A Stochastic Segmentation Model for Categorical and Continuous Features of various biological sequential data

A Dissertation Presented

by

Yifan Mo

to

The Graduate School in Partial Fulfillment of the Requirements for the

Degree of

Doctor of Philosophy

in

Applied Mathematics and Statistics

Stony Brook University

November 2012

Stony Brook University

The Graduate School

Yifan Mo

We, the dissertation committee for the above candidate for the
Doctor of Philosophy degree, hereby recommend
acceptance of this dissertation.

Haipeng Xing - Dissertation Advisor
Associate Professor, Applied Mathematics and Statistics

Michael Q. Zhang - Chairperson of Defense
Adjunct Professor, Applied Mathematics and Statistics, Stony Brook University and Cold Spring Harbor Laboratory
Cecil H. and Ida Green Distinguished Chair Professor
Department of Molecular and Cell Biology, University of Taxes at Dallas

Song Wu
Deputy Chair, Assistant Professor, Applied Mathematics and Statistics

Yixin Fang
Assistant Professor, Division of Biostatistics, School of Medicine, New York University

This dissertation is accepted by the Graduate School.

Charles Taber
Interim Dean of the Graduate School

Abstract of the Dissertation

A Stochastic Segmentation Model for Categorical and Continuous Features of various biological sequential data

by

Yifan Mo

Doctor of Philosophy

in

Applied Mathematics and Statistics

Stony Brook University

2012

Nowadays, Hidden Markov Model (HMM) has been widely used in analysis of various biological data for both smoothing and clustering. However, characterizing each hidden state by a single distribution, the classical HMM might have some limitations on the data whose hidden state is composed by a mixture of distributions (Heng Lian et al., 2006). To address this issue, we proposed a new stochastic segmentation model and an associated estimation procedure that has attractive analytical and computational properties. We combined the forward and backward filter together based on Bayes theorem to calculate the posterior mean and variance. Besides, we developed an expectation-maximization (EM) algorithm to estimate the hyper-parameters. Furthermore, we utilized a bounded complexity mixture (BCMIX) approximation whose computational complexity is linear in sequence length. Another important feature of this segmentation model is that it yields explicit formulas for posterior means and probability of categorical states, which can be used to make inference on both categorical and continuous aspects of the data. Other quantities relating to the posterior distribution that are useful for making confidence assessments of any given segmentation can also be estimated by using our method. We perform intensive simulation studies (1) to compare the Bayes and BCMIX estimates (2) to evaluate the BCMIX estimates in terms of sum square error, Kullback-Leibler divergence and the identification ratio of true segments. We also applied our model on two biological data sets: (1) reduced representation bisulfite sequencing (RRBS) data (A.Molaro et al., 2011) (2) ENCODE Nimblegen tilled arrays (Sabo et al., 2006). Our model shows good performance on segmentation of these two sequential data. In RRBS data it can further help identify differential methylation region (DMR) while in microarray data it can discover the DNAseI Hypersensitive Sites (DHSs).

To my parents, my wife Le with all my love

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my advisor, Prof. Haipeng Xing, for the strict statistical model suggested, for interesting discussions and for his guidance and continuous support.

I would also like to thank Prof. Michael Q. Zhang for his instructions on the biology background and the bioinformatics. It is my great honor to be his Ph.D. student.

I would also thank Prof. Song Wu and Prof. Yixin Fang for joining my dissertation committee.

I would like to thank all my group mates and friends, especially Will Liao, Ning Sun, Qing Dong and Jie Wu, for all their support and help.

Chapter 1

Introduction

Biological sciences have been rapidly made process over the past two decades by the development of technologies capable of performing large-scale measurements. In particular, high throughput technology has been widely used in many areas including genomics, transcriptomics, proteomics and etc. Such high throughput, high resolution techniques have generated tremendous sequential data thus, it is revolutionizing the scale and potential applications of genomic studies, and creating an enormous and urgent need for mathematical and computational tools to meet the large scale data analysis challenges such as identifying sequence variations (both single-nucleotide and segmental) and exploring their disease associations, reconstruction of RNA transcript populations, locating the sites of protein-DNA interactions, elucidating population histories and so on. And obviously, the segmentation problem is the priority and fundamental issue for most of these applications.

In this chapter, we will review two kind of typical biological sequential data: arrayCGH data for copy number variation (CNV) detection and Chromatin Immunoprecipitation sequencing (ChIP-seq) data for identifying the protein-DNA interaction locations. For each of them, we shall retrospect the most popular and the latest computation models to show how they solved the difficulties of the segmentation problem within the data. Moreover, we will introduce the interesting and unsolved questions as our motivation. In the last section we will give the outline of this dissertation.

1.1 Literature on several biological sequential data

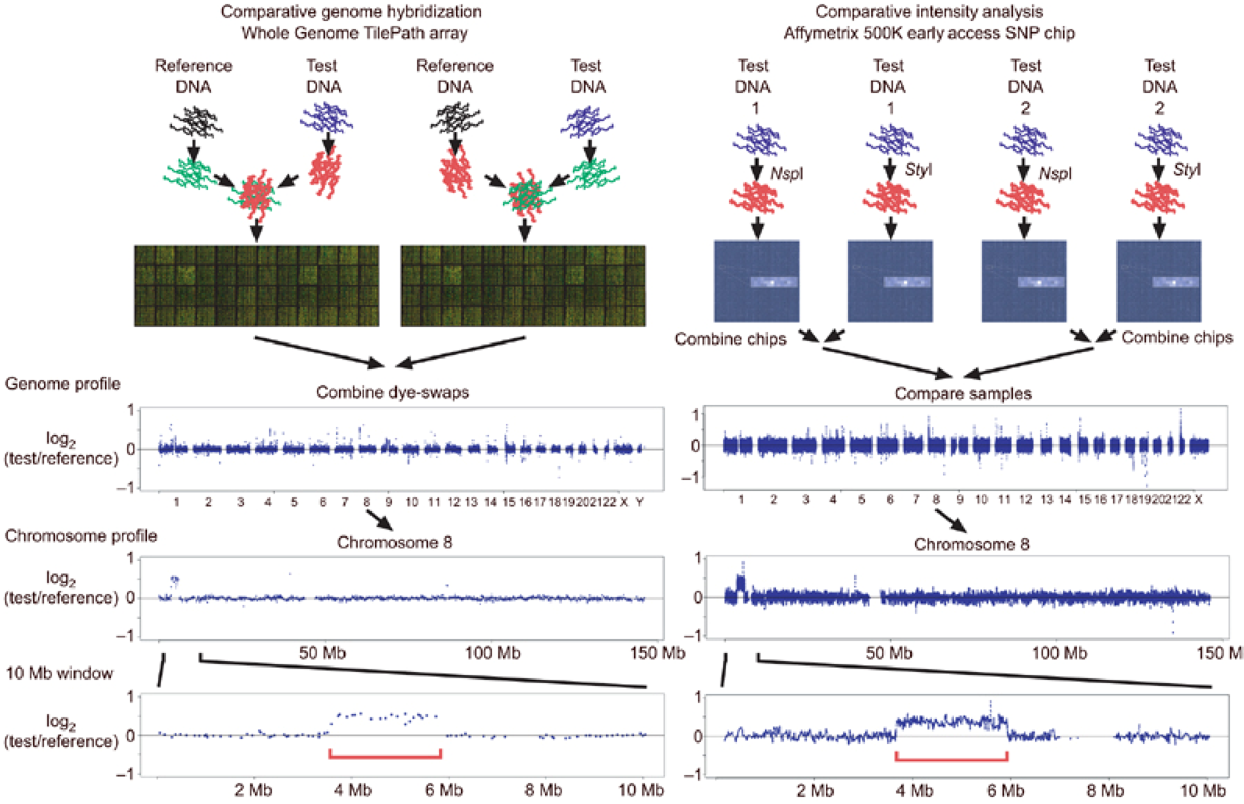
1.1.1 The segmentation of Copy Number Variation (CNV) using arrayCGH data

Genetic variation plays a crucial role in biological function. Initially, more attention was placed on studying single-nucleotide polymorphisms (SNPs). As a result, some 10 million SNPs have been identified, many of which have documented causal relationships with specific diseases. Recently, another type of genetic diversity, copy number variation (CNV), has been garnering a lot of attention. CNVs span larger regions of the genome and are not limited to a single nucleotide. They are manifested in several ways-as deletions, amplifications, inversions, or more complex configurations. A significant role for them has been elucidated in breast cancer, small cell and non-small cell lung cancer, Hodgkin's lymphoma, and various other diseases. Recent reports suggest their role has been greatly underestimated; over 1,000 CNV regions were identified, covering over 12% of the human genome (Redon *et al.,* 2006). Given the relative novelty of the field and significant clinical impact, great effort is now being directed to understanding the nature of CNVs. Coinciding with today's genomic era and widespread adoption of high-throughput technologies, microarrays have quickly been integrated and remain a fixture of the CNV data analysis pipeline.

The initial methods of comparative genomic hybridization (CGH) and representational differential analysis (RDA) for cytogenetic testing were proposed in the early 90's. Later, to reduce the complexity of whole genome samples, an enzyme digestion and PCR step was appended. These methods were effective but laborious and limited in resolution. High-throughput implementations were forged, in 1997, when Solinas-Toldo and colleagues proposed the initial concept of employing a matrix of selected DNA regions, 100kb, cloned into bacteria artificial chromosomes (BACs) to analyze genetic alterations. Today, commercial solutions from companies like Affymetrix, Illumina, and Roche offer arrays with probe lengths as short as 25-mer numbering in the millions. Furthermore, protocols supporting next-generation sequencing methods for CNV analysis have already begun trickling out. The basic protocol requires slides with probes spread across the genome. Then, fluorescent labeled genomic DNA from a test and a reference sample are added to the slide. Hybridization to the probes emits fluorescence, which can be quantified. Because the test and reference are labeled with different color dyes that can be measured independently, the ratio of the two, usually represented in $\log 2$, can serve as an indicator of copy number changes. Many different types of arrayCGH platforms now exist each with its own probe characteristics, as described by Tan and colleagues and summarized in Figure 1.1 (Tan *et al.,* 2007).

Given the raw DNA copy number arrayCGH data from a single sample, the first step in the analysis is to estimate the underlying real copy number at each probe location from the noisy log intensity value. This problem, referred to as segmentation problem, has drawn considerable attention and thus many statistical approaches have been proposed for this problem, such as: hidden Markov models (Fridlyand *et al.,* 2004), hierarchical tree clustering (Wang *et al.,* 2005), WECCA (Van Wieringen *et al.,* 2008), dynamic tree cutting (Langfelder *et al.,* 2007), HOPACH (van der Laan and Pollard, 2003), a Bayes regression approach (Wen *et al.,* 2006), wavelet approximation (Hsu *et al.,* 2005), Gaussian Mixture models (Engler *et al.,* 2006; Guha *et al.,* 2006). Lai *et al.,* 2007 , Willenbrock and Fridlyand (2005) reviewed and compared the performance of the existing approaches in 2005. Besides above, we will emphasis on two approaches: the change-point formulation under the Circular Binary Segmentation (CBS) algorithm (Olshen *et al.,* 2004)) and a stochastic segmentation

Figure 1.1: Array CGH protocols for BACS whole genome TilePath arrays (left) and Affymetrix high-density arrays (right) and proceeding data analysis (Redon *et al.*, 2006).



model (SCP) proposed by Lai *et al.*, 2007. The former one is the most widely used model for the CNV analysis. This recursive change-point formulation underlying CBS requires the least assumption and with a hybrid approach (Venkatraman and Olshen, 2007) to obtain the P-value of the likelihood ratio test (LRT), it is a fast method with very good robustness. However, most of the methods above including CBS have no means to evaluated the segments, for example, people need to decide how many segments is valid in the data. While the SCP model produces a way of assessing the confidence in the segmentation with its own attractive statistical and computational properties.

1.1.2 The “peak” calling and segmentation problems in different types of ChIP-seq data

Recent technological innovations have transformed the study of DNA-binding proteins as higher throughput techniques have come to the fore. In particular, the widely used procedure involving in vivo immunoprecipitation of chromatin-bound proteins (ChIP) has benefited from significant innovation, undergoing several reincarnations, from ChIP-qPCR to ChIP-chip (Ren, Bing *et al.,* 2000) and, most recently, to ChIP-seq (Johnson, David S. *et al.,* 2007; Robertson *et al.,* 2007). To unravel the mechanisms of gene regulation, understanding the complex interplay of protein-DNA interactions is instrumental, ChIP-seq has risen as the go-to technique for examining these interactions on a genome-wide scale. ChIP has been commonly used for illuminating transcription factor binding sites (TFBS) (Johnson, David S. *et al.,* 2007; Robertson *et al.,* 2007), but has more recently seen widespread adoption in studying epigenomic mechanisms—most notably, the role of post-translational, covalent histone modifications (Barski *et al.,* 2007; Mikkelsen *et al.,* 2007; Guttman *et al.,* 2009). As a case in point, the NIH Roadmap Epigenomics Mapping Consortium has embarked on an effort to catalogue the most comprehensive database of epigenomic data to date—including data on over 25 histone marks, along with DNA methylation, chromatin accessibility, and small RNA expression (Bernstein *et al.,* 2010). Understanding the epigenome is crucial due to its purported involvement in myriad roles from individual diversity to development to cancer and other complex diseases (Hawkins , R. David *et al.,* 2010; Kouzarides, T 2002; Widschwendter *et al.,* 2007). At the molecular level, histone modifications, in particular, have been linked to regulation of transcription, gene silencing, and chromatin reorganization (Kouzarides T *et al.,* 2002; Zhang and Pugh 2011). These associations have given rise to the “histone code” hypothesis that could perhaps be a major mechanism for modulation of the epigenome (Jenuwein, T. and Allis, C. D. 2001). Figure 1.2 shows the experiment process and several different types of ChIP-seq data profile.

ChIP can be broadly applied to study many protein-DNA interactions and on-going optimization is routinely introducing novel transcription factors and histone modifications to the diverse list of targeted proteins. From extremely sharp and punctate peaks to large, broad, and diffuse islands of enrichment, read profile signatures can span a wide range. Owing to this diversity, read profiles vary markedly and each presents its own nuanced challenges during downstream analysis. Algorithmically, punctate and diffuse enrichment have ostensibly been addressed as two mutually exclusive data types requiring distinct approaches. For instance, many transcription factors and histone acetylation modifications generate punctate profiles characterized by well-formed, sharply enriched peaks interspersed by large stretches of low signal. Several successful solutions have been introduced to address this problem (Zhang, Yong *et al.,* 2008; Rozowsky, Joel *et al.,* 2009; Qin, Zhaohui *et al.,* 2010). Here we emphasis on "MACS" since it is one of the most popular methods. They used window based approach to solve the peak calling problem. Firstly, MACS slides twice of preset bandwidth windows across the genome to search the locations with very enriched signals of the Watson strand and Crick strand, where they estimate the mean distance between the summit of these two strand as d. Then the reads will be moved to the middle of these two strand by $d/2$ base pair. They model the tag distribution along the genome by a Poisson distribution. They use one parameter λ to capture both the mean and the variance. $2d$ windows are slided across the genome to find candidate peaks. MACS uses a dynamic parameter λ_{local} defined as $\lambda_{local} = max(\lambda_{background}, \lambda_{1k}, \lambda_{5k}, \lambda_{10k})$, where $\lambda_{1k}, \lambda_{5k}, \lambda_{10k}$ are estimated λ around each candidate peak region in the control data. MACS uses λ_{local} to calculate the p-value and smooth out the potential false positives.

However, such window based approach has resolution problem and thus including lots of co-factor peaks. Moreover, as punctate peaks degenerate into more diffuse islands, read density enrichment appears far less pronounced, with much higher variance, and span much larger regions. In this scenario, peak-calling algorithms are extended beyond their intended scope and lose effectiveness (Pepke, Shirley *et al.,* 2009). Such non-punctate profiles are commonly observed when studying broad histone modifications, e.g. H3K27me3, H3K36me3, and H3K9me3. Instead, heuristic, window-based derivations have been developed to address this inadequacy (Hawkins, R. David *et al.,* 2010, Zang, CZ *et al.,* 2009). To satisfy the requirement in both peak calling and histione modification segmentation, Xing *et al.* (2012) develop a Bayesian change-point (BCP) model that is based on recent advances in infinite-state hidden Markov modeling, which has been discussed by Lai and Xing (2011). This model provides explicit formulas for posterior means of ChIP-seq read density profiles and allows a computationally efficient and fast approximation algorithm for these posterior means. An enhanced signal is generated that can then be used to identify segments with a shared read density and the "change-points" that separate them. The BCP enables analysis of whole genome ChIP-seq data with enhanced precision since read density estimates can adopt any real number value, which is more flexible than hidden Markov models with finite-state assumption. Therefore, the BCP can quickly identify islands of histone modification (HM) enrichment that correlates well with known functional associations and are both reproducible and robust at high resolution. Additionally, the BCP characterizes the diversity of ChIP-seq density profiles into and was easily adapted to segmenting sharper, punctate peaks of transcription factor (TF) with performance on par with a widely peak-calling algorithm while concurrently maintaining proficiency in diffuse data types. BCP accepts the browser extensible data (BED) format, which we transformed to read counts at every genomic region location for each chromosome. Only reads mapping to a unique genomic location were considered and only a single read per start/end coordinate was allowed to reduce spurious amplification and repetitive sequence bias. In the case of transcription factor ChIP data, adjacent positions with identical read counts were "blocked" together. For histone modification ChIP data, read counts at 200bp adjacent windows were calculated. This window size is the default setting for BCP and was chosen for two reasons. First, a single nucleosome is the expected smallest unit size for histone modification data, including wound and linker DNA, and is roughly this length. Second, 200bp is approximately the size-selected length, following DNA fragmentation, for most library preparation protocols. The user can adjust the window size, but in our experience, optimization away from the default value was rarely necessary. We assumed that read counts or average read counts on within "blocks" or windows, respectively, follow a Poisson distribution with mean θ_t, $t = 1, \ldots, n$, where n is the number of "blocks" or windows in the chromosome, and the true signal θ_t may undergo occasional change with probability p at each location t. they also assume that when θ changes to a new value at $t + 1$, the new value follows a Gamma(α, β) conjugate prior. Under this setup, the posterior distribution of θ_t given all the data is a mixture of Gamma distributions,

$$f(\theta_t|\mathcal{Y}_n) = \sum_{1\leq i\leq t\leq j\leq n} \gamma_{ijt}\text{Gamma}(\alpha_{ij}, \beta_{ij}). \tag{1.1.1}$$

Hence θ_t can be estimated by a weighted average of posterior means with different window sizes. In practical analysis, the model parameters p, α, β can be replaced by their maximum likelihood estimates, and the mixture above can be approximated by a bounded complexity mixture (BCMIX) algorithm (Lai and Xing, 2011).

BCP, as a change point model, has key differences with other similarly minded methods. Its estimate of true signal requires no prior knowledge of the number of different states of θ_t, nor the positions or magnitude of the change points. The posterior mean, as an estimator, plays an important role in peak calling (TF) and/or segmentation (HM) and we implemented it directly to finding putative TFBS and histone-mark enriched islands. Given the posterior mean of each block or window represents a piecewise constant signal, smoothed by incorporating upstream and downstream information, "false" enrichment areas caused by local noise were minimized and our ability to identify the most likely enriched region was enhanced. Consequently, “gaps” in large significant domains were marginalized and we performed segmentation using a simple cut-line across the posterior means decided from the background signal After generating candidate segments, each was substantiated as a peak or island of enrichment if the number of ChIP-seq reads within the region surpassed the 90th-quantile value expected assuming read number follows a Poisson distribution with a mean based on the number of input reads in the same region.

1.2 A Motivating Question

Although we didn’t elaborate the details of the classical finite Hidden Markov Model (HMM), it is obvious that HMM has been comprehensively used in the segmentation of these data, such as the earliest method for CNV detection(Fridlyand *et al.,* 2004) and Hpeak (Zhang, Y *et al.,* 2010) and etc. Most of these HMM based methods, although has trivial differences in preprocessing procedure or hyperparameter estimating, assume in common that the observation y_i are emitted by an underlying Markov chain with a finite state sequence S and obtains the smoothing estimation with Viterbi algorithm, Baum-Welch algorithm or similar procedure. The discrete state model works well for detecting inherited CNVs in normal sample, but is not good for the heterogeneity sample. The similar case for ChIP-seq data as we mentioned before, the large variation of the read counts also make it hard for classical finite-state HMM to smooth the parameters. That is the reason we develop the infinite-state segmentation model under the Bayesian framework. To summarize, the finite-state HMM has some drawbacks as below:

1. It is common to describe each hidden state by a single distribution. Such assumption produces a poor fit to the heterogeneity data with many outliers log ratios (Lian H

et al., 2006). Such outliers would hurt the state prediction of HMMs and have an inappropriate impact on parameter estimates.

2. Aware of the variation within the state, some modify the HMM assuming the continuous valued jumps (Guha *et al.,* 2006; Engler *et al.,* 2006; Lian, H *et al.,* 2006). But they still require pseudo-likelihood based approached or use Markov chain Monto Carlo (MCMC) to estimate the underlying states distribution. This is time-consuming for the large scale biological data such as ChIP-seq.

3. The finite-states model has no prior knowledge of the number of states, the number of segments, the magnitude of the states in advance thus, it often implement naive clustering or using sliding windows to preprocessing the sequence and cause resolution problem.

To address these drawbacks, Lai and Xing developed a novel Bayesian segmentation model with infinite-state assumption (Lai and Xing 2011) and we have apply it to the ChIP-seq data analysis and generate the tool BCP (Xing *et al.,* 2012). This infinite-state style model can handle all of the drawbacks listed above. However, as it has no constraint on the number of state we cannot estimate the posterior state probability. Put it in another word, we can generate continuous features by the posterior estimates but still need post processing step to do the hard segmentation. However, in many cases, the observations do not have such big fluctuation like the ChIP-seq data. The hidden parameter within one state still need using mixture model but we can constrain the variation of the mixture distribution in some scope. Let us take a look at the Figure 1.3 from Lian H *et al.,* 2006.

This figure is captured from the paper "Automated mapping of large-scale chromatin structure in ENCODE". In this figure, we can easily discover three segments in the first state (state 0) with the change points δ_0, δ_3 and δ_4. Yet, in this segments, their means take three different values differ from one another. Meanwhile, in other two segments from another state (state 1) with change points δ_1, δ_2 has two different means. Both means in state 1 have larger value compared to the ones of state 0. Moreover, the variance has similar scenario. Lian *et al.,* employed a Bayesian hierarchical change-point model (CPM) to deal with such problem. Similar with other mixture model they use pseudo-likelihood based approached and select the training data sets to learn the model with its distinguishing feature that the distance between change points are not geometrically distributed, which implies that a change point might appear within one state.

In contrast, we want to generate our novel stochastic segmentation model under Bayesian framework has the features as below:

1. Taking advantage of the mixture model, we character the hidden states by a continuous mixture probability distribution. Yet, instead of using MCMC or pseudo-likelihood based approached, explicit formula integrated with a linear time complexity approximation will be adopted to smoothing the parameters.

2. Unlike the infinite-state non-linear segmentation model as BCP, we estimate the posterior state probability. That is to say, we can simultaneously characterize the continuous and categorical features of the sequential data.

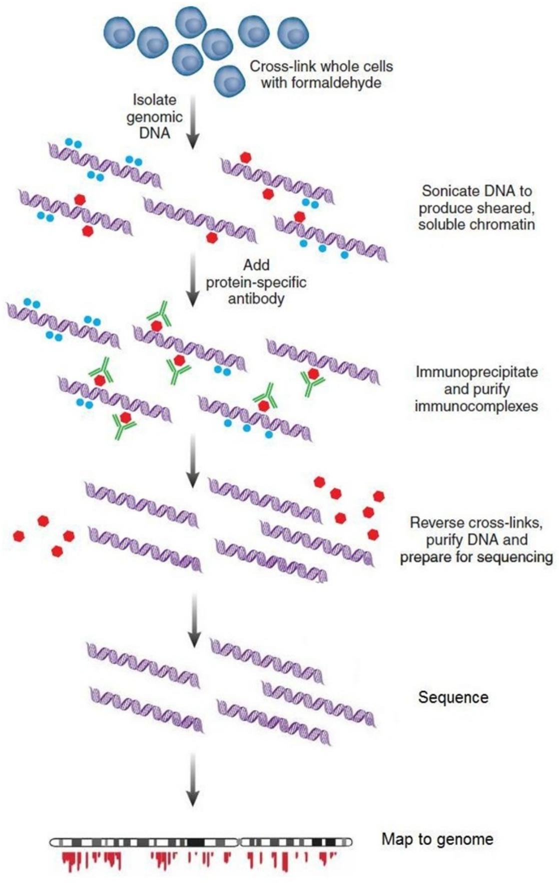
3. The distance between the change points are geometrically distributed as we assume there is no change point within one states.

4. Besides modeling the mean parameter with the hidden states, we assume the variance parameter has the hidden states as well. In another word, it is supposed to see the piecewise constant profile not only in mean parameter but also in variance.

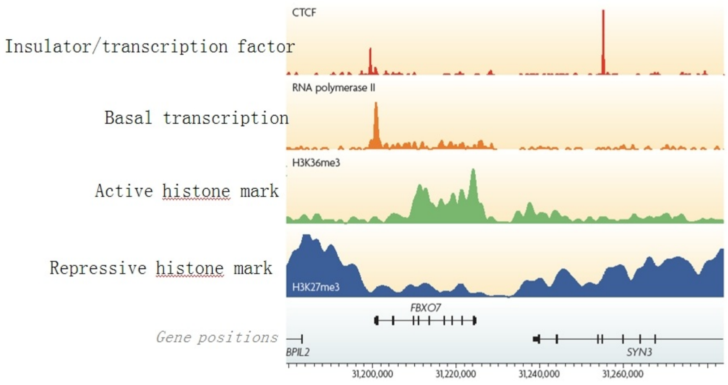
1.3 Outline

Based on the motivation above, this dissertation research which started from last November constructs a new model to solve such interesting segmentation problem. It studies the estimation of parameters in a stochastic segmentation model, exploring its application to some biological sequential data analysis. In Chapter 2, We will detailed describe this segmentation model on how to estimate the posterior distribution of mean, variance and the posterior state probability with explicit formula. Since the proposed model uses a Bayesian framework, we will use some demo graph to explain the structure of the model. Expectation Maximum (EM) are used for hyper-parameter estimation. Furthermore, to improve the computational efficiency of the estimation, a bounded complexity mixture approximation (BCMIX) is introduced. In Chapter 3, we implement the large scale simulation study to demonstrate its accuracy and robustness compared to the Bayes estimators and under different simulation settings. In Chapter 4, we will apply the model to two real biological sequential data: Nimblegen ENCODE Array for identifying DNaseI sensitivity and DNaseI hypersensitive sites over the ENCODE regions in human lymphoblastoid cells (GSE4334) and Reduced Representation Bisulfite Sequencing data (RRBS) (GSE31971) to see the directional DNA methylation changes and complex intermediate states accompany lineage specificity in the adult hematopoietic compartment. In the end, some conclusions and discussions are given in Chapter 5.

Figure 1.2: Top: the work flow for Chromatin Immunoprecipitation sequencing (ChIP-seq) experiment; Bottom: Different types of ChIP-seq data profile.

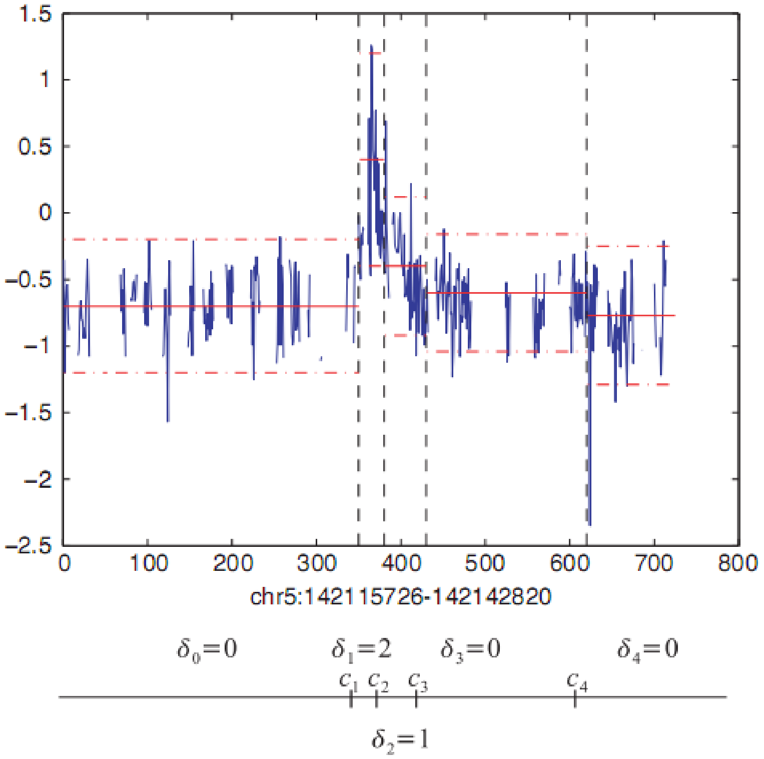

Cross-link whole cells with formaldehyde
Isolate genomic DNA
Sonicate DNA to produce sheared, soluble chromatin
Add protein-specific antibody
Immunoprecipitate and purify immunocomplexes
Reverse cross-links, purify DNA and prepare for sequencing
Sequence
Map to genome

Elaine R Mardis. ChIP-seq: welcome to the new frontier. Nature Methods - 4, 613 - 614 (2007)



Park. ChIP-seq: advantages and challenges of a maturing technology. Nat Rev Genet (2009) vol. 10 (10) pp. 669-80

Figure 1.3: A sample segmentation using Bayesian hierarchical change-point model (CPM) for a 27 kb region within ENr212.



Chapter 2

Estimation in a novel Stochastic Segmentation Model

2.1 Model specification

The classic Hidden Markov Model described in the introduction can deal with the classification problem. With the prior distribution, it can generate the posterior estimation of the target variable. Yet, in practice, the value of the unknown parameters with the same hidden state are not necessary the same. That is our motivation to consider this novel segmentation model to incorporate this feature.

We consider the following model assumptions:

1. The observations y_t follow the model $y_t = \mu_t + \sigma_t \epsilon_t$ for $t = 1, \dots, n$, where ϵ_t are independent normal random variables with mean 0 and variance 1.

2. The categorical states of $\boldsymbol{\theta}_t := (\mu_t, \sigma_t)$ are represented by a K-state irreducible hidden Markov chain $\{s_t\}$ with transition probability matrix $Q = (q_{ij})$ and stationary distribution π.

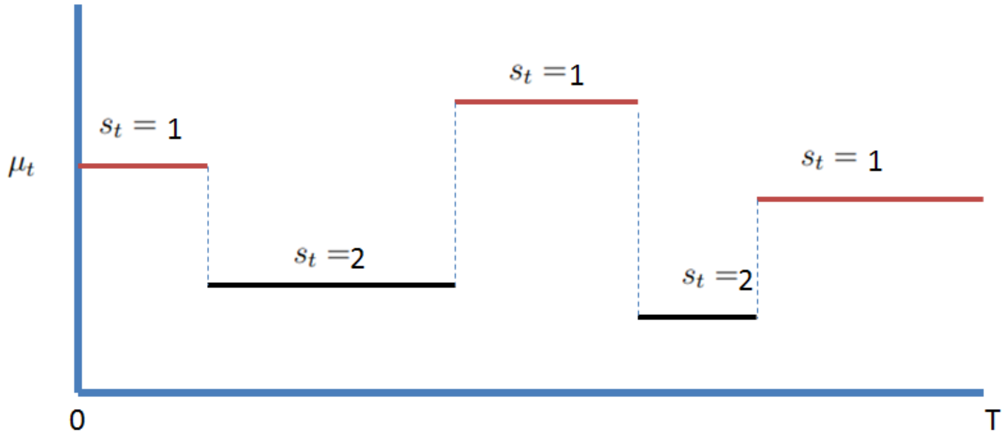
3. The dynamics of $\boldsymbol{\theta}_t$ is given by $\boldsymbol{\theta}_t = 1_{\{s_t = s_{t-1}\}} \theta_{t-1} + 1_{\{s_t \neq s_{t-1}\}} (z_t,)$, where

$$\mu_t | \sigma_t \sim N(z^k, \sigma_t^2 \kappa^k), \quad (2\sigma_t^2)^{-1} \sim \text{Gamma}(g^{(k)}, \lambda^{(k)}),$$

where $z^{(k)}, \kappa^{(k)}, \lambda^{(k)}$, and $g^{(k)}$ $(k = 1, \ldots, K)$ are hyper-parameters.

Note that the second assumption indicates $\boldsymbol{\theta}_t$ has a stationary distribution and thus a reversible Markov chain can be defined. Unlike the classic HMM models the $\boldsymbol{\theta}_t$ is a constant based on the hidden state, the third assumption allows the parameter in each state is a random variable follow a certain distribution. Let's take a two-state scenario as an example. Figure 2.1 visually demonstrates the assumptions, showing an example of possible values of a one-dimensional μ_t. Four transitions occur during the period $0 \leq t \leq T$. Within each state, μ_t take different values. The values are realizations from the state-specific distribution of $\mu(s_t)$. The transitions are governed by some hidden Markov chain.

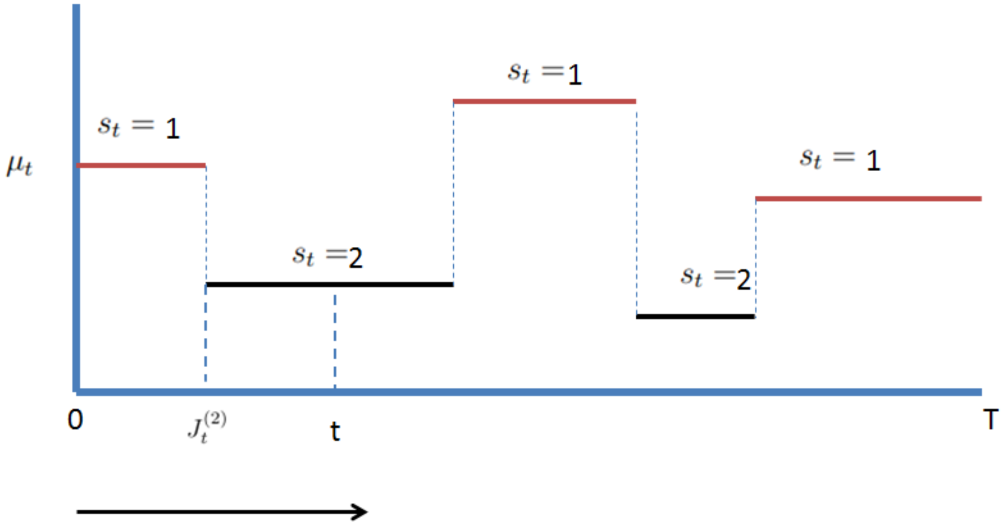
Figure 2.1: Illustration: Values of $\beta(s_t)$ in a stochastic regime switching model.



2.2 The forward Filtering estimate of parameters

Let's first discuss about the forward filter, which is the estimate of $\boldsymbol{\theta}_t$ for any time t given all the information from the beginning to t. Let $J_t^{(k)} = \max\{i \leq t : s_{i-1} \neq s_i = \cdots = s_t = k\}$ be the most recent switching time prior or equal to t and at which s_t switches from a state other than k to state k. Figure 2.2 illustrates the definition of $J_t^{(k)}$. At time t, $s_t = 2$, and the most recent switching occurs before t is at time $J_t^{(2)}$ as shown in the figure.

Figure 2.2: Illustration: Definition of $J_t^{(k)}$.



Let

$$\xi_t^{(k)} = P(s_t = k|\mathcal{Y}_t), \qquad \xi_{i,t}^{(k)} = P(J_t^{(k)} = i|\mathcal{Y}_t), \tag{2.2.1}$$

for $1 \leq i \leq t$ and $1 \leq j \leq K$, in which $\mathcal{Y}_{i,j} := \{y_i, \ldots, y_j\}$ and $\mathcal{Y}_t := \mathcal{Y}_{1t}$, then, by definition, $\xi_t^{(k)} = \sum_{i=1}^t \xi_{i,t}^{(k)}$. We define $f(x|\cdot)$ is the probability density function of distribution $[x|\cdot]$. The conditional distribution of $\boldsymbol{\theta}_t$, given $\mathcal{Y}_t$ and $J_t^{(k)} = i$ is composed by two parts:

$$\begin{aligned}
& f(\mu_t|\sigma_t, \mathcal{Y}_{i,t}) \propto f(\mathcal{Y}_{i,t}|\mu_t, \sigma_t) \cdot f(\mu_t|\sigma_t) \\
\propto \quad & \exp(-\frac{\sum_{j=i}^t (y_j - \mu_t)^2}{2\sigma_t^2}) \cdot \exp(-\frac{(\mu_t - z^{(k)})^2}{2\sigma_t^2 \kappa^{(k)}}) \\
\propto \quad & \exp\Big\{ - \frac{\mu_t^2 - 2\frac{\sum_{j=i}^t y_j \kappa^{(k)} + z^{(k)}}{(t-i+1)\kappa^{(k)}+1}\mu_t + \frac{\kappa^{(k)}}{(t-i+1)\kappa^{(k)}+1}(\sum_{j=i}^t y_j^2 + \frac{(z^{(k)})^2}{\kappa^{(k)}})}{2\sigma_t^2 \frac{\kappa^k}{(t-i+1)\kappa^k+1}} \Big\}
\end{aligned}$$

$$\propto \;\; \exp\Big\{-\frac{1}{2\sigma_t^2(t-i+1+1/\kappa^{(k)})^{-1}}\Big\{\mu_t-(t-i+1+1/\kappa^{(k)})^{-1}(\sum_{j=i}^{t} y_j+\frac{z^{(k)}}{\kappa^{(k)}})\Big\}^2\Big\}.$$

Let

$$\kappa_{it}^{(k)}=\big(\frac{1}{\kappa^{(k)}}+t-i+1\big)^{-1}, \qquad z_{it}^{(k)}=\kappa_{it}^{(k)}\big(\frac{z^{(k)}}{\kappa^{(k)}}+\sum_{j=i}^{t} y_j\big).$$

Then

$$\mu_t|\sigma_t,\mathcal{Y}_{it}\sim N(z_{it}^{(s_t)},\sigma_t^2\kappa_{it}^{(s_t)}).$$

And let $P_t=(2\sigma_t^2)^{-1}, C_0=(\Gamma(g^{(k)}))^{-1}(\lambda^{(k)})^{-g^{(k)}}(\kappa^{(k)})^{-\frac{1}{2}}$.

$$\begin{aligned}
f(P_t|\mathcal{Y}_{i,t}) &\propto f(\mathcal{Y}_{i,t}|P_t)\cdot f(P_t)\\
&= \int f(\mathcal{Y}_{i,t}|\mu_t,P_t)\cdot f(\mu_t|P_t)d\mu_t\cdot f(P_t)\\
&= C_0P_t^{g^{(k)}-1}\exp\{\frac{-P_t}{\lambda^{(k)}}\}\int P_t^{\frac{t-i+2}{2}}\exp\Big\{-P_t\sum_{j=i}^{t}(y_j-\mu_t)^2\Big\}\cdot\exp\Big\{-P_t\frac{(\mu_t-z^{(k)})^2}{\kappa^{(k)}}\Big\}d\mu_t\\
&= C_0P_t^{(g^{(k)}+\frac{t-i+2}{2})-1}\exp\Big\{-P_t(\frac{1}{\lambda^{(k)}}+\frac{(z(k))^2}{\kappa^{(k)}}+\sum_{j=i}^{t}y_j^2-\frac{(z_{it}^{(k)})^2}{\kappa_{it}^{(k)}})\Big\}\\
\cdot &\int\exp\Big\{-\frac{P_t}{\kappa_{it}^{(k)}}(\mu_t-\kappa_{it}^{(k)}(\frac{z(k)}{\kappa^{(k)}}+\sum_{j=i}^{t}y_j))^2\Big\}d\mu_t\\
&= C_0(\kappa_{it}^{(k)})^{\frac{1}{2}}P_t^{\frac{1}{2}}P_t^{-\frac{1}{2}}P_t^{(g^{(k)}+\frac{t-i+1}{2})-1}\exp\Big\{-P_t(\frac{1}{\lambda^{(k)}}+\frac{(z(k))^2}{\kappa^{(k)}}+\sum_{j=i}^{t}y_j^2-\frac{(z_{it}^{(k)})^2}{\kappa_{it}^{(k)}})\Big\}.
\end{aligned}$$

Again let

$$g_{it}^{(k)}=g^{(k)}+(t-i+1)/2, \qquad \frac{1}{\lambda_{it}^{(k)}}=\frac{1}{\lambda^{(k)}}+\frac{(z^{(k)})^2}{\kappa^{(k)}}+\sum_{j=i}^{t}y_j^2-\frac{(z_{it}^{(k)})^2}{\kappa_{it}^{(k)}}.$$

Then

$$(2\sigma_t^2)^{-1}|\mathcal{Y}_{it}\sim \mathrm{Gamma}(g_{it}^{(k)},\lambda_{it}^{(k)}).$$

Combine the derivation above we can gain the conditional distribution of $\boldsymbol{\theta}_t$, given $\mathcal{Y}_t$ and $J_t^{(k)} = i$, is

$$\mu_t|\sigma_t, \mathcal{Y}_{it} \sim N(z_{it}^{(s_t)}, \sigma_t^2\kappa_{it}^{(s_t)}), \qquad (2\sigma_t^2)^{-1}|\mathcal{Y}_{it} \sim \text{Gamma}(g_{it}^{(k)}, \lambda_{it}^{(k)}).$$

Based on the above conditional distribution, the posterior distribution of $\boldsymbol{\theta}_t$ given $\mathcal{Y}_t$ is a mixture distributions:

$$\boldsymbol{\theta}_t|\mathcal{Y}_t \sim \sum_{k=1}^{K}\sum_{i=1}^{t} \xi_{i,t}^{(k)}\big[\boldsymbol{\theta}_t|\mathcal{Y}_{it}, J_t^{(k)} = i\big]. \tag{2.2.2}$$

Let us see how to derive the mixture weight $\xi_{i,t}^{(k)}$. First note that

$$f(\boldsymbol{\theta}_t, y_t, s_{t-1} = k|\mathcal{F}_{t-1}) = \sum_{l=1}^{K} f(\boldsymbol{\theta}_t, y_t, s_{t-1} = k, s_t = l|\mathcal{F}_{t-1}).$$

When $l \neq k$,

$$\begin{aligned}
& f(\boldsymbol{\theta}_t, y_t, s_{t-1} = k, s_t = l|\mathcal{F}_{t-1}) \\
= \;& f(\boldsymbol{\theta}_t, y_t|\mathcal{F}_{t-1}, s_{t-1} = k, s_t = l)P(s_{t-1} = k, s_t = l|\mathcal{F}_{t-1}) \\
= \;& f(y_t|\mathcal{F}_{t-1}, J_t^{(l)} = t)f(\boldsymbol{\theta}_t|\mathcal{F}_t, J_t^{(l)} = t)P(s_t = l|s_{t-1} = k)P(s_{t-1} = k|\mathcal{F}_{t-1}) \\
= \;& f(y_t|\mathcal{F}_{t-1}, J_t^{(l)} = t)f(\boldsymbol{\theta}_t|\mathcal{F}_t, J_t^{(l)} = t)p_{k,l}\xi_{t-1}^{(k)}.
\end{aligned}$$

When $l = k$,

$$f(\boldsymbol{\theta}_t, y_t, s_{t-1} = k, s_t = k|\mathcal{F}_{t-1}) = \sum_{i=1}^{t-1} f(J_t^{(k)} = i, \boldsymbol{\theta}_t, y_t|\mathcal{F}_{t-1})$$

$$= \sum_{i=1}^{t-1} f(\boldsymbol{\theta}_t, y_t|\mathcal{F}_{t-1}, J_t^{(k)} = i)P(s_{t-1} = k, s_t = k|\mathcal{F}_{t-1})$$

$$= \sum_{i=1}^{t-1} f(y_t|\mathcal{F}_{t-1}, J_t^{(k)} = i) f(\boldsymbol{\theta}_t|\mathcal{F}_t, J_t^{(k)} = i) P(s_t = k|s_{t-1} = k) P(s_{t-1} = k|\mathcal{F}_{t-1})$$

$$= \sum_{i=1}^{t-1} f(y_t|\mathcal{F}_{t-1}, J_t^{(l)} = t) f(\boldsymbol{\theta}_t|\mathcal{F}_t, J_t^{(k)} = i) p_{k,k} \xi_{i,t-1}^{(k)}.$$

Define

$$\xi_{i,t}^{(k)*} = \begin{cases} \big(\sum_{l \neq k} \xi_{t-1}^{(l)} p_{lk}\big) f(y_t|J_t^{(k)} = t) & i = t, \\ p_{kk} \xi_{i,t-1}^{(k)} f(y_t|\mathcal{F}_{t-1}, J_t^{(k)} = i) & i < t, \end{cases}$$

Thus

$$f(\boldsymbol{\beta}_t|\mathcal{F}_t) \propto \sum_{k=1}^{K} f(\boldsymbol{\beta}_t, y_t, s_{t-1} = k|\mathcal{F}_{t-1})$$

$$= \sum_{k=1}^{K} \xi_{t,t}^{(k)*} f(\boldsymbol{\theta}_t|\mathcal{F}_t, J_t^{(l)} = t) + \sum_{k=1}^{K} \sum_{i=1}^{t-1} \xi_{i,t}^{(k)*} f(\boldsymbol{\theta}_t|\mathcal{F}_t, J_t^{(k)} = i).$$

So the mixture weight $\xi_{i,t}^{(k)}$ is the conditional probability which can be determined by the recursions

$$\xi_{i,t}^{(k)} \propto \xi_{i,t}^{(k)*} := \begin{cases} \big(\sum_{l \neq k} \xi_{t-1}^{(l)} p_{lk}\big) f(y_t|J_t^{(k)} = t) & i = t, \\ p_{kk} \xi_{i,t-1}^{(k)} f(y_t|\mathcal{F}_{t-1}, J_t^{(k)} = i) & i < t. \end{cases} \quad (2.2.3)$$

When $i = t$:

Recall $P_t = (2\sigma_t^2)^{-1}$ and define $\psi_{0,0}^{(k)} = (\kappa^{(k)})^{-\frac{1}{2}} \frac{(\lambda^{(k)})^{-g^{(k)}}}{\Gamma(g^{(k)})}$.

$$f(y_t|J_t^{(k)} = t) = \int_{\theta_t} f(y_t|\theta_t) f(\boldsymbol{\theta}_t|J_t^{(k)} = t) d\boldsymbol{\theta}_t$$

$$= \psi_{0,0}^{(k)} \int_{P_t} \int_{\mu_t} P_t^{\frac{1}{2}} \exp\{-P_t(y_t - \mu_t)^2\} P_t^{\frac{1}{2}} \exp\Big\{-P_t \frac{(\mu_t - z^{(k)})^2}{\kappa^{(k)}}\Big\} P_t^{g^{(k)}-1} \exp\{-\frac{P_t}{\lambda^{(k)}}\} d\mu_t dP_t$$

$$
\begin{aligned}
&= \psi_{0,0}^{(k)} \int_{P_t} P_t^{g^{(k)}} \exp\Big\{ - P_t[\frac{1}{\lambda^{(k)}} + y_t^2 + \frac{(z^{(k)})^2}{\kappa^{(k)}} - \frac{(\kappa_{tt}^{(k)})^2}{\kappa_{tt}^{(k)}}(y_t + \frac{z^{(k)}}{\kappa^{(k)}})^2]\Big\} \cdot \\
&\quad \int_{\mu_t} \exp\Big\{\frac{-P_t}{\kappa_{tt}^{(k)}}[\mu_t - \kappa_{tt}^{(k)}(y_t + \frac{z^{(k)}}{\kappa^{(k)}})]^2\Big\} d\mu_t dP_t \\
&= \psi_{0,0}^{(k)} \int_{P_t} P_t^{g^{(k)}} \exp\Big\{ - P_t[\frac{1}{\lambda^{(k)}} + y_t^2 + \frac{(z^{(k)})^2}{\kappa^{(k)}} - \frac{(z_{tt}^{(k)})^2}{(\kappa_{tt}^{(k)})}]\Big\} P_t^{-\frac{1}{2}} (\kappa_{tt}^{(k)})^{\frac{1}{2}} dP_t \\
&= \psi_{0,0}^{(k)} (\kappa_{tt}^{(k)})^{\frac{1}{2}} \int_{P_t} P_t^{(g^{(k)} + \frac{1}{2}) - 1} \exp\{\frac{-P_t}{\lambda_{tt}^{(k)}}\} dP_t \\
&= \psi_{0,0}^{(k)} (\kappa_{tt}^{(k)})^{\frac{1}{2}} \Gamma(g_{tt}^{(k)}) (\lambda_{tt}^{k})^{g_{tt}^{(k)}}.
\end{aligned}
$$

Define $\psi_{tt}^{(k)} = (\kappa_{tt}^{(k)})^{-\frac{1}{2}} \frac{(\lambda_{tt}^{k})^{-g_{tt}^{(k)}}}{\Gamma(g_{tt}^{(k)})}$, then:

$$
f(y_t | J_t^{(k)} = t) = \psi_{0,0}^{(k)} / \psi_{t,t}^{(k)}.
$$

When $i < t$:

Again $P_t = (2\sigma_t^2)^{-1}$ and define $\psi_{i,t-1}^{(k)} = (\kappa_{it-1}^{(k)})^{-\frac{1}{2}} \frac{(\lambda_{it-1}^{k})^{-g_{it-1}^{(k)}}}{\Gamma(g_{it-1}^{(k)})}$.

$$
\begin{aligned}
&f(y_t | \mathcal{F}_{t-1}, J_t^{(k)} = i) = \int_{\theta_t} f(y_t | \theta_t) f(\boldsymbol{\theta}_t | \mathcal{F}_{t-1}, J_t^{(k)} = t) d\boldsymbol{\theta}_t \\
&= \psi_{i,t-1}^{(k)} \int_{P_t} \int_{\mu_t} P_t^{g_{it-1}^{(k)}} \exp\Big\{ - P_t[(y_t - \mu_t)^2 + \frac{(\mu_t - z_{it-1}^{(k)})^2}{\kappa_{it-1}^{(k)}} + \frac{1}{\lambda_{it-1}^{(k)}}]\Big\} d\mu_t dP_t \\
&= \psi_{i,t-1}^{(k)} \int_{P_t} P_t^{g_{it}^{(k)} - 1} P_t^{\frac{1}{2}} \exp\Big\{ - P_t[\frac{1}{\lambda_{it-1}^{(k)}} + y_t^2 + \frac{(z_{it-1}^{(k)})^2}{\kappa_{it-1}^{(k)}} - \frac{(\kappa_{it}^{(k)})^2}{\kappa_{it}^{(k)}}(y_t + \frac{z_{it-1}^{(k)}}{\kappa_{it-1}^{(k)}})^2]\Big\} \\
&\cdot \quad \int_{\mu_t} \exp\Big\{ - \frac{P_t}{\kappa_{it}^{(k)}}[\mu_t - \kappa_{it}^{(k)}(y_t + \frac{z_{it-1}^{(k)}}{\kappa_{it-1}^{(k)}})]^2\Big\} d\mu_t dP_t \\
&= \psi_{i,t-1}^{(k)} (\kappa_{it}^{(k)})^{\frac{1}{2}} \int_{P_t} P_t^{g_{it}^{(k)} - 1} P_t^{\frac{1}{2}} P_t^{-\frac{1}{2}} \exp\Big\{ - P_t[\frac{1}{\lambda^{(k)}} + \frac{(z^{(k)})^2}{\kappa^{(k)}} + \sum_{j=i}^{t} y_j^2 - \frac{(z_{it}^{(k)})^2}{\kappa_{it}^{(k)}}]\Big\} dP_t \\
&= \psi_{i,t-1}^{(k)} (\kappa_{it}^{(k)})^{\frac{1}{2}} \Gamma(g_{it}^{(k)}) (\lambda_{it}^{k})^{g_{it}^{(k)}}.
\end{aligned}
$$

Define $\psi_{i,t}^{(k)} = (\kappa_{it}^{(k)})^{-\frac{1}{2}} \frac{(\lambda_{it}^{k})^{-g_{it}^{(k)}}}{\Gamma(g_{it}^{(k)})}$, then:

$$f(y_t|\mathcal{F}_{t-1}, J_t^{(k)} = i) = \psi_{i,t-1}^{(k)}/\psi_{i,t}^{(k)}.$$

Making use of $\sum_{k=1}^{K}\sum_{i=1}^{t}\xi_{i,t}^{(k)} = 1$, we can show that the conditional probabilities $\xi_{i,t}^{(k)}$ are determined by $\xi_{i,t}^{(k)} = \xi_{i,t}^{(k)*}\Big/\Big[\sum_{h=1}^{K}\sum_{j=1}^{t}\xi_{j,t}^{(h)*}\Big]$, in which

$$\xi_{i,t}^{(k)*} := \begin{cases} \big(\sum_{l\neq k}\xi_{t-1}^{(l)}q_{lk}\big)\psi_{0,0}^{(k)}/\psi_{t,t}^{(k)} & i = t, \\ q_{kk}\xi_{i,t-1}^{(k)}\psi_{i,t-1}^{(k)}/\psi_{i,t}^{(k)} & i < t, \end{cases} \tag{2.2.4}$$

and

$$\psi_{i,j}^{(k)} = \frac{1}{\sqrt{\kappa_{ij}^{(k)}}}\frac{1}{\Gamma(g_{ij}^{(k)})}\big[\lambda_{ij}^{(k)}\big]^{-g_{ij}^{(k)}}.$$

Hence expressions (2.2.1) and (2.2.2) implies

$$P(s_t = k|\mathcal{Y}_t) = \sum_{i=1}^{t}\xi_{i,t}^{(k)}, \qquad E(\boldsymbol{\theta}_t|\mathcal{Y}_t) = \sum_{k=1}^{K}\sum_{i=1}^{t}\xi_{i,t}^{(k)}E(\boldsymbol{\theta}_t|\mathcal{Y}_{it}). \tag{2.2.5}$$

2.3 The backward Filtering estimate of parameters

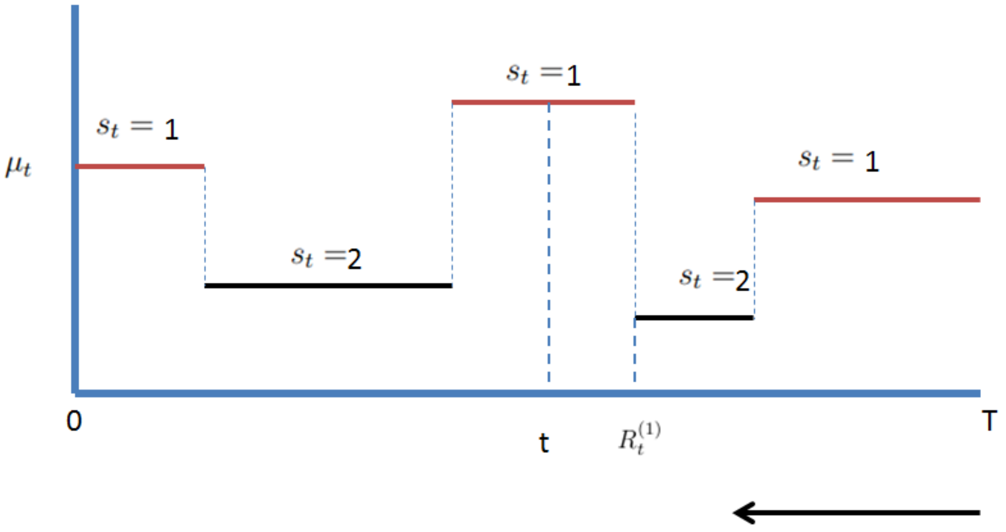
The model assumption implies that, a stationary distribution of $\boldsymbol{\theta}_t$ exists and is given by

$$\sum_{k=1}^{K}\pi_k \text{Normal}(z^{(k)}, V^{(k)}). \tag{2.3.1}$$

This indicates that, if $\boldsymbol{\theta}_t$ is initialized at the stationary distribution, its time-reversed Markov chain can be defined. This substantially simplifies the smoothing estimates of $\boldsymbol{\theta}_t$. Note that this also imposes stationarity conditions for y_t. As indicated before, $\boldsymbol{\theta}_t$ is a reversible Markov chain. Therefore we can obtain a backward filter that is analogous to the forward filter. That is, we reverse the location, starting with location T and estimating $\boldsymbol{\theta}_t$ for any position t given the "historical" information from T to t.

Define $R_t^{(k)} = \min\{j \geq t : k = s_t \cdots = s_{j-1} \neq s_j\}$ be the most recent changing position larger than or equal to t when s_t switches from the state k to another state. Figure 2.3 illustrates the definition of $R_t^{(k)}$. At time t, the regime is $s_t = 1$, and the most recent transition occurs after t is at $R_t^{(1)}$ as shown in Figure 2.3.

Figure 2.3: Illustration: Definition of $R_t^{(k)}$.



Define

$$\eta_t^{(k)} = P(s_t = k | \mathcal{F}_{t,T}), \qquad \eta_{j,t}^{(k)} = P(R_t^{(k)} = j | \mathcal{F}_{t,T}),$$

for $t \leq j \leq T$ and $1 \leq k \leq K$. The quantity $\eta_t^{(k)}$ is the conditional probability that the current state is k given information $\mathcal{F}_{t,T}$ $\eta_{i,t}^{(k)}$ is the conditional probability that the current state is k and the next transition occurs at location j given $\mathcal{F}_{t,T}$. Thus $\eta_t^{(k)} = \sum_{j=t}^{T} \eta_{t,j}^{(k)}$. If we know all the information from time t to T and that the next transition occurs at location j, we just need to use the information before the change to estimate the current value of $\boldsymbol{\theta}_t$.

We then use the time-reversed chain of $\boldsymbol{\theta}_t$ to obtain a backward analog of (2.2.2),

$$\boldsymbol{\theta}_{t+1}|\mathcal{Y}_{t+1,T} \sim \sum_{k=1}^{K} \sum_{j=t+1}^{T} \eta_{t+1,j}^{(k)} \left[\boldsymbol{\theta}_{t+1}|\mathcal{Y}_{t+1,j}\right], \tag{2.3.2}$$

in which the weights $\eta_{t+1,j}^{(k)}$ can be obtained by backward induction using the time-reversed counterpart of (2.2.4):

$$\eta_{t+1,j}^{(k)} \propto \eta_{t+1,j}^{(k)*} := \begin{cases} \left(\sum_{l\neq k} \eta_{t+2}^{(l)} \widetilde{q}_{lk}\right)\psi_{0,0}^{(k)}/\psi_{t+1,t+1}^{(k)} & j = t+1, \\ \widetilde{q}_{kk}\eta_{t+2,j}^{(k)}\psi_{t+2,j}^{(k)}/\psi_{t+1,j}^{(k)} & j > t+1, \end{cases} \tag{2.3.3}$$

where $\widetilde{Q} = (\widetilde{q}_{lk})$ is the transition matrix of the reversed chain of $\{s_t\}$, and $\widetilde{q}_{lk} = P(s_t = k|s_{t+1} = l)$. Since for $B \subset \mathbb{R}^d$, $P(\beta_t \in B|\mathcal{Y}_{t,T}) = \int P(\beta_t \in B|\beta_{t+1})dP(\beta_{t+1}|\mathcal{Y}_{t,T})$, it follows from (2.3.2) that

$$\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,T} \sim \sum_{k=1}^{K} \Big\{\widetilde{q}_{kk} \sum_{j=t+1}^{T} \eta_{t+1,j}^{(k)} \left[\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,j}\right] + \Big(\sum_{l\neq k} \widetilde{q}_{lk}\eta_{t+1}^{(l)}\Big)\boldsymbol{\theta}_t\Big\}. \tag{2.3.4}$$

2.4 Smoothing estimate of parameters

Next, we shall use Bayes' theorem to combine the forward filter (2.2.2) with its backward variant (2.3.4) to derive the posterior distribution of $\boldsymbol{\theta}_t$ given $\mathcal{Y}_T$ $(1 \leq t < T)$.

$$f(\boldsymbol{\theta}_t|\mathcal{Y}_T) = \sum_{k=1}^{K} f(\boldsymbol{\theta}_t, s_t = k|\mathcal{Y}_T) \propto \sum_{k=1}^{K} f(\boldsymbol{\theta}_t, s_t = k|\mathcal{Y}_t)\frac{f(\boldsymbol{\theta}_t, s_t = k|\mathcal{Y}_{t+1,T})}{f(\theta, s_t = k)}. \tag{2.4.1}$$

Thus we first study $f(\boldsymbol{\theta}_t, s_t = k|\mathcal{Y}_t)f(\boldsymbol{\theta}_t, s_t = k|\mathcal{Y}_{t+1,T})\big/f(\theta, s_t = k)$.

$$f(\boldsymbol{\theta}_t, s_t = k|\mathcal{Y}_t)f(\boldsymbol{\theta}_t, s_t = k|\mathcal{Y}_{t+1,T})\big/f(\theta, s_t = k)$$

$$
\begin{aligned}
&= \frac{\sum_{i=1}^{t} \xi_{i,t}^{(k)} f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) \cdot \{\widetilde{q}_{kk} \sum_{j=t+1}^{T} \eta_{t+1,j}^{(k)} f(\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,j}) + \sum_{l\neq k} \widetilde{q}_{lk} \eta_{t+1}^{(l)} f(\boldsymbol{\theta}_t|s_t = k)\}}{P(s_t = k) f(\boldsymbol{\theta}_t|s_t = k)} \\
&= \frac{\sum_{i=1}^{t} \xi_{i,t}^{(k)} f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) \cdot \widetilde{q}_{kk} \sum_{j=t+1}^{T} \eta_{t+1,j}^{(k)} f(\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,j})}{\pi_k f(\boldsymbol{\theta}_t|s_t = k)} \\
&\quad + \frac{\sum_{i=1}^{t} \xi_{i,t}^{(k)} f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) \cdot \sum_{l\neq k} \widetilde{q}_{lk} \eta_{t+1}^{(l)} f(\boldsymbol{\theta}_t|s_t = k)}{\pi_k f(\boldsymbol{\theta}_t|s_t = k)} \\
&= \sum_{i}^{t} \xi_{i,t}^{(k)} \sum_{l\neq k} \frac{\widetilde{q}_{lk}}{\pi_k} \eta_{t+1}^{(l)} f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) + \frac{\widetilde{q}_{kk}}{\pi_k} \sum_{1\leq i\leq t\leq j\leq T} \xi_{i,t}^{(k)} \eta_{t+1,j}^{(k)} \frac{f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) f(\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,j})}{f(\boldsymbol{\theta}_t|s_t = k)}.
\end{aligned}
$$

Further we study $f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) f(\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,j}) / f(\boldsymbol{\theta}_t|s_t = k)$:

$$
\begin{aligned}
& \frac{f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) f(\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,j})}{f(\boldsymbol{\theta}_t|s_t = k) f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,j})} \\
= \;& \frac{\psi_{i,t}^{(k)} \psi_{t+1,j}^{(k)}}{\psi_{i,j}^{(k)} \psi_{0,0}^{(k)}} P_t^{(g_{it}^{(k)} + g_{t+1j}^{(k)} + g^{(k)} + g_{ij}^{(k)})} \cdot \frac{\exp\{-P_t(\frac{1}{\lambda_{it}^{(k)}} + \frac{1}{\lambda_{t+1j}^{(k)}})\}}{\exp\{-P_t(\frac{1}{\lambda^{(k)}} + \frac{1}{\lambda_{ij}^{(k)}})\}} \cdot \frac{\exp\Big\{ - P_t[\frac{(\mu_t - z_{it}^{(k)})^2}{\kappa_{it}^{(k)}} + \frac{(\mu_t - z_{t+1j}^{(k)})^2}{\kappa_{t+1j}^{(k)}}]\Big\}}{\exp\Big\{ - P_t[\frac{(\mu_t - z^{(k)})^2}{\kappa^{(k)}} + \frac{(\mu_t - z_{ij}^{(k)})^2}{\kappa_{ij}^{(k)}}]\Big\}} \\
= \;& \frac{\psi_{i,t}^{(k)} \psi_{t+1,j}^{(k)}}{\psi_{i,j}^{(k)} \psi_{0,0}^{(k)}} P_t^{(g_{it}^{(k)} + g_{t+1j}^{(k)} + g^{(k)} + g_{ij}^{(k)})} \\
\cdot \;& \exp\Big\{ - P_t[(\frac{1}{\kappa_{it}^{(k)}} + \frac{1}{\kappa_{t+1j}^{(k)}} - \frac{1}{\kappa^{(k)}} - \frac{1}{\kappa_{ij}^{(k)}})\mu_t^2 - 2(\frac{z_{it}^{(k)}}{\kappa_{it}^{(k)}} + \frac{z_{t+1j}^{(k)}}{\kappa_{t+1j}^{(k)}} - \frac{z^{(k)}}{\kappa^{(k)}} - \frac{z_{ij}^{(k)}}{\kappa_{ij}^{(k)}})\mu_t]\Big\} \\
\cdot \;& \exp\Big\{ - P_t(\frac{1}{\lambda_{it}^{(k)}} + \frac{1}{\lambda_{t+1j}^{(k)}} - \frac{1}{\lambda^{(k)}} + \frac{1}{\lambda_{ij}^{(k)}} + \frac{(z_{it}^{(k)})^2}{\kappa_{it}^{(k)}} + \frac{(z_{t+1j}^{(k)})^2}{\kappa_{t+1j}^{(k)}} - \frac{(z^{(k)})^2}{\kappa^{(k)}} - \frac{(z_{ij}^{(k)})^2}{\kappa_{ij}^{(k)}})\Big\} \\
= \;& \frac{\psi_{i,t}^{(k)} \psi_{t+1,j}^{(k)}}{\psi_{i,j}^{(k)} \psi_{0,0}^{(k)}}.
\end{aligned}
$$

Thus

$$
\frac{f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) f(\boldsymbol{\theta}_t|\mathcal{Y}_{t+1,j})}{f(\boldsymbol{\theta}_t|s_t = k)} = \frac{\psi_{i,t}^{(k)} \psi_{t+1,j}^{(k)}}{\psi_{i,j}^{(k)} \psi_{0,0}^{(k)}} f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,j}). \tag{2.4.2}
$$

Put ((2.4.2)) into ((2.4.2)) and combine ((2.4.1)) we could get:

$$f(\boldsymbol{\theta}_t|\mathcal{Y}_T) = \sum_{k=1}^{K}(\sum_{i}^{t}\xi_{i,t}^{(k)}\sum_{l\neq k}\frac{\widetilde{q}_{lk}}{\pi_k}\eta_{t+1}^{(l)}f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,t}) + \frac{\widetilde{q}_{kk}}{\pi_k}\sum_{1\leq i\leq t\leq j\leq T}\xi_{i,t}^{(k)}\eta_{t+1,j}^{(k)}\frac{\psi_{i,t}^{(k)}\psi_{t+1,j}^{(k)}}{\psi_{i,j}^{(k)}\psi_{0,0}^{(k)}}f(\boldsymbol{\theta}_t|\mathcal{Y}_{i,j})) \tag{2.4.3}$$

The posterior distribution of $\boldsymbol{\theta}_t$ given $\mathcal{Y}_T$ can also be expressed as the following mixture of normal distributions

$$\boldsymbol{\theta}_t|\mathcal{Y}_T \sim \sum_{k=1}^{K}\sum_{1\leq i\leq t\leq j\leq T}\alpha_{ij,t}^{(k)}[\boldsymbol{\theta}_t|\mathcal{Y}_{i,j}], \tag{2.4.4}$$

in which the mixture weights $\alpha_{ij,t}^{(k)}$ are posterior probabilities explained below. Consider the event

$$C_{ij}^{(k)} = \{s_i = \cdots = s_j = k, s_i \neq s_{i-1}, s_j \neq s_{j+1}\}.$$

We can show that, for $i \leq t \leq j$, $\alpha_{ijt}^{(k)} = P(C_{ij}^{(k)}|\mathcal{Y}_n)$.

And from the derivative before, $\alpha_{ij,t}^{(k)}$ can be calculated recursively as

$$\begin{aligned} &\alpha_{ijt}^{(k)} = \alpha_{ijt}^{(k)*}\Big/D_t, \qquad D_t = \sum_{k=1}^{K}\sum_{1\leq i\leq t\leq j\leq T}\alpha_{ijt}^{(k)*}, \\ &\alpha_{ijt}^{(k)*} = \begin{cases} \xi_{i,t}^{(k)}\big(\sum_{l\neq k}\eta_{t+1}^{(l)}q_{kl}/\pi_l\big) & i\leq t=j, \\ q_{kk}\xi_{i,t}^{(k)}\eta_{t+1,j}^{(k)}\psi_{i,t}^{(k)}\psi_{t+1,j}^{(k)}\big/(\pi_k\psi_{i,j}^{(k)}\psi_{0,0}^{(k)}) & i\leq t<j. \end{cases} \end{aligned} \tag{2.4.5}$$

Therefore, the smoothing estimates of $\boldsymbol{\theta}_t$ and s_t given $\mathcal{Y}_T$ are given by

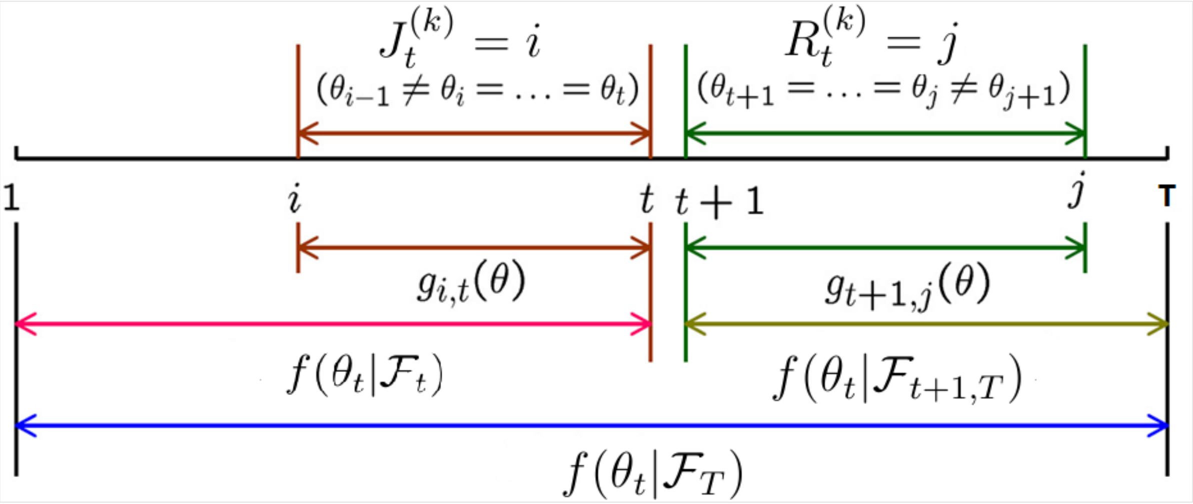
$$E(\mu_t|\mathcal{Y}_T) = \sum_{k=1}^{K}\sum_{1\leq i\leq t\leq j\leq T}\alpha_{ijt}^{(k)}z_{i,j}^{(k)}, \tag{2.4.6}$$

$$E(\sigma_t^2|\mathcal{Y}_T) = \sum_{k=1}^{K}\sum_{1\leq i\leq t\leq j\leq T}\frac{1}{2}(g_{ij}^{k}-1)^{-1}(\lambda_{ij}^{k})^{-1}, \tag{2.4.7}$$

$$P(s_t = k|\mathcal{Y}_T) = \sum_{1\leq i\leq t\leq j\leq T} \alpha_{ijt}^{(k)}. \tag{2.4.8}$$

Similarly, we use the Figure 2.4 illustrates the structure of our algorithm. At position t, we assume a possible change point to state k occurs at location i before t and at location j after t. We then derive the posterior probability density function $g_{i,t}(\theta)$ and $g_{t+1,j}(\theta)$. Moreover, the forward and backward filter at location t can be calculated based on all such g that containing all possible i and j, which are represented by $f(\theta_t|\mathcal{F}_t)$ and $f(\theta_t|\mathcal{F}_{t+1,T})$ in the figure. By combining them together we could generate the posterior distribution $f(\theta_t|\mathcal{F}_T)$at t given the whole information of the sequence.

Figure 2.4: Illustration: The structure of the algorithm.


$J_t^{(k)} = i$
$(\theta_{i-1} \neq \theta_i = \ldots = \theta_t)$
$R_t^{(k)} = j$
$(\theta_{t+1} = \ldots = \theta_j \neq \theta_{j+1})$
1
i
t
$t+1$
j
T
$g_{i,t}(\theta)$
$g_{t+1,j}(\theta)$
$f(\theta_t|\mathcal{F}_t)$
$f(\theta_t|\mathcal{F}_{t+1,T})$
$f(\theta_t|\mathcal{F}_T)$

One concern here is that, since (2.4.7) are represented as K mixtures of mixtures of normals, it is questionable whether the smoothing formula could differentiate the values of $\boldsymbol{\theta}_t$ when K states are close to each other. Such identification issue is closed related to the choice of appropriate hyper-parameters.

2.5 Bounded Complexity Mixture (BCMIX) Approximation

Although the Bayes filter (2.2.2) uses a recursive updating formula (2.2.4) for the weights $\xi_{i,t}^{(k)}$ $(1 \leq i \leq t, 1 \leq k \leq K)$, the number of weights increases dramatically with t, resulting in rapidly increasing computational complexity and memory requirements in estimating $\boldsymbol{\theta}_t$ as t keeps increasing. To address the issue of computational efficiency, we follow Xing et al. (2011) and consider a *bounded complexity mixture* (BCMIX) approximation procedure with much lower computational complexity yet comparable to the Bayes estimates in statistical efficiency. The idea of BCMIX approximation is to keep only a fixed number M of weights at every stage t, in particular, the most recent m $(1 \leq m < M)$ weights $\xi_{i,t}^{(k)}$ (with $t-m < i \leq t$) and the largest $M - m$ of the remaining weights.

Denote $\mathcal{K}_{t-1}^{(k)}$ the set of induces i for which $\xi_{i,t-1}^{(k)}$ in (2.2.4) is kept at stage $t-1$ for regime k. Note that there are at most M induces in $\mathcal{K}_{t-1}^{(k)}$ and $\mathcal{K}_{t-1}^{(k)} \supset \{t-1, \cdots, t-m\}$. When a new observation arrives at time t, we still define $\xi_{i,t}^{(k)*}$ by (2.2.4) for $i \in \{t\} \cup \mathcal{K}_{t-1}^{(k)}$ and denote i_t the index not belonging to the most recent m stages, $\{t, t-1, \ldots, t-m+1\}$, such that

$$\xi_{i_t,t}^{(k)*} = \min\{\xi_{i,t}^{(k)*} : i \in \mathcal{K}_{t-1}^{(k)} \quad \text{and} \quad i \leq t-m\}, \tag{2.5.1}$$

choosing $i_t^{(k)}$ to be the one farthest from t if the minimizing set in (2.5.1) has more than one element. Define $\mathcal{K}_t^{(k)} = \{t\} \cup (\mathcal{K}_{t-1}^{(k)} - \{i_t^{(k)}\})$, and then

$$\xi_{i,t}^{(k)} = \left(\xi_{i,t}^{(k)*} \Big/ \sum_{j \in \mathcal{K}_t^{(k)}} \xi_{j,t}^{(k)*}\right), \qquad i \in \mathcal{K}_t^{(k)}, \tag{2.5.2}$$

yields a BCMIX approximation to the forward filter.

Similarly, to obtain a BCMIX approximation to the backward filter (2.3.3), let $\widetilde{\mathcal{K}}_{t+1}^{(k)}$ denote the set of indices j for which $\eta_{j,t+1}^{(k)}$ in (2.3.3) is kept at stage $t+1$ for regime k; thus,

$\widetilde{\mathcal{K}}_{t+1}^{(k)} \supset \{t+1, \cdots, t+m\}$. At time t, define $\eta_{j,t}^{(k)}$ by (2.3.3) for $j \in \{t\} \cup \mathcal{K}_{t+1}^{(k)}$ and let j_t be the index not belonging to the most recent m stages, $\{t, t+1, \cdots, t+m-1\}$ such that

$$\eta_{j_t,t}^{(k)*} = \min\{\eta_{j,t}^{(k)*} : j \in \widetilde{\mathcal{K}}_{t+1}^{(k)} \quad \text{and} \quad j \geq t+m\}, \tag{2.5.3}$$

choosing $j_t^{(k)}$ to be the one farthest from t if the minimizing set in (2.5.3) has more than one element. Define $\widetilde{\mathcal{K}}_t^{(k)} = \{t\} \cup (\mathcal{K}_{t+1}^{(k)} - \{i_t^{(k)}\})$, and then

$$\eta_{j,t}^{(k)} = \Big(\eta_{j,t}^{(k)*} \Big/ \sum_{j \in \widetilde{\mathcal{K}}_t^{(k)}} \eta_{j,t}^{(k)*}\Big), \qquad j \in \widetilde{\mathcal{K}}_t^{(k)}, \tag{2.5.4}$$

yields a BCMIX approximation to the backward filter.

For the smoothing estimate $E(\boldsymbol{\theta}_t|\mathcal{Y}_T)$ and its associated posterior distribution, we construct BCMIX approximations by combining the preceding forward and backward BCMIX filters with index sets $\mathcal{K}_t^{(k)}$ and $\widetilde{\mathcal{K}}_{t+1}^{(k)}$, respectively, at time t. Then the BCMIX approximations to (2.4.5) are given as

$$\widetilde{\alpha}_{ijt} = \alpha_{ijt}^{*} / \widetilde{D}_t, \qquad \widetilde{D}_t = \sum_{i \in \mathcal{K}_t^{(k)}, j \in \{t\} \cup \widetilde{\mathcal{K}}_{t+1}^{(k)}} \alpha_{ijt}^{*},$$

$$\alpha_{ijt}^{(k)*} = \begin{cases} \xi_{i,t}^{(k)} \big(\sum_{l \neq k} \eta_{t+1}^{(l)} q_{kl} / \pi_l \big) & i \in \mathcal{K}_t^{(k)}, \\ q_{kk} \xi_{i,t}^{(k)} \eta_{t+1,j}^{(k)} \psi_{i,t}^{(k)} \psi_{t+1,j}^{(k)} / (\pi_k \psi_{i,j}^{(k)} \psi_{0,0}^{(k)}) & i \in \mathcal{K}_t^{(k)}, j \in \{t\} \cup \widetilde{\mathcal{K}}_{t+1}^{(k)}. \end{cases}$$

Therefore, the BCMIX smoother for $\boldsymbol{\theta}_t$ and s_t given $\mathcal{Y}_T$ are expressed as

$$E(\boldsymbol{\theta}_t|\mathcal{Y}_T) \approx \textstyle\sum_{k=1}^{K} \sum_{i \in \mathcal{K}_t^{(k)}, j \in \{t\} \cup \widetilde{\mathcal{K}}_{t+1}^{(k)}} \widetilde{\alpha}_{ijt}^{(k)} \big[\boldsymbol{\theta}_t | \mathcal{Y}_{ij}\big], \tag{2.5.5}$$

$$P(s_t = k|\mathcal{F}_T) \approx \textstyle\sum_{k=1}^{K} \sum_{i \in \mathcal{K}_t^{(k)}, j \in \{t\} \cup \widetilde{\mathcal{K}}_{t+1}^{(k)}} \widetilde{\alpha}_{ijt}^{(k)}. \tag{2.5.6}$$

The BCMIX approximation fixes the number of filters as M at each time, and keeps the m closest weights and the other $M - m$ largest weights. This greatly reduces the computational complexity $O(T^2)$ of the filter in Section 2.2 and $O(T^3)$ of the smoother in Sections 2.3 to $O(T)$. The specification of M and m are discussed in Section 3.

2.6 Estimation of Hyperparameter

The inference procedure in the above sections involve the hyper-parameters $\mathbf{\Phi} = \{Q, z^{(k)}, \kappa^{(k)}, \lambda^{(k)}, g^{(k)}; k = 1, \ldots, \}$, which can be replaced by their estimates in the empirical Bayes approach. We can show that the conditional density function of y_t given $\mathcal{Y}_{t-1}$ is expressed as

$$f(y_t|\mathcal{Y}_{t-1}) = \sum_{k=1}^{K}\sum_{i=1}^{t} \xi_{it}^{(k)*}, \tag{2.6.1}$$

where $\xi_{it}^{(k)*}$ are given by (2.2.4) and are functions of hyper-parameter vector $\mathbf{\Phi}$. Given $\mathbf{\Phi}$ and the observed data $\mathcal{Y}_n$, the log likelihood function is

$$l(\mathbf{\Phi}) = \sum_{t=1}^{n} \log f(y_t|\mathcal{Y}_{t-1}) = \sum_{t=1}^{n} \log\Big\{\sum_{k=1}^{K}\sum_{i=1}^{t} \xi_{it}^{(k)*}\Big\}, \tag{2.6.2}$$

in which $f(\cdot|\cdot)$ denotes conditional density function. Maximizing (2.6.2) over $\mathbf{\Phi}$ yields the maximum likelihood estimate $\widehat{\mathbf{\Phi}}$.

Since $\mathbf{\Phi}$ is a $[4K + K(K-1)]$-dimensional vector and the functions $\xi_{it}^{(k)}$ have to be computed recursively for $1 \leq t \leq T$, direct maximization of (2.6.2) is computationally expensive due to the curse of dimensionality. We can use the EM algorithm to exploit the much simpler structure of the log likelihood $l_c(\mathbf{\Phi})$ of the complete data $\{(y_t, s_t, \boldsymbol{\theta}_t), 1 \leq t \leq T\}$.

Recall $P_t = (2\sigma_t^2)^{-1}$.

$$
\begin{aligned}
l_c(\mathbf{\Phi}) =& \sum_{t=1}^{T} \log f(\{y_t, s_t, \boldsymbol{\theta}_t\}) \\
=& \sum_{t=1}^{T} \Big\{ \log f(y_t|\boldsymbol{\theta}_t) + \sum_{k=1}^{K} f(\boldsymbol{\theta}_t|s_t = k) 1_{\{s_t=k\}} + \sum_{k,l=1}^{K} \log(p_{kl}) 1_{\{s_{t-1}=k, s_t=l\}} \Big\} \\
=& - \sum_{t=1}^{T} \Big\{ (y_t - \mu_t)^2 P_t + \frac{1}{2} \log(\pi P_t^{-1}) \Big\} \\
& - \sum_{t=1}^{T} \sum_{k=1}^{K} \Big\{ \frac{(\mu_t - z^{(k)})^2}{\kappa^{(k)}} P_t + \frac{1}{2} \log(\pi P_t^{-1} \kappa^{(k)}) \Big\} \\
& - \sum_{t=1}^{T} \sum_{k=1}^{K} \Big\{ g^{(k)} \log(\lambda^{(k)}) - \log(\Gamma(g^{(k)})) - (g^{(k)} - 1) \log(P_t) + \frac{P_t}{\lambda^{(k)}} \Big\} 1_{\{s_t=k\}} \\
& + \sum_{t=1}^{T} \sum_{k,l=1}^{K} \log(p_{kl}) 1_{\{s_{t-1}=k, s_t=l\}}.
\end{aligned}
\tag{2.6.3}
$$

The E-step of the EM algorithm calculates $E[l_c(\mathbf{\Phi})|\mathcal{F}_t]$ which is

$$
\begin{aligned}
E[l_c(\mathbf{\Phi})|\mathcal{F}_t] =& - \sum_{t=1}^{T} E[(y_t - \mu_t)^2 P_t|\mathcal{F}_t] - \frac{1}{2} \sum_{t=1}^{T} E[\log(\pi P_t^{-1})|\mathcal{F}_t] \\
& - \sum_{t=1}^{T} \sum_{k=1}^{K} E[\frac{(\mu_t - z^{(k)})^2}{\kappa^{(k)}} P_t 1_{\{s_t=k\}}|\mathcal{F}_t] - \frac{1}{2} \sum_{t=1}^{T} \sum_{k=1}^{K} E[\log(\pi P_t^{-1} \kappa^{(k)}) 1_{\{s_t=k\}}|\mathcal{F}_t] \\
& - \sum_{t=1}^{T} \sum_{k=1}^{K} g^{(k)} \log(\lambda^{(k)}) E[1_{\{s_t=k\}}|\mathcal{F}_t] - \sum_{t=1}^{T} \sum_{k=1}^{K} \log(\Gamma(g^{(k)})) E[1_{\{s_t=k\}}|\mathcal{F}_t] \\
& + \sum_{t=1}^{T} \sum_{k=1}^{K} (g^{(k)} - 1) E[\log(P_t) 1_{\{s_t=k\}}|\mathcal{F}_t] - \sum_{t=1}^{T} \sum_{k=1}^{K} E[\frac{P_t}{\lambda^{(k)}} 1_{\{s_t=k\}}|\mathcal{F}_t] \\
& + \sum_{t=1}^{T} \sum_{k,l=1}^{K} \log(p_{kl}) E[1_{\{s_{t-1}=k, s_t=l\}}|\mathcal{F}_t].
\end{aligned}
\tag{2.6.4}
$$

It involves the computation of the conditional expectations $E[(y_t - \mu_t)^2 P_t|\mathcal{F}_t]$, $E[\log(\pi P_t^{-1})|\mathcal{F}_t]$,

$E[\frac{(\mu_t - z^{(k)})^2}{\kappa^{(k)}} P_t 1_{\{s_t=k\}}|\mathcal{F}_t]$ and $E[\log(\pi P_t^{-1}\kappa^{(k)})1_{\{s_t=k\}}|\mathcal{F}_t]$, $E[\log(P_t)1_{\{s_t=k\}}|\mathcal{F}_t]$, $E[\frac{P_t}{\lambda^{(k)}}1_{\{s_t=k\}}|\mathcal{F}_t]$, and the conditional probability $E(\mathbf{1}_{\{s_t=k\}}|\mathcal{F}_T) = P(s_t = k|\mathcal{F}_T)$ and $E(\mathbf{1}_{\{s_{t-1}=k,s_t=l\}}|\mathcal{F}_T) = P(s_{t-1} = k, s_t = l|\mathcal{F}_T)$. For the first conditional probability,

$$\begin{aligned} P(s_t = k|\mathcal{F}_T) &= \sum_{i=1}^{t} P(J_t^{(k)} = i|\mathcal{F}_T) = \sum_{i=1}^{t}\sum_{j=t}^{T} P(J_t^{(k)} = i, R_t^{(k)} = j|\mathcal{F}_T) \\ &= \sum_{1\le i\le t\le j\le T} P(C_{ij}^{(k)}|\mathcal{F}_T) = \sum_{1\le i\le t\le j\le T} \alpha_{ijt}^{(k)}. \end{aligned}$$

For the second conditional probability,

$$P(s_{t-1} = k, s_t = l|\mathcal{F}_T) = P(s_t = l|s_{t-1} = k, \mathcal{F}_T)P(s_{t-1} = k|\mathcal{F}_T). \tag{2.6.5}$$

From the above derivation, we know that

$$P(s_{t-1} = k|\mathcal{F}_T) = \sum_{1\le i\le t-1\le j\le T} \alpha_{i,j,t-1}^{(k)}.$$

Furthermore,

$$\begin{aligned} & P(s_t = j|s_{t-1} = i, \mathcal{F}_T) = \frac{P(s_t = j, s_{t-1} = i, \mathcal{F}_T)}{P(s_{t-1} = i, \mathcal{F}_T)} \\ = & \frac{P(s_t = j, s_{t-1} = i, \mathcal{F}_t|\mathcal{F}_{t+1,T})}{P(s_{t-1} = i, \mathcal{F}_t|\mathcal{F}_{t+1,T})} \\ = & \frac{P(s_{t-1} = i, \mathcal{F}_t|s_t = j)P(s_t = j|\mathcal{F}_{t+1,T})}{P(s_{t-1} = i, \mathcal{F}_t|\mathcal{F}_{t+1,T})} \\ = & \frac{P(s_{t-1} = i, \mathcal{F}_t)P(s_t = j|s_{t-1} = i, \mathcal{F}_t)}{P(s_t = j)} \frac{P(s_t = j|\mathcal{F}_{t+1,n})}{P(s_{t-1} = i, \mathcal{F}_t|\mathcal{F}_{t+1,T})} \\ = & \frac{P(s_{t-1} = i, \mathcal{F}_t)}{P(s_{t-1} = i, \mathcal{F}_t|\mathcal{F}_{t+1,T})} \frac{P(s_t = j|s_{t-1} = i, y_t)P(s_t = j|\mathcal{F}_{t+1,T})}{P(s_t = j)} \\ \propto & \frac{P(s_t = j, y_t|s_{t-1} = i)P(s_t = j|\mathcal{F}_{t+1,T})}{P(s_t = j)} \end{aligned}$$

$$
\begin{aligned}
&= \frac{f(y_t|s_t=j,s_{t-1}=i)P(s_t=j|s_{t-1}=i)\sum_{k=1}^{K}P(s_t=j,s_{t+1}=k|\mathcal{F}_{t+1,T})}{P(s_t=j)} \\
&= \frac{f(y_t|s_t=j,s_{t-1}=i)P(s_t=j|s_{t-1}=i)\sum_{k=1}^{K}P(s_t=j|s_{t+1}=k,\mathcal{F}_{t+1,T})P(s_{t+1}=k|\mathcal{F}_{t+1,T})}{P(s_t=j)} \\
&= \frac{f(y_t|s_t=j)P(s_t=j|s_{t-1}=i)\sum_{k=1}^{K}P(s_t=j|s_{t+1}=k)P(s_{t+1}=k|\mathcal{F}_{t+1,T})}{P(s_t=j)} \\
&= \frac{\psi_{t,t}^{(j)}/\psi_{0,0}^{(j)}p_{ij}\sum_{k=1}^{K}\widetilde{p}_{kj}\eta_{t+1}^{k}}{\pi_j}.
\end{aligned}
$$

Thus

$$
P(s_t=l|s_{t-1}=k,\mathcal{F}_T)=\frac{\psi_{t,t}^{(l)}/\psi_{0,0}^{(l)}p_{kl}\widetilde{P}_l'\eta_{t+1}/\pi_l}{\sum_{i=1}^{K}\left[\psi_{t,t}^{(i)}/\psi_{0,0}^{(i)}p_{ki}\widetilde{P}_i'\eta_{t+1}/\pi_i\right]}. \tag{2.6.6}
$$

Plugging (2.6.6) into (2.6.5), we have

$$
P(s_t=l,s_{t-1}=k|\mathcal{F}_T)=\frac{\psi_{t,t}^{(l)}/\psi_{0,0}^{(l)}p_{kl}\widetilde{P}_l'\eta_{t+1}/\pi_l}{\sum_{i=1}^{K}\left[\psi_{t,t}^{(i)}/\psi_{0,0}^{(i)}p_{ki}\widetilde{P}_i'\eta_{t+1}/\pi_i\right]}\sum_{1\le i\le t-1\le j\le T}\alpha_{i,j,t-1}^{(k)}.
$$

Then the conditional probabilities are:

$$
\begin{aligned}
E(\mathbf{1}_{\{s_t=k\}}|\mathcal{F}_T) &= \sum_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}, \\
E(\mathbf{1}_{\{s_{t-1}=k,s_t=l\}}|\mathcal{F}_T) &= \frac{\psi_{t,t}^{(l)}/\psi_{0,0}^{(l)}p_{kl}\widetilde{P}_l'\eta_{t+1}/\pi_l}{\sum_{i=1}^{K}\left[\psi_{t,t}^{(i)}/\psi_{0,0}^{(i)}p_{ki}\widetilde{P}_i'\eta_{t+1}/\pi_i\right]}\sum_{1\le i\le t-1\le j\le T}\alpha_{i,j,t-1}^{(k)}.
\end{aligned} \tag{2.6.7}
$$

The M-step of the EM algorithm involves calculating the partial derivatives of (2.6.4) with respect to $\mathbf{\Phi}$. Simple algebra yields the following updating formulas for $\mathbf{\Phi}$.

$$
\begin{aligned}
\widehat{q}_{kl,\text{new}} &= \frac{\sum_{t=2}^{T}P(s_{t-1}=k,s_t=l|\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}})}{\sum_{t=2}^{T}P(s_{t-1}=k|\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}})}, \\
\widehat{z}_{\text{new}}^{(k)} &= \frac{\sum_{t=1}^{T}E[\mu_tP_t1_{\{s_t=k\}}|\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}]}{\sum_{t=1}^{T}E[P_t1_{\{s_t=k\}}|\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}]},
\end{aligned}
$$

$$\begin{aligned}
\widehat{\kappa}^{(k)}_{\text{new}} &= \frac{2\sum_{t=1}^{T} E[(\mu_t - \widehat{z}^{(k)}_{\text{old}})^2 P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}]}{\sum_{t=1}^{T} E[1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}]}, \\
\widehat{\lambda}^{(k)}_{\text{new}} &= \frac{\sum_{t=1}^{T} E[P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}]}{\sum_{t=1}^{T} g^{(k)}_{\text{old}} E[1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}].}
\end{aligned} \tag{2.6.8}$$

For $\widehat{q}_{kl,\text{new}}$, (2.6.7) can be used for calculation.

For $\widehat{z}^{(k)}_{\text{new}}$, the numerator and denominator are simplified as below:

$$E[\mu_t P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}] = \sum_{t=1}^{T} \alpha^{(k)}_{ijt} E[\mu_t P_t | C^{(k)}_{ij} \mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}], \tag{2.6.9}$$

$$\begin{aligned}
& E[\mu_t P_t | C^{(k)}_{ij} \mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}] \\
={}& \int_{P_t} P_t \int_{\mu_t} \mu_t f(\mu_t | P_t, C^{(k)}_{ij}, \mathcal{F}_t) f(P_t | C^{(k)}_{ij}, \mathcal{F}_t) d\mu_t dP_t \\
={}& \int_{P_t} P_t \int_{\mu_t} \mu_t \frac{\sqrt{P_t}}{\sqrt{\pi \kappa^{(k)}_{ij}}} \exp\{-\frac{(\mu_t - z^{(k)}_{i,j})^2}{\kappa^{(k)}_{ij}} P_t\} \frac{(\lambda^{(k)}_{ij})^{-g^{(k)}_{ij}}}{\Gamma(g^{(k)}_{ij})} P_t^{g^{(k)}_{ij}-1} \exp\{-\frac{P_t}{\lambda^{(k)}_{ij}}\} d\mu_t dP_t \\
={}& \int_{P_t} P_t^{g^{(k)}_{ij}} \frac{\sqrt{P_t}}{\sqrt{\pi \kappa^{(k)}_{ij}}} \frac{(\lambda^{(k)}_{ij})^{-g^{(k)}_{ij}}}{\Gamma(g^{(k)}_{ij})} \exp\{-\frac{P_t}{\lambda^{(k)}_{ij}}\} \int_{\mu_t} \mu_t \exp\{-\frac{(\mu_t - z^{(k)}_{i,j})^2}{\kappa^{(k)}_{ij}} P_t\} d\mu_t dP_t \\
={}& \int_{P_t} P_t^{(g^{(k)}_{ij}+1)-1} \frac{(\lambda^{(k)}_{ij})^{-g^{(k)}_{ij}}}{\Gamma(g^{(k)}_{ij})} \exp\{-\frac{P_t}{\lambda^{(k)}_{ij}}\} z^{(k)}_{ij} dP_t = \lambda^{(k)}_{ij} \frac{\Gamma(g^{(k)}_{ij})}{\Gamma(g^{(k)}_{ij})} z^{(k)}_{ij} \\
={}& \lambda^{(k)}_{ij} g^{(k)}_{ij} z^{(k)}_{ij},
\end{aligned} \tag{2.6.10}$$

$$\begin{aligned}
& E[P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}] \\
={}& \sum_{1 \le i \le t \le j \le T} \alpha^{(k)}_{ijt} E[P_t 1_{\{s_t=k\}} | C^{(k)}_{ij} \mathcal{F}_T, \widehat{\mathbf{\Phi}}_{\text{old}}] \\
={}& \sum_{1 \le i \le t \le j \le T} \alpha^{(k)}_{ijt} g^{(k)}_{ijt} \lambda^{(k)}_{ijt}.
\end{aligned} \tag{2.6.11}$$

So

$$\begin{aligned}
\widehat{z}^{(k)}_{\text{new}} &= \frac{\sum\limits_{t=1}^{T} E[\mu_t P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}]}{\sum\limits_{t=1}^{T} E[P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}]} \\
&= \frac{\sum\limits_{t=1}^{T} \sum\limits_{1\le i\le t\le j\le T} \alpha^{(k)}_{ijt} \lambda^{(k)}_{ij} g^{(k)}_{ij}) z^{(k)}_{ij}}{\sum\limits_{t=1}^{T} \sum\limits_{1\le i\le t\le j\le T} \alpha^{(k)}_{ijt} \lambda^{(k)}_{ij} g^{(k)}_{ij}}. \qquad (2.6.12)
\end{aligned}$$

For $\widehat{\kappa}^{(k)}_{\text{new}}$, the posterior expectations needed in the updating formula are $E[(\mu_t - \widehat{z}^{(k)}_{\text{old}})^2 P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}]$ and $E[1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}]$. The latter one has been simplified to $\sum_{1\le i\le t\le j\le T} \alpha^{(k)}_{ijt}$ in (2.6.7).

$$\begin{aligned}
& E[(\mu_t - z^{(k)}_{\text{old}})^2 P_t 1_{\{s_t=k\}}|\mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}] \\
= & \sum_{1\le i\le t\le j\le T} \alpha^{(k)}_{ijt} E[(\mu_t^2 - 2\mu_t z^{(k)}_{\text{old}} + (z^{(k)}_{\text{old}})^2) P_t | C^{(k)}_{ij}, \mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}] \\
= & \sum_{1\le i\le t\le j\le T} \alpha^{(k)}_{ijt} \Big\{ E[(\mu_t^2 P_t | C^{(k)}_{ij}, \mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}] - 2 z^{(k)}_{\text{old}} E[\mu_t P_t | C^{(k)}_{ij}, \mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}] \\
& + (z^{(k)}_{\text{old}})^2) E[P_t | C^{(k)}_{ij}, \mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}] \Big\}, \qquad (2.6.13)
\end{aligned}$$

in which

$$\begin{aligned}
& E[(\mu_t^2 P_t | C^{(k)}_{ij}, \mathcal{F}_T, \widehat{\boldsymbol{\Phi}}_{\text{old}}] \\
= & \int_{P_t} P_t \int_{\mu_t} \mu_t^2 f(\mu_t | P_t, C^{(k)}_{ij}, \mathcal{F}_t) f(P_t | C^{(k)}_{ij}, \mathcal{F}_t) d\mu_t dP_t \\
= & \int_{P_t} P_t \int_{\mu_t} \mu_t^2 \frac{\sqrt{P_t}}{\sqrt{\pi \kappa^{(k)}_{ij}}} \exp\{-\frac{(\mu_t - z^{(k)}_{i,j})^2}{\kappa^{(k)}_{ij}} P_t\} \frac{(\lambda^{(k)}_{ij})^{-g^{(k)}_{ij}}}{\Gamma(g^{(k)}_{ij})} P_t^{g^{(k)}_{ij}-1} \exp\{-\frac{P_t}{\lambda^{(k)}_{ij}}\} d\mu_t dP_t \\
= & \int_{P_t} P_t^{g^{(k)}_{ij}} \frac{(\lambda^{(k)}_{ij})^{-g^{(k)}_{ij}}}{\Gamma(g^{(k)}_{ij})} \exp\{-\frac{P_t}{\lambda^{(k)}_{ij}}\} \int_{\mu_t} \mu_t^2 \frac{\sqrt{P_t}}{\sqrt{\pi \kappa^{(k)}_{ij}}} \exp\{-\frac{(\mu_t - z^{(k)}_{i,j})^2}{\kappa^{(k)}_{ij}} P_t\} d\mu_t dP_t
\end{aligned}$$

$$\begin{aligned}
&= \int_{P_t} P_t^{g_{ij}^{(k)}} \frac{(\lambda_{ij}^{(k)})^{-g_{ij}^{(k)}}}{\Gamma(g_{ij}^{(k)})} \exp\{-\frac{P_t}{\lambda_{ij}^{(k)}}\}\big(\frac{1}{2P_t}\kappa_{ij}^{(k)} + (z_{ij}^{(k)})^2\big)dP_t \\
&= \frac{\kappa_{ij}^{(k)}}{2} + \lambda_{ij}^{(k)} g_{ij}^{(k)} (z_{ij}^{(k)})^2. \qquad (2.6.14)
\end{aligned}$$

Using (2.6.10),

$$-2z_{\text{old}}^{(k)}E[\mu_t P_t|C_{ij}^{(k)},\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}] = -2\lambda_{ij}^{(k)}g_{ij}^{(k)}z_{\text{old}}^{(k)}z_{ij}^{(k)}, \qquad (2.6.15)$$

and

$$(z_{\text{old}}^{(k)})^2)E[P_t|C_{ij}^{(k)},\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}]] = (z_{\text{old}}^{(k)})^2\lambda_{ij}^{(k)}g_{ij}^{(k)}. \qquad (2.6.16)$$

We have

$$\begin{aligned}
\widehat{\kappa}_{\text{new}}^{(k)} &= \frac{2}{\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}} \sum_{1\le i\le t\le j\le T} \alpha_{ijt}^{(k)}\Big\{E[(\mu_t^2P_t|C_{ij}^{(k)},\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}] - 2z_{\text{old}}^{(k)}E[\mu_tP_t|C_{ij}^{(k)},\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}] \\
&\quad +(z_{\text{old}}^{(k)})^2)E[P_t|C_{ij}^{(k)},\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}]\Big\} \\
&= \frac{2}{\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}} \sum_{1\le i\le t\le j\le T} \alpha_{ijt}^{(k)}\Big\{\frac{\kappa_{ij}^{(k)}}{2} + \lambda_{ij}^{(k)}g_{ij}^{(k)}(z_{ij}^{(k)})^2 - 2\lambda_{ij}^{(k)}g_{ij}^{(k)}z_{\text{old}}^{(k)}z_{ij}^{(k)} + (z_{\text{old}}^{(k)})^2\lambda_{ij}^{(k)}g_{ij}^{(k)}\Big\} \\
&= \frac{2\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}\lambda_{ij}^{(k)}g_{ij}^{(k)}(z_{ij}^{(k)} - z_{\text{old}}^{(k)})^2}{\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}} + \frac{\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}\kappa_{ij}^{(k)}}{\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}}. \qquad (2.6.17)
\end{aligned}$$

For $\widehat{\lambda}_{\text{new}}^{(k)}$, the two posterior expectations are calculated in (2.6.11) and (2.6.7). So

$$\begin{aligned}
\widehat{\lambda}_{\text{new}}^{(k)} &= \frac{\sum\limits_{t=1}^{T} E[P_t 1_{\{s_t=k\}}|\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}}{\sum\limits_{t=1}^{T} g_{\text{old}}^{(k)}E[1_{\{s_t=k\}}|\mathcal{F}_T,\widehat{\mathbf{\Phi}}_{\text{old}}} \\
&= \frac{\sum\limits_{t=1}^{T}\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}g_{ijt}^{(k)}\lambda_{ijt}^{(k)}}{\sum\limits_{t=1}^{T} g_{\text{old}}^{(k)}\sum\limits_{1\le i\le t\le j\le T}\alpha_{ijt}^{(k)}}. \qquad (2.6.18)
\end{aligned}$$

The iteration scheme (2.6.8) is carried out until convergence or until some prescribed upper bound on the number of iterations is reached.

To speed up the computations involved in the EM algorithm, one can use the BCMIX approximations instead of the full recursions to determine the items (2.6.9)-(2.6.14). Our simulation studies shows that the EM procedure converge vary fast.

2.7 Implementation

We have shown the posterior distribution of $\boldsymbol{\theta}_t$, given the whole data information, is mixture of distributions. In this section, we describe in detail how to implement the algorithms, presenting explicit formulas. Let us start with a glance of Bayes algorithm.

Step 1 Calculating $\boldsymbol{V}_{i,j}^{(k)}$ and $\boldsymbol{z}_{i,j}^{(k)}$. Similar to (**??**), given $\mathcal{F}_T$ and $C_{ij}^{(k)}, i < j$ we use

$$\begin{aligned}
\kappa_{ij}^{(k)} &= \big(\frac{1}{\kappa^{(k)}} + j - i + 1\big)^{-1}, \\
z_{ij}^{(k)} &= \kappa_{it}^{(k)}\big(\frac{z^{(k)}}{\kappa^{(k)}} + \sum_{m=i}^{j} y_m\big), \\
g_{it}^{(k)} &= g^{(k)} + (t - i + 1)/2, \\
\lambda_{it}^{(k)} &= (\frac{1}{\lambda^{(k)}} + \frac{(z^{(k)})^2}{\kappa^{(k)}} + \sum_{j=i}^{t} y_j^2 - \frac{(z_{it}^{(k)})^2}{\kappa_{it}^{(k)}})^{-1}.
\end{aligned}$$

The results can be saved in two three-dimensional matrices for future calculation. More specifically, the posterior distribution of μ_t given σ_t and $\mathcal{Y}_{it}$ follows the normal distribution, and the posterior distribution of σ_t given $\mathcal{Y}_{it}$ satisfies the inverse-gamma distribution. If there is no information other than $s_t = k$ is given, the initial posterior distribution are normal distribution and inverse-gamma with initial parameters setting respectively. Using $\kappa_{i,j}^{(k)}$, $z_{i,j}^{(k)}$, $g_{ij}^{(k)}$ and $\lambda_{it}^{(k)}$ we can also calculate the conditional densities $\psi_{0,0}^{(k)}$ and $\psi_{i,j}^{(k)}$. They are also used to calculate the smoothing estimate of $\boldsymbol{\theta}_t$ by (2.4.5).

Step 2 Calculating the forward filter (2.2.4) in a recursive manner.

(A) Start with $t = 1$. According to (2.2.4), we have

$$\xi_{1,1}^{(k)} \propto \xi_{1,1}^{(k)*} = \big(\sum_{l \neq k} \xi_0^{(l)} p_{lk}\big)\psi_{0,0}^{(k)}/\psi_{1,1}^{(k)}.$$

Substitute $\xi_0^{(l)}$ for $l \neq k$ by the stationary distribution π_l, use $\psi_{1,1}^{(k)}$ and $\psi_{0,0}^{(k)}$ to calculate $\big(\sum_{l \neq k} \pi_l p_{lk}\big)\psi_{0,0}^{(k)}/\psi_{1,1}^{(k)}$, which gives the value of $\xi_{1,1}^{(k)*}$, and therefore $\xi_{1,1}^{(k)} = \frac{\xi_{1,1}^{(k)*}}{\sum_{k=1}^{K} \xi_{1,1}^{(k)*}}$.

(B) At $t > 1$, calculate $\xi_{t,t}^{(k)*} = \big(\sum_{l \neq k} \xi_{t-1}^{(l)} p_{lk}\big)\psi_{0,0}^{(k)}/\psi_{1,1}^{(k)}$ directly. Use $\xi_{i,t-1}^{(k)}$ to calculate $\xi_{i,t}^{(k)*} = p_{kk}\xi_{i,t-1}^{(k)}\psi_{i,t-1}^{(k)}/\psi_{i,t}^{(k)}$ for $i < t$. Normalize $\xi_{i,t}^{(k)*}$ by dividing $\sum_{1 \leq i \leq t} \xi_{i,t}^{(k)*}$ to get $\xi_{i,t}^{(k)}$. Keep doing (B) until $t = T$.

Step 3 Calculating the backward filter (2.3.3) in a recursive manner. The backward filter $\eta_{j,t+1}^{(k)}$ can be calculated similarly by starting with $t = T$.

Step 4 Calculating the smoothing mixture weight and the smoothing estimate (2.4.5).

Here comes a big problem of this procedure on computational complexity, which is caused by the three-dimension store matrix. With t increasing, the number of weights increase dramatically and thus requires a huge space for the storing especially for the large biological data sets. We use two smart ways to slove this problem and make our algorithm running efficiently.

The first modification is to making use of the BCMIX approximation procedure which can fix the number of weights as a constant M. The cost associated with the method is to keep the index set $\mathcal{K}_t^{(k)}$ for forward filter $\xi_{i,t}^{(k)}$ and $\widetilde{\mathcal{K}}_{t+1}^{(k)}$ for backward filter $\eta_{j,t+1}^{(k)}$. The basic procedure is similar to the preceding one with calculation of up to $M + 1$ weights for each stage t. The detailed procedure is as follows.

Step 1 Calculating $\kappa_{i,j}^{(k)}$, $z_{i,j}^{(k)}$, $g_{i,j}^{(k)}$ and $\lambda_{i,t}^{(k)}$.

Step 2 Calculating the BCMIX forward filter (2.5.2) in a recursive manner.

(A) For $1 \leq t \leq M$, use the Bayes procedure to calculate $\xi_{i,t}^{(k)*}$, $\xi_{i,t}^{(k)}$. The index set $\mathcal{K}_t^{(k)}$ at stage t is $\{1, \cdots, t\}$.

(B) At $t > M$, use new information at stage t to calculate $\psi_{t,t}^{(k)}$ and therefore $\xi_{t,t}^{(k)*} = \big(\sum_{l \neq k} \xi_{t-1}^{(l)} p_{lk}\big)\psi_{0,0}^{(k)}/\psi_{t,t}^{(k)}$. Use $\xi_{i,t-1}^{(k)}$ to calculate $\xi_{i,t}^{(k)*} = p_{kk}\xi_{i,t-1}^{(k)}\psi_{i,t-1}^{(k)}/\psi_{i,t}^{(k)}$ for $i \in \mathcal{K}_{t-1}^{(k)}$. Compare the weights in $\mathcal{K}_{t-1}^{(k)} - \{i_t^{(k)}\}$ and drop the smallest one. The remaining M weights form the new index set $\mathcal{K}_t^{(k)}$, and $\xi_{i,t}^{(k)} = \frac{\xi_{i,t}^{(k)*}}{\sum_{j \in \mathcal{K}_t^{(k)}} \xi_{j,t}^{(k)*}}$. Keep doing (B) until $t = T$, saving both the index sets and the BCMIX forward filters for future calculation.

Step 3 Calculating the BCMIX backward filter (2.5.4) in a recursive manner starting with $t = T$.

Step 4 Calculating the BCMIX smoothing mixture weight (2.5) and the smoothing estimate (2.5.5), (2.5.6).

Let us takes a second look at the Bounded Complexity Mixture (BCMIX) procedure, it shows only a small part of the huge precalculated storing matrices $\kappa_{i,j}^{(k)}$, $z_{i,j}^{(k)}$, $g_{ij}^{(k)}$ and $\lambda_{it}^{(k)}$ have been used. As a result, it wastes lots of space and time to calculate all such storing matrices. However, we do not know which matrices to use before calculating the index sets. A better idea is to calculate $\kappa_{i,j}^{(k)}$, $z_{i,j}^{(k)}$, $g_{ij}^{(k)}$ and $\lambda_{it}^{(k)}$ when we need them. Among them, the $g_{ij}^{(k)}$ can easily obtained by $g_{ij-1}^{(k)} + \frac{1}{2}$. One more challenge is that the formulas to calculate $\kappa_{i,j}^{(k)}$, $z_{i,j}^{(k)}$ and $\lambda_{it}^{(k)}$ involve matrix inversion, which will take a long time to implement. Instead of calculating $\kappa_{i,j}^{(k)}$, $z_{i,j}^{(k)}$ and $\lambda_{it}^{(k)}$ directly, we can calculate $KI_{i,j}^{(k)} := (\kappa_{i,j}^{(k)})^{-1}$, $KIZ_{i,j}^{(k)} := (\kappa_{i,j}^{(k)})^{-1} z_{i,j}^{(k)}$ and $LI_{i,j}^{(k)} := (\lambda_{it}^{(k)})^{-1}$ by the following simple recursive formulas if we know $KI_{i,j-1}^{(k)}$, $KIZ_{i,j-1}^{(k)}$

and $LI_{i,j-1}^{(k)}$, then

$$\begin{aligned} KI_{i,j}^{(k)} &= (\kappa_{i,j}^{(k)})^{-1} = \kappa(k)^{-1} + j - i + 1 = KI_{i,j-1}^{(k)} + 1, \\ KIZ_{i,j}^{(k)} &= (\kappa_{i,j}^{(k)})^{-1} \dot{z}_{ij}^{(k)} = \frac{z^{(k)}}{\kappa^{(k)}} + \sum_{j=i}^{t} y_j = KIZ_{i,j-1}^{(k)} + y_j, \\ LI_{i,j}^{(k)} &= (\lambda_{it}^{(k)})^{-1} = \frac{1}{\lambda^{(k)}} + \frac{(z^{(k)})^2}{\kappa^{(k)}} + \sum_{j=i}^{t} y_j^2 - \frac{(z_{it}^{(k)})^2}{\kappa_{i,t}^{(k)}} = LI_{i,j-1}^{(k)} + y_j^2 + \frac{(KIZ_{i,j}^{(k)})^2}{KI_{i,j}^{(k)}} - \frac{(KIZ_{i,j-1}^{(k)})^2}{KI_{i,j-1}^{(k)}}. \end{aligned} \tag{2.7.1}$$

So the BCMIX algorithm can be further simplified by adding this recursive updating feature. The detailed procedure is as follows.

Step 1 Calculating the BCMIX forward filter (2.5.2) in a recursive manner from $t = 1$. Follow Step 2 in the above BCMIX algorithm. Assume at stage $t-1$ we have finished calculating $\xi_{i,t-1}^{(k)}$ and $\mathcal{K}_{t-1}^{(k)}$, and saved all the $VI_{i,t-1}^{(k)}$ and $VIZ_{i,t-1}^{(k)}$ for $i \in \mathcal{K}_{t-1}^{(k)}$. At stage t, $KI_{i,t}^{(k)}$, $KIZ_{i,t}^{(k)}$ and $LI_{i,j}^{(k)}$ for $i \in \mathcal{K}_{t-1}^{(k)}$ can be calculated by (2.7.1). $KI_{t,t}^{(k)} = \kappa(k)^{-1} + 1$, $KIZ_{t,t}^{(k)} = \frac{z^{(k)}}{\kappa^{(k)}} + y_t$ and $LI_{t,t}^{(k)} = \frac{1}{\lambda^{(k)}} + \frac{(z^{(k)})^2}{\kappa^{(k)}} + y_t^2 - \frac{(KIZ_{t,t}^{(k)})^2}{KI_{t,t}^{(k)}}$. They are used to calculate $\psi_{i,t}^{(k)}$ by

$$\begin{aligned} \psi_{i,t}^{(k)} &= \frac{1}{\sqrt{\kappa_{it}^{(k)}}} \frac{1}{\Gamma(g_{it}^{(k)})} \left[\lambda_{it}^{(k)}\right]^{-g_{it}^{(k)}} \\ &= (KI_{i,}^{(k)})^{\frac{1}{2}} \frac{1}{\Gamma(g_{it}^{(k)})} (LI_{i,j}^{(k)})^{g_{it}^{(k)}}, \end{aligned}$$

and therefore $\xi_{i,t}^{(k)*}$ are calculated for all $i \in \{t\} \cup \mathcal{K}_{t-1}^{(k)}$. A small weight is dropped by the BCMIX rule and the remaining index set $\mathcal{K}_t^{(k)}$, $\xi_{i,t}^{(k)}$, $KI_{i,t}^{(k)}$, $KIZ_{i,t}^{(k)}$ and $LI_{i,t}^{(k)}$ are saved.

Step 2 Calculating the BCMIX backward filter (2.5.4) in a recursive manner starting with $t = T$. If we know $KI_{i-1,j}^{(k)}$ $KIZ_{i-1,j}^{(k)}$ and $LI_{i-1,j}^{(k)}$, and want to calculate $KI_{i,j}^{(k)}$, $KIZ_{i,j}^{(k)}$

and $LI_{i,j}^{(k)}$ by the recursive formulas

$$\begin{aligned}
KI_{i,j}^{(k)} &= KI_{i-1,j}^{(k)} + 1, \\
KIZ_{i,j}^{(k)} &= KIZ_{i-1,j}^{(k)} + y_i, \\
LI_{i,j}^{(k)} &= LI_{i-1,j}^{(k)} + y_i^2 + \frac{(KIZ_{i,j}^{(k)})^2}{KI_{i,j}^{(k)}} - \frac{(KIZ_{i-1,j}^{(k)})^2}{KI_{i-1,j}^{(k)}}.
\end{aligned}$$

Using these updating formulas, we can recursively calculate $KI_{t+1,j}^{(k)}$, $KIZ_{t+1,j}^{(k)}$ and $LI_{t+1,j}^{(k)}$ for $j \in \widetilde{\mathcal{K}}_{t+1}^{(k)}$ and conduct Step 3 in the above BCMIX algorithm.

Step 3 Calculating the BCMIX smoothing mixture weight $\widetilde{\alpha}_{ijt}^{(k)}$ and the smoothing estimate $\hat{\boldsymbol{\theta}}_{t|T}$. We can evaluate $KI_{i,j}^{(k)}$, $KIZ_{i,j}^{(k)}$, $g_{i,j}^{(k)}$and $LI_{i,j}^{(k)}$ for $i \in \mathcal{K}_t^{(k)}, j \in \widetilde{\mathcal{K}}_{t+1}^{(k)}$ by

$$\begin{aligned}
KI_{i,j}^{(k)} &= KI_{i,t}^{(k)} + KI_{t+1,j}^{(k)} - (\kappa^{(k)})^{-1}, \\
KIZ_{i,j}^{(k)} &= KIZ_{i,t}^{(k)} + KIZ_{t+1,j}^{(k)} - \frac{z^{(k)}}{\kappa^{(k)}}, \\
g_{i,j}^{(k)} &= g_{i,t}^{(k)} + g_{t+1,j}^{(k)} - g^{(k)}, \\
LI_{i,j}^{(k)} &= LI_{i,t}^{(k)} + LI_{t+1,j}^{(k)} - \frac{1}{\lambda^{(k)}} - \frac{(z^{(k)})^2}{\kappa^{(k)}} + \frac{(KIZ_{i,t}^{(k)})^2}{KI_{i,t}^{(k)}} + \frac{(KIZ_{t+1,j}^{(k)})^2}{KI_{i,j}^{(k)}} - \frac{(KIZ_{i,j}^{(k)})^2}{KI_{i,j}^{(k)}}.
\end{aligned} \tag{2.7.2}$$

Then we could use the items above to calculate the posterior mean, posterior variance and the posterior state probability followed the formula in the previous sections.

Chapter 3

Simulation Studies

In this chapter, we will implement intensive simulation experiments. Firstly, some general criterion are introduced, including sum of squared error (SSE), Kullback-Leibler (KL) divergence and the identification ratio (IR) of true state calling. Then we make comparisons between the Bayes estimates and BCMIX approximation procedure estimates through the Monte Carlo simulations. We will show the BCMIX is statistically and computational efficient. Afterwards, We examine the effect of BCMIX approximation by different simulation settings. The evaluations are made based on the three criterion and shows a very good performance of our segmentation model. And a quite fuzzy scenario is discussed in the end.

3.1 Comparison Criterion

There are three criterion by which we assess the performance of the estimation of parameter $\boldsymbol{\theta}_t$: the sum of squared errors, the Kullback-Leibler divergence and the L_2 errors between the true and estimated parameters. In our model, as $y_t = \mu_t + \sigma_t \epsilon_t$, the sum of squared error and L_2 errors actually is the same, which defines as below

$$SSE = \frac{1}{T}\sum_{t=1}^{T}(\mu_t - \hat{m}u_t)^2 = \sum_{t=1}^{T}(\mu_t - E(\mu_t|\mathcal{Y}_\sqcup)^\in,$$

which measures the discrepancy between the true and estimation of the mean variables. Here note in other segmentation model for regression dependent variables, the SSE is different from the L_2 errors. The Kullback-Leibler (KL) divergence is a equation of information theory or a measure in statistics (Cover and Thomas, 1991) that quantifies how close of two probability distribution. For example, we have two measure space with probability distribution $p = p_i$ and $q = q_i$, the KL divergence is defined by $KL(p||q) = \sum_i p_i log_2(\frac{p_i}{q_i})$. In our Bayesian model the KL divergence is used as a measure of the information gain of loss in changing form a prior distribution to a posterior distribution. For example, assuming the data information is considered, we could update the probability distribution of $\boldsymbol{\theta}_t$ given parameter space to a new posterior distribution which is the the probability distribution of $\boldsymbol{\theta}_t$ given both parameter space and the data information. Here in our model, the KL divergence is calculated by the formula as below:

$$KL(\boldsymbol{\theta}_t, \hat{\boldsymbol{\theta}}_t) = \frac{(\mu_t - \hat{\mu}_t)^2}{\hat{\sigma}_t^2} + \frac{\sigma_t^2}{\hat{\sigma}_t^2} - 1 - log(\frac{\sigma_t^2}{\hat{\sigma}_t^2}),$$

which measures the discrepancy between models with $\boldsymbol{\theta}_t$ and $\hat{\boldsymbol{\theta}}_t$. We use κ, the average of KL over the whole sample period, defined by

$$\kappa := \frac{1}{T} \sum_{t=1}^{T} KL(\boldsymbol{\theta}_t, \hat{\boldsymbol{\theta}}_t)$$

From the formula of our KL divergence we could find that it not only consider the difference of the mean variable but also check the ratio term of the volatility. Put it in another way, it measures more explicitly and accurately compared to the SSE when the variance also has the hidden states, thus κ should be a more appropriate criterion.

We also need to evaluate the performance of the smoothed probability $\hat{r}_{t|T}^{(k)} = P(s_t = k|\mathcal{F}_t)$ as discussed in previous chapter. We use this probability to provide assessment of the hidden state s_t belonging to regime k. However, this is not a logical variable only taking a value of 1 or 0, but a probability theoretically close to 1 or 0. When there is a transition from some regime to another one, the probability might show some fuzziness. An intuitive and simple way to make the inference on s_t is to compare the smoothed probability $\hat{r}_{t|T}^{(k)}$ with 0.5. If for any $1 \leq k \leq K$, $\hat{r}_{t|T}^{(k)} > 0.5$, we identity $s_t = k$. More specifically, to evaluate the performance of this procedure, we define an identification ratio as

$$IR := \frac{\sum_{t=1}^{T} \sum_{k=1}^{K} \mathbf{1}_{(\hat{r}_{t|T}^{(k)} > 0.5) \cap (s_t = k)}}{T},$$

where $\mathbf{1}$ denotes an indicator function, and T is the length of the sequence. If the true regime is k, and a probability reasonably close to 1, $\hat{r}_{t|T}^{(k)} > 0.5$, is obtained from the procedure, then $(\hat{r}_{t|T}^{(k)} > 0.5) \cap (s_t = k)$ is true, and the indicator function returns 1 for stage t.

3.2 Simulation 1: Comparison between Bayes and BCMIX Estimates

In last chapter, we mention that the Bayes method is quite accurate at the cost of time consuming computation and heavy burden on the memory requirement since the number of weights probability increase with t. In another word, it might cause big challenge in estimating $\boldsymbol{\theta}_t$ while the t is larger at some extent. For the biologic data nowadays, it is nearly impossible to use the Bayes method to do the estimation. BCMIX approximation procedure is much faster with less memory demand, which can carry out our model efficiently. In this section, the simulation experiment will display the comparison of the performance of the Bayes method and BCMIX in several aspects.

We use 2 states in our model, ie, $K = 2$. The values of the parameter $\boldsymbol{\theta}_t := \mu_t, \sigma_t$ depend on the hidden state s_t. In all the examples shown in this section, data are generated according to hyperparameter values:$z^{(1)} = 2.0, \kappa^{(1)} = 0.8, \lambda^{(1)} = 0.8, g^{(1)} = 2.5$; $z^{(2)} = 4.0, \kappa^{(2)} = 1.0, \lambda^{(2)} = 0.5, g^{(2)} = 1.8$ and $P = \begin{pmatrix} 0.99 & 0.01 \\ 0.01 & 0.99 \end{pmatrix}$. Furthermore, given s_t, $\boldsymbol{\theta}_t$ is a realization from a truncated distribution such that $|\mu_t| < 8$ and the ratio of variance of different state to make the series stationary. We generate $N = 500$ series, each of length $T = 1000$, and consider s_t changing over location in four scenarios:

Scenario 1. There is only one transition from regime 1 to regime 2. $s_t = 1$ for $1 \leq t \leq 200$; $s_t = 2$ for $201 \leq t \leq 1000$.

Scenario 2. There is only one transition from regime 1 to regime 2. $s_t = 1$ for $1 \leq t \leq 800$; $s_t = 2$ for $801 \leq t \leq 1000$.

Scenario 3. There are two transitions between regime 1 and regime 2. $s_t = 1$ for $1 \leq t \leq 350$; $s_t = 2$ for $351 \leq t \leq 700$; $s_t = 1$ for $701 \leq t \leq 1000$.

Scenario 4. There are three transitions between regime 1 and regime 2. $s_t = 1$ for $1 \leq t \leq 200$; $s_t = 2$ for $201 \leq t \leq 400$; $s_t = 1$ for $401 \leq t \leq 600$; $s_t = 2$ for $601 \leq t \leq 1000$.

In each scenario, we assume the true hyper-parameters are given, and compute both BCMIX and Bayes estimates. As mentioned in Section 2.7, the performance of the BCMIX procedure depends on the specification of M and m. This dependence is examined here, choosing $M = 2m$ and M =10, 20, 30 and 40. Furthermore, to access the performance of both methods, we consider a simple benchmark under the condition that the hidden state of each position is known in advance. so that the Bayes estimates of $\boldsymbol{\theta}_t$ between two transitions are given by the standard Bayesian formulas (Section 2.7 of Box and Tiao (1973)). It is called "fictitious Bayes" estimate. Tables 3.1 and 3.2 compare fictitious Bayes estimate (fBayes), Bayes estimate (Bayes), and the BCMIX estimate (BCMIX) in terms of the SSE, κ respectively.

The first three columns in the Table 3.1 display that the fictitious Bayes estimates

Table 3.1: Performance of Sum of squared errors (SSE) for fBayes, Bayes and BCMIX estimates. Standard errors are given in parentheses below the estimates.

			BCMIX			
Scenarios	*fBayes*	*Bayes*	(10,5)	(20,10)	(30,15)	(40,20)
Scenario 1	0.0023	0.0031	0.0025	0.0025	0.0025	0.0025
	(1.28E-04)	(1.63E-04)	(1.55E-04)	(1.55E-04)	(1.55E-04)	(1.55E-04)
Scenario 2	0.0024	0.0031	0.0024	0.0024	0.0025	0.0025
	(1.26E-04)	(1.52E-04)	(1.44E-04)	(1.44E-04)	(1.44E-04)	(1.44E-04)
Scenario 3	0.0027	0.0035	0.0030	0.0030	0.0030	0.0030
	(1.29E-04)	(1.69E-04)	(1.63E-04)	(1.63E-04)	(1.63E-04)	(1.63E-04)
Scenario 4	0.0027	0.0051	0.0045	0.0045	0.0045	0.0045
	(1.29E-04)	(2.25E-04)	(2.21E-04)	(2.21E-04)	(2.21E-04)	(2.21E-04)

has the smallest SSE while BCMIX(10,5) is general better than Bayes estimates. As we known for this variance-varying model, SSE is not an perfect criterion for evaluating the performance of different procedure. Yet, we can still tell that these three method don't have fundamental difference no matter the SSE or the standard deviation. Moreover, the relative differences between BCMIX(10,5) and fBayes is less than 2% in all scenarios, which demonstrates BCMIX has very promising results in segmentation. On the other hand we surprisingly discover that BCMIX(10,5) is obvious better than Bayes results regardless of its much better time complexity in the big data sets. The last four columns in Table 3.1 show that the average SSE over 500 sequences changes very slight with respect to the different values of M and m. In last chapter we knew that the approximation should improve as M and m become larger since more filters are kept in estimating. However, at least, from this table, we couldn't find any necessary clear trend since in each scenario the SSE is almost the same. In summary, the estimation results do not change dramatically when M and m are getting larger, which can demonstrate the BCMIX procedure is a very robust for the model with enough accuracy compared with the Bayes estimates.

Table 3.2: Performance of Kullback-Leibler divergence ($10^3\kappa$) for fBayes, Bayes and BCMIX estimates. Standard errors are given in parentheses below the estimates.

			BCMIX			
Scenarios	*fBayes*	*Bayes*	(10,5)	(20,10)	(30,15)	(40,20)
Scenario 1	3.973 (1.23E-04)	5.461 (2.10E-04)	4.269 (2.01E-04)	4.268 (2.01E-04)	4.268 (2.01E-04)	4.268 (2.01E-04)
Scenario 2	4.027 (1.25E-04)	5.394 (1.78E-04)	4.200 (1.68E-04)	4.199 (1.68E-04)	4.198 (1.68E-04)	4.198 (1.68E-04)
Scenario 3	5.882 (1.43E-04)	7.434 (2.80E-04)	6.245 (2.75E-04)	6.244 (2.75E-04)	6.244 (2.75E-04)	6.242 (2.75E-04)
Scenario 4	7.883 (1.43E-04)	9.557 (2.77E-04)	8.365 (2.72E-04)	8.365 (2.72E-04)	8.364 (2.72E-04)	8.362 (2.72E-04)

As mentioned in last section, Kullback-Leibler divergence is a more accurate measure of the difference between the true and estimated parameters. Table 3.2 shows the comparison in terms of Kullback-Leibler divergence for the four scenarios. Again, as the benchmark, the fictitious Bayes estimates give out the best results than the other ones. We can find the same trend of the first three columns as the previous table: BCMIX(10,5) is less accurate than fBayes and the relative difference around 6% while BCMIX is significantly better than the Bayes estimates. Moreover, the values in each scenario with different M and m have a slight difference. For example, comparing the results of BCMIX(20,10) and BCMIX(40,20), the relative differences of these two are just less than 1%. In short, from this table, we can conclude that BCMIX is a reliable procedure as accurate as the benchmark. And the larger M and m, the smaller KL divergence, although the difference is very subtle. Considering the computation time of BCMIX(20,10) is one fourth of BCMIX(40,20), it suggests that $M = 20$ and $m = 10$ is an ideal choice.

Table 3.3 compares the Bayes and the BCMIX estimates in terms of identification ratio (IR). The first two columns show that both the Bayes and BCMIX methods give an average

Table 3.3: Performance of Identification Ratio (IR*100%) for Bayes and BCMIX estimates. Standard errors are given in parentheses below the estimates.

		BCMIX			
Scenarios	*Bayes*	(10,5)	(20,10)	(30,15)	(40,20)
Scenario 1	99.9892	99.9992	99.9992	99.9992	99.9992
	(2.023588e-05)	(3.987958e-06)	(3.987958e-06)	(3.987958e-06)	(3.987958e-06)
Scenario 2	99.9898	99.9994	99.9994	99.9994	99.9994
	(1.938927e-05)	(4.46855e-06)	(4.46855e-06)	(4.46855e-06)	(4.46855e-06)
Scenario 3	99.9894	99.9999	99.9999	99.9999	99.9999
	(1.95523e-05)	(4.454175e-06)	(4.454175e-06)	(4.454175e-06)	(4.454175e-06)
Scenario 4	99.9884	99.9984	99.9984	99.9984	99.9984
	(2.054098e-05)	(5.617037e-06)	(5.617037e-06)	(5.617037e-06)	(5.617037e-06)

IR larger than 99%, with slight absolute differences of less than 0.1%. And in general, BCMIX has better identification ratio than Bayes method about 0.01%. The standard deviation shows that BCMIX has less variation indicating its better stability. The last four columns in Table 3.3 don't have any changing since the IR values have already reach a very high level even under the combination $M = 10$ and $m = 5$. This table further demonstrates the effectiveness and robustness of BCMIX. As a results, we will implement much more simulation works using a more complicated model in the next section. We will display more scenarios, longer sequence and thus the Bayes model has too much burden on computation and become "mission impossible". Here as learning from the three table, we will take the combination $M = 20$ and $m = 10$ in the next step simulation to estimate the parameters and make the inference.

In Table **??**, the fBayes just increase from 0.0024 to 0.0027 between Scenario 2 to 4, while the Bayes and BCMIX has an obvious increase from 0.0031 to 0.0051 and from 0.0024 to 0.0045, respectively. More significant differences are shown in Table 3.2. In Scenario 3, $10^3\kappa$ of fBayes estimate, Bayes estimates and BCMIX all have a significant increase from

4.027 to 7.833, 5.394 to 9.557 and 4.200 to 8.365, respectively. However, as shown in Table 3.3, the methods can identify the correct hidden state and regime more efficiently when there are more transitions in Scenario 3 but Scenario 4 shows it might not be always the case. The associated standard errors become a little larger in the last two scenarios. However the 0.0001% doesn't not really influence its robustness. We still believe the Bayes and BCMIX procedures are robust and efficient to make inference on regimes when number of change points increases.

To visualize the simulation results, Let us investigate the following figures. Figure 3.1 shows a randomly selected simulation path y_t in each scenario. From the figure we find some obvious changing patterns in each series. For example, in the second plot (Scenario 1), the points after the position 200 general larger than the ones before 200, indicating a change in the pattern around $t = 200$. In Scenario 4, since there are more transitions, it is clear there are some change points within the series but we cannot identify their precise position. Figure 3.2 shows the true μ_t and estimated $\hat{\mu}_{t|T}$ (the posterior mean) of the corresponding series. Before we analyze the estimates, let us observe the true parameters in different states to have a better understanding of the model. In the last plot (Scenario 4), there are two regimes and three transitions from regime 1 to 2, then back to regime 1, and then to regime 2 again. However, values of μ_t within each regime are not the same. For regime 1, $\mu_t = 1.14$ before the first transition, and $\mu_t = 1.59$ between the second and third transition. For regime 2, $\mu_t = 4.31$ between the first and second transitions, and $\mu_t = 4.12$ after the third transition. The similar situations appear in the Figure3.3. For regime 1, $\sigma_t^2 = 0.42$ before the first transition, and $\sigma_t^2 = 0.56$ between the second and third transitions. For regime 2, $\sigma_t^2 = 2.10$ between the first and second transition, and $\sigma_t^2 = 2.2$ after the third transition.This is the new feature of our model as specified in the third assumption. Different from the classic HMM model in which μ_t or σ_t^2 should be a constant within each state, in our model both are random variables following some distribution within each state.

Now let us look at the estimation results. In Figure 3.2, we cannot tell the difference between Bayes estimate (dotted line), BCMIX estimate (dashed line) and the real value (solid line). In the first two scenarios (top two plots), the estimated parameters are very close to the true μ_t. In the last two scenarios (bottom two plots) there are some minor difference between the BCMIX/Bayes estimates and the real value, but there are no difference between BCMIX and Bayes. Similar cases in Figure3.3, there are some obvious but minor differences between the BCMIX/Bayes estimates and the real value, especially in the state 2 which has larger σ_t^2. But the differences between BCMIX and Bayes are very small even both is exact the same. These two figures demonstrate our model can correctly estimates the mean variable while less accurate estimates are generated for the variance variable.

Figure 3.4 shows the true and estimated $P(s_t = 1)$ (posterior state probability) of each series. Specifically, if the true state is 1, the true probability of $P(s_t = 1) = 1$; if the true state is 2, the true probability of $P(s_t = 1) = 0$. There are two states in our simulation model, hence $P(s_t = 2) = 1 - P(s_t = 1)$ for $1 \leq t \leq T$. For convenient, we only show the probability of state 1. Very slight differences between the estimated probabilities in Bayesian procedure (dotted line) and BCMIX procedure (dashed line) can be observed. When there are enough observations between two consecutive transitions, as in the first three plots (Scenarios 1, 2 and 3), both procedures capture the transitions very quickly. But when there are more frequent transitions, as in the last plot, the estimated probabilities show some fuzziness around transitions. That is why we use $P(s_t = 1) > 0.5$ to make inference on the unknown regime.

Figure 3.1: A selected series y_t in Scenarios 1 (top-left), 2 (top-right), 3 (bottom-left) and 4 (bottom-right).

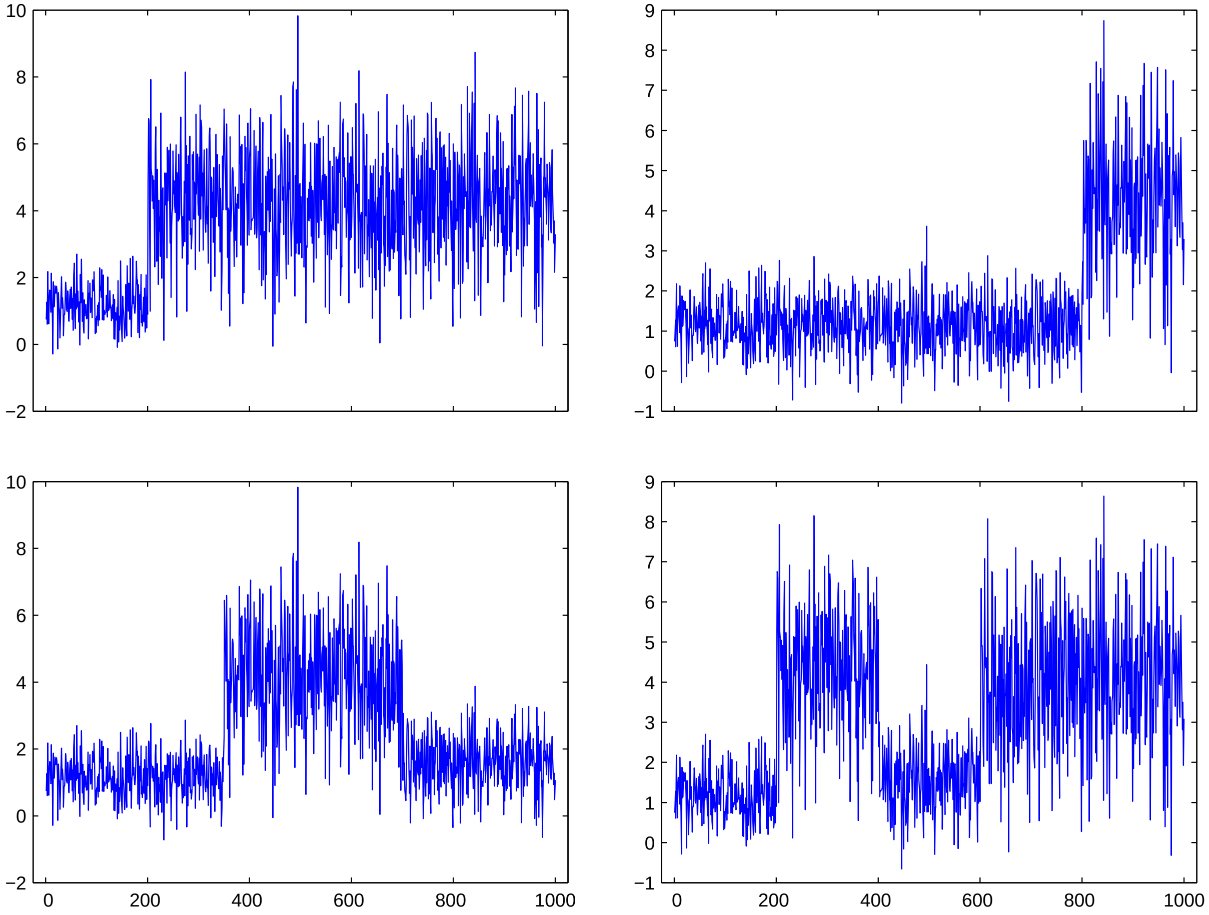


Figure 3.2: Bayes estimates (dotted line), BCMIX estimates (dashed line) of $\hat{\mu}_{t|T}$ and true μ_t (solid line) of the selected series in Scenarios 1 (top-left), 2 (top-right), 3 (bottom-left) and 4 (bottom-right)

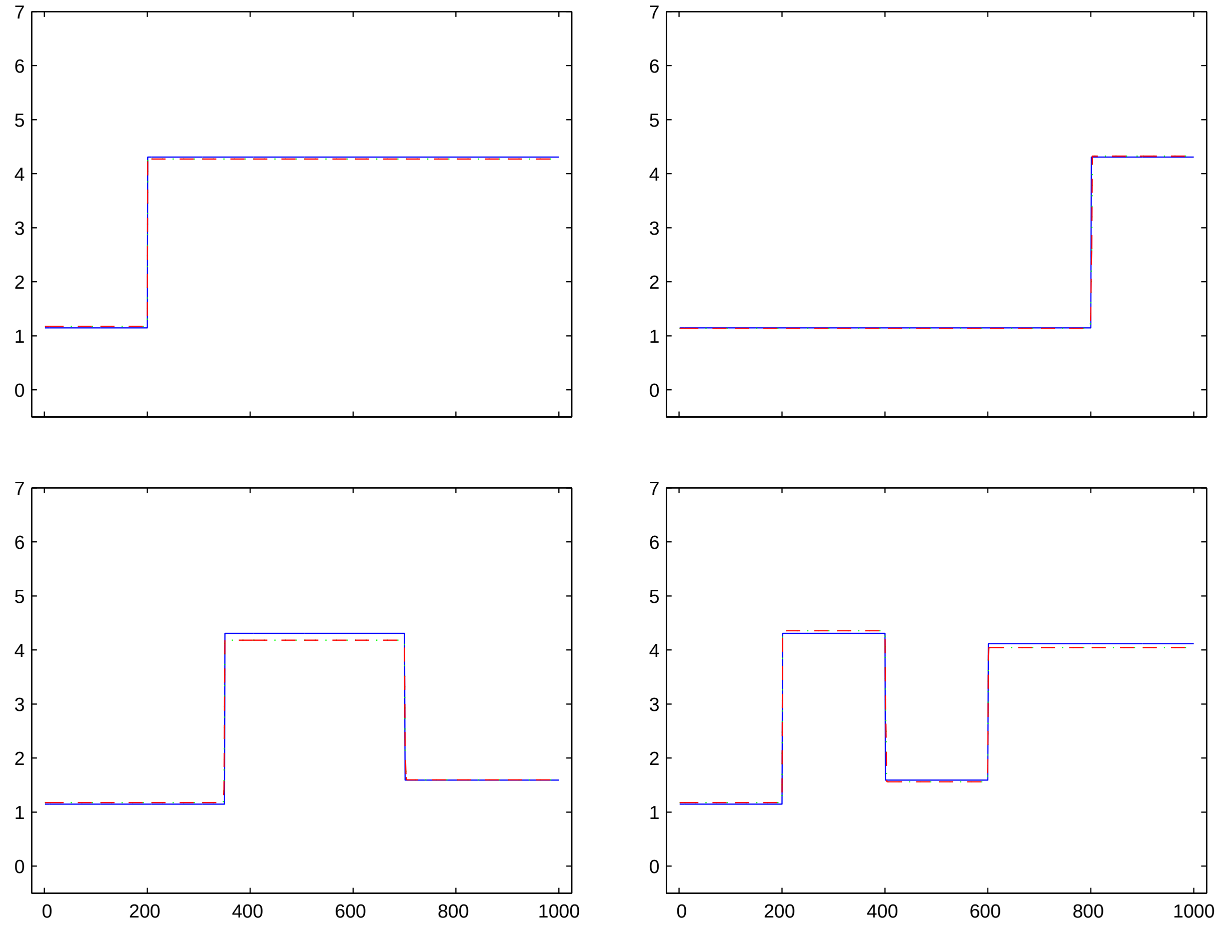


Figure 3.3: Bayes estimates (dotted line), BCMIX estimates (dashed line) of $\hat{\sigma}^2_{t|T}$ and true σ^2_t (solid line) of the selected series in Scenarios 1 (top-left), 2 (top-right), 3 (bottom-left) and 4 (bottom-right)

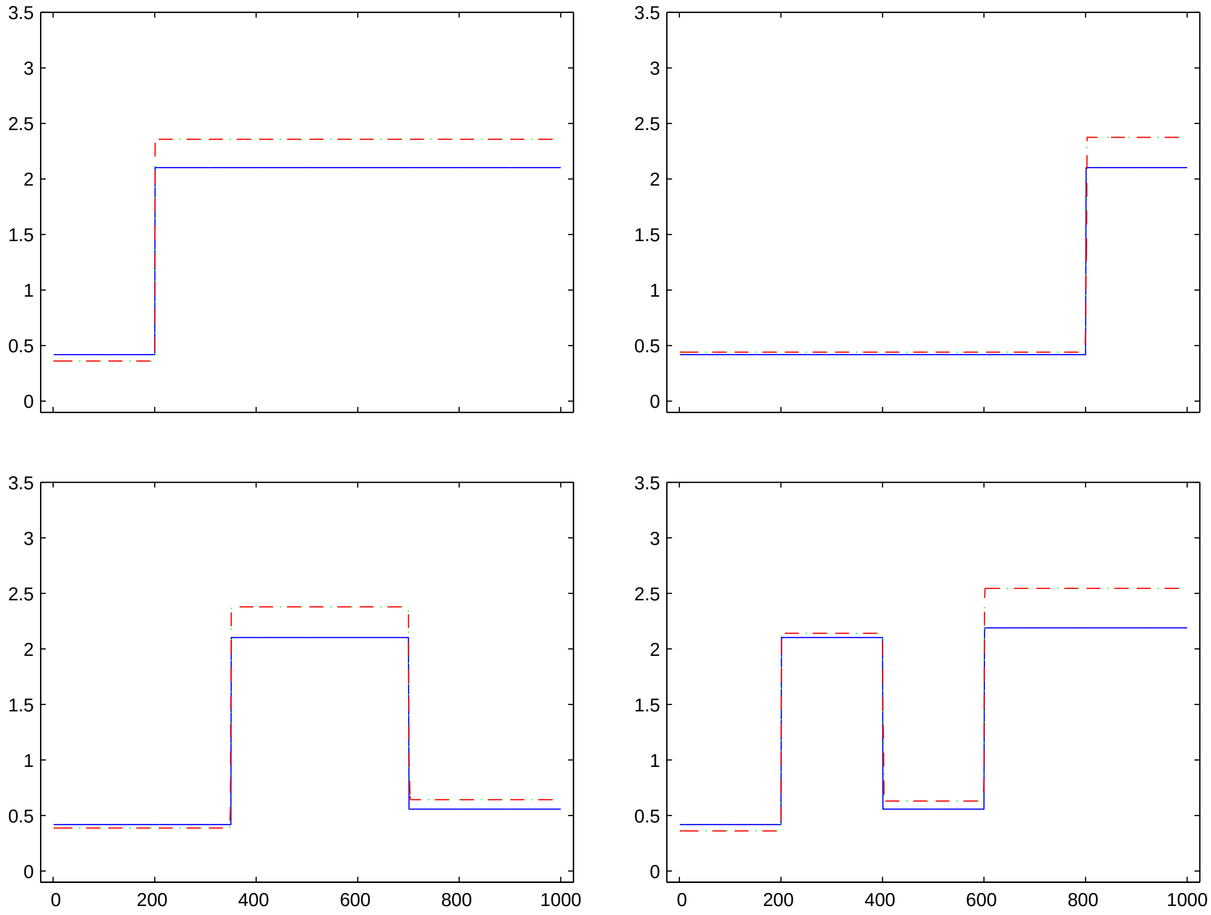
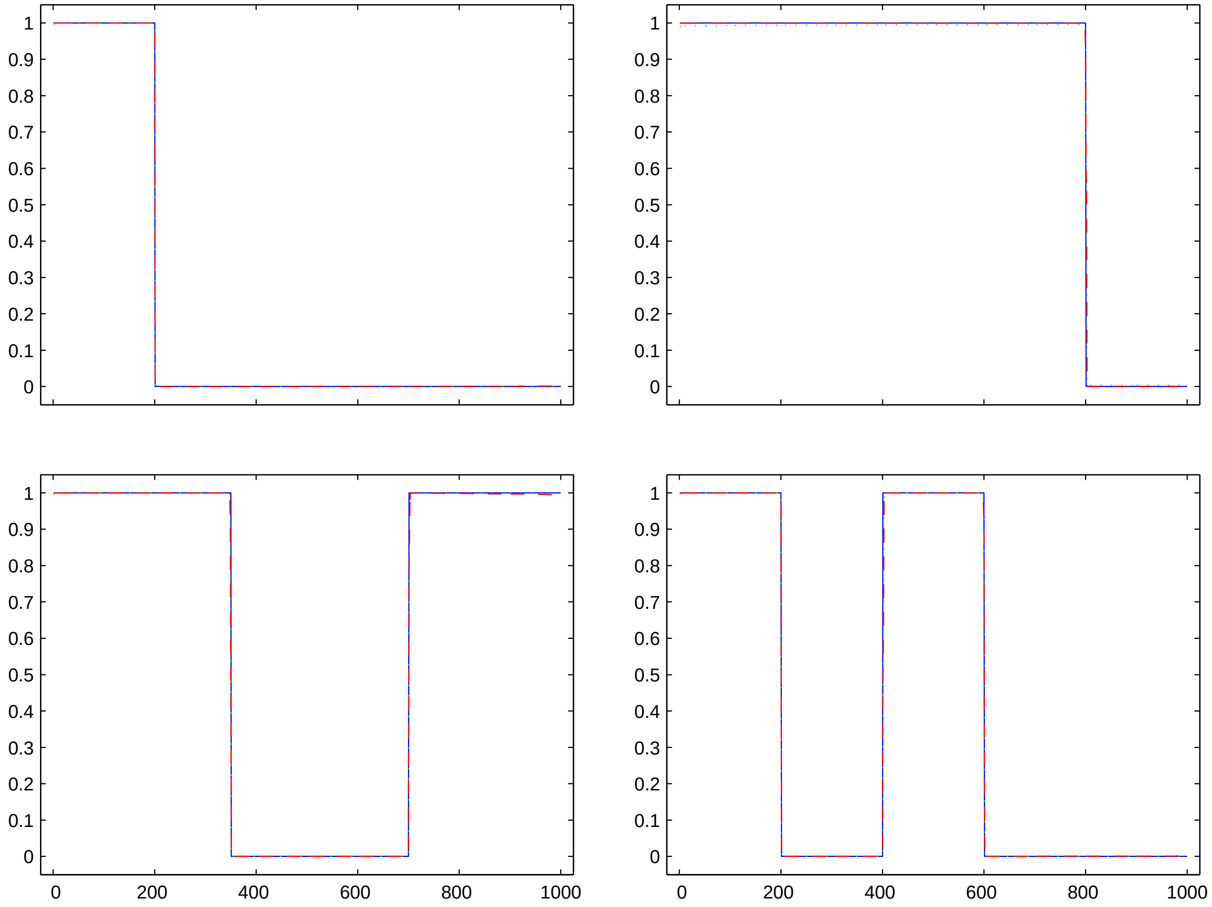


Figure 3.4: Bayes estimates (dotted line), BCMIX estimates (dashed line) of $\hat{r}_{t|T}^{(1)}$ and true $P(s_t = 1)$ (solid line) of the selected series in Scenarios 1 (top-left), 2 (top-right), 3 (bottom-left) and 4 (bottom-right)



3.3 Simulation 2: Large simulation with different simulation settings

In this section, we will examine the effects of different simulation settings on the estimates. In the experiment, we will only use the BCMIX procedure for large scale simulation studies with specific M and m. We assume we have the exact same model described in chapter 2. Again there are two states, $K = 2$. We still use the same parameter settings with the previous simulation study: $z^{(1)} = 2.0, \kappa^{(1)} = 0.8, \lambda^{(1)} = 0.8, g^{(1)} = 2.5$, $z^{(2)} = 4.0, \kappa^{(2)} = 1.0, \lambda^{(2)} = 0.5, g^{(2)} = 1.8$. And in order to make the series stationary we did the same truncated procedure. The transition matrix is $P = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}$, which has the following settings:

Scenario 1. $(p, q) = (0.001, 0.001)$.

Scenario 2. $(p, q) = (0.002, 0.001)$.

Scenario 3. $(p, q) = (0.002, 0.002)$.

Scenario 4. $(p, q) = (0.004, 0.001)$.

Scenario 5. $(p, q) = (0.004, 0.002)$.

Scenario 6. $(p, q) = (0.008, 0.004)$.

Scenario 7. $(p, q) = (0.008, 0.008)$.

Scenario 8. $(p, q) = (0.016, 0.008)$.

Scenario 9. $(p, q) = (0.016, 0.016)$.

Let $N = 500$ and T takes the values of 3000, 4000, 5000, 6000, 7000 and 8000 for each scenario. In each scenario, we give the hyper-parameters some initial value as below: $z^{(1)} = 1.5, \kappa^{(1)} = 1.0, \lambda^{(1)} = 1.0, g^{(1)} = 2.5$, $z^{(2)} = 3.5, \kappa^{(2)} = 1.2, \lambda^{(2)} = 0.4, g^{(2)} = 1.8$ and $(p, q) = (0.01, 0.01)$. The hyper-parameters are estimated by the EM algorithm described in last chapter until convergence. Then the estimates are computed. The BCMIX procedure with $M = 20$ and $m = 10$ is adopted to estimate the smoothing parameters and give inference on the states. Tables 3.4, 3.5 compare the estimates in different scenarios in terms of the SSE and κ, respectively. Each table has 6 columns and 9 rows, in which every “cell” is the result of 500 times simulation for that specific scenario.

Let’s first take an overall view of these the Tables 3.4 and 3.5 column by column. Within each column, the sample size T is the same, but the value of p and q is different. This infers that the transition matrix $P = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}$ is different and thus, the mean number of the change point is different for each row. Although we cannot guarantee the number of transitions are the same for each pair of p and q since the they are generated by the Markov chain, we knew the more transitions should be expected once the p and q become larger from top to bottom in these tables.

Table 3.4 shows two different trend in two direction. On one hand, the larger are p and q, the larger is the SSE. We can explain it as the big differences between μ_t and $\hat{\mu}_t$ is more likely to happen around the transitions. Once the number of the transitions become larger, the SSE has more chance to increase. Here note the cell of $p, q = 0.002, 0.002$ and the cell of $p, q = 0.004, 0.001$, it is unclear which has more transitions. Yet, from the SSE, it seems the former one should has more change points. One the other hand, the SSE becomes smaller once the sequence become longer. Under the condition that having same probability of having transitions, the longer series or more data information should make the prediction of the posterior mean more accurately. But from the table we could tell the differences between columns are quite small. For example, when $p, q = 0.016, 0.016$ there is no distinction between $T = 7000$ and $T = 8000$. In short, we can tell BCMIX has very good performance on all of these p, q settings since the largest SSE is only about 0.56%.

Table 3.5 shows a similar trend: the larger are p and q, the larger are κ. For example,

Table 3.4: Performance of Sum of squared errors (SSE) for BCMIX estimates. Standard errors are given in parentheses.

Scenarios	$T = 3000$	$T = 4000$	$T = 5000$	$T = 6000$	$T = 7000$	$T = 8000$
$p = 0.001$ $q = 0.001$	0.00179 (7.10E-05)	0.00167 (6.623E-05)	0.00160 (5.87E-05)	0.00166 (6.50E-05)	0.00150 (5.32E-05)	0.00163 (6.96E-05)
$p = 0.002$ $q = 0.001$	0.00223 (9.05E-05)	0.00212 (7.589E-05)	0.00195 (6.45E-05)	0.00198 (7.04E-05)	0.00179 (6.33E-05)	0.00191 (7.52E-05)
$p = 0.002$ $q = 0.002$	0.00263 (9.93E-05)	0.00255 (8.421E-05)	0.00144 (5.34E-05)	0.00240 (7.89E-05)	0.0022 (7.03E-05)	0.00232 (8.35E-05)
$p = 0.004$ $q = 0.001$	0.00242 (9.47E-05)	0.00230 (7.860E-05)	0.00217 (7.02E-05)	0.00222 (7.98E-05)	0.00202 (6.89E-05)	0.00210 (7.62E-05)
$p = 0.004$ $q = 0.002$	0.00301 (1.08E-04)	0.00289 (9.195E-05)	0.00286 (8.82E-05)	0.00272 (8.44E-05)	0.00262 (8.63E-05)	0.00264 (8.96E-05)
$p = 0.008$ $q = 0.004$	0.00410 (1.39E-04)	0.00396 (1.193E-04)	0.00387 (1.08E-04)	0.00376 (1.24E-04)	0.00355 (1.07E-04)	0.00357 (1.17E-04)
$p = 0.008$ $q = 0.008$	0.00478 (1.48E-04)	0.00460 (1.308E-04)	0.00454 (1.32E-04)	0.00428 (1.31E-04)	0.00403 (1.28E-04)	0.00409 (1.35E-04)
$p = 0.016$ $q = 0.008$	0.00521 (1.57E-04)	0.00490 (1.373E-04)	0.00483 (1.37E-04)	0.00457 (1.34E-04)	0.00431 (1.33E-04)	0.00437 (1.47E-04)
$p = 0.016$ $q = 0.016$	0.00567 (1.72E-04)	0.00519 (1.426E-04)	0.00518 (1.51E-04)	0.00496 (1.49E-04)	0.00476 (1.56E-04)	0.00476 (1.70E-04)

Table 3.5: Performance of average Kullback-Leibler divergence ($10^3\kappa$) for BCMIX estimates. Standard errors are given in parentheses.

Scenarios	$T = 3000$	$T = 4000$	$T = 5000$	$T = 6000$	$T = 7000$	$T = 8000$
$p = 0.001$	3.829	3.541	3.372	3.441	3.191	3.200
$q = 0.001$	(1.46E-04)	(1.18E-04)	(1.06E-04)	(1.09E-04)	(8.88E-05)	(8.93E-05)
$p = 0.002$	4.517	4.377	4.138	4.177	3.817	3.833
$q = 0.001$	(1.63E-04)	(1.35E-04)	(1.17E-04)	(1.15E-04)	(9.55E-05)	(9.74E-05)
$p = 0.002$	5.555	5.459	4.358	5.161	4.813	4.858
$q = 0.002$	(1.76E-04)	(1.52E-04)	(7.81E-05)	(1.18E-04)	(9.80E-05)	(9.77E-05)
$p = 0.004$	4.890	4.973	4.715	4.668	4.346	4.355
$q = 0.001$	(1.68E-04)	(1.59E-04)	(1.28E-04)	(1.20E-04)	(9.91E-05)	(9.81E-05)
$p = 0.004$	6.434	6.344	6.323	6.043	5.727	5.757
$q = 0.002$	(1.90E-04)	(1.74E-04)	(1.55E-04)	(1.27E-04)	(1.12E-04)	(1.06E-04)
$p = 0.008$	8.810	8.772	8.728	8.645	8.239	8.306
$q = 0.004$	(2.12E-04)	(1.92E-04)	(1.64E-04)	(1.64E-04)	(1.37E-04)	(1.43E-04)
$p = 0.008$	10.246	10.324	10.333	9.979	9.708	9.645
$q = 0.008$	(2.09E-04)	(2.21E-04)	(1.83E-04)	(1.66E-04)	(1.63E-04)	(1.44E-04)
$p = 0.016$	11.122	11.297	11.309	10.848	10.551	10.505
$q = 0.008$	(2.16E-04)	(2.47E-04)	(1.93E-04)	(1.69E-04)	(1.66E-04)	(1.54E-04)
$p = 0.016$	12.439	12.144	12.217	12.005	11.859	11.701
$q = 0.016$	(2.49E-04)	(2.19E-04)	(2.05E-04)	(1.83E-04)	(1.80E-04)	(1.64E-04)

when $T = 3000$, $p = 0.016$, and $q = 0.016$, $10^3\kappa$ is 12.439. The quantity $10^3\kappa$ decreases to 11.122 when p remains at 0.016 and q changes to 0.008, and decreases to 6.434 when q and q change to 0.004 and 0.002 respectively. Comparing them column by column, we have the opposite tendency as in SSE table: the longer the sequence the small the κ value except a few cell in the first rows. Again, the difference between columns are not as big as them between the rows. When the sample size is large enough, the measured divergence κ should become stable. As we mentioned before, the quantity κ is the average Kullback-Leibler divergence which is a more appropriate measurement of the difference between the model with true parameter and the model with estimated parameter (posterior mean and variance). We can conclude that BCMIX is an efficient method with small KL divergence.

Let's take a look at the standard deviation of these two table 3.4 and 3.5. We can get similar tendency as SSE and κ. The standard deviation become larger when p, q increase and become smaller when T increase. And the largest standard deviation in SSE is $1.72E - 04$ while it is $2.49E - 04$ in KL divergence. Both is quite small, which shows BCMIX is a robust method besides its accuracy.

Table 3.6 summarizes the identification ratio (IR) in each scenario. The IR value becomes larger when p, q increases and also T increases. It infers that BCMIX has higher probability of correct identification when there are more transitions. For example, when $p, q = 0.001, 0.001$, the IR is around 93% with a little increase with T, and it rapidly reaches about 98% with $p, q = 0.004, 0.002$. After that it increase gently to around 99% when $p, q = 0.008, 0.008$ and finally becomes quite stable with even larger p, q. Again as the previous two table, the changes in the rows are placid. The relative difference between $T = 3000$ and $T = 8000$ is less than 1% and tend to stable with T after IR reaches 99%. The standard deviation still remains on a very low level less than 0.01. It has the opposite tendency compared to the IR, that is decreasing with p, q and T increasing. In summary, Table 3.6 support that our

Table 3.6: Performance of identification ratio (IR) for BCMIX estimates. Standard errors are given in parentheses.

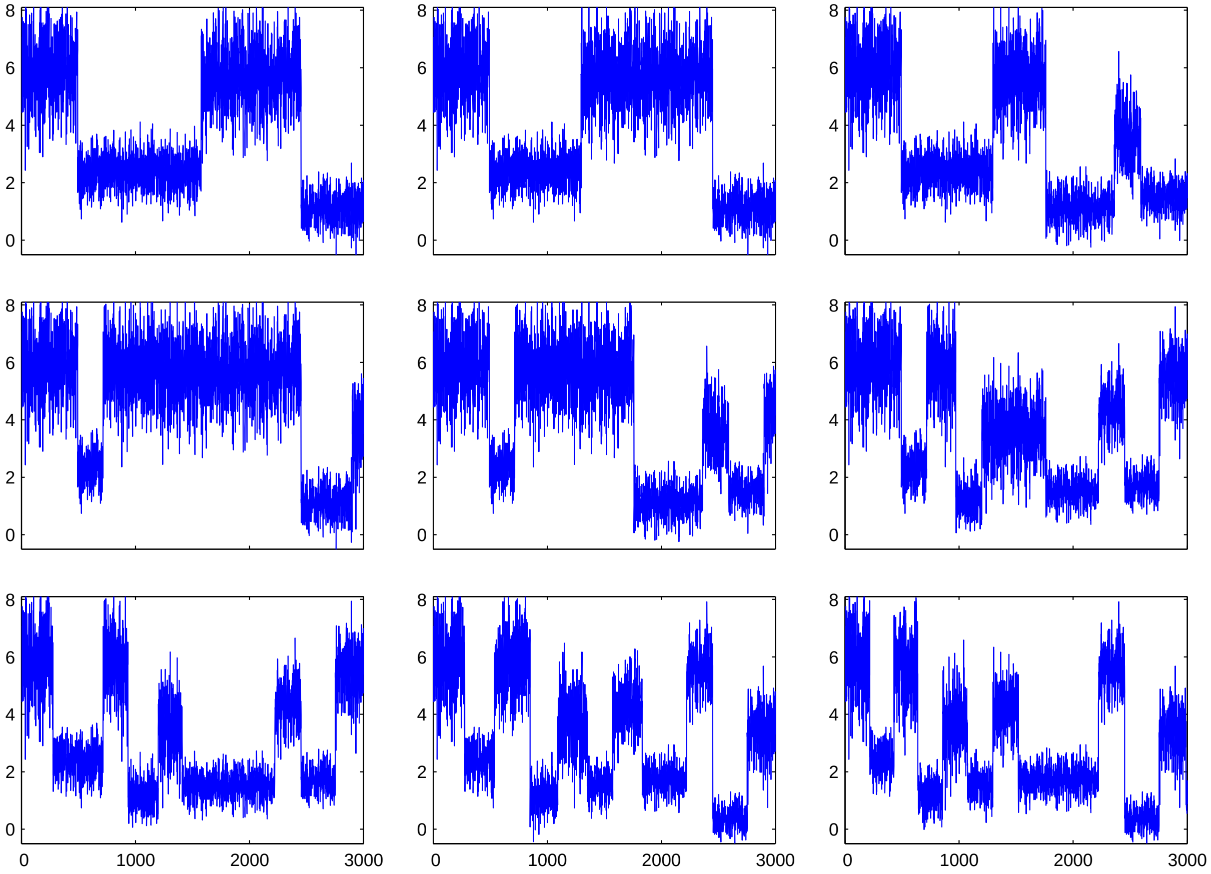
Scenarios	$T = 3000$	$T = 4000$	$T = 5000$	$T = 6000$	$T = 7000$	$T = 8000$
$p = 0.001$ $q = 0.001$	0.932 (9.04E-03)	0.932 (7.80E-03)	0.932 (7.77E-03)	0.933 (7.64E-03)	0.935 (7.18E-03)	0.935 (7.09E-03)
$p = 0.002$ $q = 0.001$	0.932 (7.02E-03)	0.949 (5.52E-03)	0.937 (6.02E-03)	0.940 (5.53E-03)	0.953 (4.93E-03)	0.953 (4.64E-03)
$p = 0.002$ $q = 0.002$	0.944 (6.52E-03)	0.955 (5.46E-03)	0.959 (5.80E-03)	0.961 (4.95E-03)	0.962 (4.51E-03)	0.970 (3.82E-03)
$p = 0.004$ $q = 0.001$	0.961 (5.15E-03)	0.970 (3.70E-03)	0.962 (3.74E-03)	0.962 (4.38E-03)	0.966 (3.98E-03)	0.968 (3.59E-03)
$p = 0.004$ $q = 0.002$	0.976 (3.59E-03)	0.977 (3.81E-03)	0.980 (3.19E-03)	0.982 (3.35E-03)	0.983 (2.95E-03)	0.980 (3.24E-03)
$p = 0.008$ $q = 0.004$	0.984 (3.22E-03)	0.987 (2.58E-03)	0.990 (2.12E-03)	0.991 (1.87E-03)	0.991 (2.03E-03)	0.989 (2.16E-03)
$p = 0.008$ $q = 0.008$	0.983 (3.50E-03)	0.986 (2.92E-03)	0.991 (2.02E-03)	0.991 (1.78E-03)	0.992 (1.96E-03)	0.992 (1.69E-03)
$p = 0.016$ $q = 0.008$	0.986 (2.80E-03)	0.989 (2.29E-03)	0.995 (1.28E-03)	0.994 (1.36E-03)	0.993 (1.53E-03)	0.992 (1.58E-03)
$p = 0.016$ $q = 0.016$	0.987 (2.86E-03)	0.989 (2.42E-03)	0.993 (1.74E-03)	0.993 (1.54E-03)	0.994 (1.32E-03)	0.994 (1.14E-03)

model could not only generate accurate continuous variable estimation but also has good performance on identification of different states which we call categorical feature of the data.

Similar to the first simulation, we will show some figures of a randomly selected simulation path in each scenario to visualize the simulation results. Figure 3.5 shows the series y_t in each scenario with $T = 3000$. Different from the series shown in Figure 3.1 in last section, the series in Figure 3.5 are longer with more frequent transitions between two regimes. As we changed the initial value for the simulation. We could find that the mean of two states are more close and the variance of each state become larger. Furthermore, we find more fluctuations in magnitude in each series when p and q become larger. Figures 3.6 and 3.7 compare the true μ_t and σ_t^2 with $\hat{\mu}_{t|T}$ and $\hat{\sigma}^2_{t|T}$ of the same series in each scenario. From Figure 3.6 it is clear that when p and q become larger, the series experiences more frequent transitions. For example, in the first plot with $p = 0.001$ and $q = 0.001$, there are 3 transitions in total, while in the last plot with $p = 0.016$ and $q = 0.016$, there are 11 transitions. In each plot there are two states, but the values of μ_t within each regime are not constant. For example, in the last plot, the true μ_t within state 1 follows a normal distribution with mean 4.0 and take realized values of 5.89, 5.68, 3.64, 4.26, 5.49 and 3.41 over time, while the true μ_t within state 2 follows another normal distribution with mean 2.5 and take realized values of 2.38, 1.13, 1.52, 1.74, 0.40 and 0.93. Similar cases happen in 3.7, the true value of σ_t^2 within state 1 follows an inverse gamma distribution with two two parameters of $2.5, 0.8$ and take realized value of 1.27, 1.06, 0.85, 0.57, 0.58 and 0.5, while the true σ_t^2 within state 2 follows the inverse gamma distribution with two parameters of $1.8, 0.5$ and take realized value of 0.29, 0.23, 0.17, 0.17, 0.12 and 0.15.

Let's take at the figure about the difference between the true value and the estimates. In Figures 3.6, there are barely no difference between the BCMIX estimates and the true value. Only in the last three figures, when the number of transitions increasing, there are

Figure 3.5: A selected series y_t in Scenarios 1-9 (from left to right and top to bottom).



some segments in the middle part of the sequence appear small differences. Yet, in Figures 3.7, there are some obvious differences, especially for the state 1 with larger realized value. These results are similar with the conclusion we get from the previous simulation.

Figure 3.8 shows the true and estimated $P(s_t = 1)$ of the same series in each scenario. Under the simulation settings in these cases, we can find the model could quickly call the state even when the number of transitions increasing. But in the last figure we can see there are some few red dots which represent the posterior state probability appears in the middle of 0 and 1. Even though their value still indicate the correct state calling, but we might consider how it would be like if we change the simulation with more “strict” settings.

Figure 3.6: BCMIX estimates $\hat{\mu}_{t|T}$ (dashed line) and true μ_t (solid line) of the selected series in Scenarios 1-9 (from left to right and top to bottom).

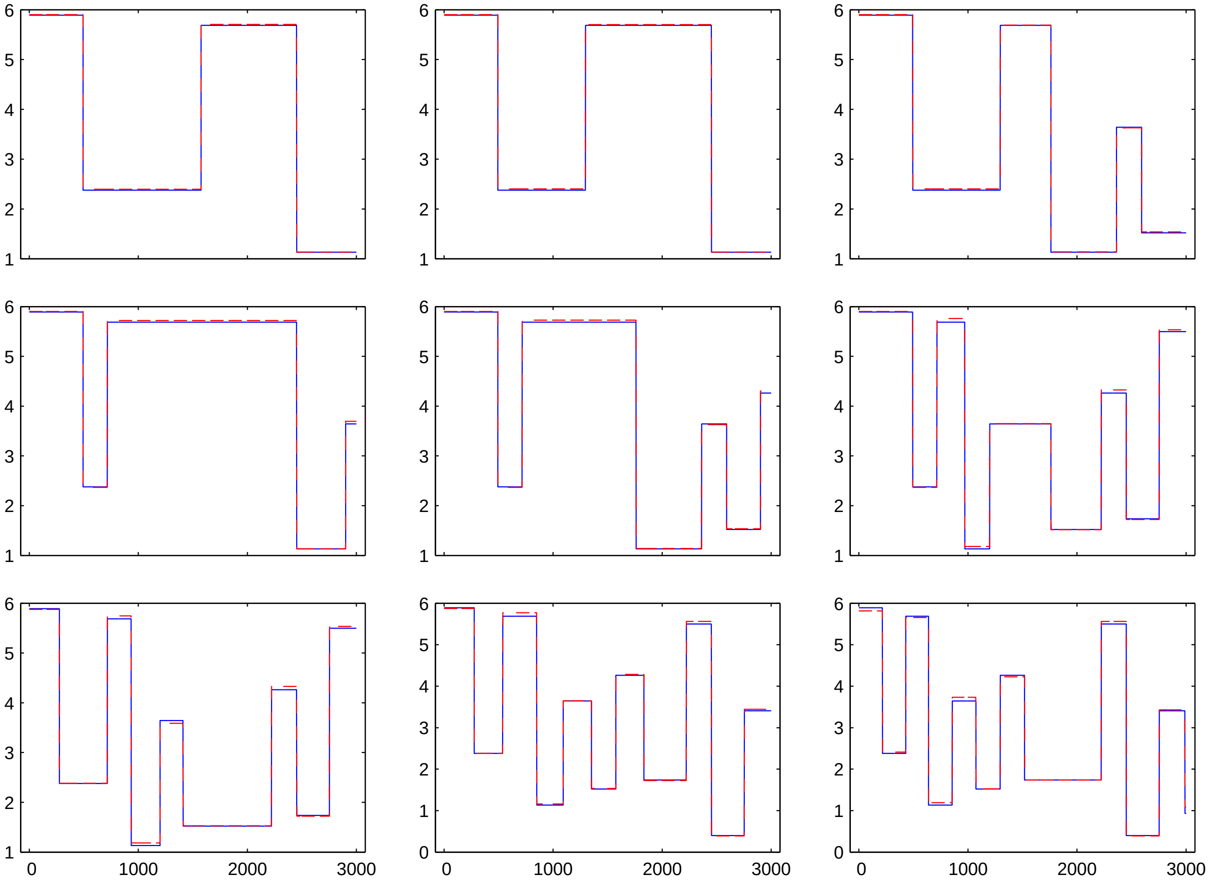


Figure 3.7: BCMIX estimates $\hat{\sigma}^2_{t|T}$ (dashed line) and true σ^2_t (solid line) of the selected series in Scenarios 1-9 (from left to right and top to bottom).

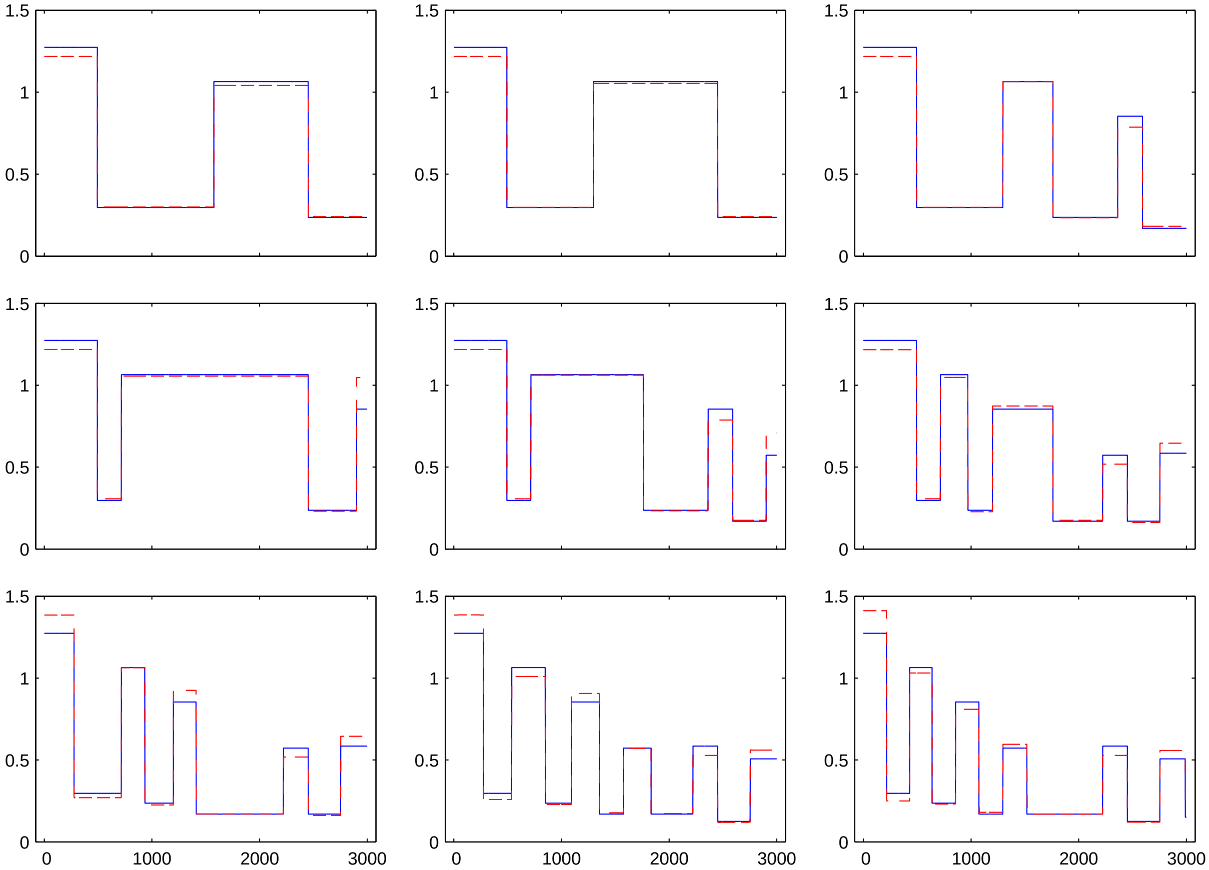
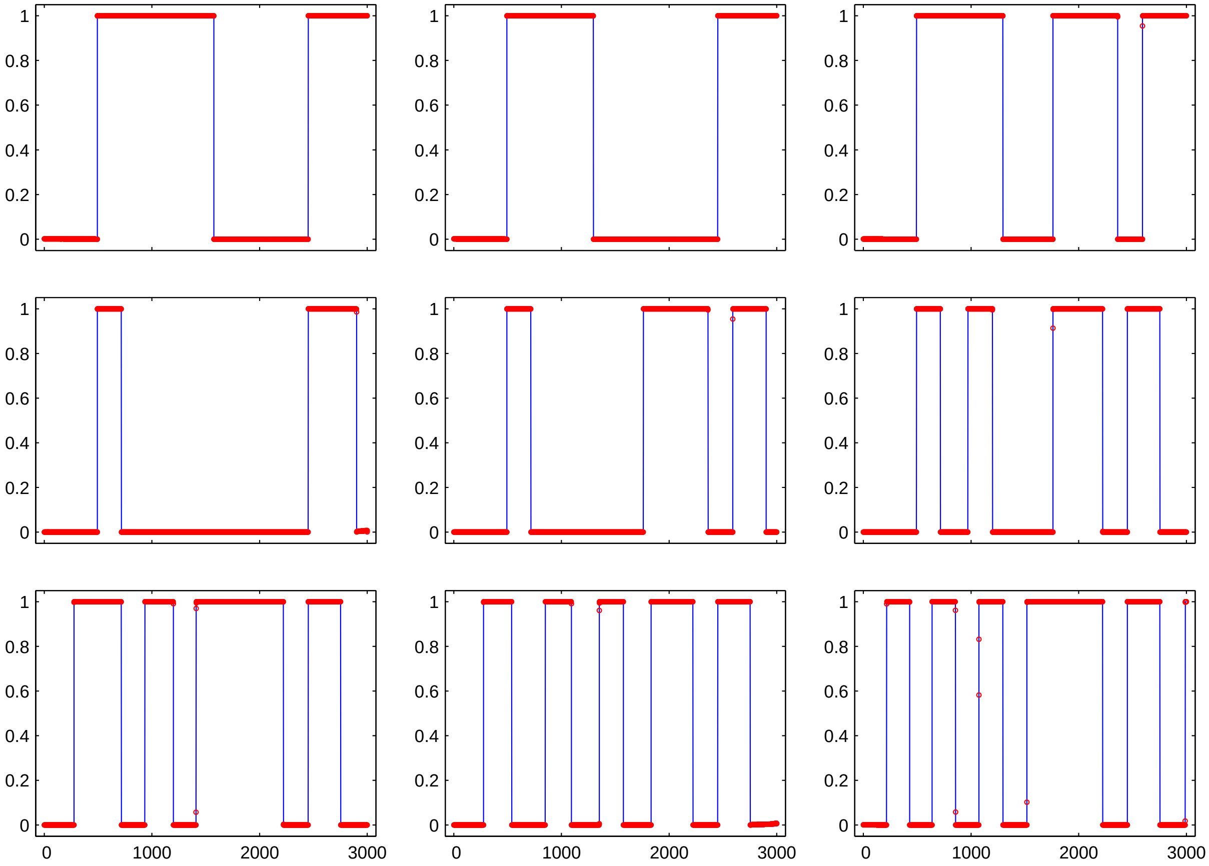


Figure 3.8: BCMIX estimates $\hat{r}^{(1)}_{t|T}$ (dashed line) and true $P(s_t = 1)$ (solid line) of the selected series in Scenarios 1-9 (from left to right and top to bottom).

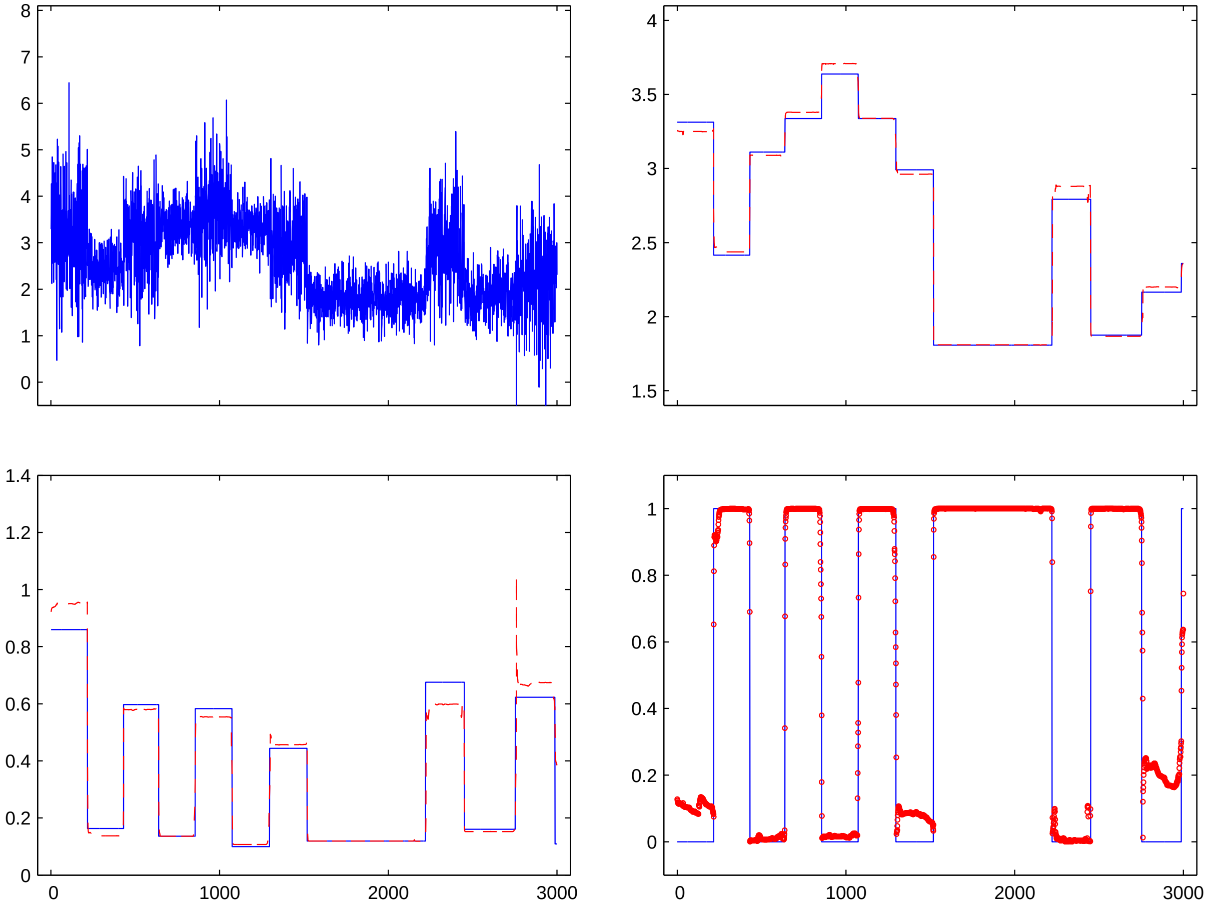


At the end of this section, we will display a special simulation to test our model. We still use the BCMIX procedure specific $M = 20$ and $m = 10$ and 2 states. But instead using the previous simulation parameter settings a we will set some "strict" conditions on the setting, which are $z^{(1)} = 2.5, \kappa^{(1)} = 0.8, \lambda^{(1)} = 0.6, g^{(1)} = 2.0$, $z^{(2)} = 3.5, \kappa^{(2)} = 1.0, \lambda^{(2)} = 0.5, g^{(2)} = 1.8$. The $p, q = 0.016, 0.016$. From it we could discover two changes: the smaller mean difference and larger mean of σ_t^2. Put it in another word, the two states are more closer with larger variance, thus make the model more difficulty to correctly estimate the posterior mean and variance. We choose the largest p, q value means it still keeps many transitions in the sequence. Consider the visualization, we set $T = 3000$ and do 500 simulation with different seeds.

To visualize the results, we displays the figures Figure 3.9. In the first observation figure y_t, we can discover some transitions with "blur" boundaries and we cannot know the number of transitions and the magnitude of each states. The figure $\mu_t and \hat{\mu}_t$ demonstrates that the "shape" of the means are much different with the previous ones. In this scenario, the mean of state 1 sometimes are larger than the mean of state 2 since the large variation of the hidden variable. Moreover, the figure of IR has clearly shows the fuzziness around each transitions. It is clear that when there is a transition, it takes a while to recognize it. So the probability of $P(s_t = 1)$ does not jump directly from 1 to 0 or 0 to 1. Instead it adjusts step by step and takes some values in between. These "middle" points may affect the identification ratio. Moreover, there are more middle points when there are more frequent transitions, although the IR is higher.

As before we calculate the three criterion SSE, Kullback-Leibler divergence and identification ratio. Not surprisingly, the SSE and KL divergence has increased to 0.008 and 0.02 with the standard deviation of 0.00027 and 0.0003. The IR has decreased to 75% with standard deviation of 0.01. Therefore, we can find that the model still has quite good per-

Figure 3.9: A selected series y_t in Scenarios 1 (top-left), μ_t vs. $\hat{\mu}_t$ 2 (top-right), σ_t^2 vs. $\hat{\sigma}_t^2$ 3 (bottom-left) and identification ratio 4 (bottom-right).



formance on estimating parameter and state calling under some extreme parameter settings.

Chapter 4

Real Data Analysis

In this section, we will apply the stochastic segmentation model to two real data sets: Nimblegen ENCODE Array for identifying DNaseI sensitivity and DNaseI hypersensitive sites over the ENCODE regions in human lymphoblastoid cells (GSE4334) and Reduced Representation Bisulfite Sequencing data (RRBS) (GSE31971) to see the directional DNA methylation changes and complex intermediate states accompany lineage specificity in the adult hematopoietic compartment. We will display some relative results of our model such as posterior mean, variance and state probability. The genome browser screenshots are also used for the biological explanation of the results.

4.1 ENCODE Array for detecting DNaseI hypersensitive sites (DHS)

This data is published on July 27th, 2006 with the series number GSE4334 in GEO database. The goal of this study was to map DNaseI sensitivity and DNaseI hypersensitive sites over the ENCODE regions in human lymphoblastoid cells (GM06990, Coriell). The DNase I accessibility assay used by Sabo, et al. is a "quantitative chromatin profiling" method first introduced previously by Dorschner, et al. Intact nuclei are first isolated and

Table 4.1: Hyperparameter estimate using EM algorithm for selected six chromosomes

	chr 1	chr 5	chr 7	chr 8	chr 9	chr 12
z_1	1.62865	1.4385	1.39987	1.33204	1.60695	1.41574
z_2	0.361365	0.338669	0.310175	0.278436	0.216164	0.303773
z_3	-0.0165594	-0.03391	-0.02376	-0.04424	-0.01739	-0.02035
κ_1	0.971606	1.40201	1.40413	1.04643	0.923544	1.22338
κ_2	2.01044	2.03293	2.18995	2.50551	2.41756	2.24939
κ_3	1.33345	1.05668	1.05509	0.861222	1.24738	1.02158
λ_1	1.38604	1.49653	1.75998	1.9908	1.2697	1.88609
λ_2	4.45153	4.51314	5.20456	5.34755	4.32053	5.39408
λ_3	10.2532	7.95578	8.5532	8.13503	8.24995	8.66786

divided into two fractions, one which will be treated with DNase I, another which will not. In a departure from the Dorschner, et al. method, Sabo, et al. furthered size selected for small fragments presumably cut twice by DNase I in close proximity, rather than just once. Using a custom-designed Nimblegen array which employed around 39,000 50-mer probes tiled with 12-mer overlaps falling within 44 genomic ENCODE segments. Signal-to-noise ratios were then calculated at each probe position by comparing DNase-I-treated versus untreated samples. We used these signal-to-noise ratios as the input for our algorithm.

Rather than using two states model, we take 3 states in this study. We run our model chromosome by chromosome. We randomly choose six chromosomes (Chr1, Chr5, Chr7, Chr8, Chr9, Chr12) to display the results. Table 4.1 shows the estimated hyperparameters by the EM algortihm . We can find the same state have similar results of each hyperparameter. The corresponding estimated transition probability matrices are showed in Table 4.2. Moreover, Table 4.3 displays some general statistics of the results, where we can find that the major and minor hypersensitive sites have very few coverage (7%) compared to the insensitive region (93%).

We choose two series to visualize the estimation of posterior mean, variance and state

Table 4.2: Estimated transition probabilities for selected six chromosomes

chr 1	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.863676	0.100086	0.036239
State 2	0.032456	0.798508	0.169035
State 3	0.004811	0.024998	0.970191
chr 5	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.810422	0.142955	0.046623
State 2	0.02342	0.811846	0.164734
State 3	0.002498	0.021419	0.976083
chr 7	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.816241	0.129291	0.054468
State 2	0.016209	0.821953	0.161838
State 3	0.001998	0.020323	0.977679
chr 8	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.827226	0.119964	0.05281
State 2	0.010763	0.802488	0.186749
State 3	0.001661	0.012828	0.985511
chr 9	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.825319	0.113304	0.061377
State 2	0.012774	0.774094	0.213133
State 3	0.002048	0.031269	0.966683
chr 12	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.816969	0.121484	0.061547
State 2	0.020417	0.834264	0.145319
State 3	0.001951	0.017313	0.980736

Table 4.3: Base level coverage and segment lengths for three states

	Major DHS (State 1)	Minor DHS (State 2)	Insensitive (State 3)
Number of segs	815	5127	31104
Number of bases	234kb	802kb	13326kb
Percent of bases	1.60%	5.60%	92.80%
Mean of seg length	287	156	428

probability. We choose the first 600 probes in Chromosome 1 which cover 55024 base pair (chr1:148374643-148429666) and another 600 probes in Chromosome 6 which covers 37164 base pair (chr6:41537432-41574595). Figure 4.1 displays the observation along the probes. Upper is the series from Chromosome 1 and bottom is the series from Chromosome 6. Figure 4.2 displays the posterior estimation for the series from Chromosome 1. Upper is posterior mean; middle is posterior variance and bottom is the posterior state probability (Red: State 1; Blue: State 2; Green: State 3). Figure 4.3 is a similar plot for the series from Chromosome 6. From these two figures, we can conclude that our model has good performance on smoothing the signal and generate reasonable state call. Although the posterior state probabilities here have many fuzziness, it can correct call the state once we use a cut line $p = 0.5$. Figure 4.4 is the genome browser screen-shot for the series from Chromosome 1, in which some of the major and minor hypersensitive sites called by our model is close to TSS or overlapping with gene body.

To assess DNase I hypersensitive (DHS) island accuracy, we operated on the assumption that read densities falling within a DHS island would be enriched and flanked by significantly reduced densities outside of the island boundaries–forming a plateau-shaped profile. Given that, we examined the read density profiles of our and Sabo, et al.'s DHS island calls, including flanking regions up- and downstream half the island length. We divided the islands, plus flanks, into 100 equal-sized bins and calculated average read densities within

Figure 4.1: The observation of two selected series with 600 probes.

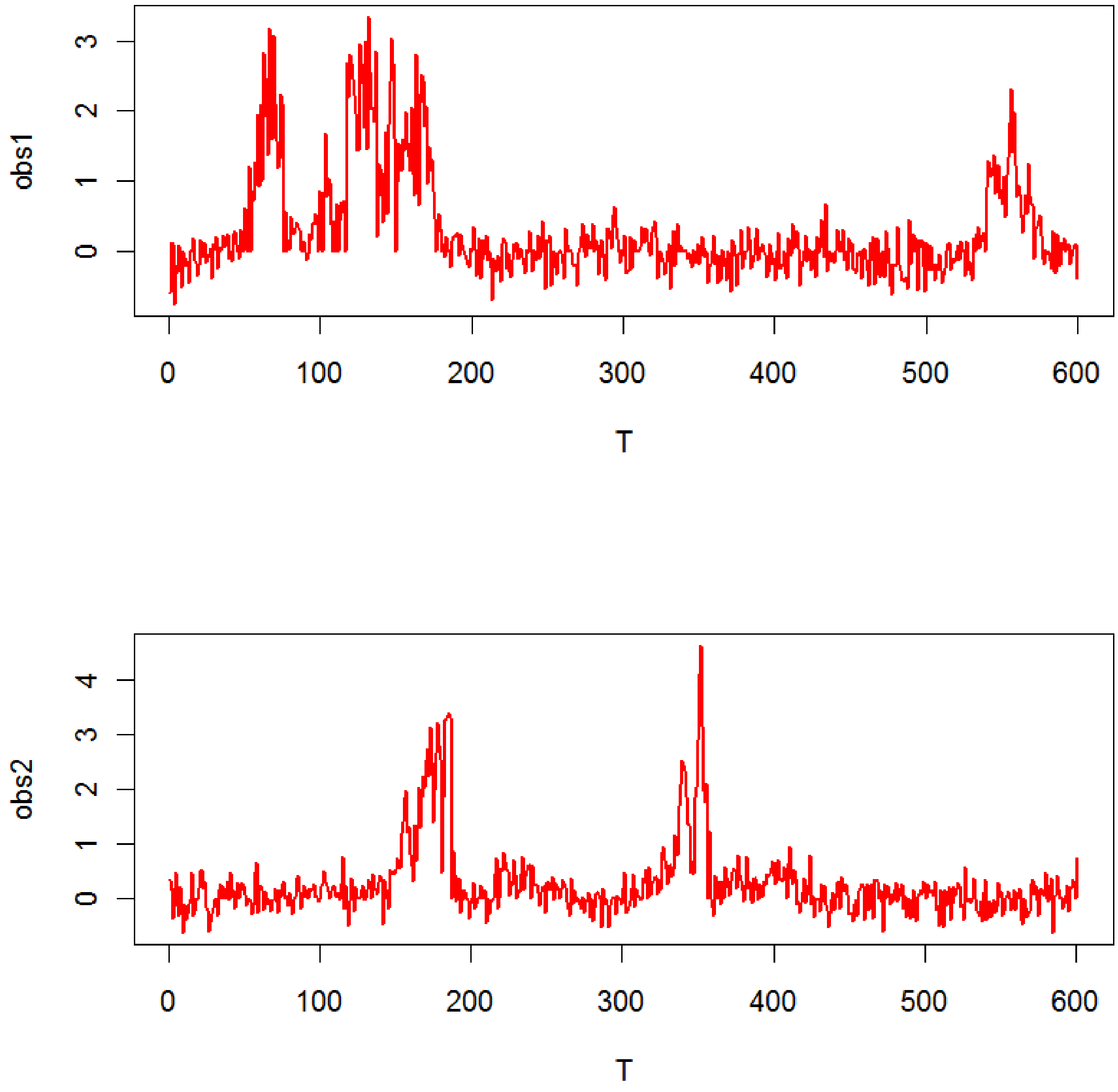


Figure 4.2: The posterior estimation of mean, variance and state probability for the series from Chromosome 1.

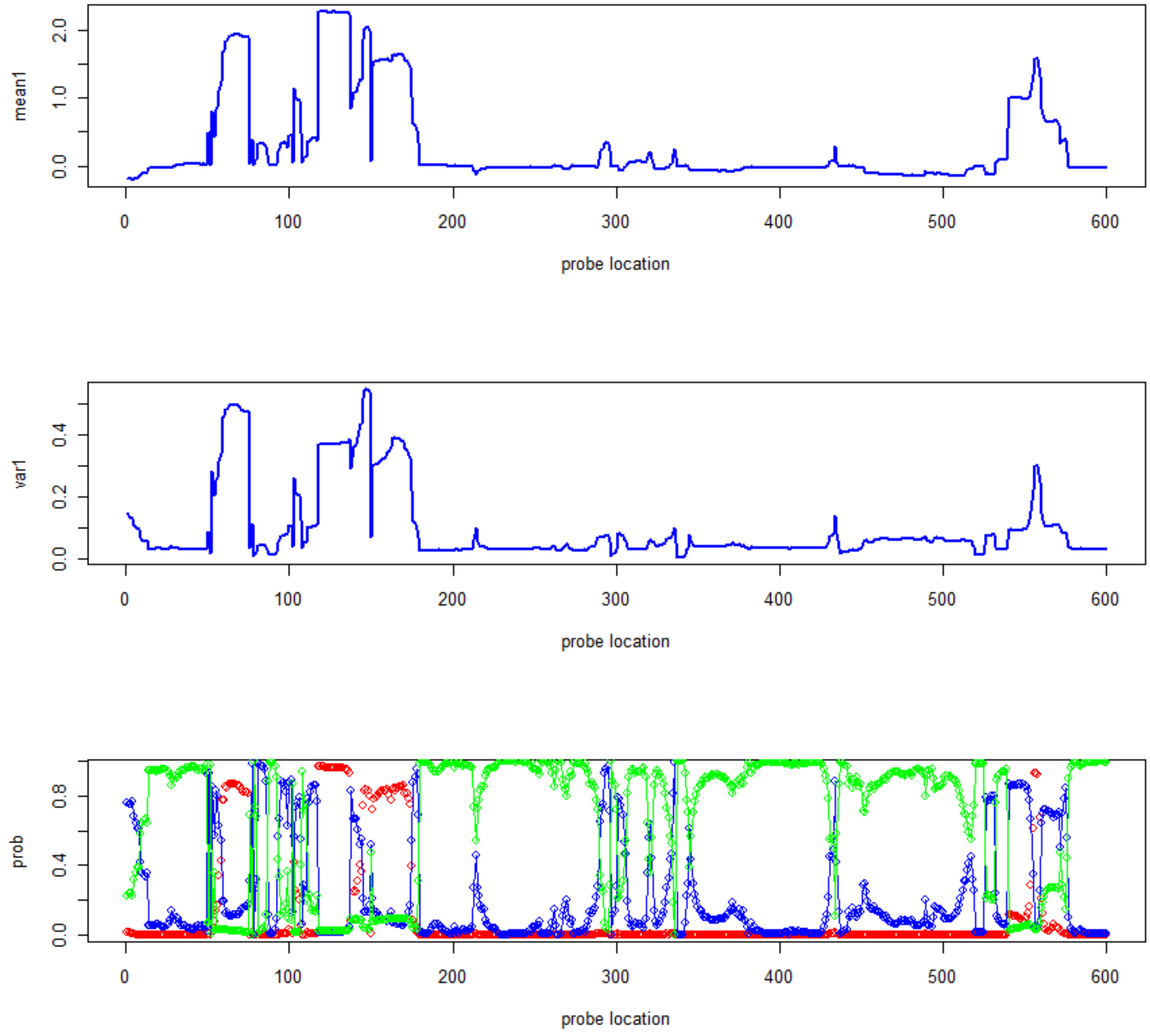


Figure 4.3: The posterior estimation of mean, variance and state probability for the series from Chromosome 6.

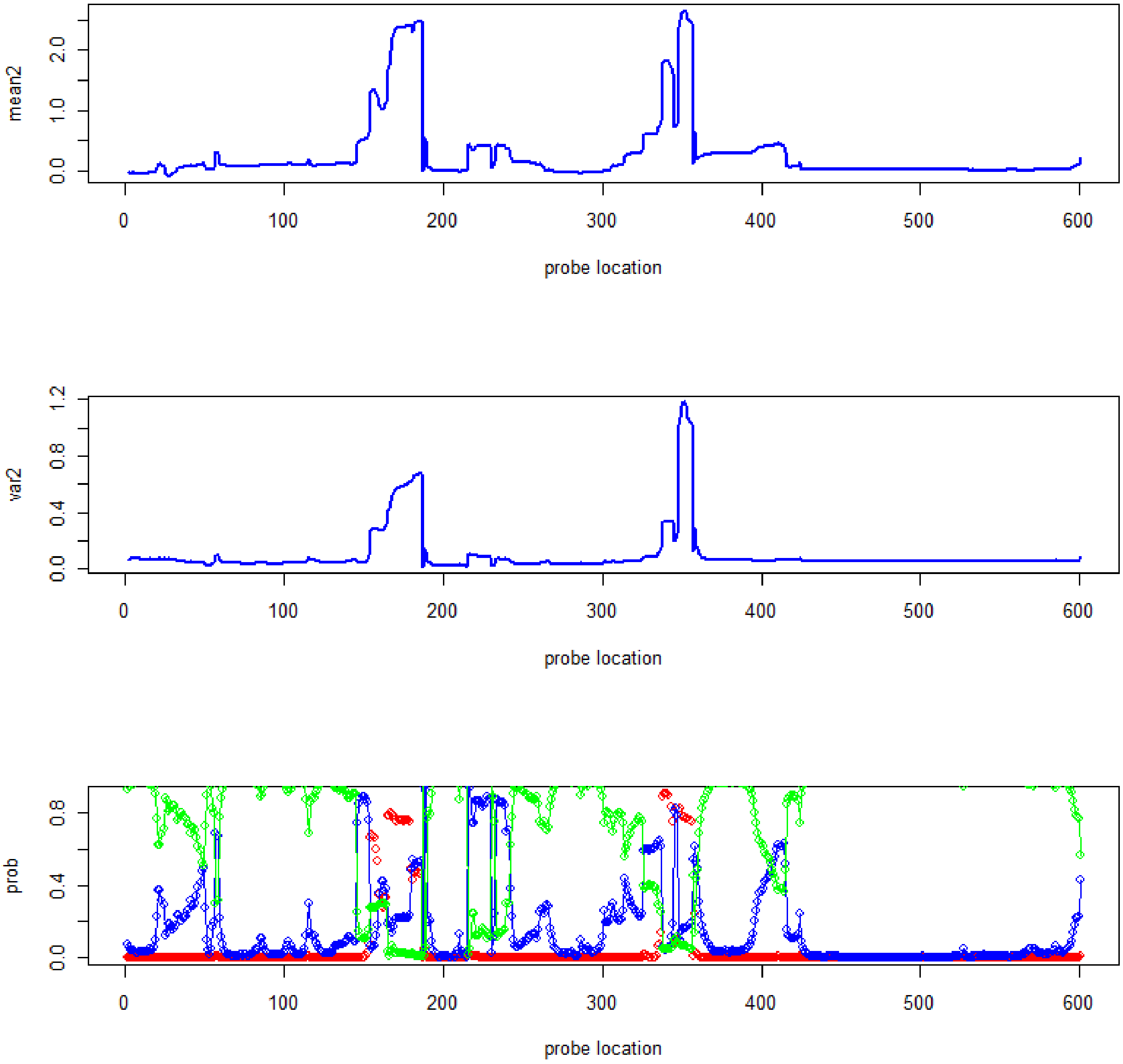
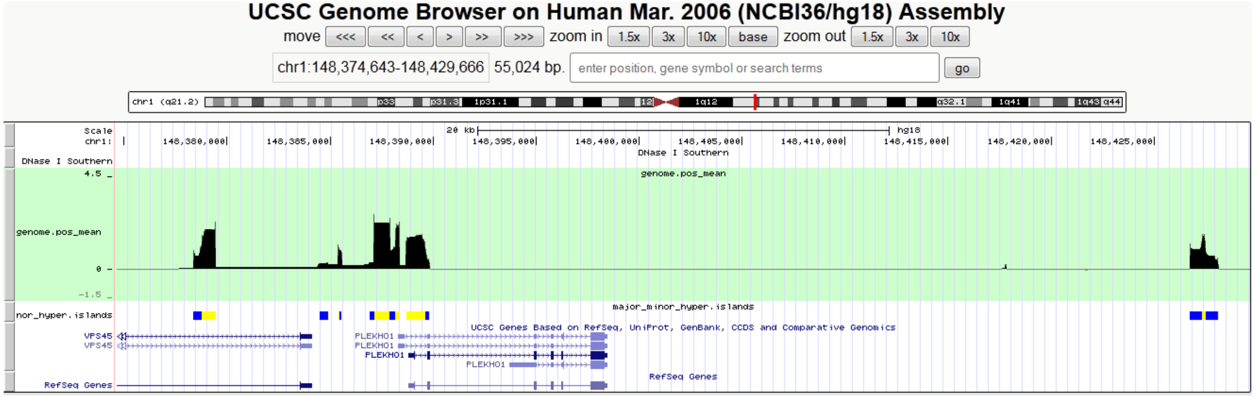


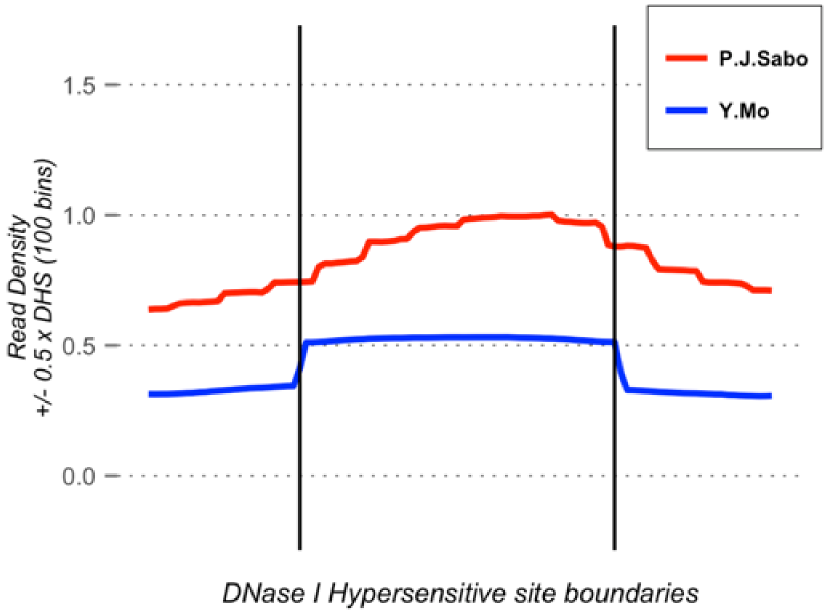
Figure 4.4: A screenshot corresponding to the selected series of Chromosome 1 from UCSC genome browser.


UCSC Genome Browser on Human Mar. 2006 (NCBI36/hg18) Assembly
move <<< << < > >> >>> zoom in 1.5x 3x 10x base zoom out 1.5x 3x 10x
chr1:148,374,643-148,429,666 55,024 bp.
enter position, gene symbol or search terms
go
chr1 (q21.2)
p33
p31.3
1p31.1
12
1q12
q32.1
1q41
1q43
q44
Scale
chr1:
20 kb
hg18
148,380,000
148,385,000
148,390,000
148,395,000
148,400,000
148,405,000
148,410,000
148,415,000
148,420,000
148,425,000
DNase I Southern
4.5
genome.pos_mean
0
-1.5
nor_hyper.islands
major_minor_hyper.islands
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics
VPS45
VPS45
PLEKHO1
PLEKHO1
PLEKHO1
PLEKHO1
RefSeq Genes
RefSeq Genes

each bin. Our method showed a much more pronounced plateau (Figure 4.5), which implies our DHS islands more accurately defined island boundaries around regions of higher DNase I accessibility.

Common wisdom suggests functionally relevant regions of DNA are more susceptible to enzymatic digestion by DNase treatment due to increased accessibility at cis-regulatory elements. We assessed the degree of enrichment of our DHS islands within regions of known functional importance, including CpG islands, known genes, mRNA transcripts, spliced ESTs, and regions enriched for histone modification marks. We adopt the same enrichment calculation as Lian Heng et al., 2006. Compared to the DHS calling from Sabo et al., 2006, our result has better enrichment on major DHS and similar enrichment on major and minor DHS (Figure 4.6). The method of calculation of enrichment is same as Lian et al., 2006.

Figure 4.5: The assessment of DHS island accuracy.



4.2 Reduced Representation Bisulfite-seq data for detecting Differential Methylation Region (DMR)

This data is published on July 27th, 2006 with the series number GSE31971 in GEO database. The goal of this study is to provide insights into directional changes in DNA methylation as cells adopt terminal fates. DNA was extracted from the sperm tissue and fragmented to 150-200bp nt by sonication. Samples were treated with bisulfite, which converts unmethylated cytosine nucleotides to thymines, leaving methylated cytosines unchanged. Following this treatment, sequencing was performed and reads were mapped to the reference genome using RMAPBS, revealing the converted and unconverted cytosine positions. The number of reads methylated and unmethylated at each base position were determined. The M-value, log2 (methylated reads / unmethylated reads), at each position was calculated and used as input for our algorithm.

Again, we take 3 states in our model. We choose six genes (CD19, PIWIL1, PIWIL3,

Figure 4.6: Enrichment of annotation functional elements.

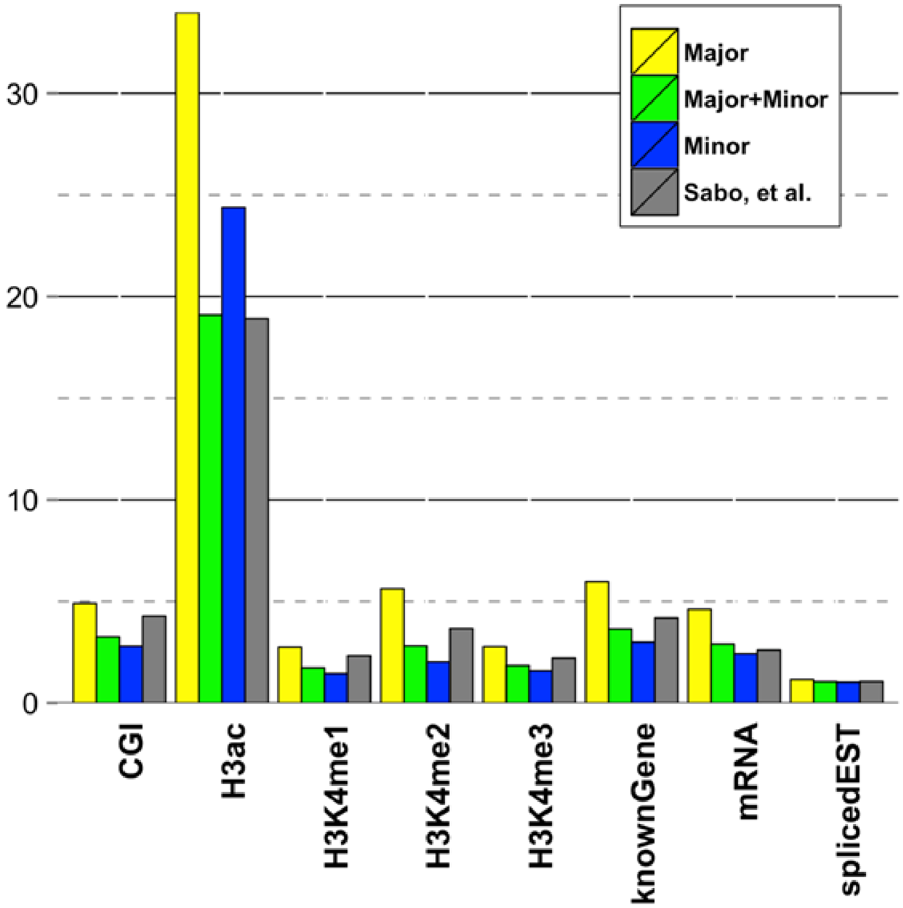


Table 4.4: Hyperparameter estimate using EM algorithm for six chosen genes under Sperm cell line.

	CD19	PIWIL1	PIWIL3	PLD6	TDRD1	TDRD9
z_1	2.15513	1.72502	1.96827	1.89935	2.03974	2.04597
z_2	0.00900467	0.007067	0.013866	-0.05837	0.010556	0.010434
z_3	-1.58032	-0.8042	-0.06966	-2.21303	-0.40859	-0.26453
κ_1	0.0832219	0.571049	0.63258	0.523774	0.596497	0.403964
κ_2	0.452864	0.46103	0.40019	1.61268	0.3009	0.446317
κ_3	1.77884	1.92117	2.55403	2.98552	3.02786	3.1302
λ_1	0.678908	0.431715	0.399965	0.243351	0.328238	0.403964
λ_2	20.7468	41.2349	47.2962	10.6345	47.4145	50.3631
λ_3	0.429416	0.641349	0.261904	1.56773	0.246383	0.313745

PLD6, TDRD1 and TDRD9) under five different cell lines (B cell, HSPC, Sperm, CD133 and Neutrophil) in this study. Table 4.4 shows the estimated hyperparameters by the EM algortihm for different genes in the Sperm cell line. The corresponding estimated transition probability matrices are showed in Table 4.5. Since there is no strong evidence for the changing variance, a genome browser screenshot will display the signal profile and the posterior mean for gene CD19 under Bcell (Figure 4.7).

Then we display the state calling figures for each gene under all of the five cell lines via the genome browser screenshot. From these figures we could further identify the changing of hypo methylatoin regions (HMRs) for a specific gene across different cells (Figure 4.8, 4.9, 4.10, 4.11, 4.12, 4.13). As mentioned by Emily Hodges et al., 2011, promoter HMRs shared across diverse cell-types typically display a constitutive core that expands and contracts in a lineage-specific manner to fine-tune the expression of associated genes. Our results also well demonstrate such scenario. We can discover in most of these genes, the HMRs are shared common part but their width differ. Take CD19 as the case, the sperm cell has the most expansive hypomethylation with the core of known CGI, while the B cell marker CD19 displays a broader HMR at its transcriptional start site (TSS) and does not overlap any

Table 4.5: Estimated transition probabilities for six chosen genes.

CD19	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.953219	0.035787	0.010993
State 2	0.133322	0.760572	0.106105
State 3	0.053074	0.049814	0.897112
PIWIL1	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.942224	0.049052	0.008724
State 2	0.177194	0.765404	0.057402
State 3	0.108773	0.044784	0.846442
PIWIL3	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.93319	0.058242	0.008568
State 2	0.269287	0.672779	0.057935
State 3	0.129217	0.067086	0.803697
PLD6	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.970319	0.014387	0.015294
State 2	0.065539	0.778561	0.1559
State 3	0.023315	0.014169	0.962516
TDRD1	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.921836	0.073046	0.005118
State 2	0.330461	0.634812	0.034727
State 3	0.198612	0.086456	0.714932
TDRD9	Major DHS	Minor DHS	Insensitive
	(State 1)	(State 2)	(State 3)
State 1	0.941722	0.052617	0.005662
State 2	0.321165	0.652795	0.02604
State 3	0.190382	0.065817	0.743801

Figure 4.7: The screenshot for gene CD19 under Sperm cell. Top: M value; middle: posterior mean; bottom: state call.

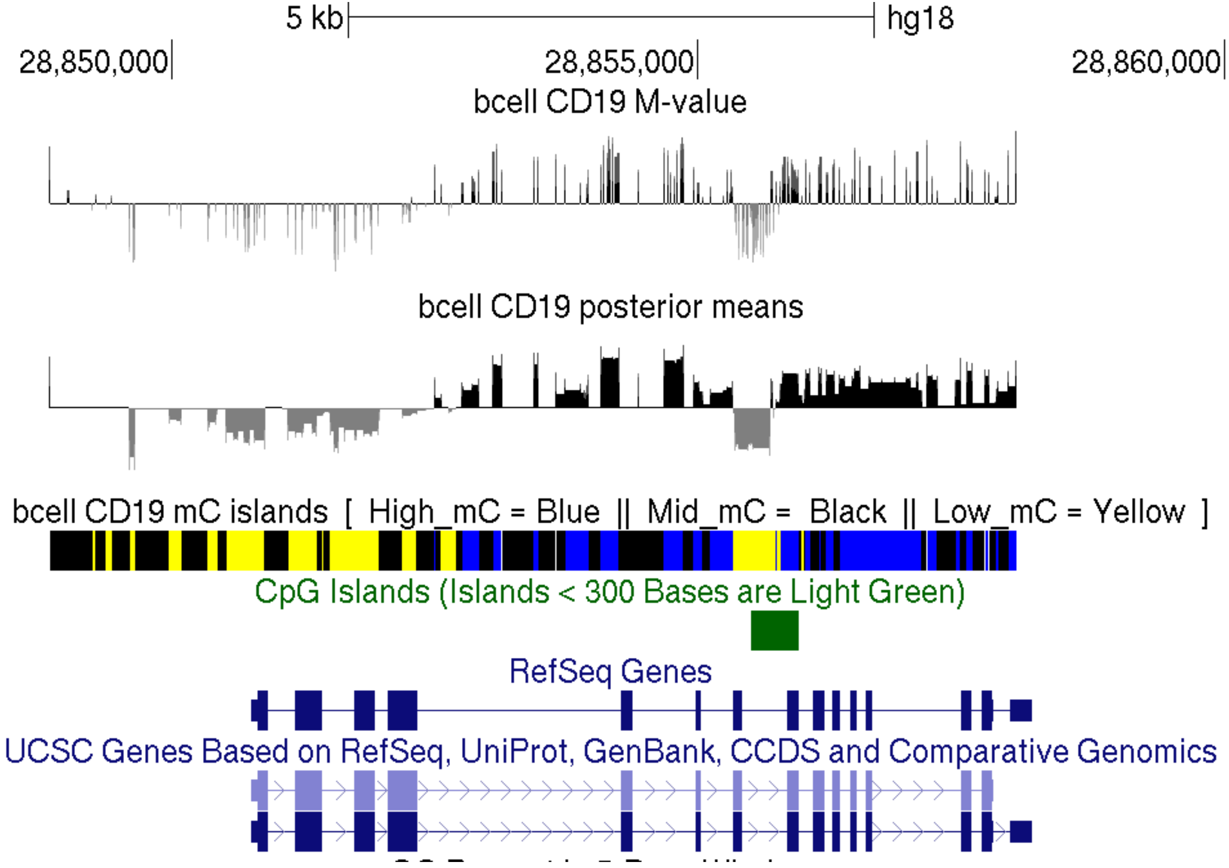
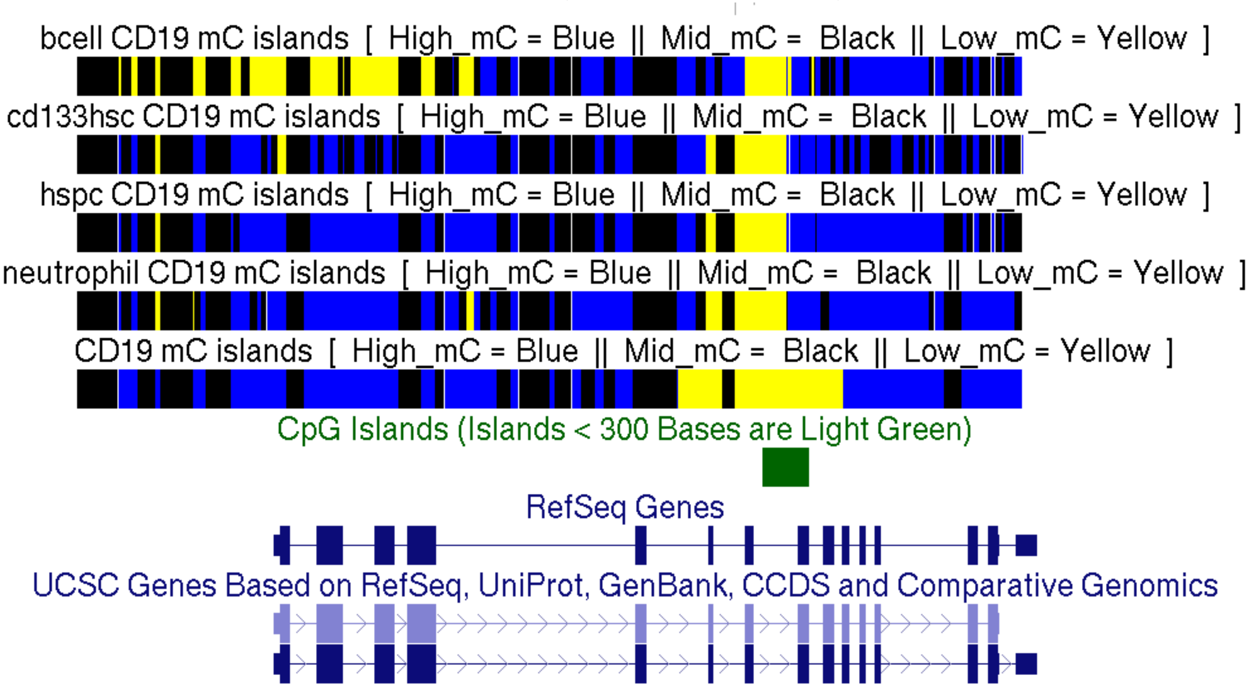

5 kb
hg18
28,850,000
28,855,000
28,860,000
bcell CD19 M-value
bcell CD19 posterior means
bcell CD19 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CpG Islands (Islands < 300 Bases are Light Green)
RefSeq Genes
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

Figure 4.8: Genome browser tracks depict the segmentation across gene CD19 under B cell, CD133+, HSPC, Neutrophil and Sperm (from top to bottom).


bcell CD19 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
cd133hsc CD19 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
hspc CD19 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
neutrophil CD19 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CD19 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CpG Islands (Islands < 300 Bases are Light Green)
RefSeq Genes
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

known CGI. These suggest the boundaries of HMRs vary in a cell type manner.

Figure 4.9: Genome browser tracks depict the segmentation across gene PIWIL1 under B cell, CD133+, HSPC, Neutrophil and Sperm (from top to bottom).

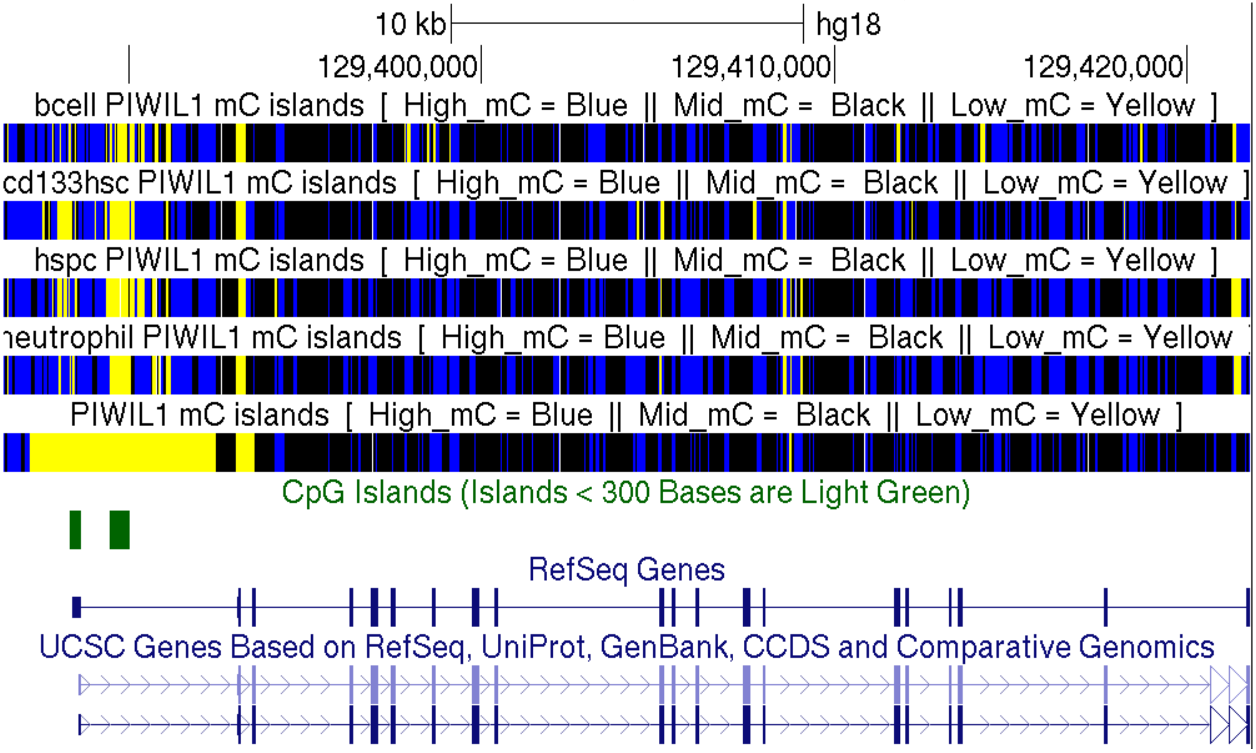

10 kb
hg18
129,400,000
129,410,000
129,420,000
bcell PIWIL1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
cd133hsc PIWIL1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
hspc PIWIL1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
ıeutrophil PIWIL1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow
PIWIL1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CpG Islands (Islands < 300 Bases are Light Green)
RefSeq Genes
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

Figure 4.10: Genome browser tracks depict the segmentation across gene PIWIL3 under B cell, CD133+, HSPC, Neutrophil and Sperm (from top to bottom).

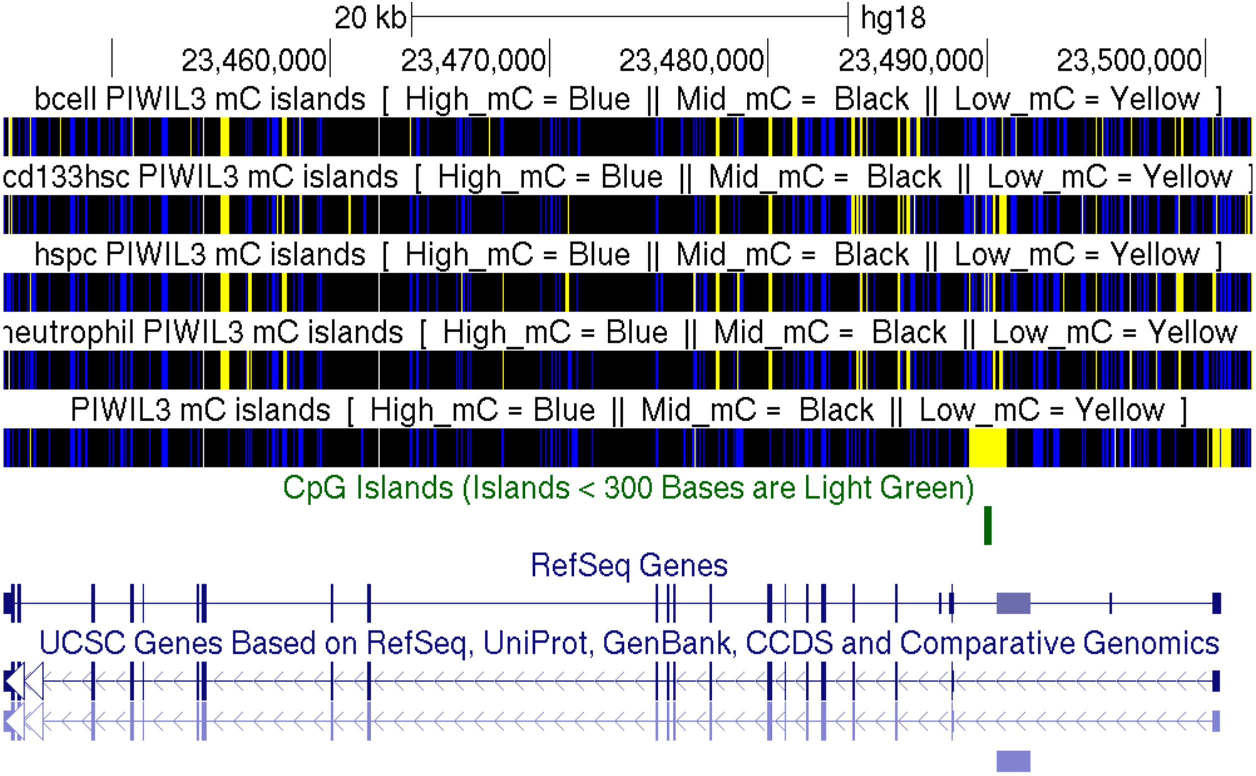

20 kb
hg18
23,460,000
23,470,000
23,480,000
23,490,000
23,500,000
bcell PIWIL3 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
cd133hsc PIWIL3 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
hspc PIWIL3 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
ıeutrophil PIWIL3 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow
PIWIL3 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CpG Islands (Islands < 300 Bases are Light Green)
RefSeq Genes
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

Figure 4.11: Genome browser tracks depict the segmentation across gene PLD6 under B cell, CD133+, HSPC, Neutrophil and Sperm (from top to bottom).

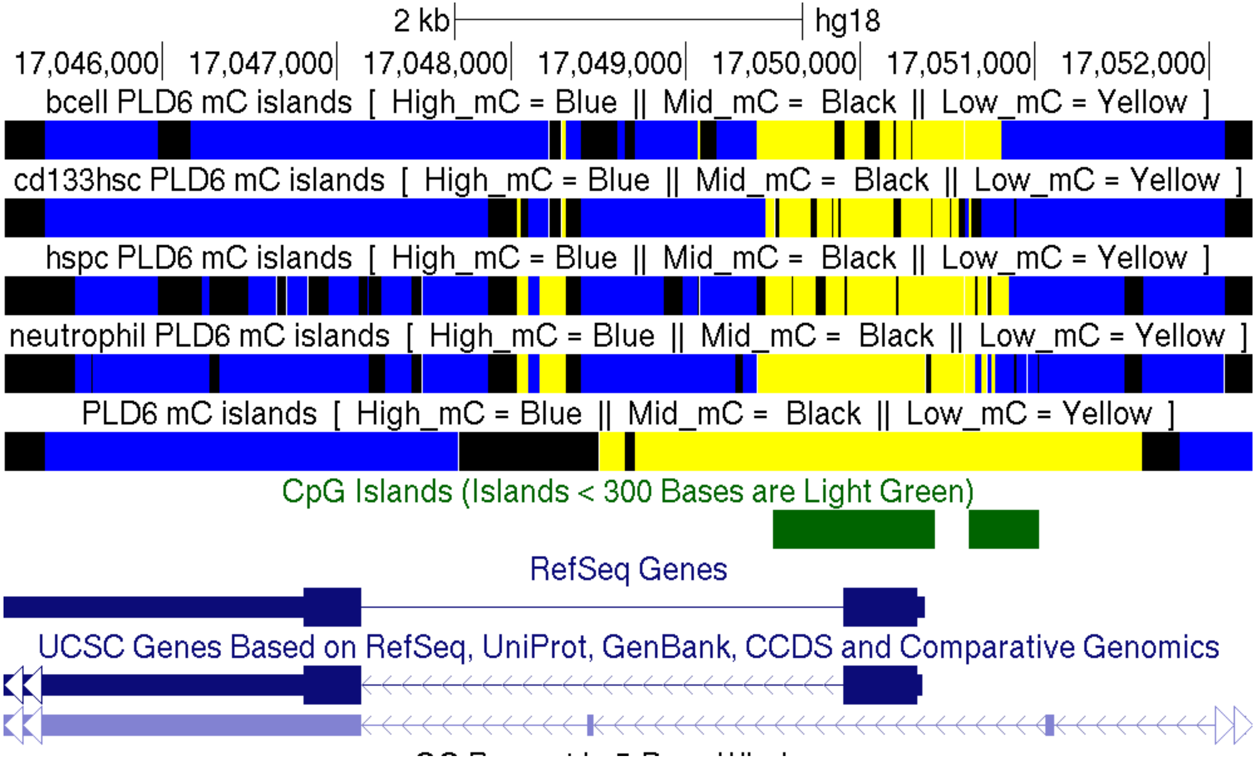

2 kb
hg18
17,046,000
17,047,000
17,048,000
17,049,000
17,050,000
17,051,000
17,052,000
bcell PLD6 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
cd133hsc PLD6 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
hspc PLD6 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
neutrophil PLD6 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
PLD6 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CpG Islands (Islands < 300 Bases are Light Green)
RefSeq Genes
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

Figure 4.12: Genome browser tracks depict the segmentation across gene TDRD1 under B cell, CD133+, HSPC, Neutrophil and Sperm (from top to bottom).

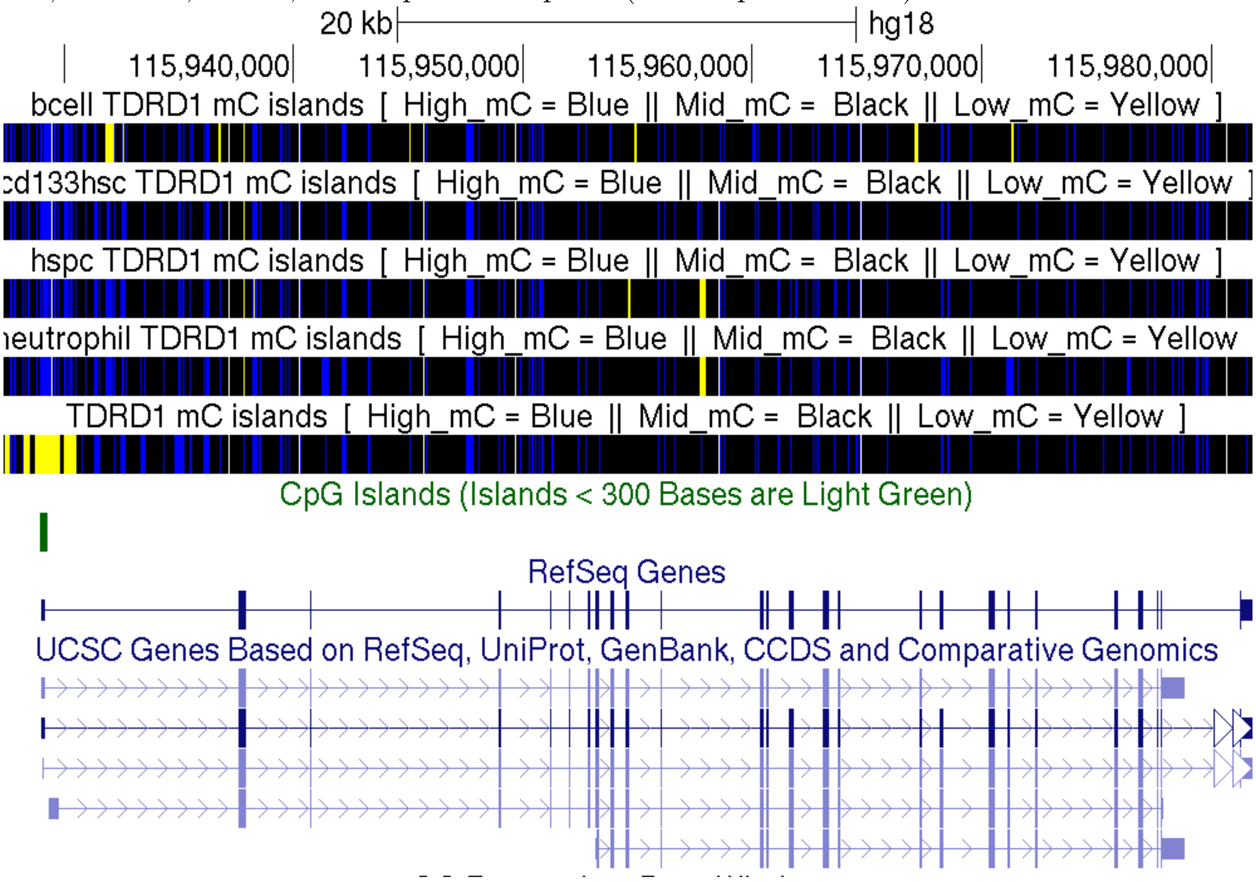
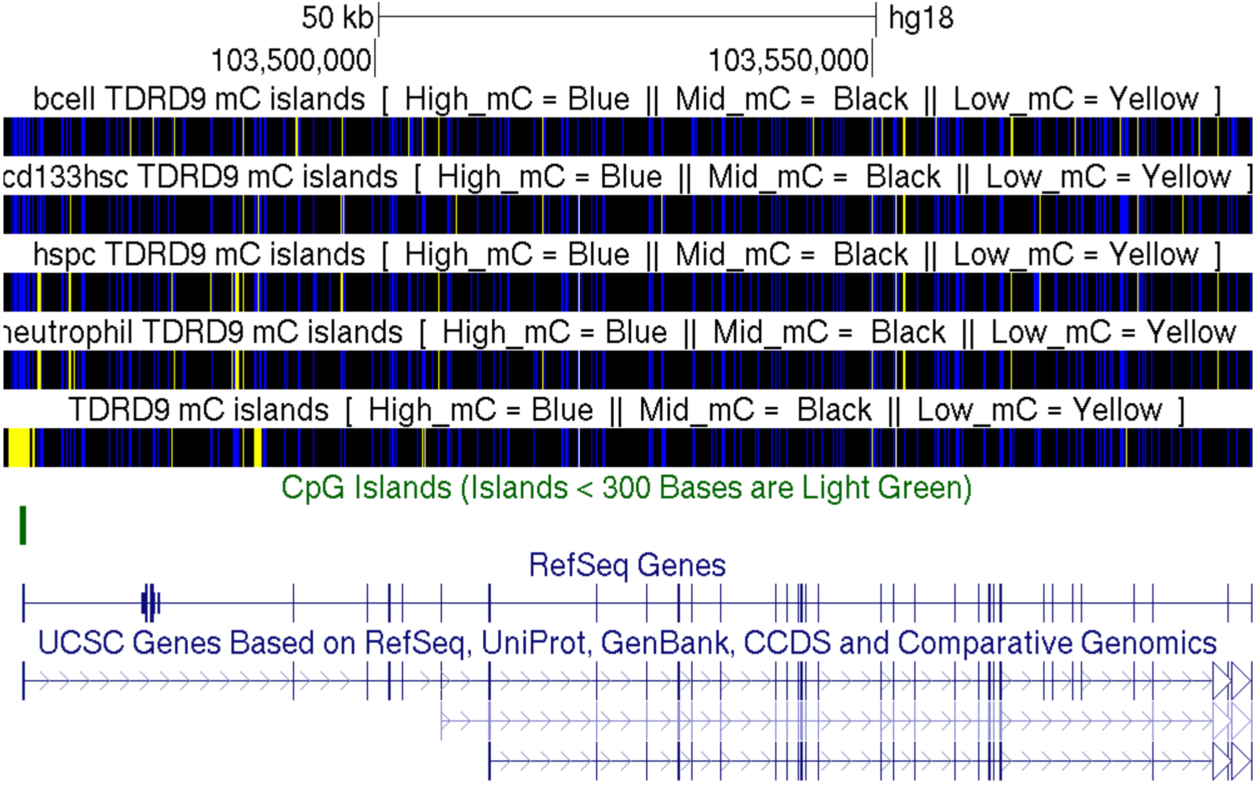

20 kb
hg18
115,940,000
115,950,000
115,960,000
115,970,000
115,980,000
bcell TDRD1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
cd133hsc TDRD1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
hspc TDRD1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
ıeutrophil TDRD1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow
TDRD1 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CpG Islands (Islands < 300 Bases are Light Green)
RefSeq Genes
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

Figure 4.13: Genome browser tracks depict the segmentation across gene TDRD9 under B cell, CD133+, HSPC, Neutrophil and Sperm (from top to bottom).


50 kb
hg18
103,500,000
103,550,000
bcell TDRD9 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
cd133hsc TDRD9 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
hspc TDRD9 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
ıeutrophil TDRD9 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow
TDRD9 mC islands [High_mC = Blue || Mid_mC = Black || Low_mC = Yellow]
CpG Islands (Islands < 300 Bases are Light Green)
RefSeq Genes
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

Chapter 5

Conclusions

For the analysis of biological sequential data, we proposed a class of stochastic segmentation models and an associated inference framework that has attractive statistical and computational properties. The stochastic regime switching model in Chapter 2 assumes that $y_t = \mu_t + \sigma_t \epsilon_t$ for $t = 1, \dots, n$, where ϵ_t are independent normal random variables with mean 0 and variance 1, and the categorical states of $\boldsymbol{\theta}_t := (\mu_t, \sigma_t)$ is an unknown step function whose prior distribution depends on a finite state hidden Markov chain s_t. After the hidden state shifts from one regime to another regime, the model parameters jump to another set of values, which are generated by state-dependent prior distributions and hence are not necessarily same as those within the same state during the past.

A forward filtering procedure shows the posterior distribution of the parameter as a mixture distribution with explicit weights which can be calculated recursively. Furthermore, based on the reversibility of the hidden Markov chain, a backward filtering procedure can be conducted in a similar way. Based on Bayes' theorem, both the smoothing estimate of parameter and probability of regimes can be calculated explicitly to save a time-consuming numerical filtering procedure. The hyperparameters in the model can be estimated by the Expectation-Maximum (EM) algorithm. Furthermore, a Bounded Complexity Mixture Approximation (BCMIX) is shown to have much lower computational complexity yet comparable to the Bayes estimates in statistical efficiency. Simulation studies evaluate the Bayes and BCMIX estimates in terms of the sum of squared errors (SSE) and the Kullback-Leibler divergence (κ). Moreover, the accuracy of identifying the transitions is evaluated by an Identification Ratio (IR). Applying this model to two biological data sets: Nimblegen ENCODE Array for identifying DNaseI sensitivity and DNaseI hypersensitive sites over the ENCODE regions in human lymphoblastoid cells (GSE4334) and Reduced Representation Bisulfite Sequencing data (RRBS) (GSE31971) to see the directional DNA methylation changes and complex intermediate states accompany lineage specificity in the adult hematopoietic compartment, it generates promising results in biological explanation.

An important benefit of our Bayesian model is that we can derive analytical filtering and smoothing formulas for the posterior distributions of model parameters and make inference on segments. The BCMIX estimate has much lower computational complexity yet comparable to the Bayes estimate in statistical efficiency.

Reference

Barski A, Cuddapah S, Cui K, Roh TY, Schones DE, Wang Z, Wei G, Chepelev I, Zhao K. High-resolution profiling of histone methylations in the human genome. *Cell* 129, 823837 (2007).

Bernstein BE, Stamatoyannopoulos JA, Costello JF, Ren B, Milosavljevic A, Meissner A, Kellis M, Marra MA, Beaudet AL, Ecker JR, Farnham PJ, Hirst M, Lander ES, Mikkelsen TS, Thomson JA. The NIH Roadmap Epigenomics Mapping Consortium. *Nat Biotechnol* 28, 10451048 (2010).

Blankenberg D, Von Kuster G, Coraor N, Ananda G, Lazarus R, Mangan M, Nekrutenko A, Taylor J. Galaxy: a web-based genome analysis tool for experimentalists. *Curr Protoc Mol Biol* Chapter 19, Unit 19.10.121 (2010).

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

Brot P, Richardson S. Detection of gene copy number changes in CGH microarrays using a spatially correlated mixture model. *Bioinformatics* 22, 911-918(2006).

Chen X, Xu H, Yuan P, Fang F, Huss M, Vega VB, Wong E, Orlov YL, Zhang W, Jiang J, Loh YH, Yeo HC, Yeo ZX, Narang V, Govindarajan KR, Leong B, Shahab A, Ruan Y, Bourque G, Sung WK, Clarke ND, Wei CL, Ng HH. Integration of external signaling pathways with the core transcriptional network in embryonic stem cells. Cell 133, 11061117 (2008).

Cui K, Zang C, Roh TY, Schones DE, Childs RW, Peng W, Zhao K. Chromatin signatures in multipotent human hematopoietic stem cells indicate the fate of bivalent genes during differentiation. *Cell Stem Cell* 4, 8093 (2009).

Dorschner MO, Hawrylycz M, Humbert R, Wallace JC, Shafer A, Kawamoto J, Mack J, Hall R, Goldy J, Sabo PJ, Kohli A, Li Q, McArthur M, Stamatoyannopoulos JA. High-throughput localization of functional elements by quantitative chromatin profiling. Nature methods, 1(3), 219225. (2004).

Engler DA, Mohapatra G, Louis DN, Betensky RA. A pseudolikelihood approach for simultaneous analysis of array comparative genomic hybridizations.*Biostatistics* 7, 399-421(2006).

Fabarius A, Hehlmann R, Duesberg PH. Instability of chromosome structure in cancer cells increases exponentially with degrees of aneuploidy. *Cancer Genetics and Cytogenetics*143, 59-72(2003).

Fridlyand J, Snijders A, Pinkel D, Albertson DG, Jain AN. Application of hidden Markov models to the analysis of the array-CGH data. *Journal of Multivariate Analysis* 90, 132-153(2004).

Goecks J, Nekrutenko A, Taylor J; Galaxy Team.Galaxy Team Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biol* 11, R86 (2010).

Guha S, Li Y, Neuberg D. Harvard University Biostatistics Working Paper Series. 2006. Bayesian hidden Markov modeling of array CGH data. Working paper 24. Available at: http://www.bepress.com/harvardbiostat/paper24.

Guttman M, et al. Chromatin signature reveals over a thousand highly conserved large non-coding RNAs in mammals. *Nature* 458, 223227 (2009).

Hardenbol P, Baner J, Jain M, Nilsson M, Namsaraev EA, Karlin-Neumann GA, Fakhrai-Rad H, Ronaghi M, Willis TD, Landegren U. Multiplexed genotyping with sequence-tagged molecular inversion probes. *Nature Biotechnology. and others.* 21, 673-678(2003).

Hawkins RD, Hon GC, Lee LK, Ngo Q, Lister R, Pelizzola M, Edsall LE, Kuan S, Luu Y, Klugman S, Antosiewicz-Bourget J, Ye Z, Espinoza C, Agarwahl S, Shen L, Ruotti V, Wang W, Stewart R, Thomson JA, Ecker JR, Ren B. Distinct epigenomic landscapes of pluripotent and lineage-committed human cells. *Cell Stem Cell* 6, 479491 (2010).

Heintzman ND, Hon GC, Hawkins RD, Kheradpour P, Stark A, Harp LF, Ye Z, Lee LK, Stuart RK, Ching CW, Ching KA, Antosiewicz-Bourget JE, Liu H, Zhang X, Green RD, Lobanenkov VV, Stewart R, Thomson JA, Crawford GE, Kellis M, Ren B. et al. Histone modifications at human enhancers reflect global cell-type-specific gene expression. *Nature* 459, 108112 (2009).

Hicks J, Muthuswamy L, Krasnitz A, Navin N, Riggs M, Grubor V, Esposito D, Alexander J, Troge J, Wigler M. High-resolution ROMA CGH and FISH analysis of aneuploid and diploid breast tumors. *Cold Spring Harbor Symposia on Quantitative Biology. and others* 70, 51-63(2005).

Hsu L, Self SG, Grove D, Randolph T, Wang K, Delrow JJ, Loo L, Porter P. Denoising array-based comparative genomic hybridization data using wavelets. *Biostatistics* 6, 211-226(2005).

Hup P, Stransky N, Thiery J, Radvanyi F, Barillot E. Analysis of array CGH data: from signal ratio to gain and loss of DNA regions. *Bioinformatics* 20, 3413-3422(2004).

Jenuwein T, Allis CD. Translating the histone code. *Science* 293, 10741080 (2001).

Johnson DS, Mortazavi A, Myers RM, Wold B. Genome-wide mapping of in vivo protein-DNA interactions. *Science* 316, 14971502 (2007).

Karolchik D, Hinrichs AS, Furey TS, Roskin KM, Sugnet CW, Haussler D, Kent WJ. The UCSC Table Browser data retrieval tool. *Nucleic Acids Res* 32, D4936 (2004).

Kolasinska-Zwierz P, Down T, Latorre I, Liu T, Liu XS, Ahringer J. Differential chromatin marking of introns and expressed exons by H3K36me3. *Nat. Genet.* 41, 376381 (2009).

Komura D, Shen F, Ishikawa S, Fitch KR, Chen W, Zhang J, Liu G, Ihara S, Nakamura H, Hurles ME. Genome-wide detection of human copy number variations using high density DNA oligonucleotide arrays. *Genome Research. and others* 10.1101/gr.5629106(2006).

Kouzarides T. Histone methylation in transcriptional control. *Curr. Opin. Genet. Dev.* 12, 198209 (2002).

Lai TL, Liu H, Xing H. Autoregressive models with piecewise constant volatility and regression parameters. *Statistica Sinica*15, 279-301(2005).

Lai TL, Xing H. A simple Bayesian approach to multiple change-points. *Statistica Sinica* (2011).

Lai WR, Johnson MD, Kucherlapati R, Park PJ. Comparative analysis of algorithms for identifying amplifications and deletions in array CGH data. *Bioinformatics* 21, 3763-3770(2005).

Li B, Carey M, Workman JL. The role of chromatin during transcription.*Cell* 128, 707719 (2007).

Metzker ML. Sequencing technologies - the next generation. *Nat Rev Genet* 11, 3146 (2010).

Mikkelsen TS, Ku M, Jaffe DB, Issac B, Lieberman E, Giannoukos G, Alvarez P, Brockman W, Kim TK, Koche RP, Lee W, Mendenhall E, O'Donovan A, Presser A, Russ C, Xie X, Meissner A, Wernig M, Jaenisch R, Nusbaum C, Lander ES, Bernstein BE. Genome-wide maps of chromatin state in pluripotent and lineage-committed cells. *Nature* 448, 553560 (2007).

Olshen AB, Venkatraman ES, Lucito R, Wigler M. Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics* 5, 557-572(2004).

Park PJ. ChIP-seq: advantages and challenges of a maturing technology. *Nat Rev Genet* 10, 669680 (2009).

Pauler FM, Sloane MA, Huang R, Regha K, Koerner MV, Tamir I, Sommer A, Aszodi A, Jenuwein T, Barlow DP. H3K27me3 forms BLOCs over silent genes and intergenic regions and specifies a histone banding pattern on a mouse autosomal chromosome. *Genome Res* 19, 221233 (2009).

Peng W, Zhao K. An integrated strategy for identification of both sharp and broad peaks from next-generation sequencing data. *Genome Biol* 12, 120 (2011).

Pepke S, Wold B, Mortazavi A.Computation for ChIP-seq and RNA-seq studies. *Nat Methods* 6, S2232 (2009).

Picard F, Robin S, Lavielle M, Vaisse C, Daudin J. A statistical approach for array CGH data analysis. *BMC Bioinformatics* 6, 27(2005).

Pinkel D, Albertson DG. Array comparative genomic hybridization and its applications in cancer. *Nature Genetics* 37, Suppl.:11-17(2005).

Pinkel D, Seagraves R, Sudar D, Clark S, Poole I, Kowbel D, Collins C, Kuo W-L, Chen C, Zhai Y. High resolution analysis of DNA copy number variation using comparative genomic hybridization to microarrays.*Nature Genetics. and others.* 20, 207-211(1998).

Pollack JR, Perou CM, Alizadeh AA, Eisen MB, Pergamenschikov A, Williams CF, Jeffrey SS, Botstein D, Brown PO. Genome-wide analysis of DNA copy-number changes using cDNA microarrays.*Nature Genetics* 23, 41-46(1999).

Pollack JR, Sorlie T, Perou CM, Rees CA, Jeffrey SS, Lonning PE, Tibshirani R, Botstein D, Brresen-Dale A, Brown PO. Microarray analysis reveals a major direct role of DNA copy number alternation in the transcriptional program of human breast tumors. *Proceedings of the National Academy of Sciences* 99(20), 12963-12968(2002).

Pruitt KD, Tatusova T, Klimke W, Maglott DR. NCBI Reference Sequences: current status, policy and new initiatives. *Nucleic Acids Res* 37, D326 (2009).

Qin ZS, Yu J, Shen J, Maher CA, Hu M, Kalyana-Sundaram S, Yu J, Chinnaiyan AM. HPeak: an HMM-based algorithm for defining read-enriched regions in ChIP-Seq data. *BMC Bioinformatics* 11, 369 (2010).

Ren B, Robert F, Wyrick JJ, Aparicio O, Jennings EG, Simon I, Zeitlinger J, Schreiber J, Hannett N, Kanin E, Volkert TL, Wilson CJ, Bell SP, Young RA. Genome-wide location and function of DNA binding proteins. *Science* 290, 23062309 (2000).

Robertson G, Hirst M, Bainbridge M, Bilenky M, Zhao Y, Zeng T, Euskirchen G, Bernier B, Varhol R, Delaney A, Thiessen N, Griffith OL, He A, Marra M, Snyder M, Jones S. Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nat Methods* 4, 651657 (2007).

Rozowsky J, Euskirchen G, Auerbach R, Zhang Z, Gibson T, Bjornson R, Carriero N, Snyder M, Gerstein MB. PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. *Nat Biotechnol* 27, 6675 (2009).

Sabo P, Kuehn MS, Thurman R, Johnson BE, Johnson EM, Cao H, Yu M, et al. Genome-scale mapping of DNase I sensitivity in vivo using tiling DNA microarrays. Nature methods, 3(7), 511518 (2006).

Snijders AM, Fridlyand J, Mans DA, Segraves R, Jain AN, Pinkel D, Albertson DG. Shaping of tumor and drug-resistant genomes by instability and selection. *Oncogene* 22, 4370-4379(2003).

Snijders AM, Nowak N, Segraves R, Blackwood S, Brown N, Conroy J, Hamilton G, Hindle AK, Huey B, Kimura K. Assembly of microarrays for genome-wide measurement of DNA copy number. *Nature Genetics. and others*29, 263-264(2001).

Song Q, Smith AD. Identifying dispersed epigenomic domains from ChIP-Seq data. *Bioinformatics* 27, 870871 (2011).

Stitzel ML, Sethupathy P, Pearson DS, Chines PS, Song L, Erdos MR, Welch R, Parker SC, Boyle AP, Scott LJ; NISC Comparative Sequencing Program, Margulies EH, Boehnke M, Furey TS, Crawford GE, Collins FS. Global epigenomic analysis of primary human pancreatic islets provides insights into type 2 diabetes susceptibility loci. *Cell Metab.* 12, 443455 (2010).

The ENCODE Project Consortiu A User's Guide to the Encyclopedia of DNA Elements (ENCODE). *PLoS Biol* 9, 21 (2011).

Wang P, Kim Y, Pollack J, Narasimhan B, Tibshirani R. A method for calling gains and losses in array-CGH data. *Biostatistics* 6, 45-58(2005).

Wen C, Wu Y, Huang Y, Chen W, Liu S, Jiang S, Juang J, Lin C, Fang W, Hsiung CA. A Bayes regression approach to array-CGH data. *Statistical Applications in Generics and Molecular Biology* 5, Article 3. and others(2006). Available at: http://www.bepress.com/sagmb/vol5/iss1/art3.

Widschwendter M, Fiegl H, Egle D, Mueller-Holzner E, Spizzo G, Marth C, Weisenberger DJ, Campan M, Young J, Jacobs I, Laird PW. Epigenetic stem cell signature in cancer. *Nat. Genet.* 39, 157158 (2007).

Willenbrock H, Fridlyand J. A comparison study: applying segmentation to arrayCGH data for downstream analyses. *Bioinformatics* 21, 4084-4091(2005).

Zang C, Schones DE, Zeng C, Cui K, Zhao K and Peng W. A clustering approach for identification of enriched domains from histone modification ChIP-Seq data. *Bioinformatics* 25, 19521958 (2009).

Zhang N, Siegmund D. A modified Bayes information criterion with applications to comparative genomic hybridization data. *Biometrics* 63, 22-32(2006).

Zhang Y, Liu T, Meyer CA, Eeckhoute J, Johnson DS, Bernstein BE, Nusbaum C, Myers RM, Brown M, Li W, Liu XS. Model-based Analysis of ChIP-Seq (MACS). *Genome Biol* 9, R137 (2008).

Zhang Z, Pugh BF. High-resolution genome-wide mapping of the primary structure of chromatin. *Cell* 144, 175186 (2011).