A Brief Introduction to the Matlab package StepSignalMargiLike

Chu-Lan Kao, Chao Du

Jan 2015

This article contains a short introduction to the Matlab package for estimating change-points in stepwise signals. In the first section, we will outline the theoretical framework of the marginal Likelihood estimator as formulated in Du, Kao and Kou (2015). The second section contains a brief introduction of the Matlab-implementation of this method. In the third section, we will demonstrate the package with a walk-through example.

1 Theoretical Framework of the Marginal Likelihood Estimator

Problems We define the stepwise signal as a series of observations $\boldsymbol{x} = \{x_1, x_2, \cdots, x_n\}$, measured at successive times (or spatial locations) $\boldsymbol{t} = \{t_1, t_2, \ldots, t_n\}$, $t_1 < t_2 < \cdots < t_n$. The probability distribution of the observations are determined by parameter $\theta \in \Theta$ through a family of densities $f(x|\theta)$. The signal appears to be stepwise due to the fact that θ is a step function of time whose transitions are determined by $m-1$ change-points $\boldsymbol{\tau}_{1:(m-1)} = \{\tau_1, \cdots, \tau_{m-1}\}$:

$$\theta(t) = \theta_j \quad \text{if } t \in (\tau_{j-1}, \tau_j], \tag{1.1}$$

where $\tau_j \in [t_1, t_n]$ for all $j \in \{1, \cdots, m-1\}$. The $m-1$ change-points split the signal into m segments. We refer $\boldsymbol{\theta}_{1:m} = \{\theta_j\}_{j=1}^m$ as the segment parameters and further assume that the adjacent θ_j's are distinguishable.

Given the change-points $\boldsymbol{\tau}_{1:(m-1)}$ and the associated segment parameters $\boldsymbol{\theta}_{1:m}$, the observations are assumed to be independently distributed:

$$P(\boldsymbol{x}|\boldsymbol{\tau}_{1:(m-1)}, \boldsymbol{\theta}_{1:m}) = \prod_{j=1}^{m} \prod_{t_i \in (\tau_{j-1}, \tau_j]} f(x_i|\theta_j). \tag{1.2}$$

The observations up to time τ_1 have density $f(\cdot|\theta_1)$; the distribution of observations after time τ_1 but up to τ_2 has parameter θ_2; ...; the observations after time τ_{m-1} are characterized by parameter

θ_m. Please note that we set $\tau_0 \equiv 0$ and $\tau_m \equiv t_n$ for notational ease. We further assume that the change-points can only take discrete values from the set $\{t_i\}_{i=1}^n$, that is, $\tau_j \in \{t_1, \cdots, t_{n-1}\}$ for $j \in \{1, \cdots, m-1\}$.

The main goal of our estimator is to determine the number m and positions $\{\tau_j\}_1^{m-1}$ of the change-points in the stepwise signal $\{x_i\}_{i=1}^n$. With the change-points determined, the segment parameters $\boldsymbol{\theta}_{1:m}$ can often be estimated with relative ease.

Marginal Likelihood Estimator Our method utilizing the marginal likelihood in which $\boldsymbol{\theta}_{1:m}$ are integrated out (Chib, 1998; Yang and Kuo, 2001; Fearnhead, 2005). Given the set of change-points $\boldsymbol{\tau}_{1:(m-1)}$, we assume that the segment parameters $\boldsymbol{\theta}_{1:m}$ are independently and identically drawn from a prior distribution $\pi(\cdot|\alpha)$ with pre-determined hyperparameter(s) α. Then we can express the conditional marginal likelihood given the set of change-points, as

$$\begin{aligned} P(\boldsymbol{x}|\boldsymbol{\tau}_{1:(m-1)}) &= \prod_{j=1}^{m} \int_{\theta_j} \prod_{t_i \in (\tau_{j-1}, \tau_j]} f(x_i|\theta_j)\pi(\theta_j|\alpha)d\theta_j \\ &:= \prod_{j=1}^{m} D(\boldsymbol{x}_{(\tau_{j-1},\tau_j]}|\alpha), \end{aligned} \tag{1.3}$$

where, in general, $D(\boldsymbol{x}_{(a,b]}|\alpha)$ denotes the probability of obtaining the observations during the period $(a, b]$ with no change-point in between. A closed form of $D(\boldsymbol{x}_{(a,b]}|\alpha)$ can be obtained if conjugate priors are used. Otherwise, $D(\boldsymbol{x}_{(a,b]}|\alpha)$ can be calculated by numerical methods.

We estimate the set of change-points as the maximizer of $P(\boldsymbol{x}|\boldsymbol{\tau}_{1:(m-1)})$ over all the feasible combinations of change-points, restricted by an upper bound $M \leq n$ on the number of segments. In the extreme case of $M = n$, every observation t_i can be a segment itself.

Computation We handle the computational burden with a dynamic programming algorithm (Bellman and Roth, 1969; Bement and Waterman, 1977; Auger and Lawrence, 1989). Suppose that $M \leq n$ is an upper bound for the number of segments, we suggest the following algorithm.

Define

$$H(x_1, \cdots, x_i|m) = \max_{\boldsymbol{\tau_{1:(m-1)}} \subseteq \{t_1, \cdots, t_{i-1}\}} P(x_1, \cdots, x_i|\boldsymbol{\tau}_{1:(m-1)}).$$

Step 1 For $1 \leq i \leq n$: $H(x_1, \cdots, x_i|1) = D(x_1, \cdots, x_i|\alpha)$

$\vdots$

Step m For $m \leq i \leq n$: $H(x_1, \cdots, x_i|m) = \max_{m-1 \leq j \leq i-1} H(x_1, \cdots, x_j|m-1)D(x_{j+1}, \cdots, x_i|\alpha)$

$\vdots$

Step M For $M \leq i \leq n$: $H(x_1, \cdots, x_i|M) = \max_{M-1 \leq j \leq i-1} H(x_1, \cdots, x_j|M-1)D(x_{j+1}, \cdots, x_i|\alpha)$

Using the above recursive functions, we can obtain the following estimators with computational cost $O(n^2M)$ and storage $O(nM)$:

- the maximum marginal likelihood estimator $\hat{\boldsymbol{\tau}}_{1:(m-1)}$ with exactly m segments ($m \leq M$)

$$\hat{\boldsymbol{\tau}}_{1:(m-1)} = \underset{\boldsymbol{\tau_{1:(m-1)}}}{\arg\max}\, P(\boldsymbol{x}|\boldsymbol{\tau}_{1:(m-1)}), \tag{1.4}$$

- the maximum marginal likelihood estimator $\hat{\boldsymbol{\tau}}_M$ with up to M segments

$$\hat{\boldsymbol{\tau}}_M = \underset{\boldsymbol{\tau_{1:(m-1)}},\, 1\leq m\leq M}{\arg\max} P(\boldsymbol{x}|\boldsymbol{\tau}_{1:(m-1)}). \tag{1.5}$$

Jackson *et al.* (2005) developed another more efficient but less flexible algorithm in which the unrestrictive (with up to n segments) maximum marginal likelihood estimator $\hat{\boldsymbol{\tau}}$ can be computed with computational cost $O(n^2)$ and storage $O(n)$. This algorithm is based on the following recursive functions.

Define

$$G(x_1, \cdots, x_i) = \max_{\boldsymbol{\tau}\subseteq\{t_1,\cdots,t_{i-1}\}} P(x_1, \cdots, x_i|\boldsymbol{\tau}).$$

$$\begin{aligned}
&\text{Step } 1 \quad G(x_1) = D(x_1|\alpha)\\
&\qquad\vdots\\
&\text{Step } i \quad G(x_1, \cdots, x_i) = \max_{1\leq j\leq i-1} G(x_1, \cdots, x_j) D(x_{j+1}, \cdots, x_i|\alpha)\\
&\qquad\vdots\\
&\text{Step } n \quad G(x_1, \cdots, x_n) = \max_{1\leq j\leq n-1} G(x_1, \cdots, x_j) D(x_{j+1}, \cdots, x_n|\alpha)
\end{aligned}$$

Generally speaking, for a large value of M, we expect that $\hat{\boldsymbol{\tau}}_M$ from the first algorithm is identical to $\hat{\boldsymbol{\tau}}$ from the second algorithm. Thus, the second algorithm is the algorithm of choice for large M. On the other hand, if there is a strong restriction on the number of segments M or one needs to compare models with different number of change-points, the first algorithm should be used.

Choice of hyperparameters α The performance marginal likelihood estimator depends on the choice of prior distribution $\pi(\cdot|\alpha)$, represented by the particular choice of hyperparameters α. So long as the prior used is relatively consistent with the data, our estimator is quite robust and there is some flexibility in choosing a good prior. Still, a strong prior tends to over-fit the data, yielding too many change-points, while a weak prior tends to under-fit the data, missing the real change-points.

A reasonable choice of hyperparameters α can be chosen based on expert knowledge (Chib, 1998; Fearnhead, 2005, 2006). However, such choice is often ambiguous and may not always be practical. In contrast, our formulation uses an empirical Bayes approach to set the hyperparameters, as explained in the following guidelines:

1) Derive the expectation and variance of a single observation as functions of α, the hyperparameter: $E(x|\alpha)$ and $Var(x|\alpha)$.

2) Set the value of α so that $E(x|\alpha) = \hat{\mu}$, the sample average, and that $Var(x|\alpha)$ is a large multiple of $\hat{\sigma}^2$, the sample variance.

In particular, for normal and Poisson data, we recommend the following priors:

Normal Data: For normal data $x_i|(\mu_j, \sigma_j^2) \sim N(\mu_j, \sigma_j^2)$, we use the conjugate prior: $\sigma_j^2|m \sim$ scaled $Inv\text{-}\chi^2(\nu_0, \sigma_0^2)$, $\mu_j|(\sigma_j^2, m) \sim N(\mu_0, \sigma_j^2/\kappa_0)$.

(a) When the variability of the segment means μ_j is low or moderate (for example, if it is known that the range of μ_j is moderate), we recommend two conjugate priors with hyperparameters:

$$\text{Norm-A} \quad : \qquad \mu_0 = \bar{x},\ \sigma_0^2 = \hat{\sigma}^2,\ \kappa_0 = \frac{1}{2},\ \nu_0 = 3; \tag{1.6}$$

$$\text{Norm-B} \quad : \qquad \mu_0 = \bar{x},\ \sigma_0^2 = 2.5\hat{\sigma}^2,\ \kappa_0 = \frac{1}{2},\ \nu_0 = 3 \tag{1.7}$$

The prior Norm-A is good at locating short segments, but may over fit the data when outliers are common. The Norm-B prior is a more conservative choice. In practice, it is recommended to apply the Norm-A prior first. If the resulting step function appears to be over-fitting, then Norm-B prior can be applied to re-analyze the data. The final decision should be made based on the scientific understanding of the applied problem.

(b) When the variability of the segment means μ_j is large (for example, if the range of μ_j is large), we recommend the following conjugate prior:

$$\text{Norm-C:} \quad \mu_0 = \bar{x},\ \sigma_0^2 = \frac{3}{5}\hat{\tau}^2,\ \kappa_0 = \frac{5}{12}\frac{\hat{\tau}^2}{\hat{\sigma}^2},\ \nu_0 = 3, \tag{1.8}$$

where $\hat{\tau}^2$ is the average within-segment sample variance based on the change-points estimator obtained through prior Norm-A (i.e., $\hat{\tau}^2$ is the average of $\hat{\sigma}_j^2$, where $\hat{\sigma}_j^2$ is the sample variance within the jth segment identified by first applying the prior Norm-A).

Remark 1. Under the conjugate prior, which has density

$$\pi(\mu_j, \sigma_j^2|\mu_0, \kappa_0, \nu_0, \sigma_0^2) = \frac{(\sigma_0^2\nu_0/2)^{\nu_0/2}(\sigma_j^2)^{-(\nu_0/2+1)}}{\Gamma(\nu_0/2)(2\pi\sigma_j^2/\kappa_0)^{1/2}} \exp(-\frac{1}{2\sigma_j^2}\left(\kappa_0(\mu_j-\mu_0)^2 + \nu_0\sigma_0^2\right)),$$

the marginal likelihood has a closed form in that

$$\begin{aligned} D(\boldsymbol{x}_{(\tau_{j-1},\tau_j]}|\mu_0, \kappa_0, \nu_0, \sigma_0^2) \quad &\propto \quad (\sigma_0^2\nu_0)^{\nu_0/2}\frac{\Gamma(\frac{\nu_0+n_j}{2})}{\Gamma(\nu_0/2)}\sqrt{\frac{\kappa_0}{\kappa_0+n_j}} \\ &\quad \times \left(\nu_0\sigma_0^2 + \sum x_i^2 - \frac{1}{n_j}(\sum x_i)^2 + \frac{\kappa_0(\sum x_i - n_j\mu_0)^2}{n_j(\kappa_0+n_j)}\right)^{-(\nu_0+n_j)/2}, \end{aligned}$$

where all the sums are over the set $\{i : t_i \in (\tau_{j-1}, \tau_j]\}$, and n_j represents the number of observations within the interval $(\tau_{j-1}, \tau_j]$.

Poisson Data: When the data consist of counts, such as fluorescence or photon counts from biological or chemical experiments, modeling them as Poisson, $x_i|\lambda_j \sim Poisson(\lambda_j)$, is more appropriate. We use the conjugate prior $\lambda_j|\alpha, \beta \sim \Gamma(\alpha, \beta)$. We recommend the following choice of the hyperparameters

$$\text{Pois-P:} \quad \alpha = \bar{x}\beta, \ \beta = \frac{1}{2\hat{\sigma}^2}. \tag{1.9}$$

With this prior we have $E(x|\alpha, \beta) = \bar{x}$ and $Var(x|\alpha, \beta) = \bar{x}(1 + 2\hat{\sigma}^2)$.

Remark 2. Under the conjugate prior $\lambda_j|\alpha, \beta \sim \Gamma(\alpha, \beta)$, which has density $\pi(\lambda_j|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\lambda_j^{\alpha-1}e^{-\beta\lambda_j}$, the marginal likelihood has a closed form in that

$$D(\boldsymbol{x}_{(\tau_{j-1},\tau_j]}|\alpha, \beta) \propto \frac{\Gamma(\sum x_i + \alpha)}{\Gamma(\alpha)}\beta^\alpha/(n_j + \beta)^{\alpha+\sum x_i},$$

where the sums are over $\{i : t_i \in (\tau_{j-1}, \tau_j]\}$, and n_j is the number of observations within $(\tau_{j-1}, \tau_j]$.

2 Matlab Implementation

2.1 Installation Instruction

To install this Matlab package, please run the installation script file `InstallScript.m` which contains the following commands:

```
mex ChangePointAnalyzeNorm.cc;
mex ChangePointAnalyzeNormUnRes.cc;
mex ChangePointAnalyzePoiss.cc;
mex ChangePointAnalyzePoissUnRes.cc;
```

These commands are used to compile the C++ codes that constitute the core of the dynamic programing algorithm. This procedure only need to be done once. You may also need to use the command

```
mex -setup
```

to make sure that the default complier associated with `mex` supports C++ language.

In addition to the standard C++ library, our implementation also utilized standalone functions [1] by Dr. John D. Cook for evaluating the logarithm of Gamma function.

[1] `Gamma.h, Gamma.cpp`, in public domain, author: Dr. John D. Cook, available at `www.johndcook.com/blog/stand_alone_code/`.

2.2 A Guide to the Implemented Functions

In our current implementation, the function for estimating change-points in a given stepwise signal using the aforementioned method is:

```
[index_chPTs, log_H, max_j] = est_changepoints(data_x, model, prior, max_segs, @logMD)
```

This function requires the stepwise signals $\boldsymbol{x}$, the name of distributional family of the data (Distribution families including univariate Normal and Poisson are implemented in the current version. For other cases, users need to provide specific log-marginal likelihood functions $\log(D(\boldsymbol{x}_{(a,b]}|\alpha))$), the hyper parameters α as well as an upper bound M (optional) on the number of change-points. It returns the maximum marginal likelihood estimator $\hat{\boldsymbol{\tau}}_M$ as in equation 1.5, a matrix that contains the log value of the H matrix used in the algorithm and an index matrix that records the j that maximizes the marginal likelihood in each step. Note that in our current implementation, the time series $\boldsymbol{t}$ only functions as label, so the convention that $t_i = i$ for $i \in \{1, 2, \cdots, n\}$ is used. The arguments used in this function are explained in detail below:

`data_x`	Observed data $\boldsymbol{x}$ in vector or matrix form. When the data is in matrix form, each column should represent a single observation.
`model`	The specified distributional assumption. Currently we have implemented two arguments: 'normal' (data follows one dimensional Normal distribution with unknown mean and variance) and 'poisson' (data follows Poisson distribution with unknown intensity). A third argument 'user' is also accepted, given that the prior, upper bound on the number of change-points and the log marginal likelihood function are specified in the arguments `prior`, `max_segs` and `@logMD`.
`prior`	The hyperparameter(s)α, which are used for calculating log marginal likelihood.
`max.segs`	Optional argument. The upper bound M on the number of change-points, which must be a positive integer greater then 1. If missing, the function would process using the algorithm by Jackson et al.(2005).
`@logMD`	Optional argument. Log marginal likelihood function $\log(D(\boldsymbol{x}_{(a,b]}|\alpha))$, which takes two arguments, the observed signal and the hyperparameters.

The outputs are:

`index_chPTs` A vector that represents the ordered set of estimated change-points. Each element in the vector represents the index of the end point of a segment. If the result is no change-points, an empty vector is returned.

`log_H` The log value of the H matrix used in the dynamical computation algorithm, where `log_H(m,i)` $= \log H(x_1, x_2, ..., x_i|m)$. In case that the optional argument `max.segs` is unspecified, this argument instead returns the log value of the G vector used in the algorithm by Jackson et al.(2005), where `log_H(i)` $= \log G(x_1, x_2, ..., x_i)$.

`max_j` An index matrix (vector if optional argument `max_segs` is unspecified) which records the j that maximizes the marginal likelihood in each step.

The estimation results can be visualized using the following supplement function:

```
plot_changePTs(data_x, data_t, index_chPTs, est_means)
```

`data_t` The one-dimensional array $\boldsymbol{t}$, serves as label.

`index_chPTs` The set of the index of change-points in an ascending array, which is the result of `est_changepoints`.

`est_means` The array of estimated segment means, whose length must be one plus the length of `index_chPTs`.

Functions for estimating the posterior segmental means in Normal and Poisson distributed signals.

```
est_mean_norm(data_x, index_chPTs, prior)
est_mean_pois(data_x, index_chPTs, prior)
```

Functions for calculating the hyperparameters α with the empirical Bayesian method outlined in the pervious section. These functions can be used to calculated the Norm-A, Norm-B, Norm-C and Pois-P hyperparameters, respectively.

```
prior_norm_A(data_x)
prior_norm_B(data_x)
prior_norm_C(data_x)
prior_pois(data_x)
```

3 A Walk-through Example: Array CGH data

Background Locating the aberration regions in a genomic DNA sequence is important for understanding the pathogenesis of cancer and many other diseases. Array Comparative Genomic Hybridization (CGH) is a technique developed for such a purpose. A typical array CGH data sequence

consists of the log-ratios of normalized intensities from disease versus control samples, indexed by the genome numbers. The regions of concentrated high or low log-ratios departing from 0 indicate amplification or loss of chromosomal segments. Thus, the central question in analyzing array CGH data is to detect those abnormal regions.

Example Here we will use our marginal likelihood method to study one sample of array CGH data analyzed in Lai *et al.* (2005). The data are normalized based on the raw glimoa data from Bredel *et al.* (2005), which concerns primary glioblastoma multiforme (GBM), a malignant type of brain tumor. In particular, this sample represents chromosome 13 in GBM29. The normalized data in xls format is available at `http://compbio.med.harvard.edu/Supplements/Bioinformatics05b.html`.

The following codes contained in the script `Example.m` can be used to estimate the change-points in this sample. Both Norm-A and Norm-B priors are used to analyze the data.

```
load('Chrom_13_GBM29.mat')
max_segs = 10;

#### This signal only contains a few segments, so the maximum number of segments is set to 10.

prior_A = prior_norm_A(data_x);
[index_chPT_A,log_H_A,max_j_A] = est_changepoints(data_x, 'normal', prior_A, max_segs);

#### Estimate the change-points using Norm-A prior.

prior_B= prior_norm_B(data_x);
[index_chPT_B,log_H_B,max_j_B] = est_changepoints(data_x, 'normal', prior_B, max_segs);

#### Estimate the change-points using Norm-B prior.

est_means_A = est_mean_norm(data_x, index_chPT_A, prior_A);
est_means_B = est_mean_norm(data_x, index_chPT_B, prior_B);

subplot(2,1,1);
```

```
plot_changePTs(data_x, data_t, index_chPT_A, est_means_A);
subplot(2,1,2);
plot_changePTs(data_x, data_t, index_chPT_B, est_means_B);

#### Estimate and draw the step functions of the means.
```

The estimated step functions along with the CGH data are shown in Figure 1 for sample GBM29.

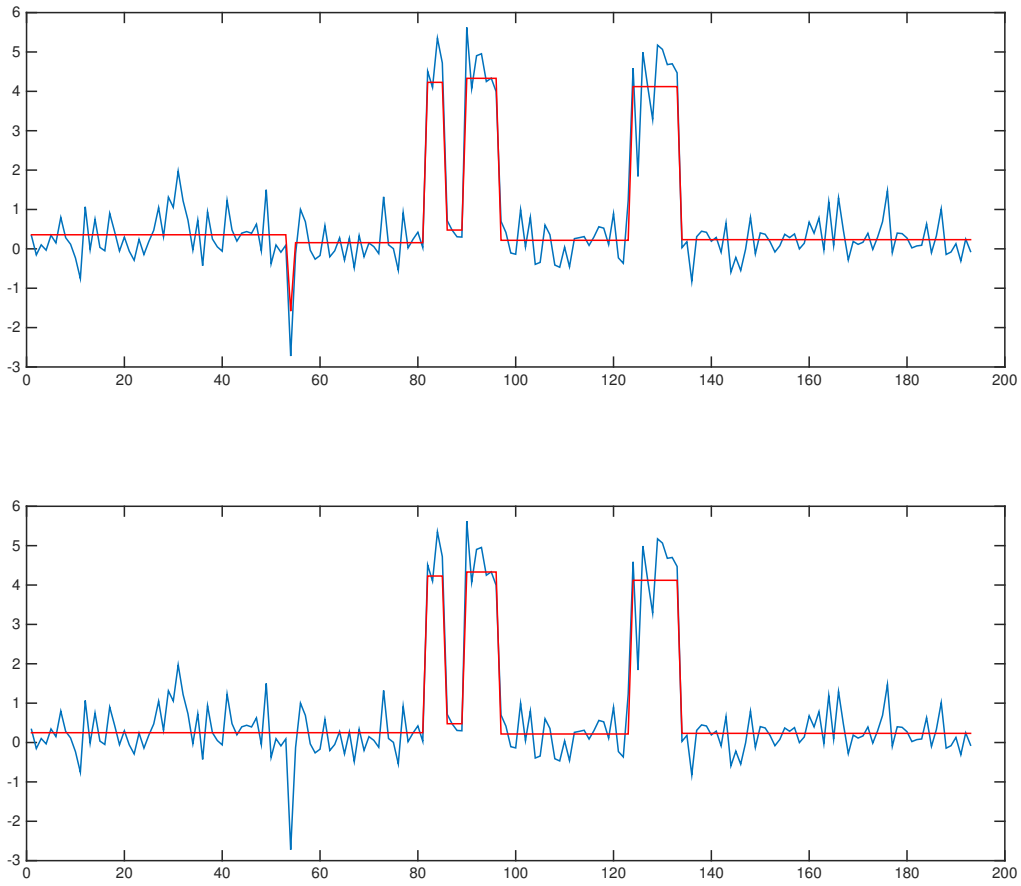

Figure 1: Array CGH data of GBM29, with the estimated step functions based on Norm-A and Norm-B priors.

In sample GBM29, three regions of high amplitude amplifications exist and have been well studied. Based on Figure 1, both estimators successfully identify these three high amplifications even though the first two regions are separated only by four probes. The estimator based on the Norm-A prior identified a single-probe outlier, which could be a real local aberration or an experimental error.

References

[1] Auger, I. E., and Lawrence, C. E. (1989), Algorithms for the optimal identification of segment neighborhoods. *B. Math. Biol.*, 51, 39-54.

[2] Bellman, R., and Roth, R. (1969). Curve fitting by segmented straight lines. *J. Amer. Statist. Assoc.*, 64, 1079-1084.

[3] Bement, T. R., and Waterman, M. S. (1977). Locating maximum variance segments in sequential data. *Math. Geol.*, 9, 55-61.

[4] Bredel, M., Bredel, C., Juric, D., Harsh, G. R., Vogel, H., Recht, L. D., and Sikic, B. I. (2005). High-resolution genome-wide mapping of genetic alterations in human glial brain tumors. *Cancer Res.*, 65, 4088-4096.

[5] Chib, S. (1998). Estimation and comparison of multiple change-point models. *Journal of Econometrics*, 86, 221-241.

[6] Du, C., Kao, C-L. M., and Kou, S.C. (2015). Stepwise Signal Extraction via Marginal Likelihood. *J. Amer. Statist. Assoc.*, in press.

[7] Fearnhead, P. (2005). Exact Bayesian curve fitting and signal segmentation. *Signal Processing, IEEE Transactions*, 53, 6, 2160–2166.

[8] Fearnhead, P. (2006). Exact and efficient bayesian inference for multiple changepoint problems. *Statistics and Computing*, 16, 203-213.

[9] Jackson, B., Scargle, J. D., Barnes, D., Arabhi, S., Alt, A., Gioumousis, P., Gwin, E., Sangtrakulcharoen, P., Tan, L., and Tsai, T. T. (2005). An algorithm for optimal partitioning of data on an interval. *Signal Processing Letters, IEEE*, 12, 2, 105–108.

[10] Lai, W. R., Johnson, M. D., Kucherlapati, R., and Park, P. J. (2005). Comparative analysis of algorithms for identifying amplifications and deletions in array CGH data. *Bioinformatics*, 21, 3763-3770.

[11] Yang, T. Y., and Kuo, L. (2001). Bayesian binary segmentation procedure for a poisson process with multiple changepoints. *Jour. Comput. Graph. Statist.*, 10, 772-785.