rate is driven down while the opposite holds true if σ_0^2 is small relative to the added noise.

in the data set. As for the three detection criteria being studied, the mean of the posterior distribution of the number of change points had the highest error rate, followed by the median, and then the mode. In addition, if on the same data set the mean, median, and mode identified the change point at different times, the mean was often the first to detect. In other words, the mean was the most aggressive criteria for detecting change points in a data set.

To put these numbers in perspective, the simulation was also run with several existing multiple change point algorithms including:

- BOCPD: Bayesian Online Change Point Detection (Adams and MacKay 2007), a sequential Bayesian algorithm available as Matlab code
- WBS: Wild Binary Segmentation (Fryzlewicz 2014), a batch algorithm available in the R package 'wbs'
- BS: Binary Segmentation, a batch algorithm that uses a likelihood ratio test statistic to decide whether or not a change point exists in the data set. [Note: A version of BS that uses a CUSUM statistic as its testing criteria is also available in the R package 'wbs'.]
- CUSUM: A sequential implementation of the CUSUM method as found in the R package 'strucchange' (Zeileis et al. 2002). Threshold criteria for the CUSUM statistic can be found in Zeileis (2004)
- BCP: The batch Bayesian change point algorithm of Ruggieri (2013) [see Section 3], available as Matlab code

One thousand data sets were generated and Gaussian white noise of variance 0.1, 1, and 10 was added to each data set [a total of 3000 data sets]. For the frequentist algorithms [WBS, BS, and CUSUM], the error rate can be lowered by raising the threshold at which the test statistic shows significant evidence in favor of a change point. Thresholds were set to achieve a reasonable error rate and will be carried through the rest of the simulations to see how this parameter affects each algorithm's ability to detect true change points. In general, WBS had a much higher error rate than the other frequentist algorithms for this first simulation. The BS equivalent of WBS [which only considers global, rather than local CUSUM statistics] has an error rate approximately one-tenth that of WBS, making it equivalent to the other frequentist methods under consideration. In contrast, Bayesian algorithms can lower their error rates by adjusting the values of the prior parameters. For BOCPD, the parameter for average run length was set to 100 to match the length of each data set, the prior variance was set to be twice the level of the added noise, and the value for the prior parameter k was set to be 0.01. Of the 1000 simulated data sets for a noise level of 1, 83 had a point at which a homogenous segment was not the most probable outcome. This is analogous to checking the mode of the posterior distribution on the number of change points in the current approach. Saatci et al (2010) previously commented that BOCPD can be sensitive to the hand-selected prior parameters, an effect also seen in this simulation. For example, using the default values for these parameters [error variance = 1, $k = 1$] produced a false positive for 0, 167, and 252 of the 1000 simulated

data sets when the added noise was 0.1, 1, and 10, respectively. In general, increasing the value of the prior for the error variance tended to reduce the number of false positives at the expense of being able to accurately detect true change points. Therefore, it is possible that a different set of prior parameters might produce both fewer false positives and increased accuracy for this method. Overall, BCP [which is equivalent to the sequential algorithm proposed here after the full data set has been revealed, parameters $v_0 = 1$, $\sigma_0^2 = 0.01$, $d_{min} = 5$, $k_0 = 0.01$] had the fewest number of falsely detected change points using its default set of parameters. Nevertheless, it is clear that the error rates seen in Table 1 [$\sim 3\text{-}5\%$ for the default parameters] are reasonable for the type of simulation being done.

For the second simulation, data sets are generated according to $Y = \beta_1 + \beta_2 X$. Both the intercept and the trend parameters were selected from uniform distributions, with $\beta_1 \sim U(-2, 2)$ and $\beta_2 \sim U(-0.2, 0.2)$. Gaussian white noise of a specified magnitude was then added to each data set. The default parameter values were again set as: $v_0 = 1$, $\sigma_0^2 = 1$, and $d_{min} = 5$, but this time $k_0 = 0.1$ so as to create error rates large enough to see differences between the various parameter settings and detection criteria. Since the effect of k_0 on the probability function depends on the size of the model, it does not have to be chosen as small to achieve the desired error rate. Table 2 shows the error rates for the linear model for each of the three detection criteria under a variety of parameter settings as well as error rates for several other change point methods.

Table 2: Number of Times a Change Point was Detected, $Y = \beta_1 + \beta_2 X$

	Mean	Median	Mode		Mean	Median	Mode
$k_0 = 1$	45	40	39	Noise = 0.1, $\sigma_0^2 = 1$	0	0	0
$k_0 = 0.1$	22	19	19	Noise = 1, $\sigma_0^2 = 1$	22	19	19
$k_0 = 0.01$	6	6	5	Noise = 10, $\sigma_0^2 = 1$	101	96	96
$k_0 = 0.001$	1	1	1				
$d_{min} = 2$	32	25	23	Noise = 0.1, $\sigma_0^2 = 0.1$	13	13	13
$d_{min} = 5$	22	19	19	Noise = 1, $\sigma_0^2 = 1$	22	19	19
$d_{min} = 10$	6	6	6	Noise = 10, $\sigma_0^2 = 10$	22	20	20
<i>Method</i>	Noise = 0.1	Noise = 1	Noise = 10	<i>Method</i>	Noise = 0.1	Noise = 1	Noise = 10
<i>BOCPD</i>	169	93	129	<i>CUSUM</i>	3	5	2
<i>WBS</i>	0	0	0	<i>BCP</i>	0	0	0
<i>BS</i>	9	12	17				

Table 2: The number of times that a change point was erroneously detected in the 1000 simulated data sets. The default parameter values are Noise=1, $\sigma_0^2 = 1$, $k_0 = 0.1$, and $d_{min} = 5$. For each simulation, any parameters not listed are assumed to take on their default value. Error rates for five other multiple change point algorithms are given in the bottom three rows of the table.

The results in Table 2 are nearly identical to those in Table 1 in terms of how each parameter affects the inference. Specifically, the error rate of the algorithm will decrease as the value of k_0 decreases, as d_{min} increases, and as the added noise decreases. As before, this final effect is mitigated if the value of σ_0^2 is chosen to be the same magnitude as the error variance. In addition, the mean of the posterior distribution is again seen as the most aggressive of the detection criteria in terms of the number of false positives, while the mode is the most conservative of the three detection criteria.

Again, to put the error rates in perspective, the simulation was also run with five other multiple change point algorithms. However, many of the existing change point algorithms [for which the code is publically available] are designed only to detect changes in the mean, including BOCPD and WBS. As such, these algorithms were modified from their original form to be able to search for change points in a more general setting. Conversely, BCP and CUSUM are applicable to a general regression model, and so retain their original form. One important result to note is that due to the extra parameter in the regression model, error rates were in general lower than when simulating a constant model for a given set of parameters. This difference was most substantial for the WBS algorithm in which the rate of false positives dropped from ~11% to zero. Note that both WBS and CUSUM use a CUSUM statistic to search for change points. The difference in the rate of false positives for the two algorithms has to do with how the threshold criteria for the existence of a change point is defined. See Zeileis (2004) for more information.

5.2 Multiple Change Points – A Change in Trend

For this simulation, the length of the data sets was increased to $N = 175$ to accommodate the placement of three change points. The locations of the change points were selected as uniform random variables: $U(65, 75)$, $U(85, 95)$ and $U(120, 130)$, creating four segments of varying length. The intercept for the model was selected $U(-2, 2)$ and the trend for the first line segment was selected $N(0.15, 0.05^2)$, negated with probability 0.5. To avoid overly obvious change point locations, the piecewise function was made to be continuous. The change in trend from the first to the second line segment was selected $N(0.4, 0.05^2)$, from the second to the third line segment $N(0.4, 0.025^2)$, and from the third to the fourth segment $N(0.25, 0.01^2)$. Each change in trend was negated with probability 0.5. Gaussian white noise was then added to each data set. In total, one thousand data sets were generated for each simulation. Figure 2 illustrates four different data sets generated according to this procedure, providing a good representation of the types of data sets being analyzed. As seen in Figure 2, some of the change points are quite obvious, while others are much more subtle. Because of this, the Sequential Bayesian Change Point algorithm [or any change point algorithm for that matter] should not be expected to detect all three change points every single time.

Figure 2: Examples of Simulated Data Sets – Three Change Points

Figure 2: Sample data sets from the multiple change point simulation. Each data set is piecewise continuous with change points signifying a change in trend. The dashed vertical lines represent the actual placement of the three change points in each data set.

Table 3: Speed of Detection in the Multiple Change Point Simulation

$k_0 = 0.1$	Detection Speed	Detection Rate	$d_{min} = 2$	Detection Speed	Detection Rate
Mean	9.64/17.14/11.23	98.1/88.7/78.8	Mean	10.22/18.85/13.13	96.7/88.5/79.4
Median	9.64/15.44/9.79	98.1/89.6/82.6	Median	9.87/16.28/10.52	96.7/88.4/81.6
Mode	9.52/14.59/10.11	98.1/88.6/82.2	Mode	9.65/15.76/10.81	96.9/87.8/80.5
$k_0 = 0.01$	Detection Speed	Detection Rate	$d_{min} = 5$	Detection Speed	Detection Rate
Mean	10.13/24.78/20.49	99.2/81.8/73.1	Mean	9.64/17.14/11.23	98.1/88.7/78.8
Median	10.15/24.19/20.48	99.1/82.1/74.7	Median	9.64/15.44/9.79	98.1/89.6/82.6
Mode	9.88/24.18/20.53	99.2/81.7/74.7	Mode	9.52/14.59/10.11	98.1/88.6/82.2
$k_0 = 0.001$	Detection Speed	Detection Rate	$d_{min} = 10$	Detection Speed	Detection Rate
Mean	9.93/20.30/17.41	99.2/89.7/90.3	Mean	12.50/20.56/16.96	99.9/79.1/65.1
Median	9.96/20.39/17.33	99.1/90.0/91.1	Median	12.36/20.06/15.55	99.8/80.3/68.6
Mode	9.95/20.39/17.37	99.3/89.6/89.8	Mode	12.26/20.18/15.03	99.8/79.9/69.1
Noise = 0.1, $\sigma_0^2 = 1$	Detection Speed	Detection Rate	Noise = 10, $\sigma_0^2 = 1$	Detection Speed	Detection Rate
Mean	4.81/17.55/11.45	100/93.6/70.6	Mean	25.39/22.14/15.25	57.3/41.6/24.3
Median	4.84/13.44/9.51	100/97.5/73.0	Median	24.78/21.67/15.62	57.0/40.5/27.1
Mode	4.85/13.14/9.62	100/92.8/70.6	Mode	24.13/21.78/15.66	57.4/40.7/28.3

Noise = 0.1, $\sigma_0^2 = 0.1$	Detection Speed	Detection Rate	Noise = 10, $\sigma_0^2 = 10$	Detection Speed	Detection Rate
Mean	4.73/10.50/8.33	100/85.9/72.7	Mean	27.28/26.35/18.97	57.6/40.1/26.6
Median	4.76/7.56/6.97	100/89.9/73.1	Median	25.69/25.82/19.61	57.1/40.2/28.4
Mode	4.66/7.05/6.51	100/86.6/73.0	Mode	25.57/26.48/19.75	57.7/40.7/27.5

Table 3: The average detection speed for each of the three change points in the 1000 simulated data sets for each of the three detection criteria [mean, median, and mode of the posterior distribution of the number of change points]. Detection speed indicates the number of data points after the true location of the change point at which it was detected by the algorithm [first change point / second change point / third change point]. Detection rate give the percentage of times that a particular change point was identified by the algorithm. The default parameter values are $Noise=1$, $\sigma_0^2 = 1$, $k_0 = 0.1$, and $d_{min} = 5$. For each simulation, any parameters not listed are assumed to take on their default value. See the online Appendix for more complete results from this simulation.

Table 3 summarizes the algorithm's ability to identify each of the three change points using the three detection criteria described in Section 4.2 [mean, median, and mode of the posterior distribution of the number of change points]. Parameter settings for each of the simulations mimic those chosen for the simulations of Section 5.1. Again, since the goal of these simulations is to show how the detection rate and speed change in response to changes in the parameter values, the parameter values are not chosen optimally. In general, the second change point took the longest to detect, while the third was the most difficult to detect. As expected, change points became harder to detect as the magnitude of the noise added to each data set increased. This is shown both by a slower detection speed and by a lower detection rate. Changing the value of σ_0^2 to match the level of the added noise did not seem to affect the results as it did when identifying false positives. Since each change point [in general] took 10 or more data points to identify, changing the minimum segment length did not have a significant impact on the results until d_{min} approached 10. As seen previously, requiring longer segments reduces the number of false alarms by forcing that algorithm to take in more information before making a decision about a change point. However, requiring longer segments also appears to slow detection speed and hinder detection ability. Finally, lowering the value of k_0 was shown in the previous section to decrease the rate of false positives. Here, it is associated with a slower detection speed, illustrating the trade-off that exists between minimizing the number of false positives while trying to have the fastest detection rate possible.

Among the three detection criteria, there was not a large difference in the speed or rate of detection for the three change points. However, when a difference did exist, the mean of the posterior distribution tended to be slower and slightly less capable of detecting change points than the median and mode, which on the whole appear to be equivalent. Combining these results with those from the previous section, it appears that using the mean of the posterior distribution of the number of change points is the least desirable of the three detection criteria proposed. Additionally, when the parameter values are chosen so that error rates are low, the

median and mode of the posterior distribution are nearly equivalent in terms of their ability to detect true change points in a data set.

Table 4 shows how the Sequential Bayesian Change Point algorithm [Sequential BCP] compares to five other multiple change point algorithms in terms of detection ability. The median is used as the detection criteria for Sequential BCP, and its parameters were chosen to have the same values as BCP [$v_0 = 1$, $\sigma_0^2 = 1$, $d_{min} = 5$, $k_0 = 0.001$]. One thousand data sets were simulated with a white noise level of one. Undetected change points were most often the result of a particular algorithm identifying too few change points in a data set [rather than incorrectly identifying change points]. One interesting finding is that BS, CUSUM, and BOCPD had a greater proportion of false positives than was seen in the previous simulation of a linear function. In other words, these algorithms often detected more than the true number of change points in the data set. One explanation for this phenomenon is that these algorithms did not place one of the three change points correctly and so had to add another in order to correct its mistake. On the other hand, WBS, BCP and Sequential BCP were more conservative in their detection of change points, each with zero false positives. However, BCP and Sequential BCP had slightly lower detection rates than CUSUM, while WBS was significantly lower. On the positive side, it would seem that these algorithms were able to correctly place the change points and / or avoid false positives. Finally, the Bayesian algorithms (BOCPD, BCP, and Sequential BCP) are able to provide uncertainty bounds on both the number and location of change points [not shown], a benefit over a frequentist approach.

Table 4: Speed and Accuracy of Detection, A Comparison

Method	Detection Rate	Detection Speed	# Simulations <3	# Simulations =3	# Simulations >3
WBS	82.3/47.1/96.9	N/A	680	320	0
BS	94.7/88.1/97.8	N/A	111	725	164
BCP	95.8/84.4/99.4	N/A	221	779	0
CUSUM	98.3/93.8/99.3	7.26/23.31/17.92	26	902	72
BOCPD	94.6/88.7/83.5	8.66/14.25/18.70	96	370	534
Seq. BCP	99.1/90.0/91.1	9.96/20.39/17.33	284	714	0

Table 4: The average detection rate for each of the three change points in the 1000 simulated data sets for each of the six algorithms being compared as a part of the simulation study [$Noise=1$]. Detection rate give the percentage of times that a particular change point was identified by the algorithm [first change point / second change point / third change point]. Detection speed indicates the number of data points after the true location of the change point at which it was detected by the algorithm, which applies only to sequential algorithms. The final three columns indicate the proportion of simulations which over or underestimated the true number of change points.

5.3 The Well-Log Data Set – A Change in Mean

The well-log data is a geophysical data set that consists of nuclear magnetic response measurements of underground rocks. Measurements are taken at discrete time points by a probe that is lowered through a bore-hole. The data set contains 4050 observations and is

684 piecewise constant, with each segment representing a single rock type that has constant
685 physical properties. Thus, change points in this data set represent the boundary between
686 different underground rock structures. These measurements are used to interpret the
687 geophysical structure of the rock surrounding the well, as the ability to detect changes in rock
688 type is important in the prevention of blowouts when drilling for oil (see Fearnhead and Clifford
689 (2003)). The data set originated with Ó Ruanaidh and Fitzgerald (1996), but has since been
690 used as a standard data set in multiple change point analysis.

691 A small number of outliers were manually removed from the data set. The same was done
692 in the analysis performed by Ó Ruanaidh and Fitzgerald (1996), Fearnhead and Clifford (2003),
693 Adams and MacKay (2007), and Whiteley et al. (2011). The data set was then standardized for
694 numerical stability and parameters were set at $k_0 = 0.001$, $d_{min} = 10$, $v_0 = 1$, and $\sigma_0^2 =$ variance of
695 data set excluding outliers. The median of the posterior distribution on the number of change
696 points was used as the detection criteria for the presence of a change point. Figure 3 shows the
697 well-log data set [including outliers] the inferred piecewise constant model, and the posterior
698 distribution of the locations of change points identified by the Sequential Bayesian Change
699 Point algorithm. In total, the algorithm was able to detect approximately 17 change points.

700 The results are highly consistent with that of Ó Ruanaidh and Fitzgerald (1996) [who fixed
701 the number of change points at 13], Fearnhead and Clifford (2003) [16 change points],
702 Fearnhead (2006) [a batch algorithm, 16 change points], and Whiteley et al. (2011) [16 change
703 points]. Small differences in the posterior distribution are likely due, at least in part, to how the
704 outliers were removed, which was not specified in any of these analyses. For example, not
705 removing the long string of outliers around $t = 3950$ would result in the algorithm identifying an
706 additional pair of change points, while removing more of the data points at the beginning of the
707 data set would likely eliminate that change point. Adams and MacKay (2007) showed only a
708 small subsection of the data set in their paper. The change points identified by BOCPD between
709 $t = 1600$ and $t = 2700$ are consistent with the ones found here as well.

710
711 **Figure 3: The Well-Log Data Set**

Figure 3: The well-log data set. (a) The solid green line represents the piecewise constant model inferred by the Sequential Bayesian Change Point algorithm. The posterior probability of a change point at a specific location is indicated by the red spikes at the bottom of the figure. (b) The posterior distribution for the number of change points in the well-log data set.

5.4 Global Surface Temperature Anomalies – A Change in Trend

The National Oceanic and Atmospheric Administration [NOAA] maintains several data sets related to the Earth's historical temperature. Here we consider the global surface temperature anomalies [combined land and ocean] from 1880 to 2013. This 134 year data set describes the departure of the Earth's temperature from its long term average, in this case the 20th century. Positive anomalies indicate that the observed temperatures were warmer than the reference period while negative anomalies represent that the observed temperatures were cooler than

the reference period. See the NOAA website for more details on this data set

(<http://www.ncdc.noaa.gov/monitoring-references/faq/anomalies.php>)

A visual inspection of the global surface temperature anomalies data set [Figure 4] clearly shows that the increase in temperature is not constant, but that most of the warming takes place during two distinct periods, one beginning around 1910 and the other during the 1970s (Karl et al., 2000; Seidel and Lanzante, 2004; Menne, 2006). This suggests three change points due to the fairly flat regime between the two warming periods. To fit the global surface temperature record, it is assumed that each temperature regime can be fit by a single linear trend, $Y = \beta_1 + \beta_2 X$, and $k_{max} = 6$ was chosen as a reasonable maximum for the number of change points. Parameters were set at $k_0 = 0.01$, $v_0 = 1$, $\sigma_0^2 = 0.01$, and $d_{min} = 5$. The median of the posterior distribution on the number of change points was again used as the detection criteria for the presence of a change point.

Figure 4 shows the global surface temperature anomaly data set, the inferred model, and the posterior distribution of the locations of change points identified by the Sequential Bayesian Change Point algorithm. In total, the algorithm was able to detect 3 change points, which matches well with previous analyses of this data set [Table 5]. The third change point is the most difficult to identify as its posterior distribution is tri-modal. Interestingly, the most recent mode [1996] has only become evident with the addition of the last several years' data and represents a stabilization of temperatures as opposed to the increase in temperatures represented by the modes at 1963 and 1976. Because of the additional observation added to the data set each year, earlier analyses would not have detected a change point this late in the record. See Ruggieri (2013) for a more complete discussion of prior analyses on this data set.

Figure 4: Global Surface Temperature Anomalies

Figure 4: Global surface temperature anomalies [combined land and ocean] as compiled by NOAA. The dashed green line represents the piecewise linear model inferred by the Sequential Bayesian Change Point algorithm. Posterior probability of a change point at a specific location is given by the red spikes at the bottom of the figure. Three change points are identified in this data set, the third being tri-modal.

Table 5: Global Surface Temperature Anomalies, A Comparison

Study	Change Points		
Karl et al., 2000	1910	1941	1975
Seidel and Lanzante, 2004	n/a	1945	1977
Menne, 2006	1902	1945	1963
Ruggieri et al., 2009	1906	1945	1963
Sequential Bayesian Change Point	1906	1945	1963/1976/1996

Table 5: Comparing the placement of change points by the Sequential Bayesian Change Point algorithm to previous studies of the annual global surface temperature anomalies time series [1880-2013]. Seidel and Lanzante (2004) do not identify a change in the first decade of the 20th century. In this study, a third change point is selected for only 78.0% of the sampled solutions and is tri-modal, located at either 1963, 1976, or 1996. Note that previous studies analyzed a shorter data set.

6. Discussion and Conclusions

The ideal change point algorithm would have a minimal number of false positives combined with the quickest possible detection rate, giving us two obvious measures of performance: rate of false positives and delay in detection. However, there is an obvious trade-off between these two measures of performance as quick detection is often linked with a high rate of false alarms. Monitoring the posterior distribution of the number of change points appears to be the best

way to detect the existence of a new change point in a data set. Three detection criteria were proposed and studied in this paper, namely the mean, median, and mode of this posterior distribution. Of the three, the mean of the posterior distribution appears to be the least desirable of the detection criteria as it had the highest error rate and slowest detection speed. While the median had a slightly higher error rate than the mode, the two were nearly equivalent in terms of their ability to detect actual change points.

One potential difficulty in monitoring the posterior distribution of the number of change points is that it is not a monotonic function. For example, outliers or even random variation in the data set can make it appear that a change point has occurred, when in fact further data collection would indicate otherwise. Thus, a delicate balance has to be struck between the speed and accuracy of detection if the algorithm is going to prove itself useful.

Section 5.1 described simulations used to study the false alarm rate of the Sequential Bayesian Change Point algorithm. Errors in detecting change points are a product of the random variation in the data sets and often resulted from a small number of consecutive data points that are all significantly above or significantly below the baseline signal. In this situation, the addition of more data eliminates this spurious change point. Of the 20,000 simulations summarized in this section, only 11 were erroneously determined to contain a change point once the full data set was revealed. Of these, the vast majority [9/11] were in the simulation of the constant mean when k_0 was set to its largest value. The other two occurred in the simulation of the constant mean when d_{min} was set to its smallest value. In general, both error rates and discovery rates were consistent with those of existing change point algorithms, and the algorithm was found to be conservative in its change point detection. For a given set of parameter values, a lack of false positives was accompanied by less than perfect detection.

If these simulations are placed in the context of a classification problem [change point vs. no change point], then each choice of parameter settings would represent a different classifier with its own ‘discovery rate’ and ‘false alarm rate’. The Receiver Operator Characteristic [ROC] curve specifies the maximum detection rate for a given false alarm rate. Real-world classifiers will never have 100% detection without any false alarms, so it should not be surprising that the algorithm proposed in this paper has both false alarms and less than perfect detection ability. See Fawcett (2006) for more information on ROC curves.

The efficient algorithm described here can update itself in linear time with each new observation. However, for very long data sets, even a linear update can become computationally prohibitive. One approach to addressing this problem is to expand upon the product partition model that forms the foundation of this algorithm. The product partition model assumes that segments separated by a change point are independent of each other (Barry and Hartigan, 1993). If the locations of the change points are also assumed to be independent of one another, then the complexity of the algorithm can be reduced from linear

to the average segment length. However, by doing this the exact nature of the algorithm is sacrificed. Work is ongoing to study the effects of this assumption.

For the purpose of illustration, the simulated and real data sets presented in Section 5 highlight only the simplest types of change point analysis, namely a change in mean and a change in linear trend. For example, including sinusoidal predictor variables in the regression model would allow a user to search for change points in a periodic data set [e.g. Ruggieri et al. (2009); Ruggieri (2013)]; a multinomial model allows examination of a genomic data set [e.g. Liu and Lawrence (1999)]. In short, the applicability of the Sequential Bayesian Change Point algorithm extends far beyond these two scenarios and is appropriate for any data set in which the likelihood of the data can be calculated for a given predictive model.

Finally, as written the Sequential Bayesian Change Point algorithm assumes an independent error structure. This is a common assumption in change point analysis (e.g. Barry and Hartigan 1993; Stephens 1994; Western and Kleykamp 2004; Adams and MacKay 2007; Ruggieri et al. 2009; Ruggieri 2013; Ruggieri and Lawrence 2014; among others), especially when the goal is to detect a shift in the mean signal (Lucas and Saccucci, 1990; Carlin et al. 1992; Siegmund and Venkatraman, 1995; Lavielle and Lebarbier, 2001; Fearnhead and Clifford, 2003; Hawkins et al., 2003; Whiteley et al., 2011). Future work will consider models with a correlated error structure.

While many of the existing sequential change point algorithms only approximate the posterior distribution of the location of the change points, the efficient and exact Sequential Bayesian Change Point algorithm draws samples directly from the posterior distribution without imposing any assumptions on the distance between adjacent change points. The algorithm has the ability to quickly update itself with each new observation and can provide uncertainty bounds both on the number and locations of the change points in a data set. Simulation studies show a low rate of falsely detected change points while at the same time giving confidence that existing change points will quickly be detected. The well- log and global surface temperature anomalies data sets give a small indication as to the many possible applications of this algorithm.

Availability: Matlab code for the Sequential Bayesian Change Point algorithm and the associated data sets described in this manuscript can be obtained by contacting the first author, Eric Ruggieri, at eruggier@holycross.edu

843 **Acknowledgements:** The authors would like to thank the associate editor and reviewers for
844 their helpful feedback on this manuscript. This project was supported by a grant from the
845 National Science Foundation, DMS-1407670 (E. Ruggieri, PI).

846

847

References:

- Adams, R.P. and MacKay, D.J.C. (2007) Bayesian online changepoint detection. (Available at: <http://arxiv.org/pdf/0710.3742.pdf>.)
- Auger, I.E. and Lawrence, C.E. (1989) Algorithms for the Optimal Identification of Segment Neighborhoods. *Bulletin of Mathematical Biology*, **51**, 39-54.
- Bai, J. and Perron, P. (2003) Computation and Analysis of Multiple Structural Change Models. *Journal of Applied Econometrics*, **18**, 1-22. doi:10.1002/jae.569
- Barry, D. and Hartigan, J.A. (1993) A Bayesian Analysis for Change Point Problems. *Journal of the American Statistical Association*, **88**(421), 309-319.
- Carlin, B.P., Gelfand, A.E., and Smith, A.F.M. (1992) Hierarchical Bayesian analysis of changepoint problems. *Applied Statistics*, **41**, 389-405.
- Chib, S. (1998) Estimation and comparison of multiple change-point models. *Journal of Econometrics*, **86**, 221-241.
- Chopin, N. (2007) Dynamic detection of change points in line time series. *Annals of the Institute of Statistical Mathematics*, **59**, 349-366.
- Fawcett, T. (2006) An introduction to ROC analysis. *Pattern Recognition Letters*, **27**, 861-874. doi: 10.1016/j.patrec.2005.10.010
- Fearnhead, P. and Clifford, P. (2003) On-line inference for hidden Markov models via particle filters. *Journal of the Royal Statistical Society, Series B*, **65**(4), 887-899.
- Fearnhead, P. (2006) Exact and Efficient Bayesian Inference for Multiple Changepoint problems. *Statistical Computing*, **16**, 203-213. doi 10.1007/s11222-006-8450-8.
- Fearnhead, P. and Liu, Z. (2007) On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society, Series B*, **69**(4), 589-605.
- Fryzlewicz, P. (2013) Wild binary segmentation for multiple change-point detection. (Available at: <http://stats.lse.ac.uk/fryzlewicz/wbs/wbs.pdf>)
- Gallagher, C., Lund, R., and Robbins, M. (2012) Changepoint detection in daily precipitation data. *Environmetrics*, **23**(5), 407-419. doi: 10.1002/env.2146
- Green, P.J. (1995) Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, **82**(4), 711-732. doi:10.1093/biomet/82.4.711
- Grigg, O.A. and Spiegelhalter, D.J. (2008) An Empirical Approximation to the Null Unbounded Steady-State Distribution of the Cumulative Sum Statistic. *Technometrics*, **50**(4), 501-511.
- Hawkins, D.M., Qiu, P., and Kang, C.W. (2003) The Changepoint Model for Statistical Process Control. *Journal of Quality Technology*, **35**(4), 355-366.
- Karl, T.R., Knight, R.W., and Baker B. (2000) The record breaking global temperatures of 1997 and 1998: Evidence for an increase in the rate of global warming? *Geophysical Research Letters* **27**(5), 719-722.
- Koop, G. and Potter, S.M. (2007) Estimation and forecasting in models with multiple breaks. *Review of Economic Studies*, **74**, 763-789.
- Koop, G. and Potter, S.M. (2009) Prior elicitation in multiple change-point models. *International Economic Review*, **50**, 751-772.
- Lavielle, M. and Lebarbier, E. (2001) An application of MCMC methods for the multiple change-points problem. *Signal Processing*, **81**(1), 39-53. doi:10.1016/S0165-1684(00)00189-4

- Liu, J.S. and Lawrence, C.E. (1999) Bayesian Inference on Biopolymer Models. *Bioinformatics*, **15**(1), 38-52.
- Lucas, J.M. and Saccucci, M.S. (1990) Exponentially Weighted Moving Average Control Schemes: Properties and Enhancements. *Technometrics*, **32**(1), 1-12.
- Menne, M.J. (2006) Abrupt Global Temperature Change and the Instrumental Record. In: *18th Conference on Climate Variability and Change*, AMS, Atlanta, GA.
- O’Ruanaidh, J. and Fitzgerald, W.J. (1996) *Numerical Bayesian methods applied to signal processing*. New York: Springer-Verlag.
- Page, E.S. (1954) Continuous Inspection Schemes. *Biometrika*, **41**(1/2), 100-115. doi: 10.2307/2333009
- Ross, G.J. (2013) Parametric and Nonparametric Sequential Change Detection in R: The cpm package. (Available at: <http://www.gordonjross.co.uk/cpm.pdf>)
- Ruggieri, E. (2013) A Bayesian Approach to Detecting Change Points in Climatic Records. *International Journal of Climatology*, **33**, 520-528. doi: 10.1002/joc.3447
- Ruggieri, E. and Lawrence, C.E. (2014) The Bayesian Change Point and Variable Selection Algorithm: Application to the $\delta^{18}\text{O}$ Record of the Plio-Pleistocene. *Journal of Computational and Graphical Statistics*, **23**(1), 87-110. doi: 10.1080/10618600.2012.707852
- Ruggieri, E., Herbert, T., Lawrence, K.T., and Lawrence, C.E. (2009) Change point method for detecting regime shifts in paleoclimatic time series: Application to d18O time series of the Plio-Pleistocene. *Paleoceanography*, **24**, PA1204. doi:10.1029/2007PA001568.
- Saatci, Y., Turner, R., and Rasmussen, C.E. (2010) Gaussian Process Change Point Models. In Proceedings of the 27th International Conference on Machine Learning, pp. 927-934.
- Scott, A.J., and Knott, M. (1974) A Cluster Analysis Method for Grouping Means in the Analysis of Variance. *Biometrics*, **30**, 507-512.
- 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* **109**: D14108. doi:10.1029/2003JD004414
- Siegmund, D. and Venkatraman, E.S. (1995) Using the Generalized Likelihood Ratio Statistic for Sequential Detection of a Change-Point. *Annals of Statistics*, **23**(1), 255-271.
- Stephens, D.A. (1994) Bayesian Retrospective Multiple-changepoint Identification. *Applied Statistics*, **43**(1), 159-178.
- Western, B. and Kleykamp, M. (2004) A Bayesian Change Point Model for Historical Time Series Analysis. *Political Analysis*, **12**(4), 354-374. doi:10.1093/pan/12.4.mph023
- Whiteley, N., Andrieu, C., and Doucet, A. (2011) Bayesian Computational Methods for Inference in Multiple Change-point Models. (Available at: www.maths.bris.ac.uk/~manpw/change_points_2011.pdf)
- Wilson, R.C., Nassar, M.R., and Gold, J.I. (2010) Bayesian On-line Learning of the Hazard Rate in Change-Point Problems. *Neural Computation*, **22**(9), 2452-2476. doi:10.1162/NECO_a_00007.
- Yildirim, S., Singh, S.S., and Doucet, A. (2013) An Online Expectation-Maximization Algorithm for Changepoint Models. *Journal of Computational and Graphical Statistics*, **22**(4), 906-926.
- Zeileis, A., Leisch, F., Hornik, K., Kleiber, C. (2002) strucchange: An R Package for Testing for Structural Change in Linear Regression Models, *J. Statistical Software*, **7**(2), 1-38.
- Zeileis, A. (2004) Alternative boundaries for CUSUM tests. *Statistical Papers*, **45**(1), 123-131.