STATISTICAL METHODS AND SOFTWARE FOR CHIP-SEQ DATA ANALYSIS

By

Dongjun Chung

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy
(Statistics)

at the
UNIVERSITY OF WISCONSIN–MADISON
2012

Date of final oral examination: 7/24/2012

The dissertation is approved by the following members of the Final Oral Committee:
Sündüz Keleş. Associate Professor, Statistics
Emery Bresnick. Professor, Cell and Regenerative Biology
Colin Dewey. Associate Professor, Biostatistics and Medical Informatics
Michael Newton. Professor, Statistics
Kam-Wah Tsui. Professor, Statistics

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

ACKNOWLEDGMENTS

I would like to express my overwhelming gratitude to my advisor, Professor Sündüz Keleş, for her unconditional support, advice, and encouragement from very early stage of my research work. Her passion for research and sharp insights have been inspiration and motivation to me through my PhD studies. Her outstanding guidance has made every step in my research enjoyable and rewarding.

I thank my thesis committee members, Professor Emery Bresnick, Professor Colin Dewey, Professor Michael Newton, and Professor Kam-Wah Tsui, for their reading of my dissertation and providing valuable comments. I would like to thank Professor Emery Bresnick, Professor Robert Landick, and Professor Patricia Kiley, who generated interesting genomic data and shared sharp biological insights. It was great experience for me to work with their labs. I also would like to thank them for providing their precious time and valuable suggestions on my thesis work and their generous support for my job application.

I would like to thank previous and current members in Keleş' Research Group, Wahba's Thursday Research Group, and Multi-omics Research Group, who provided academically stimulating and exciting discussions on statistics, machine learning, and biology. I am grateful to my friends in the Department of Statistics for the emotional support, care and entertainment, with whom I could enjoy my life in Madison.

I would like to thank my wife, Jeeyeon Kim, who has shared the happy times and hard ones together with me. It is her love and support that accompanied me during my PhD studies. Lastly, and most importantly, I want to thank my parents, Young Sup Chung and Jun Im Song, for their unconditional love, support, and encouragement. This work is dedicated to them.

TABLE OF CONTENTS

Page

Page

APPENDICES

LIST OF TABLES

LIST OF FIGURES

Appendix
Figure Page

Appendix
Figure Page

Appendix
Figure Page

ABSTRACT

Chromatin immunoprecipitation followed by high throughput sequencing (ChIP-Seq) has been successfully used for genome-wide profiling of transcription factor binding sites, histone modifications, and nucleosome occupancy in many model organisms and humans. This thesis focuses on developing statistical methodologies and software to analyze ChIP-Seq data in an unbiased way.

This thesis is composed of three major parts. In the first part, we discuss statistical challenges in identification of binding events in repetitive regions. The state of the art for analyzing ChIP-Seq data relies only on using reads that map uniquely to a relevant reference genome (uni-reads). We developed CSEM, a general statistical approach for utilizing reads that map to multiple locations on the reference genome (multi-reads). Our computational and experimental results establish that multi-reads can be of critical importance for studying transcription factor binding in highly repetitive regions of genomes with ChIP-Seq experiments.

In the second part, we investigate statistical challenges in identification of closely spaced binding events. Because the compact prokaryotic genomes harbor binding sites some of which are separated by only a few base pairs, applications of ChIP-Seq in this domain have not reached their full potential. Although paired-end tag (PET) assay enables higher resolution identification of binding events than single-end tag (SET) assay, standard ChIP-Seq analysis methods are not equipped to utilize PET-specific features of the data. To address this problem, we developed dPeak, a high resolution binding site identification algorithm, that is applicable with PET and SET data. Our computational and experimental results show that when coupled with PET data, dPeak can identify closely spaced binding sites with high accuracy.

In the third part, we describe our three novel ChIP-Seq data analysis software, `csem`, `mosaics`, and `dpeak`. These three software address each of three important problems in ChIP-Seq data analysis, which are identification of binding events in repetitive regions, consideration of important sequence biases in peak calling, and identification of closely spaced binding events, respectively. Through applications to real ChIP-Seq data, we illustrate how these software can reveal novel biological insights that are currently ignored in standard ChIP-Seq data analysis.

Chapter 1

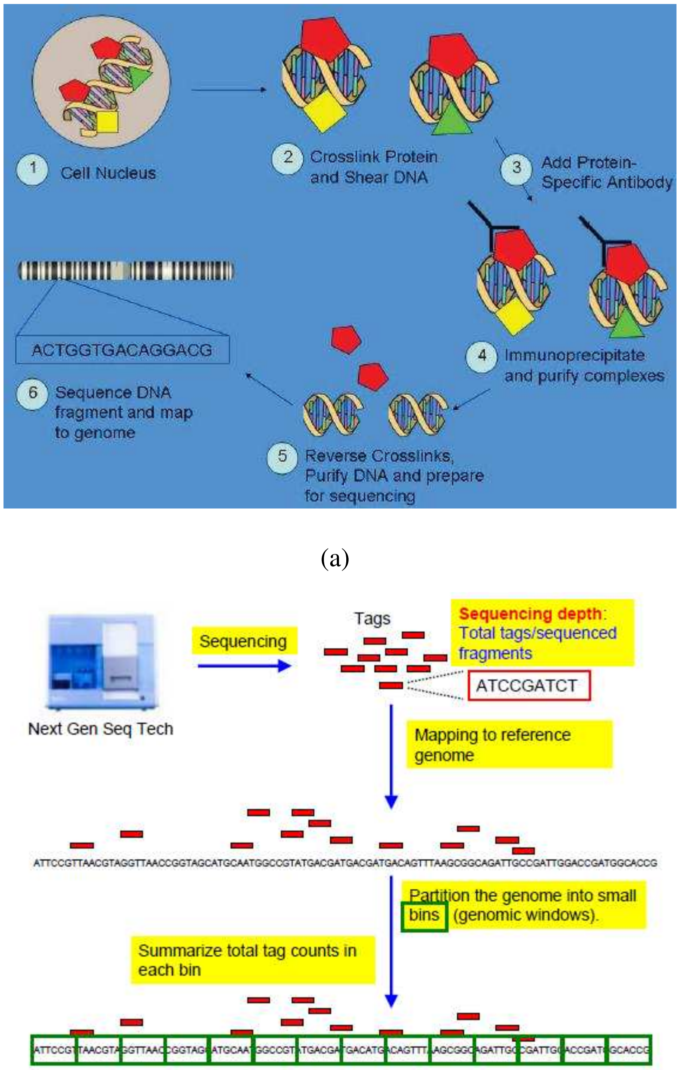
General Overview

1.1 Background

The introduction of next generation sequencing has enabled a myriad of creative ways to answer important biological questions in genome-wide scale. Chromatin immunoprecipitation coupled with high-throughput next generation sequencing (ChIP-Seq) has become a powerful technique for large scale profiling of transcription factor binding and chromatin modifications and is offering a powerful alternative to ChIP on microarrays (ChIP-chip). ChIP-Seq is currently one of the best methods for genome-wide investigation of protein-DNA interactions and has been widely used across a wide range of organisms (Mikkelsen et al., 2007; Barski et al., 2007; Johnson et al., 2007; Seo et al., 2009).

Figure 1.1 summarizes usual procedures to generate ChIP-Seq data. ChIP-Seq experiments capture millions of *DNA fragments* (150 $\sim$ 250 bp in length) that the protein under study interacts with, using random fragmentation of DNA and a protein-specific antibody. Then, high throughput sequencing of a small region (25 $\sim$ 100 bp) at one or both ends of each fragment generates millions of *reads* or *tags*. Sequencing one end and both ends are referred to as *single-end tag (SET)* and *paired-end tag (PET)* assays, respectively.

Standard preprocessing of ChIP-Seq data involves mapping reads to a reference genome and retaining the uniquely mapping ones (*uni-reads*) (Ji et al., 2008; Rozowsky et al., 2009). In PET data, paired reads from both ends of each DNA fragment are simultaneously used for the alignment and this, in turn, reduces the alignment ambiguity significantly and improves mapping rates of the reads. In spite of such advantages of using the PET assay, the SET assay dominates ChIP-Seq


1 Cell Nucleus
2 Crosslink Protein and Shear DNA
3 Add Protein-Specific Antibody
4 Immunoprecipitate and purify complexes
5 Reverse Crosslinks, Purify DNA and prepare for sequencing
6 Sequence DNA fragment and map to genome
ACTGGTGACAGGACG
Next Gen Seq Tech
Sequencing
Tags
Sequencing depth: Total tags/sequenced fragments
ATCCGATCT
Mapping to reference genome
ATTCCGTTAACGTAGGTTAACCGGTAGCATGCAATGGCCGTATGACGATGACGATGACAGTTTAAGCGGCAGATTGCCGATTGGACCGATGGCACCG
Partition the genome into small bins (genomic windows).
Summarize total tag counts in each bin
ATTCCGTTAACGTAGGTTAACCGGTAGCATGCAATGGCCGTATGACGATGACATGACAGTTTAAGCGGCAGATTGCCGATTGGACCGATGGCACCG

(a)

(b)

Figure 1.1 **Generation of ChIP-Seq data.** (a) ChIP-Seq experiments capture millions of *DNA fragments* (150 $\sim$ 250 bp in length) that the protein under study interacts with, using random fragmentation of DNA and a protein-specific antibody. (b) High throughput sequencing of a small region (25 $\sim$ 100 bp) at one or both ends of each fragment generates millions of *reads* or *tags*. Sequencing one end and both ends are referred to as *single-end tag (SET)* and *paired-end tag (PET)* technologies, respectively. Standard preprocessing of ChIP-Seq data involves mapping reads to reference genome and retaining the uniquely mapping ones (*uni-reads*). For visualization and analysis purposes, a genome is often partitioned into small bins (genomic windows) and total tag counts in each bin are summarized.

studies mainly because the PET assay costs 1.5 $\sim$ 2 times more than the SET assay. The PET assay is still considered only for specific applications such as studies of protein binding in repetitive regions.

After aligning reads to the reference genome, in PET data, length of each DNA fragment can be obtained by connecting positions of paired reads (Fullwood et al., 2009). In contrast, in the SET data, we do not know lengths of DNA fragments because we can observe location of only one end of each DNA fragment. Hence, for SET data, each read is either extended to or shifted by an estimate of the library size. This estimate is often specified as the average library size which is a parameter set in the experimental procedure (Rozowsky et al., 2009) or estimated from ChIP-Seq data (Zhang et al., 2008). For PET data, each connected read pair contributes to the whole region it overlaps (Kuan et al., 2011) or only certain part of the connected read pair (e.g., its midpoint or 5' end shifted by certain length) is considered (Zhang et al., 2008). For visualization and analysis purposes, a genome is often partitioned into small bins (genomic windows) and total tag counts in each bin are summarized. Finally, genomic regions with large numbers of aligning reads are identified as binding sites using one or more of the many available statistical approaches (Ji et al., 2008; Kuan et al., 2011; Rozowsky et al., 2009; Zhang et al., 2008, 2011).

1.2 Outline of the thesis

In this thesis, we discuss novel statistical methods (Chapter 2 and 3) and their software (Chapter 4) to analyze ChIP-Seq data. Each chapter is self-contained and includes relevant background for each statistical problem. This thesis is organized as follows.

In Chapter 2, we propose a statistical framework to utilize reads mapping to multiple locations on the genome (*multi-reads*). Specifically, we approach the multi-read problem as nonparametric estimation of a mixing density and propose a novel algorithm, CSEM. By applying the proposed algorithm to human STAT1 and mouse GATA1 data, we show that incorporating multi-reads leads to detection of novel peaks and majority of them are located in repetitive regions. We further validate these novel discoveries by experimental validation.

In Chapter 3, we propose a statistical framework to resolve closely spaced binding events. This project is motivated by the problem of identification of binding site in prokaryotic genomes, some of which are separated by only a few base pairs. In order to address this research question, we developed dPeak, a high-resolution binding site identification algorithm. The dPeak algorithm implements a probabilistic model that accurately describes each of PET and SET ChIP-Seq data generation process. Using simulation studies and analysis of *E. coli* σ^{70} data, we show that the dPeak algorithm coupled with PET data identifies each of closely spaced binding events while standard ChIP-Seq data analysis methods (window-based peak calling algorithms) do not provide resolutions sufficient for this problem. We further show that the dPeak algorithm outperforms its competing algorithms for SET ChIP-Seq data. Finally, we analytically investigate PET and SET ChIP-Seq data and discuss why PET data has advantages over SET data in resolution.

In Chapter 4, we describe three of our software that address important issues in ChIP-Seq data analysis. Specifically, we illustrate using the `csem` software (Chapter 2) (Chung et al., 2011) for multi-read allocation, using the `mosaics` software (Kuan et al., 2011) for peak calling that considers various sequencing biases inherent in ChIP-Seq data, and using the `dpeak` software (Chapter 3) for deconvolution of closely spaced binding events. This analysis workflow allows biological researchers investigate various issues in ChIP-Seq data analysis, which are usually ignored in practice. In addition, our software provide user-friendly interface and computationally efficient implementation. These software are available from public repository such as bioconductor and we also developed corresponding tools for Galaxy, an open web-based platform for genomic research, which provides user-friendly interface for integration of multiple bioinformatics tools.

Chapter 2

Identifying Protein-DNA Interaction Sites in Repetitive Regions

2.1 Introduction

The first step of ChIP-Seq data analysis is to map reads to the reference genome and retain reads that map to unique locations (*uni-reads*) (Blahnik et al., 2009; Ji et al., 2008; Rozowsky et al., 2009). Although constraining the analysis to uni-reads by discarding reads that map to multiple locations (*multi-reads*) leads to reduced coverage and sequencing depth, this may not render a serious problem in most cases. This is because many uni-reads might be adjacent to discarded multi-reads and can lead to identification of underlying peaks. However, discarding multi-reads poses a significant challenge for identifying binding locations residing in genomic regions that have been duplicated over evolutionary time since these regions will not have many uni-reads.

Shortcomings of discarding multi-reads have been recognized in transcriptome sequencing (RNA-Seq) (Faulkner et al., 2008; Li et al., 2010; Mortazavi et al., 2008; Taub et al., 2010). These studies demonstrated that discarding multi-reads leads to inaccurate estimation of expression of genes that reside in repetitive regions. Gene repetitiveness may be due to either low complexity segments or recent gene duplications. Numerous studies have highlighted the biological importance of segmental duplications (Bailey and Eichler, 2006; Marques-Bonet et al., 2009). Duplicated genes could retain their original functions or acquire new functions by changes in coding sequences and regulatory regions (Hurles, 2004; Rowen et al., 2005). Gonzalez et al. (2005) and Bailey et al. (2002) showed that segmental duplications in human genomes are selectively enriched for genes associated with disease susceptibility, immunity, and defense. Overall, these studies highlighted that annotating segmental duplications in terms of transcription factor binding might aid in understanding functions of genes within these regions. In addition, retrotransposon, a major class of transposable elements which duplicate through RNA intermediates that are reverse transcribed and are inserted at new genomic locations, also carry transcription factor binding sites that regulate gene expression (Polak and Domany, 2006; Roman et al., 2008). For example, Wang et al. (2007) showed with *in vivo* ChIP experiments that transcription factor p53 binds to human endogenous retrovirus (ERV) long terminal repeats (LTR), that are 100 bp to 5 kb long, with a p53 binding site.

There has been little work in the literature that investigates the effects of multi-reads in ChIP-Seq data analysis. Blahnik et al. (2009) provided an example of how discarding multi-reads could result in potential false negatives. Rozowsky et al. (2009) briefly discussed that by randomly assigning multi-reads to one of their mapping locations, one can increase the number of detected binding sites. Day et al. (2010) recently studied enrichment of the known repetitive elements as described by the Repbase database (Jurka et al., 2005) and the RepeatMasker (Smit et al., 2010) scans in ChIP-Seq data of histone modifications. Similarly, Wang et al. (2010) developed an algorithm for genomic mapping of ambiguous tags which increased read coverage for highly repetitive sequences. However, neither of these thoroughly investigated the effects of multi-reads on overall peak (binding location) detection. Currently, none of the popular ChIP-Seq data analysis software Blahnik et al. (2009); Ji et al. (2008); Kharchenko et al. (2008); Qin et al. (2010); Valouev et al. (2008); Zhang et al. (2008) takes multi-reads into account.

This chapter is organized as follows. In Section 2.2, we describe a statistical framework for utilizing multi-reads and discuss its implementation. Briefly, we consider the multi-read problem as a nonparametric estimation problem of a mixing density (Laird, 1978) and derive an Expectation-Maximization-Smoothing (EMS) algorithm (Silverman et al., 1990). In Section 2.3, we investigate the effects of discarding multi-reads on two different ChIP-Seq datasets: STAT1 binding in interferon-γ-stimulated HeLa S3 cells (Rozowsky et al., 2009) and GATA1 binding in mouse G1E-ER4 cells (Cheng et al., 2009). Briefly, incorporation of multi-reads can lead to an increase of up to 25% in the sequencing depth and identify high quality novel peaks. Location analysis of these peaks reveals that they are likely to be critical for constructing comprehensible genetic networks with members in repetitive regions of the genomes. Our computational experiments demonstrate that multi-reads can not only lead to detection of novel peaks in low mappable regions but also improve peak identification in moderate to highly mappable regions. We support our computational experiments by experimental validation of a subset of GATA1 peaks that were only identifiable when multi-reads were incorporated. This leads to identification of novel GATA1 target genes. In Section 2.4, we summarize our main findings and discuss implications of our studies.

2.2 Methods

2.2.1 Multi-read problem as nonparametric estimation of a mixing density

ChIP-Seq analysis is essentially based on the density generating reads at each position on a genome. Let n be sample size, i.e., sequencing depth, and m be the number of possible positions, i.e., number of nucleotide bases over the genome. Also, let i be the index for reads and j be the index for positions on the genome, i.e., $i = 1, \cdots, n$ and $j = 1, \cdots, m$. We define π to be the density function for generating reads and $\pi(j)$ to be the value of π at j-th position, i.e., the probability that a read is generated from j-th position. For notational convenience, $\pi(j)$ is also denoted as π_j. Let Z_i be a random variable indicating true origin of i-th read and $Z_i = j$ if i-th read is generated from j-th position, where $Z_i \in \{1, \cdots, m\}$. We assume that $Z \sim \pi$ and specifically, $P(Z = j) \equiv \pi(j)$. Here, our overall goal is to estimate density π.

In ChIP-Seq analysis, we do not observe Z_i directly because some reads can align to multiple positions on the genome (multi-reads). Hence, Z_i are un-observed (hidden) random variables. Instead, we observe $Y_i = (Y_{i1}, \cdots, Y_{im})$, mapping of i-th read, where $Y_{ij} = 1$ if i-th read aligns to j-th position on the genome, and 0 otherwise. Because we consider only alignable reads, at least one element of Y_i should be one. i-th read is called an uni-read if only one element of Y_i is one, and is called a multi-read if more than one elements of Y_i is one. Y_i are observed random variables. Let $h(y|z)$ be the conditional density of observable y given hidden z. If we consider each read as a mixture of reads that are originated from each position on the genome, i.e.,

$$p(y) = \sum_{j=1}^{m} h(y|z = j)\pi(j),$$

then multi-read problem can essentially be formulated as a nonparametric estimation of a mixing density, as in Laird (1978).

A common choice to estimate π is to set $\pi(j)$ as a constant π_j and estimate π_j at each position using the maximum likelihood (ML) estimation implemented using the Expectation-Maximization (EM) algorithm (Dempster et al., 1977). However, ML reconstruction of π is unsatisfactory because this ML estimation yields 'noisy' and 'spiky' estimates that do not fully reflect knowledge about the structure of the problem. Specifically, ChIP-Seq literature indicates that this density changes smoothly across the genome and reads located nearby correspond to one protein binding event (Kharchenko et al., 2008; Park, 2009; Zhang et al., 2008). Essentially, this unsatisfactory estimation problem occurs due to high dimensionality of the parameter space.

A *smoothed EM approach* or *EMS (expectation-maximization-smoothing) algorithm* (Silverman et al., 1990) provides an effective and efficient solution to overcome such limitations of the ML estimation for estimating a mixing density. Essentially, in the EMS algorithm, a smoothing step follows each M-step, to smooth the ML results obtained from the M-step. The EMS algorithm has been shown to be an effective tool for estimating a mixing density and it outperforms existing methods (Liu et al., 2009). The EMS algorithm has also been used in diverse applications (Becker et al., 1991; Becker and Marschner, 1993; Goedert et al., 2007; Hedgepeth et al., 1999; Silverman et al., 1990). Empirical studies of simulated and real data showed that the EMS algorithm usually converges more rapidly than the usual EM algorithm and the EMS algorithm results are less sensitive to starting configurations (Becker et al., 1991; Becker and Marschner, 1993; Silverman et al., 1990). Nychka (1990) showed that an approximation to the EMS algorithm can be viewed as a penalized likelihood approach.

2.2.2 The CSEM algorithm

Let $1\{A\}$ be an indicator function of event A For notational convenience, we define $Z_{ij} = 1\{Z_i = j\}$, i.e., $Z_{ij} = 1$ and $Z_{ij'} = 0, \forall j' \neq j$ if $Z_i = j$. Because we assume that a read can be originated from only one location on the genome, we have $\sum_{j=1}^{m} Z_{ij} = 1, \forall i = 1, \cdots, n$. Then, $(Z_{i1}, \cdots, Z_{im})$ independently follows a multinomial distribution with parameters $(\pi(1), \cdots, \pi(m))$, by the set-up in the previous section. Let A_i be the set of positions that i-th read aligns to (i.e., the alignment results) and we denote $j \in A_i$ if i-th read aligns to j-th position. Then, $Y_{ij} = 1$ if $j \in A_i$ and $Y_{ij} = 0$ otherwise.

In this section, we make further simplification assumptions as follows. We assume that we know all the bases in the reference genome (i.e., no ambiguous bases) and we can sequence every base on the reads without errors (i.e., no sequencing error). In addition, we consider only exact match in the alignment. Let's define S_j as a set of positions with k-mers identical to those starting at j-th position, where k is the read length. In this setting, if i-th read is generated from j-th position, then we should observe alignment of i-th read to all the positions belonging to S_j. Then, we have $h(y_i|z_i) = \prod_{j=1}^{m} \left[1\left\{y_{ij} = 1, j \in S_{z_i}\right\} + 1\left\{y_{ij} = 0, j \notin S_{z_i}\right\}\right]$. Under these assumptions, the E-, M-, and S-steps in the EMS algorithm are obtained as follows, where we denote $\pi_j = \pi(j)$ below.

E-step:

$$z_{ij}^{(t)} = \frac{\pi_j^{(t)}}{\sum_{j' \in A_i} \pi_{j'}^{(t)}} 1\left(j \in A_i\right). \qquad [2.1]$$

This E-step includes the following two special cases:

Special case 1: If i-th read does not align to j-th position, then $z_{ij}^{(t)} = 0$, i.e., $z_{ij}^{(t)}$ is zero at the positions that i-th read does not align to.

Special case 2: If i-th read is an uni-read and aligns to j-th position, then $z_{ij}^{(t)} = 1$, i.e., $z_{ij}^{(t)}$ equals one at a single position if i-th read is an uni-read.

M-step:

In the M-step of the EMS algorithm, we first obtain initial ML estimates for $\pi_j^{(t+1)}$, say $\mu_j^{(t+1)}$. If we replace π_j with μ_j in $\log L_c(\pi)$ and differentiate $\log L_c(\pi)$ with respect to μ_j under the sum constraint $\sum_{j=1}^m \mu_j = 1$, then we have $\mu_j^{(t+1)} = \frac{1}{n}\sum_{i=1}^n z_{ij}^{(t)}$.

S-step:

In the S-step of the EMS algorithm, we smooth $\mu^{(t+1)}$ to obtain $\pi^{(t+1)}$. This step can accommodate multiple choices of smoothing algorithms. Some possible choices are as follows. First, we can partition the genome into fixed non-overlapping *bins* and assume that π is constant within each bin. This approach corresponds to a *bin smoother* (Hastie and Tibshirani, 1990). In this case, we have

$$\pi_j^{(t+1)} = \frac{1}{w}\sum_{j' \in B_j} \mu_{j'}^{(t+1)} = \frac{1}{n}\sum_{i=1}^n \frac{1}{w}\sum_{j' \in B_j} z_{ij'}^{(t)},$$

where B_j is a bin that j-th position belongs to and w is the bin size.

Second, a *moving average* or a *running mean* (Hastie and Tibshirani, 1990) can be used, i.e., instead of fixing non-overlapping bins, we move a *window* by one base each time and estimate π at each position. The moving average approach was also used in peak calling algorithms for ChIP-chip experiments (Kuan et al., 2008). In this case, we have

$$\pi_j^{(t+1)} = \frac{1}{2w+1}\sum_{j'=j-w}^{j+w} \mu_{j'}^{(t+1)} = \frac{1}{n}\sum_{i=1}^n \frac{1}{2w+1}\sum_{j'=j-w}^{j+w} z_{ij'}^{(t)}, \qquad [2.2]$$

where w is size of a half window (number of nucleotide bases within half window). Equation 2.2 indicates that $\pi_j^{(t+1)}$ is the relative ratio of number of reads within the window of size $(2w + 1)$ around position j, among all reads.

Third, a *kernel regression estimator* (Fan and Gijbels, 1996) is another popular tool for smoothing. If we use the *Nadaraya-Watson estimator* (Nadaraya, 1964; Watson, 1964), one of the simplest types of kernel regression estimators, we estimate $\pi(j)$ as

$$\pi_j^{(t+1)} = \frac{\sum_{j'=1}^{m} K_w(j'-j)\mu_{j'}^{(t+1)}}{\sum_{j'=1}^{m} K_w(j'-j)} = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{\sum_{j'=1}^{m} K_w(j'-j)} \sum_{j'=1}^{m} K_w(j'-j) z_{ij'}^{(t)},$$

where $K_w(j'-j)$ is a *kernel* that assigns a (non-negative) weight to j' based on the distance between j' and j, with a *bandwidth* w. We assume that $\int K_w(u)du = 1$ and K is an even function. In the case of an uniform kernel, this estimator coincides with the moving average estimator.

Our proposed framework provides connection to other multi-read allocation methods. First, the uniform allocation method allocates each multi-read to its possible locations uniformly. The uniform allocation method corresponds to our approach assuming uniform distribution for read generation, i.e., $\pi_1 = \cdots = \pi_m = \pi$. Second, the proportional allocation method allocates each multi-read to its possible positions, where allocated fraction to each position is proportional to the counts of uni-reads within the window around each of possible locations. The proportional allocation method corresponds to the first iteration of our approach.

2.2.3 Implementation

A simplified version of the multi-read allocation algorithm is summarized in Algorithm 1, where we use a moving average method for the S-step. For Step 1 in Algorithm 1, we used the Bowtie aligner (Langmead et al., 2009) to align reads against the appropriate reference genome (human HG18 or mouse MM9 in our studies). The parameters for Bowtie were set such that for each read, all alignments with at most 2 mismatches were reported. Reads with 100 or more such alignments were filtered out. During the algorithm, we maintain a list of (possibly non-integer) counts of the number of reads assigned at each position in the genome. Strand information for each alignment was ignored such that the left-most coordinate of each alignment was defined as its starting position. The algorithm starts with a uniform π and we allow at most 200 iterations.

Algorithm 1. Multi-read allocation algorithm. n and m denote the number of reads and the number of positions (number of nucleotide bases) on the genome, respectively. w is a half window size which is a tuning parameter of the algorithm.

```
1. A short-read alignment tool is used to establish a set of candidate alignments for each read against the reference genome.
2. Initialize $\pi_j = 1/m$ for all positions $j = 1, \cdots, m$ and $z_{ij} = 0, \forall i = 1, \cdots, n, \forall j = 1, \cdots, m$.
3. Until convergence,
    (a) For each read $i = 1, \cdots, n$, if $i$-th read has $a_i$ possible starting positions, $s_1, \cdots, s_{a_i}$, then update $z_{is_l}$ as $\pi_{s_l} / \sum_{k=1}^{a_i} \pi_{s_k}$, $l = 1, 2, \cdots, a_i$.
    (b) For each position $j = 1, \cdots, m$, update $\pi_j$ as $\sum_{i=1}^{n} \sum_{j'=j-w}^{j+w} z_{ij'} / n(2w+1)$.
```

In our simplified multi-read allocation algorithm, the choice of w (a half window size of the moving average estimator) controls the degree at which multi-read allocation is affected by uni-reads. Therefore, setting $2w + 1 \approx L$, where L is the expected fragment size (200 bp for both STAT1 and GATA1 datasets), ensures that uni-reads and multi-reads within a given window correspond to the same binding event. We considered setting w to 25, 50, and 100 bp, respectively, and observed that, although there is high overlap among the peak sets obtained with different w ($>$ 70 % overlap), $w = 100$ captures the largest number of true positives and the smallest number of false positives in our validation set.

2.3 Results: Human STAT1 and Mouse GATA1 ChIP-Seq Data

2.3.1 Contribution of multi-reads to the sequencing depth of ChIP-Seq data

The two datasets, STAT1 binding in interferon-γ-stimulated HeLa S3 cells (Rozowsky et al., 2009) and GATA1 binding in mouse GATA1-null erythroid cells (G1E-ER4) after genetic complementation with a conditionally active allele of GATA1 (ER-GATA1) (Cheng et al., 2009), and their input DNA controls were downloaded from GEO (`http://www.ncbi.nlm.nih.gov/geo/`) (accession numbers GSM320736, GSM320737, GSM453997, GSM453998 for STAT1 ChIP and input, and GATA1 ChIP and input samples, respectively). Data from different lanes within an experiment were pooled together. The STAT1 dataset has a higher sequencing depth than most published ChIP-Seq datasets and is therefore especially suited for studying the effects of multi-reads. Both datasets utilize single end short reads (30mers for STAT1 and 36mers for GATA1), which is still the state of the art for ChIP-Seq experiments.

In Table 2.1 are the total number of reads, percentages of alignable reads, uni-reads, multi-reads, and the rescued-reads (gain in sequencing depth by incorporating multi-reads) for both of the datasets we study. We observe that utilizing multi-reads leads to an increase of 22% (25%) and 17% (19%) in the sequencing depths of STAT1 and GATA1 ChIP (input) samples, respectively. The increase in sequencing depths due to multi-reads is substantial for short read datasets. The last four rows in Table 2.1 present results for longer reads (unpublished longer read datasets are courtesy of Prof. Qiang Chang at UW-Madison). Summaries on MECP2-SET ChIP and input datasets (75mer single end tags (SETs) from mouse) indicate that multi-reads still constitute a significant issue even with longer reads and they can lead to an increase in sequence depth comparable to the increase in short read datasets. The last two rows are from an experiment with 75mer paired-end tags (PETs) in mouse. Although there is a significant drop in the percentage of multi-reads, utilizing these reads increases the sequencing depth by 7% for these 75mer PETs datasets.

Table 2.1 **Impact of multi-reads on sequencing depth.** In the first column, "(C)" and "(I)" refer to ChIP and input samples, respectively. Percentages in the third to fifth columns are calculated with respect to the total number of reads (the second column). "% Rescued" in the last column is obtained as the number of multi-reads divided by the number of uni-reads and it indicates the gain in sequencing depth due to multi-reads.

Dataset	# of reads	% Alignable	% Uni-reads	% Multi-reads	% Rescued
STAT1(C)	76,913,219	36.64	29.92	6.72	22.46
STAT1(I)	49,771,625	47.90	38.31	9.59	25.03
GATA1(C)	33,124,216	79.27	67.81	11.46	16.90
GATA1(I)	20,711,007	82.37	69.38	12.99	18.73
MECP2-SET(C)	15,253,906	79.23	65.06	14.16	21.76
MECP2-SET(I)	21,870,009	90.35	78.14	12.21	15.63
MECP2-PET(C)	18,622,331	68.55	64.24	4.31	6.70
MECP2-PET(I)	18,498,899	84.26	78.92	5.34	6.77

2.3.2 Novel discovery by utilizing multi-reads

We next evaluated the effect of increase in sequencing depth due to multi-reads in terms of peak detection. We divided the genome into small non-overlapping intervals, i.e., bins, of size 50 - 250 bp as in CisGenome (Ji et al., 2008) for the downstream analysis of peak detection. We excluded bins which consisted of only the ambiguous base N. Then, for each factor, two bin-level datasets were created using (1) uni-reads only (UR sample) and (2) both uni-reads and multi-reads (MR sample). Further preprocessing involved extending each read to the expected fragment length (200 bp for both datasets) as in Rozowsky et al. (2009) and summarizing the total number of reads overlapping each bin. The bin size was selected to match the expected fragment length. The final bin counts were rounded to the nearest integer for modeling purposes since fractional counts were not continuous enough for fitting with a continuous distribution such as the Gamma distribution. We also considered applications of the ceiling and floor functions to fractional counts as alternatives to rounding. These provided upper and lower bounds on the number of reads obtainable from fractional counts, respectively. The overlap of the peak sets under these three strategies were more than 95%. Matching input control samples of the ChIP-Seq data were processed similarly to generate matching UR and MR input samples.

We analyzed UR and MR bin-level data for each experiment using our recently developed method MOSAiCS (Kuan et al., 2011) to identify peaks. This method accounts for non-specific sequence biases such as mappability (Rozowsky et al., 2009) and GC content (Dohm et al., 2008). It performs comparable to or better than some of the commonly used peak finders such as MACS (Zhang et al., 2008), CisGenome (Ji et al., 2008), and PeakSeq (Rozowsky et al., 2009). Another reason for using MOSAiCS is that currently none of the peak finders readily allow incorporation of multi-reads. The MOSAiCS model fits the UR and MR samples well.

The final peak lists were obtained by controlling the false discovery rate (FDR) at level 0.05 and filtering out bins with less than 30 ChIP tag counts. Conclusions presented below remain robust to various choices of this tag count threshold. Since the number and the quality of the peaks rely on the FDR level used, we first implemented a sensitivity analysis to evaluate the recovery rate of the UR and MR analysis peaks. We declared peaks at FDR levels of 0.005, 0.01, 0.05, 0.1, and 0.2 and classified the peaks detected at FDR level of 0.005 as UR-only (specific to UR analysis), MR-only (specific to MR analysis), and common peaks. This resulted in 23424 and 3378 MR-only peaks for STAT1 and GATA1, respectively. Then, we evaluated the percentage of the MR-only peaks identified at FDR level of 0.005 and recovered by the UR analysis at higher FDR levels. We did not calculate the recovery rate of UR-only peaks by the MR analysis since the numbers of UR-only peaks were negligible (2 for STAT1 and 10 for GATA1). Figure 2.1 displays the results of the sensitivity analysis. As the FDR level increases, the UR analysis can at most recover 30% and 5% of the MR-only peaks for STAT1 and GATA1, respectively.

Figures 2.2a,b display two examples of MR-only peaks with their MR ChIP, MR input, UR ChIP, UR input, and mappability tracks. The first is a STAT1 peak (Figure 2.2a) that resides in a poorly mappable region with a peak level mappability of 0.04 and therefore cannot be recovered by the UR analysis that relies only on uni-reads. The second is a GATA1 peak (Figure 2.2b) and it is located in a region with moderate mappability (average peak level mappability of 0.72). It is

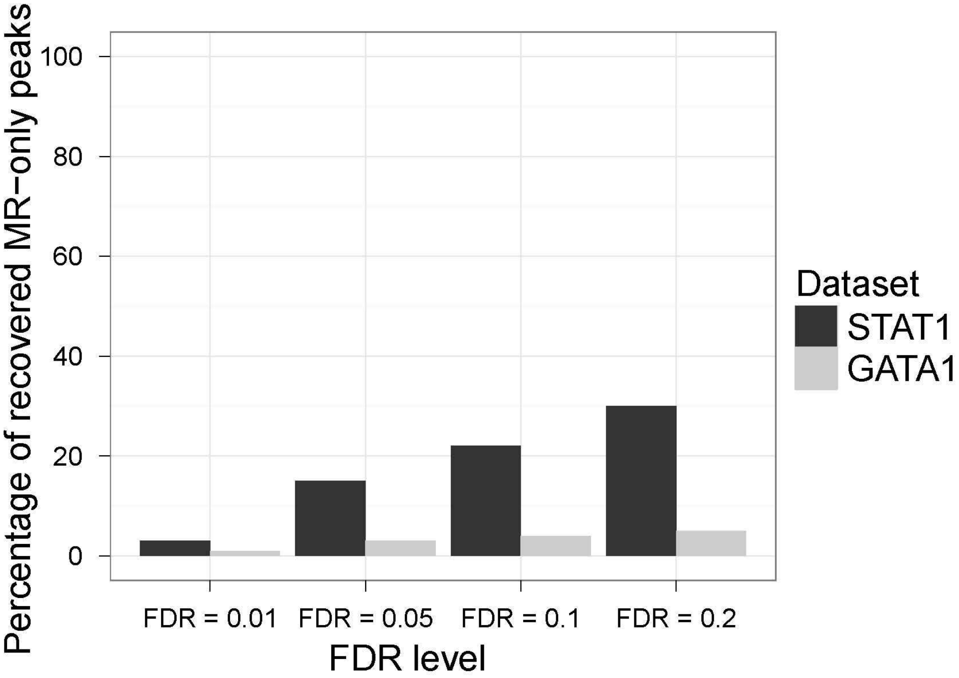
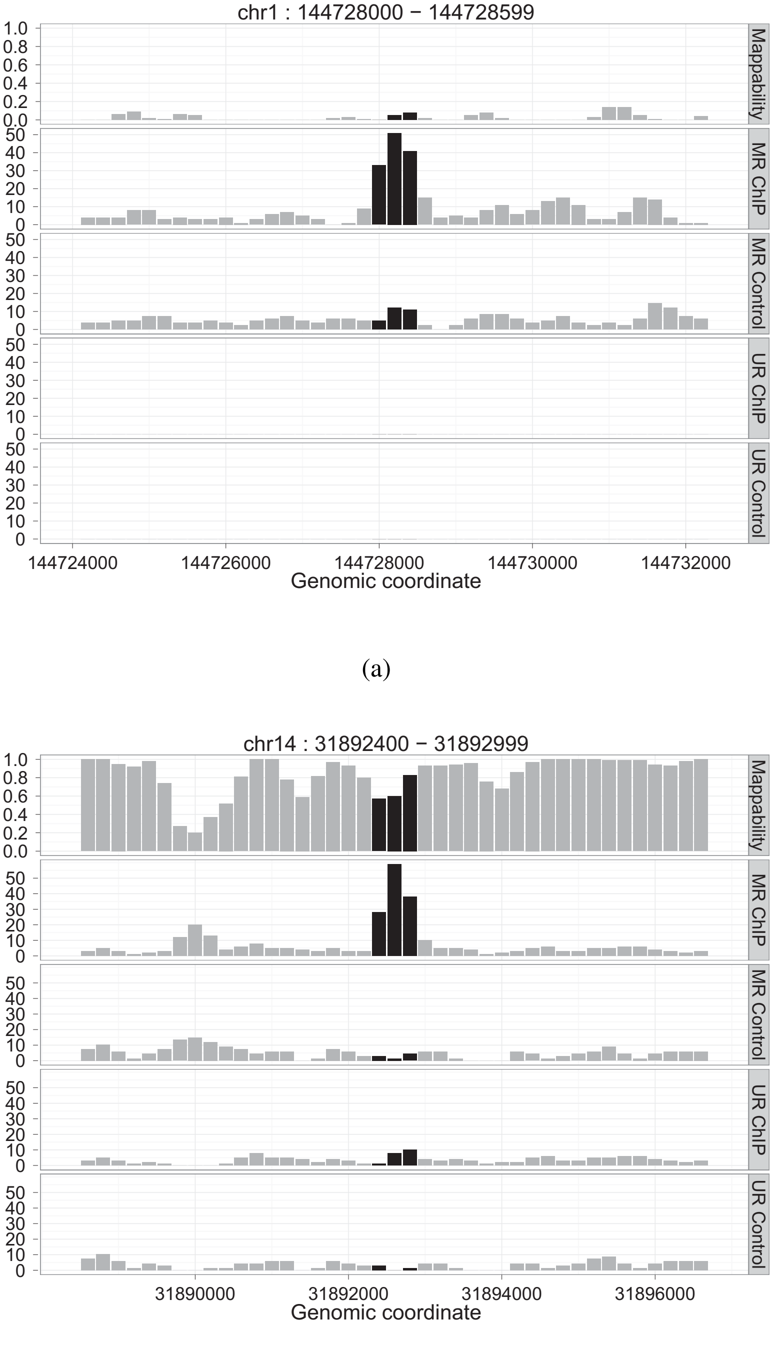


Figure 2.1 **Sensitivity analysis.** "Recovered MR-only peaks" refer to MR-only peaks that are defined at FDR level of 0.005 and are detectable by the UR analysis at higher FDR levels.

not identified as a peak in the UR analysis; however since the MR analysis boosts up the tag count of the region by utilizing multi-reads, this peak reaches the required statistical significance level in the MR sample. Of the MR-only STAT1 and GATA1 peaks, 32% and 74% are located in regions with mappability below 0.5, therefore these peaks are not likely to be detected by UR analysis regardless of the sequencing depth.

To further quantify the advantage of incorporating multi-reads beyond novel peaks that are not identifiable with only uni-reads, we performed the following computational experiment for the STAT1 sample that had higher sequencing depth and was therefore more suitable for this experiment. The idea is similar to the study of saturation in ChIP-Seq experiments where the number of identified peaks is plotted as a function of the sequencing depth (Kharchenko et al., 2008; Rozowsky et al., 2009). We defined peaks identified using all the uni-reads as the UR gold standard peak set. Then, we constructed smaller datasets by sampling from uni-reads and multi-reads and identified peaks using these datasets with lower sequencing depths. Figure 2.3 plots the percentage



(a)

(b)

Figure 2.2 **Tag count profiles of MR-only peaks with corresponding mappability scores.** (a) STAT1 MR-only peak in a poorly mappable region. (b) GATA1 MR-only peak in a moderately mappable region. Peak regions are depicted with black bars.

of the gold standard UR peaks identified at lower sequencing depths by the UR and MR analysis. We observe that utilizing multi-reads recovers UR gold standard peaks at a faster rate than using only uni-reads. In particular, peak calling using MR sample recovers all the UR gold standard peaks using only 80% of the full dataset. This experiment further solidifies the gains in sequencing depth in Table 2.1 by illustrating the practical utility of multi-reads in terms of peak finding. When we performed a similar experiment using MR peaks from the full dataset as the gold standard peak set, a significant percentage of MR-only peaks were not detected using only UR sample at any sequencing depth.

For the rest of the comparisons among the MR and UR peak sets, we focused on the highest quality peaks called at FDR level of 0.05 by further filtering UR-only or MR-only peaks. A peak identified only in the MR (UR) analysis is labelled as an MR-(UR-)only peak if its corresponding UR (MR) read count is less than 20 making it highly unlikely for the UR (MR) analysis to identify this peak as a high quality peak. This filtering is further justified by examining MR-only peaks with low and high UR ChIP read counts in more detail. MR-only peaks with low UR ChIP read counts exhibit stronger signal than those with high UR ChIP read counts in the MR sample. Among the STAT1 MR-only peaks, the peaks with low UR ChIP read counts are ranked higher than MR-only peaks with high UR ChIP read counts based on their posterior probability of ChIP enrichment in the MR peak list. Moreover, the average log base 2 bin-level ratio of ChIP over input tag counts of MR-only peaks with low UR ChIP tag counts is 1.72 while those of MR-only peaks with high UR ChIP tag counts is 0.89 (enrichments are computed after scaling ChIP and input to the same total sequencing depth within each sample). Results are similar for the GATA1 MR-only peaks.

Table 2.2 summarizes the final number of peaks retained in the subsequent downstream analysis. There are no UR-only peaks and multi-reads identify 11% and 36% more high quality peaks for STAT1 and GATA1, respectively. In order to assess the robustness of these results to the peak calling algorithm used, we implemented the conditional binomial (CB) test of CisGenome (Ji et al., 2008) to handle multi-reads. We processed the CB peaks identified at FDR level of 0.05 with the same procedure applied to MOSAiCS peaks and arrived at the same conclusion: although a large fraction of the peaks are common between UR and MR analysis, MR analysis identifies

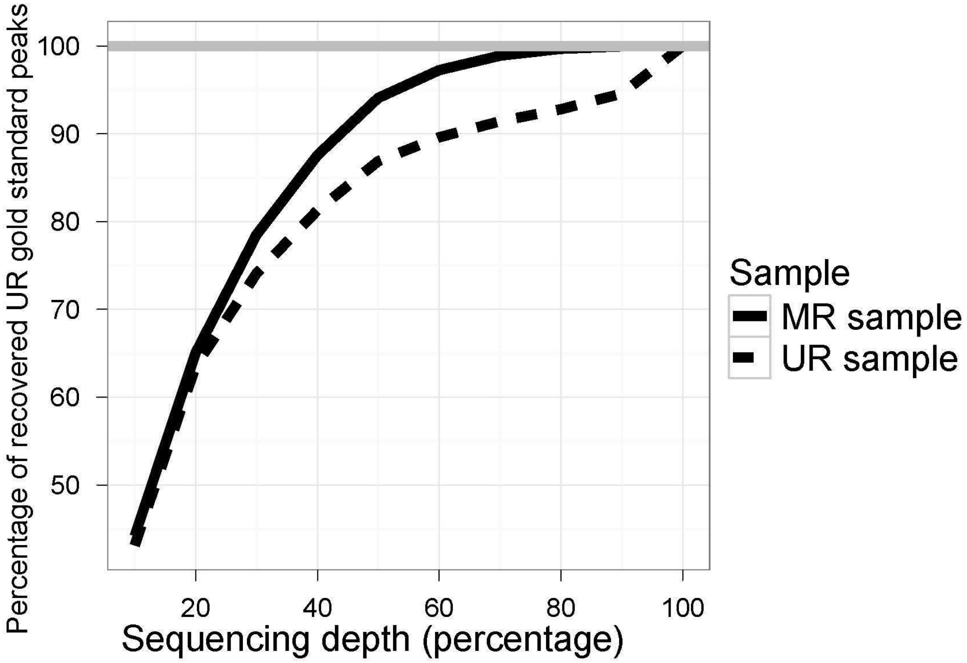


Figure 2.3 **Saturation plot of the STAT1 sample.** Percentage of STAT1 UR gold standard peaks recovered by MOSAiCS using sub-sampled UR and MR samples with lower sequencing depths. x-axis refers to the percentage of reads sampled from the full dataset.

a significant number of additional peaks. We further compared GATA1 MR-only peaks detected by MOSAiCS with the MACS peaks that were reported in (Cheng et al., 2009) and utilized only uni-reads. Only 127 out of 2146 MOSAiCS MR-only peaks (6%) were in the MACS peak list. In contrast, 97% of MOSAiCS common peaks (5878 out of 6038) were in the MACS list. The 127 MR-only peaks that were detectable by MACS had an average mappability of 0.72 which was significantly higher than the average mappability of 0.31 for the peaks that were not detectable (p-value $< 2.2\text{e}^{-16}$). Furthermore, these MACS detectable MR-only peaks had a median ranking of 851 with an interquartile range (IQR) of 609 among the 2146 MR-only peaks, indicating that they are not the strongest signal MR-only peaks. These peaks were also detectable by the MOSAICS analysis of the UR sample at a higher FDR level.

Table 2.2 **Summary of UR and MR peaks detected by MOSAiCS.**

Dataset	# of UR-only peaks	# of common peaks	# of MR-only peaks
STAT1	0	23175	2546
GATA1	0	6038	2146

2.3.3 Properties of MR-only peaks

Figure 2.4 displays boxplots of mappability (left panel) and GC content (middle panel) of STAT1 common and MR-only peaks (Similar patterns are observed in GATA1 peaks). The GC content levels are comparable between MR-only and common peaks; however, as expected, MR-only peaks cover a much broader range of mappability and have, on average, lower mappability than common peaks. Next, we investigated how the MR-only peaks would be affected by using longer reads because larger fractions of genomes become uniquely mappable when longer reads are utilized (Rozowsky et al., 2009). To assess this, we studied how mappability changes when 75mer SETs are used instead of 36mer SETs. Even though mappability improves significantly when longer reads are utilized, indicating that these peaks might eventually become detectable with the uni-read analysis, more than 50% of the GATA1 MR-only peaks still reside in low mappability regions even with 75mer SETs. The median mappability of GATA1 MR-only peaks increase from 0.27 (IQR = 0.40) to 0.67 (IQR = 0.59) when 75mer SETs are used instead of the 36mer SETs.

We also compared the peak sets in terms of their enrichments for the known binding consensus sequence of the corresponding factors. We scanned the STAT1 peaks with FIMO (Bailey et al., 2009) using the two known STAT1 position weight matrices from JASPAR (Portales-Casamar et al., 2010). For GATA1, we counted the occurrence of the consensus motif [A/T]GATA[A/G] (Evans et al., 1988) in the peak regions. Right panel of Figure 2.4 displays the STAT1 motif occurrences of common and MR-only peaks. Results are similar for GATA1. Motif enrichments in the MR-only peak sets are lower compared to the enrichments observed for the common peaks. This is potentially due to uncertainty in the mapping of reads that contribute to these peaks. However,

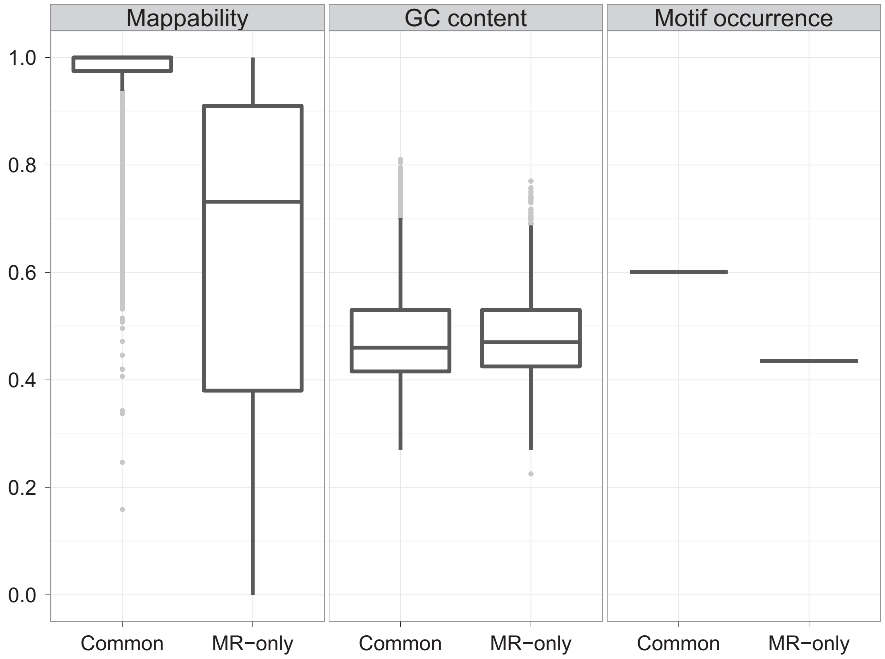


Figure 2.4 **Mappability, GC content, and STAT1 motif occurrence of the STAT1 common and MR-only peaks.** "Common" refers to common peaks identified by both the MR and the UR samples; "MR-only" peaks are unique to the MR sample. For the motif occurrence panel, y-axis represents the proportion of peaks with the consensus binding site.

the observed motif enrichments in both the STAT1 and the GATA1 MR-only peak sets are much higher than one would expect by chance (both p-values $<< 1\text{e}^{-4}$).

2.3.4 Biological annotation of MR-only and common peaks

As we discussed in the Introduction, there is a growing literature that highlights the biological importance of segmental duplications (Gonzalez et al., 2005; Bailey et al., 2002; Nicholas et al., 2009). One of the findings is that segmental duplications are enriched for genes involved in immunity and, therefore, could be potential targets for transcription factor binding. We next assessed to what extent common and MR-only peaks appear in segmental duplications of the genomes. For this analysis, we utilized segmental duplication data from the UCSC Genome Browser database

(Rhead et al., 2010) and carried out a location analysis on the peak lists. Pie charts in Figure 2.5 display location analysis results for STAT1 and GATA1, respectively. MR analysis identifies peaks in all categories. The percentages of MR-only peaks that are in the “None” category are not drastically different from that of the common peaks. A large percentage of MR-only peaks reside in segmental duplication regions (54.91% for STAT1 and 60.58% for GATA1) with a substantial amount located in promoter (10.60% and 4.19% for STAT1 and GATA1, respectively) and genic regions of genes (15.71% and 14.17% for STAT1 and GATA1) within these segmental duplications. Next, we annotated the peaks in the ”None” category in terms of interspersed repeats and low complexity DNA sequences in the human and mouse genomes utilizing RepeatMasker (Smit et al., 2010). For STAT1, 67% of the 8782 common peaks and 95% of the 667 MR-only peaks map to at least one of these types of repeats. In particular, MR-only peaks are enriched in the long terminal repeats (LTR) category compared to common peaks. Percentages of peaks in the LTR category are 22.6% and 58.5% for the common and MR-only peaks, respectively. For GATA1, 54% of the 1347 common peaks and 76% of the 526 MR-only peaks map to at least one of these types of repeats. In addition, MR-only peaks are enriched in the long interspersed repetitive elements (LINE) (9.3% of the common peaks, 22.6% of the MR-only peaks) and LTR (16.5% of the common peaks, 45.6% of the MR-only peaks) categories compared to common peaks. In contrast, common peaks are enriched in simple repeat and short interspersed repetitive elements (SINE) category. Percentages of common peaks among the ”None” category that are in simple repeat and SINE categories are 12.5% and 24.3% compared to 7.2% and 7.4% for the MR-only peaks in these categories.

To further explore STAT1 peaks in UR and MR samples in terms of segmental duplications, we compared the average tag count profiles at promoters of expressed genes (Auerbach et al., 2009) in duplicated and unduplicated regions. Of the 10913 expressed genes, 1862 (17%) are located within segmental duplications. Both the UR and MR samples have comparable tag counts at promoters of expressed genes in unduplicated regions. In contrast, STAT1 ChIP MR sample exhibits increased signal at promoters in duplicated regions relative to input MR sample. Specifically, 1 kb regions around the transcription start site (TSS) of the expressed genes in duplicated regions gain on average 24.69 and 9.93 tags in MR ChIP and input samples, respectively, compared to UR ChIP and input samples. In contrast, the gains in unduplicated regions are only 3.08 and 0.39 tags compared to UR ChIP and input samples.

We used the DAVID tools of Dennis et al. (2003) and Huang et al. (2009) to further annotate the MR-only peaks. For STAT1, we applied DAVID to the group of 102 expressed genes with at least one MR-only peak and no common peaks in their promoters. This analysis revealed significant enrichment of these genes for response to DNA damage, transcription activity, regulation of gene expression, apoptosis, programmed cell death, and intercellular signaling cascade. For GATA1, the set of expressed genes with an MR-only peak was too small. Instead, we applied DAVID to the set of genes with at least one MR-only peak and no common peaks within 10 kb upstream of TSS and 2 kb downstream of transcription end site (TES) excluding exons. DAVID analysis of such 340 genes identified significant enrichment for immune/defense response, immune system development, regulation of apoptosis, hemopoiesis, and SAND domain. These results agreed well with the observation that the segmental duplications are selectively enriched for genes associated with immunity and defense (Gonzalez et al., 2005; Bailey et al., 2002).

2.3.5 Experimental validation of MR-only peaks

We selected 13 GATA1 MR-only peaks for experimental validation for GATA1 occupancy with quantitative ChIP assays and real-time PCR. Peaks selected for validation contained a [A/T]GATA[A/G] motif, resided within promoter or genic regions of a RefSeq gene, and had a mappability value between 0.5 and 1. Eighteen percent of the GATA1 MR-only peaks satisfied the two former requirements. The mappability constraint was necessary for designing unique primers for the real-time PCR analysis of the peaks. We observe a significant increase in GATA1 occupancy in +EST compared to -EST for these MR-only GATA1 targets. In addition to our validation experiments of MR-only peaks, we also performed validation experiments for 7 [A/T]GATA[A/G] sites that resided in low mappable regions and were not predicted to be MR peaks. None of these

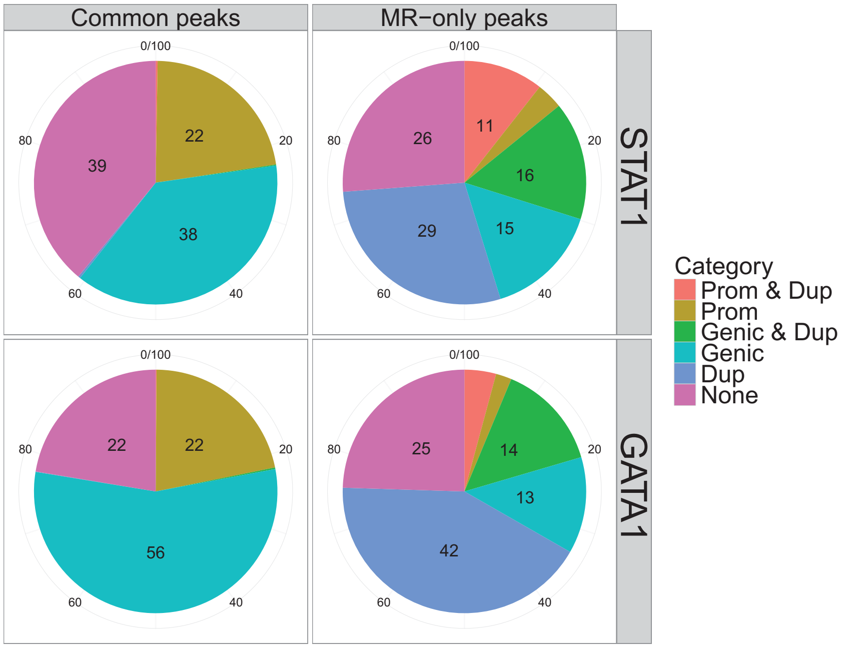


Figure 2.5 **Annotation of common and MR-only peaks with respect to TSS and duplicated regions.** Categories are: Prom & Dup: peaks that are in promoter regions (± 2.5 kb of TSS) of RefSeq genes that reside in segmental duplications; Prom: Peaks in promoter regions (excludes peaks in Prom & Dup); Genic & Dup: peaks that are within [-10 kb of TSS, +1 kb of TES] of RefSeq genes that are in segmental duplications (excludes peaks in Prom & Dup); Genic: peaks that are within [-10 kb of TSS, +1 kb of TES] of RefSeq genes (excludes peaks in Genic & Dup, Prom, and Prom & Dup); Dup: peaks that are in segmental duplications (excludes Prom & Dup and Genic & Dup) ; None: peaks that do not fall into any of the other defined categories. Numbers within the pie charts indicate the percentages of peaks in each category.

7 regions exhibited an increase in GATA1 occupancy in +EST compared to -EST, confirming that they are true negatives.

We next analyzed the expression of the genes corresponding to the above validated peaks in 24 hr β-estradiol treated G1E-ER-GATA1 cells using microarray data generated from β-estradiol treated and control cells as described in Fujiwara et al. (2009). Upon β-estradiol-treatment and GATA1 activation, these genes exhibited a fold change of 0.9 to 4.9 in expression. This confirms that GATA1 binds to these MR-only peaks and triggers expression of their corresponding genes during GATA1 mediated maturation of pro-erythroblasts. Even though these genes are not direct erythroid maturation factors, the megacaryocyte-erythrocyte progenitors express these factors at substantial levels as evidenced in the BioGPS analysis (Wu et al., 2009). These validated genes are chromatin modifiers, m-RNA splicing factors, zinc finger proteins and are mainly involved in transcriptional regulation and signal transduction. They may further contribute to the expression of erythroid specific genes/factors after being activated in the early phase of erythroid maturation.

2.4 Discussion

We investigated the shortcomings of discarding multi-reads in ChIP-Seq analysis and illustrated how incorporating multi-reads can improve detection of binding sites in highly repetitive regions of genomes. Multi-reads lead to identification of novel binding sites that are located in highly repetitive and low mappability regions and are not identifiable with uni-reads alone. They further contribute to effective utilization of uni-reads so that more peaks can be detected with lower sequencing depths. Utilizing location analysis and biological annotation, we further showed that a substantial fraction of peaks specific to multi-read analysis are located in segmental duplications of the human and mouse genomes, and attributed to genes that are well associated with immunity and defense.

Since multi-reads arise automatically in ChIP-Seq experiments, our analysis pipeline does not require any additional experiments for utilizing these reads. Once multi-reads are appropriately converted into counts by an application of the multi-read allocation algorithm, peak calling might be performed with any method that can handle bin or nucleotide level count data; however, since many of existing software start the analysis with aligned or raw tag files, these would need to be modified. To accommodate some of the existing software that rely on aligned read files (or alignment results in the bed format), we developed a script that rounds the multi-read weights to the nearest integer and adds the ones that round up to 1 to the original alignment files as pseudo reads so that they can be utilized. This procedure is equivalent to (1) allocating each multi-read to the location that it maps to with the largest weight; (2) filtering out multi-reads with weights < 0.5 since they round to 0; and (3) ignoring weight information (degree of confidence for multi-read allocation). Although this implementation decreases the number of utilized multi-reads by about a half (for GATA1), it still leads to a significant increase in the sequencing depth compared to using uni-reads alone. An application of this strategy with the MACS algorithm (Zhang et al., 2008) was able to identify 37% of the MR-only peaks identified by the MOSAiCS MR analysis. This set included the 3 true positive and 2 false positive peaks that we validated with the quantitative real time ChIP analysis.

We showed that the overall conclusions of utilizing multi-reads agree well when peak calling is performed either with MOSAiCS (Kuan et al., 2011) or CisGenome's conditional binomial model (Ji et al., 2008). Almost all of the the published ChIP-Seq studies in GEO (`http://www.ncbi.nlm.nih.gov/geo/`) utilize short reads (25-36mers) and we have observed that multi-reads can lead to a substantial increase in the sequencing depths of such datasets. A thorough investigation of peaks that were detectable only with multi-reads highlighted that a significant fraction of these peaks still have low mappability even when 75mer SETs are used. Therefore, utilization of multi-reads is also likely to improve the analysis of data from longer and/or paired-end reads. The two factors we studied are not particularly known to bind to repetitive regions. However, there are many examples of DNA binding proteins that selectively bind to repetitive regions, e.g., MECP2, KAP1. Our analysis pipeline should even have a higher impact in the analysis of such datasets. We have initially implemented our methodology for use with ChIP-Seq data from the Illumina Genome Analyzer platform; however, it is straightforward to adapt it for use with other high-throughput sequencing platforms.

A related question is whether utilizing a more flexible definition of uni-reads by relaxing the alignment criterion can provide a similar utility to that of multi-reads in terms of sequencing depth. Our computational experiments (data not shown) indicate that more flexible definitions of uni-reads (e.g. single best alignment of each read with at most 3 mismatches) can increase the sequencing depth and lead to identification of more peaks. However, such applications fail to identify high signal MR-only peaks that a multi-read analysis can identify. For example, defining uni-reads by considering the single best alignment of each read with at most 3 mismatches increases the sequencing depth by 12 % for GATA1 but can identify only 2.6 % of the GATA1 MR-only peaks.

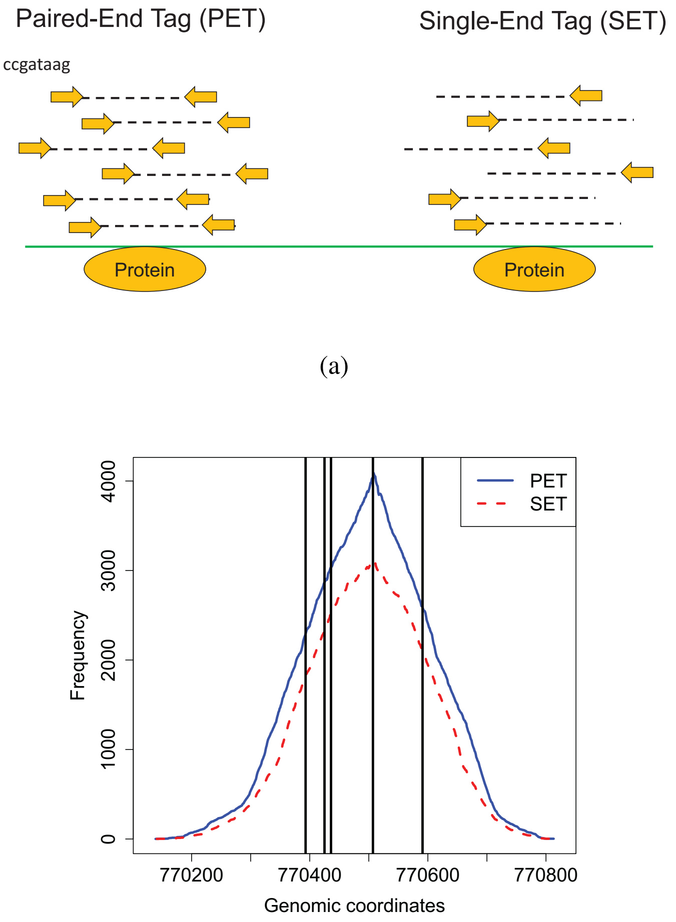
Significant fractions of eukaryotic genomes are composed of repetitive regions, e.g., more than half of the human genome. Therefore, functional properties of the repetitive regions of genomes are of significant biological interest. In particular, genomic repeats play important roles in functioning and evolution of transcriptional regulatory networks (Feschotte, 2008; Bourque et al., 2008). Bourque et al. (2008) illustrated that binding sites embedded in genomic repeats are associated with significant regulatory expansions throughout the mammalian phylogeny. Therefore, analysis and/or re-analysis of available or future ChIP-Seq datasets with our multi-read approach is expected to reveal fundamental insights into functional properties of highly repetitive regions of the genomes.

Chapter 3

Resolving Closely Spaced Protein-DNA Interaction Events

3.1 Introduction

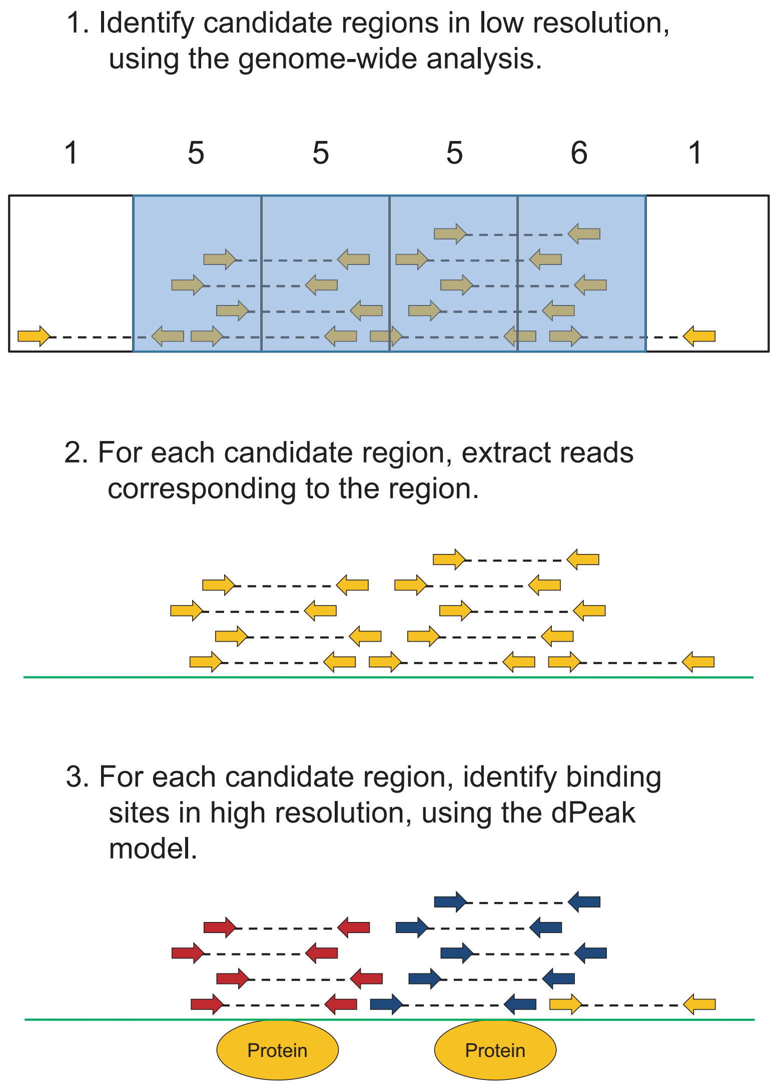
Since its introduction, chromatin immunoprecipitation followed by high throughput sequencing (ChIP-Seq) has revolutionized the study of gene regulation. ChIP-Seq is currently one of the best methods for studying protein-DNA interactions genome-wide and has been widely used across a wide range of organisms (Mikkelsen et al., 2007; Barski et al., 2007; Johnson et al., 2007; Seo et al., 2009). ChIP-Seq experiments capture millions of *DNA fragments* (150 $\sim$ 250 bp in length) that the protein under study interacts with, using random fragmentation of DNA and a protein-specific antibody. Then, high throughput sequencing of a small region (25 $\sim$ 100 bp) at one or both ends of each fragment generates millions of *reads* or *tags*. Sequencing one end and both ends are referred to as *single-end tag (SET)* and *paired-end tag (PET)* technologies, respectively (Fig. 3.1a). Standard preprocessing of these data involves mapping reads to reference genome and retaining the uniquely mapping ones (Ji et al., 2008; Rozowsky et al., 2009). In PET data, start and end positions of each DNA fragment can be obtained by connecting positions of paired reads (Fullwood et al., 2009). In contrast, location of only one end of each DNA fragment is known in the SET data. The usual practice for SET data is to either extend each read to the average library size which is a parameter set in the experimental procedure (Rozowsky et al., 2009) or shift start position of each read by an estimate of the library size (Zhang et al., 2008). Then, genomic regions with large numbers of aligning fragments are identified as binding sites using one or more of the many available statistical approaches (Ji et al., 2008; Kuan et al., 2011; Rozowsky et al., 2009; Zhang et al., 2008, 2011) (the first step in Fig. 3.2).



(a)

(b)

Figure 3.1 **SET and PET ChIP-Seq data structure.** (a) Description of paired-end tag (PET) and single-end tag (SET) ChIP-Seq data. Directions of arrows indicate strands of reads. (b) Promoter region of the cydA gene contains five closely spaced σ^{70} binding sites. Blue solid and red dotted curves indicate the number of DNA fragments and extended reads mapping to each genomic coordinate in σ^{70} PET and SET ChIP-Seq data, respectively. Black vertical lines mark σ^{70} binding sites annotated in the RegulonDB database.



1. Identify candidate regions in low resolution, using the genome-wide analysis.

2. For each candidate region, extract reads corresponding to the region.

3. For each candidate region, identify binding sites in high resolution, using the dPeak model.

Figure 3.2 **Pictorial depiction of the dPeak algorithm.**

Currently, the SET assay dominates all the ChIP-Seq experiments despite the fact that PET has several obvious, albeit less studied, advantages over SET for genome-wide binding site identification. In PET data, paired reads from both ends of each DNA fragment are simultaneously used for the alignment. This, in turn, reduces the alignment ambiguity significantly and improves mapping rates of the reads. This is especially advantageous for studying regulatory roles of repetitive regions of genomes. Although many eukaryotic genomes, such as human and mouse, are rich in repetitive elements, PET technology has not been extensively used with eukaryotic genomes (Chung et al., 2011; Fullwood et al., 2009). One of the main reasons for this is that ChIP-Seq data is information rich even when the repetitive regions are not profiled (Chen et al., 2012) and that PET assay costs 1.5 $\sim$ 2 times more than the SET assay. Put differently, given a fixed cost, PET sequencing results in a lower sequencing depth compared to SET sequencing.

In contrast to eukaryotic genomes, prokaryotic genomes are highly mappable, e.g., 97.8 % of the *Escherichia coli* (*E. coli*) genome is mappable with $32bp$ reads. This decreases the higher mapping rate appeal of the PET assay in the study of prokaryotic genomes. In this chapter, we systematically investigate advantages of the PET assay from a new perspective and demonstrate that it significantly improves the resolution of protein binding site identification. Improving resolution in identifying protein-DNA interaction sites is a critical issue in the study of prokaryotic genomes because prokaryotic transcription factors have closely spaced binding sites, some of which are only 10 to $100bp$ apart from each other (Bulyk et al., 2004; Mendoza-Vargas et al., 2009; Reznikoff et al., 1985). These closely spaced binding sites are considered to be multiple "switches" that differentially regulate gene expression under diverse growth conditions (Mendoza-Vargas et al., 2009). Therefore, identification and differentiation of closely spaced binding sites are invaluable for elucidating the transcriptional networks of prokaryotic genomes.

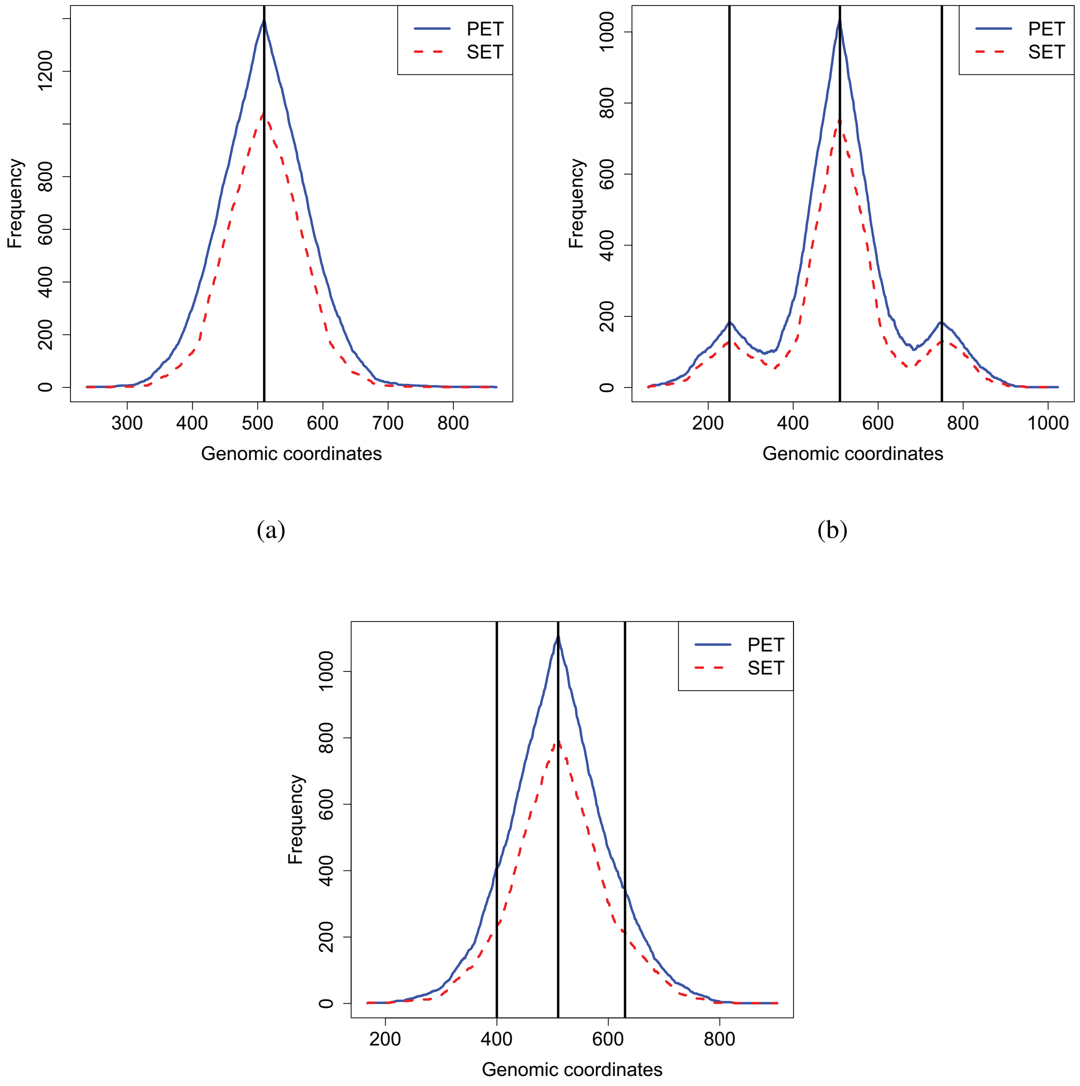
Although many methods have been proposed to identify peaks from ChIP-Seq data (reviewed in Wilbanks and Facciotti (2010)), such as MACS (Zhang et al., 2008), CisGenome (Ji et al., 2008), and MOSAiCS (Kuan et al., 2011), these approaches reveal protein binding sites only in low resolution, i.e., at an interval of hundreds to thousands of base pairs. Furthermore, they report only one "mode" or "predicted binding location" per peak. More recently, deconvolution algorithms such as CSDeconv (Lun et al., 2009), GPS (Guo et al., 2010), and PICS (Zhang et al., 2011) have been proposed to identify binding sites in higher resolution. However, these methods are specific to SET ChIP-Seq data and are not equipped to utilize main features of PET ChIP-Seq data. Motivated by the limitation of currently available methods, we developed dPeak, a high resolution binding site identification (deconvolution) algorithm that can utilize both PET and SET ChIP-Seq data.

This chapter is organized as follows. In Section 3.2, we describe a statistical framework for high resolution binding site identification and propose the dPeak algorithm. Briefly, the dPeak algorithm implements a probabilistic model that accurately describes ChIP-Seq data generation process and analytically illuminates the differences in resolution between the PET and SET ChIP-Seq assays. In Section 3.3, we demonstrate that dPeak outperforms SET-specific competing algorithms such as PICS and GPS. More importantly, we illustrate that dPeak coupled with PET ChIP-Seq data improves the resolution of binding site identification significantly in comparison to dPeak coupled with SET data. In Section 3.4, we analyze σ^{70} factor PET and SET ChIP-Seq data from *E. coli* grown under aerobic and anaerobic conditions as a case study. The results reveal the power of dPeak algorithm in identifying closely located binding sites. This study further emphasizes the importance of high resolution binding site identification when studying the same factor under diverse biological conditions. In Section 3.5, we summarize our main findings and discuss implications of our studies.

3.2 Methods

3.2.1 Motivation: *E. coli* σ^{70} ChIP-seq data

The σ^{70} factor is the transcription initiation factor for house keeping genes in *E. coli*. Kiley Lab (Department of Biomolecular Chemistry, University of Wisconsin-Madison) generated both PET and SET ChIP-Seq data of σ^{70} factor from *E. coli* grown under aerobic ($+O_2$) and anaerobic ($-O_2$) conditions. PET experiments yielded 13.8 million (M) and 22.3M 36mer paired reads and SET yielded 7.4M and 11.5M mappable 32mer reads for aerobic and anaerobic conditions on the Illumina GA II platform, respectively. Control input experiments, generated with SET sequencing, resulted in 4.6M and 10.2M mappable reads for the aerobic and anaerobic conditions, respectively.



(a)

(b)

(c)

Figure 3.3 **Coverage plots of simulated read data generated based on cydA promoter parameters estimated by dPeak:** (a) single binding event; (b, c) three binding events. dPeak analysis of PET data under aerobic condition generated three binding event predictions for the cydA promoter region. Distances between these binding events are 110 *bp* and 120 *bp*, respectively. The numbers of DNA fragments corresponding to each event are 180, 1035, and 180 (total of 1395), respectively. (a) One simulated binding event with 1395 DNA fragments. (b) Three simulated binding events at locations 250, 510, and 750, and with numbers of DNA fragments 180, 1035, and 180. (b) Three simulated binding events at locations 400, 510, and 630, and with numbers of DNA fragments 180, 1035, and 180.

Experiment	PET		SET	
	MACS	MOSAiCS	MACS	MOSAiCS
$+O_2$	270/3202/22	950/450/11.3	534/2550/34	1023/450/11.3
$-O_2$	132/4327/14	993/450/11.8	469/2890/34	1014/450/11.4

Table 3.1 **Analysis of the PET and SET data with MACS and MOSAiCS.** Reported numbers a/b/c refer to a: number of peaks; b: median peak width; c: genome coverage.

Experiment	Mean (Std)
PET, $+O_2$	1.82 (0.93)
PET, $-O_2$	2.23 (1.10)
SET, $+O_2$	1.54 (0.80)
SET, $-O_2$	1.65 (0.93)

Table 3.2 **Mean number of MOSAiCS peaks overlapping each MACS peak.**

We use these experimental data for comparisons of PET and SET assays and evaluation of our high resolution binding site detection method, dPeak, throughout the chapter.

Fig. 3.1b displays PET and SET ChIP-Seq coverage plots for the promoter region of the $cydA$ gene under the aerobic condition. The height at each position indicates the number of DNA fragments mapping to that position. $cydA$ promoter contains five known σ^{70} binding sites with distances between consecutive sites ranging from 11 to 84 bp (Gama-Castro et al., 2011). As evidenced in Fig. 3.1b, coverage plots for PET and SET appear almost indistinguishable visually. To further understand the nature of peaks that multiple binding events in this region would result in, we simulated read data with parameters matching to those of this region. Figures 3.3a,b,c display SET and PET coverage plots of this region when it harbors one, three, and three binding events, respectively. These plots support that when the binding events are in close proximity and the distances among them are less than the average library size, they appear as uni-modal peaks regardless of the library preparation protocol (Fig. 3.3c).

We next evaluated two peak callers, MACS (Zhang et al., 2011) (version 1.3.4) and MOSAiCS (Kuan et al., 2011) (version 1.4.0), both of which are specifically developed for SET data, on our SET and PET datasets (Table 3.1). In the versions of MACS and MOSAiCS that we consider for the current analysis, each method is modified so that it can handle PET ChIP-Seq data as follows. For PET ChIP-Seq data, MACS first finds the best pairs of $5'$ and $3'$ reads from multiple alignment results. Then only $5'$ read position is kept for every pair and shifted to its $3'$ direction by $100bp$ without estimation of the shift parameter. Then, the standard MACS analysis (Zhang et al., 2008) is applied to the processed data. In MOSAiCS, when bin-level files are constructed, each read pair is connected and this connected read pair contributes to all the bins it overlap. The standard MOSAiCS analysis (Kuan et al., 2011) is applied to this bin-level data.

The analysis results indicate that both methods identified broad regions. The median width of MACS peaks were 5 to 10 times larger than that of the MOSAiCS peaks. Detailed comparison of the MACS and MOSAiCS peaks revealed that each MACS peak on average has 1.54 to 2.23 MOSAiCS peaks (Table 3.2). Next, we evaluated the number of annotated σ^{70} binding events from RegulonDB (Gama-Castro et al., 2011) (`http://regulondb.ccg.unam.mx/`) in each of the MACS and MOSAiCS peaks and found that MACS peaks, on average, had 1.86 to 2.02 annotated binding events whereas MOSAiCS peaks had 1.47 to 1.48. Overall, we did not observe any differences in the peak sizes of the PET and SET assays with MOSAiCS whereas MACS peaks from PET data tended to be wider than those of the SET data. These findings indicate that the potential advantages of the PET assay for elucidating closely located binding sites are not simply revealed from visual inspection and by analysis with methods developed specifically for SET data. Hence, deciphering the advantages of PET over SET for high resolution binding site identification warrants a statistical assessment. Furthermore, these findings motivate development of our algorithm, dPeak, that can specifically utilize local read distributions from SET and PET assays. This algorithm enabled unbiased evaluation of the SET and PET assays using our *E. coli* SET and PET ChIP-Seq data.

3.2.2 The dPeak model

Consider a candidate region with n DNA fragments (i.e., n paired reads for PET data and n reads for SET data) and let 1 and m denote the start and end positions of this region, respectively. Let g^* denote the number of binding events within the region and μ_g be the position of g-th binding event, $g = 1, 2, \cdots, g^*$. Without loss of generality, assume that $1 \leq \mu_1 < \mu_2 < \cdots < \mu_{g^*} \leq m$ for identifiability. In both PET and SET data, some reads can be generated from background and we assume that background distribution is the uniform distribution over the whole candidate region. The background component is denoted as $g = 0$.

Let π_g denote the strength of g-th binding event, $g = 0, 1, 2, \cdots, g^*$, where π_0 indicates degree of non-specific binding in the candidate region. Let Z_i be group index of i-th DNA fragment and $Z_i \in \{0, 1, 2, \cdots, g^*\}$. For notational convenience, let's denote $Z_{ig} = 1\{Z_i = g\}$, where $1\{A\}$ is an indicator function of event A. We assume that $P(Z_i = g) = P(Z_{ig} = 1) = \pi_g$, $g = 0, 1, 2, \cdots, g^*$ and $\sum_{g=0}^{g^*} \pi_g = 1$.

Generative model for paired-end tag (PET) data

Let S_i and L_i be the start position and length of i-th DNA fragment, respectively. If we denote end position of i-th fragment as E_i, then $E_i = S_i + L_i - 1$ by definition. In the PET data, we directly observe S_i and E_i (equivalently, S_i and L_i) for each DNA fragment. Moreover, distribution of library size, $P(L)$, can be empirically estimated from the PET data and hence, we will assume that $P(L)$ is known. We denote the region corresponding to g-th binding event as $B_g = [\mu_g - L_i + 1, \mu_g]$ and the whole candidate region as $C = [1 - L_i + 1, m]$ for i-th DNA fragment of length L_i. This is because the leftmost position of DNA fragment can occur between $1 - L_i + 1$ and m, given that its length is L_i. If i-th fragment is generated from g-th binding event ($Z_i = g$), then for given L_i, we assume that S_i is generated from the uniform-like distribution

$$f(s|l; \mu_g, \gamma) = \left[\frac{(1-\gamma)}{l}\right]^{1\{s \in B_g\}} \left[\frac{\gamma}{m}\right]^{1\{s \in C \setminus B_g\}}, \qquad [3.1]$$

where γ denote the weight assigned to the area outside of the region corresponding to g-th binding event. We can summarize fragment generating process as follows:

1. Draw group index of the DNA fragment, $(Z_{i0}, Z_{i1}, Z_{i2}, \cdots, Z_{ig^*})$, from $Multinomial(1, (\pi_0, \pi_1, \pi_2, \cdots, \pi_{g^*}))$.
2. Draw library size, L_i, from $P(L)$.
3. Draw start position of the DNA fragment, S_i, conditional on Z_i and L_i:
   (a) If this DNA fragment belongs to g-th binding event ($Z_{ig} = 1, 1 \leq g \leq g^*$), draw start position of the fragment, S_i, from the density 3.1.
   (b) If this DNA fragment is from background ($Z_{i0} = 1$), draw S_i from Uniform$(1 - L_i + 1, m)$.

Generative model for single-end tag (SET) data

In the SET data, one of two ends of each DNA fragment is randomly selected and sequenced. Hence, L_i for each fragment is not observable; however, position and strand of the read corresponding to one end of each DNA fragment are observed (denoted by R_i and D_i, respectively). We denote $D_i = 1$ if i-th read is in the forward strand and $D_i = 0$ if i-th read is in the reverse strand. We assume that D_i follows Bernoulli distribution with known parameter p_D and p_D can empirically be estimated as proportion of forward strand reads in SET data.

Exploratory analysis indicates that read distributions can be well approximated by normal distribution. Specifically, we assume that

$$(R|Z = g, D = 1; \mu_g, \delta, \sigma^2) \sim N(\mu_g - \delta, \sigma^2),$$

and

$$(R|Z = g, D = 0; \mu_g, \delta, \sigma^2) \sim N(\mu_g + \delta, \sigma^2).$$

Note that δ corresponds to the half of the distance between modes of the forward and reverse strand reads belonging to each binding event. We can summarize SET read generating process as follows:

1. Draw group index of the read, $(Z_{i0}, Z_{i1}, Z_{i2}, \cdots, Z_{ig^*})$, from $Multinomial(1, (\pi_0, \pi_1, \pi_2, \cdots, \pi_{g^*}))$.

2. Draw strand of the read, D_i, from $Bernoulli(p_D)$.

3. Draw position of the read, R_i, conditional on Z_i and D_i:

   (a) If this read belongs to g-th binding event ($Z_{ig} = 1, 1 \leq g \leq g^*$) and it is in the forward strand ($D_i = 1$), draw position of the read, R_i, from $N(\mu_g - \delta, \sigma^2)$.

   (b) If this read belongs to g-th binding event ($Z_{ig} = 1, 1 \leq g \leq g^*$) and it is in the reverse strand ($D_i = 0$), draw position of the read, R_i, from $N(\mu_g + \delta, \sigma^2)$.

   (c) If this read is from background ($Z_{i0} = 1$) and it is in the forward strand ($D_i = 1$), draw position of the read, R_i, from Uniform$(1 - \beta + 1, m)$.

   (d) If this read is from background ($Z_{i0} = 1$) and it is in the reverse strand ($D_i = 0$), draw position of the read, R_i, from Uniform$(1, m + \beta - 1)$.

3.2.3 The dPeak algorithm

We can estimate parameters of the model for each PET and SET data using the Expectation-Maximization (EM) algorithm (Dempster et al., 1977). In our EM implementation, we considered following important issues for efficient computation and stable estimation. First, we do not have explicit solutions in the M-step for the PET model. Maximization with respect to $(\mu_1, \mu_2, \cdots, \mu_{g^*})$ requires searching over g^*-dimensional space and $O(m^{g^*})$ operations, which is computationally prohibitive. In order to boost up computation and stabilize estimation, we employ the Expectation-Conditional-Maximization (ECM) algorithm (Meng and Rubin, 1993). By employing the ECM algorithm, we only need to search over one-dimensional space, $[1, m]$, for the maximization with respect to each μ_g (while other parameters are fixed). This reduces the computation time to $O(mg^*)$ operations. Simulation studies shows that this approach is computationally efficient and provides fast convergence with accurate and stable estimation (data not shown). We have explicit solutions in the M-step for the SET model.

Second, although the EM algorithm has desirable convergence properties, it does not guarantee convergence to the global maximum when there are multiple maxima. As a result, the final estimates depend upon the initial values (McLachlan and Peel, 2000; McLachlan and Krishnan, 2008).

In order to address this issue, we consider the stochastic EM algorithm (Celeux and Diebolt, 1985), which is a special case of Monte Carlo EM (McLachlan and Peel, 2000; McLachlan and Krishnan, 2008), for the first half of iterations. The stochastic EM algorithm allows a chance of escaping from a current path of convergence to a local maximizer to other paths (McLachlan and Peel, 2000). After certain number of iterations, we switch to the ordinary version of our EM algorithm because the stochastic EM is not desirable when the process is near to convergence to a suitable local maximizer (McLachlan and Peel, 2000).

Third, in the EM implementation, non-identifiability due to overfitting (fitting too many components in the model) is problematic and should be avoided (Crawford, 1994; McLachlan and Peel, 2000). In order to address this issue, during the EM iterations, if the distance between two binding events is shorter than the size of the binding site (defined by the length of the consensus motif), we combine these two components and consider it as one component during the remaining iterations. For the σ^{70} application, we set this parameter to $20bp$ since σ^{70} binds to $-35bp$ and $-10bp$ from transcription start site. Moreover, if strength of a binding event is too weak ($\pi_g < 0.01$), this component is also removed from further consideration in the remaining iterations.

The dPeak algorithm for PET data

The EM algorithm for the PET data can be summarized as follows. Derivation of the EM algorithm is given in Appendix A.

E-step:

For $g = 1, 2, \cdots, g^*$,

$$z_{ig}^{(t)} = \frac{\pi_g^{(t)}}{A} \left[\frac{(1 - \gamma^{(t)})}{L_i} \right]^{1\left\{S_i \in B_g^{(t)}\right\}} \left[\frac{\gamma^{(t)}}{m-1} \right]^{1\left\{S_i \in C \setminus B_g^{(t)}\right\}},$$

and for $g = 0$,

$$z_{i0}^{(t)} = \frac{\pi_0^{(t)}}{A(m + L_i - 1)},$$

where A is an appropriate normalizing constant.

M-step:

For $g = 1, 2, \cdots, g^*$, we obtain

$$\mu_g^{(t+1)} \quad = \quad argmax_{\mu_g} \sum_{i=1}^{n} z_{ig}^{(t)} \left[1\left\{S_i \in B_g\right\} \log \frac{(1-\gamma^{(t)})}{L_i} + 1\left\{S_i \in C \backslash B_g\right\} \log \frac{\gamma^{(t)}}{m-1} \right].$$

and

$$\pi_g^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} z_{ig}^{(t)}.$$

Similarly,

$$\pi_0^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} z_{i0}^{(t)}.$$

Moreover,

$$\gamma^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} \sum_{g=1}^{g^*} z_{ig}^{(t)} 1 \left\{ S_i \in C \backslash B_g^{(t+1)} \right\}.$$

This algorithm has the following intuitive interpretation. In the E step, each DNA fragment is allocated to each binding event or background, based on whether the fragment overlaps positions of binding events or not. When the fragment overlaps with more than one binding event, it is assigned to each of these binding events in a fractional manner, where the fractions are proportional to relative strengths of each binding event (π_g). In the M step, location of each binding event (μ_g) is essentially updated to the position that the DNA fragments corresponding to this binding event align to the most. In this step, fragments with shorter library size (L_i) have more voting power. This is intuitive from the experimental procedure point of view because it is easier to identify the actual position of a binding event with shorter fragments.

The dPeak algorithm for SET data

The EM algorithm for the SET data can be summarized as follows. Derivation of the EM algorithm is given in Appendix B.

E-step:

For $g = 1, 2, \cdots, g^*$,

$$z_{ig}^{(t)} = \frac{\pi_g^{(t)}}{A\sqrt{2\pi(\sigma^2)^{(t)}}} \left[p_D \exp\left\{-\frac{1}{2(\sigma^2)^{(t)}}(R_i - (\mu_g^{(t)} - \delta^{(t)}))^2\right\}\right]^{1\{D_i=1\}} \left[(1-p_D)\exp\left\{-\frac{1}{2(\sigma^2)^{(t)}}(R_i - (\mu_g^{(t)} + \delta^{(t)}))^2\right\}\right]^{1\{D_i=0\}},$$

and for $g = 0$,

$$z_{i0}^{(t)} = \frac{\pi_0^{(t)} p_D^{1\{D_i=1\}}(1-p_D)^{1\{D_i=0\}}}{A(m+\beta-1)},$$

where A is an appropriate normalizing constant.

M-step:

For $g = 1, 2, \cdots, g^*$, we obtain

$$\mu_g^{(t+1)} = \frac{1}{\sum_{i=1}^n z_{ig}^{(t)}} \sum_{i=1}^n z_{ig}^{(t)} \left[(R_i + \delta^{(t)})1\{D_i = 1\} + (R_i - \delta^{(t)})1\{D_i = 0\}\right],$$

and

$$\pi_g^{(t+1)} = \frac{1}{n}\sum_{i=1}^n z_{ig}^{(t)}.$$

Similarly,

$$\pi_0^{(t+1)} = \frac{1}{n}\sum_{i=1}^n z_{i0}^{(t)}.$$

Moreover,

$$\delta^{(t+1)} = \frac{1}{n}\sum_{i=1}^n \sum_{g=1}^{g^*} z_{ig}^{(t)} \left[(\mu_g^{(t+1)} - R_i)1\{D_i = 1\} + (R_i - \mu_g^{(t+1)})1\{D_i = 0\}\right],$$

and

$$(\sigma^2)^{(t+1)} = \frac{1}{n}\sum_{i=1}^n \sum_{g=1}^{g^*} z_{ig}^{(t)} [\left\{R_i - (\mu_g^{(t+1)} - \delta^{(t+1)})\right\}^2 1\{D_i = 1\} + \left\{R_i - (\mu_g^{(t+1)} + \delta^{(t+1)})\right\}^2 1\{D_i = 0\}].$$

This algorithm has the following intuitive interpretation. In the E step, each read is allocated to each binding event or background, based on the distance between the binding events and the read shifted by δ to its $3'$ direction. Both peak shape (p_D, δ, and σ^2) and relative strength of each binding event (π_g) are considered in this allocation. In the M step, location of each binding event (μ_g) is essentially updated to the averaged position of reads corresponding to the binding event, after they are shifted by δ to their $3'$ direction. One peak shape is estimated for each candidate region through δ and σ^2. Optimal shift of read from its binding event, δ, is updated to the averaged distance between the location of each binding event and the reads corresponding to this binding event, averaged over binding events in the region. Dispersion of the reads around its binding event, σ^2, is updated to the variance of the positions of reads corresponding to the binding event around location of each binding event (μ_g), after they are shifted by δ to their $3'$ direction, averaged over binding events in the region.

3.2.4 Model selection

In practice, determining the optimal number of binding events, g^*, in each candidate region can be cast as a model selection problem. Model selection based on the Bayesian Information Criterion (BIC) (Schwarz, 1978) is a popular choice in mixture modeling and has shown superior performance in diverse applications (Fraley and Raftery, 1998, 2000). Therefore, for pre-specified g^{max}, we fit models for each of $g^* = 1, 2, \cdots, g^{max}$ binding event components and choose the model with the BIC value corresponding to the first local minimum, as the final model.

Choice of g^{max} is an important issue in model selection. g^{max} should be large enough so that all binding events in each candidate region can be considered. On the other hand, setting g^{max} larger than necessary should also be avoided as well, in order to prevent choosing the model due to ill-conditioning rather than a genuine indication of a better model (Fraley and Raftery, 1998, 2000). For appropriate choice of g^{max} in the current application, we checked the number of known binding events in each candidate region of σ^{70} data from the RegulonDB database(Gama-Castro et al., 2011) (`http://regulondb.ccg.unam.mx`) and found that 92% of the candidate regions contain either one or two binding sites. Based on this exploratory analysis, we set $g^{max} = 5$ as a default value and use it for all the analysis described in this chapter. For other applications, appropriate choice of g^{max} might depend on the protein type and experimental conditions.

3.2.5 Comparison with competing algorithms

dPeak has two unique features compared to the other peak deconvolution algorithms (Table 3.3). First, it can accommodate both SET and PET data and it explicitly utilizes specific features of both types. Second, it incorporates a background component, which accommodates reads due to non-specific binding. Consideration of non-specific binding is especially critical for deeply sequenced ChIP-Seq data because degree of non-specific binding becomes more significant as the sequencing depth gets higher. An additional unique feature of dPeak is the treatment of unknown library size for SET data. As discussed earlier, each read is either extended to or shifted by an estimate of the library size in most peak calling algorithms (Wilbanks and Facciotti, 2010) to account for unknown library size. This estimate is often specified by the users (Kuan et al., 2011; Rozowsky et al., 2009) or estimated from ChIP-Seq data (Zhang et al., 2008, 2011). Currently available algorithms with the exception of PICS use only one extension/shift estimate for all the regions in the genome. However, our exploratory analysis of real ChIP-Seq data and the empirical distribution of the library size from PET data (Figure 3.13a) indicate that using single extension/shift length might be suboptimal for peak calling (data not shown). Therefore, dPeak estimates optimal extension/shift length for each candidate region. Comparison of empirical distribution of the library size from PET data with the estimates of the region specific extension/shift lengths indicates that dPeak estimation procedure handles the heterogeneity of the peak-specific library sizes well (Figure 3.13b,c,d). This advancement ensures that dPeak is well tuned for deconvolving SET peaks as well, which then enables an unbiased computational comparison between the SET and PET assays.

3.2.6 Implementation

We identified candidate regions using the MOSAiCS algorithm (Kuan et al., 2011) (two-sample analysis with false discovery rate 0.001). In each candidate region, we fitted the dPeak model with

	dPeak	PICS	GPS
Support PET data	Yes	No	No
Support SET data	Yes	Yes	Yes
Consider non-specific binding	Yes	No	No
Consider local shift of reads	Yes	Yes	No
Construct candidate regions using	both ChIP and control samples	only ChIP sample	only ChIP sample
Peak shape estimation	Parametric	Parametric	Nonparametric
Normalization	Using non-specific binding	By sequencing depth	Regression approach
Merging	No	Yes	No
Filtering	Yes	Yes	Yes
Software interface	R	R	Java
Support parallel computing	Yes	Yes	Yes

Table 3.3 **Comparison of deconvolution algorithms.**

# of predicted events	0	1	> 1	Average # of events
dPeak	0%	86 %	14%	1.16 (0.42)
PICS	1%	97%	2%	1.02 (0.16)
GPS	82%	6%	12%	2.72 (1.69)

Table 3.4 **Prediction accuracy for** $20,000$ **candidate regions with single binding events.** Columns 2-4 report percentages of candidate regions with various numbers of predicted binding events. Column 5 reports the average number of binding events across regions with at least one predicted binding event.

up to five binding event components ($g^{max} = 5$) and the background component was always retained. The optimal number of binding events was chosen with BIC for each candidate region.

3.3 Simulation Studies

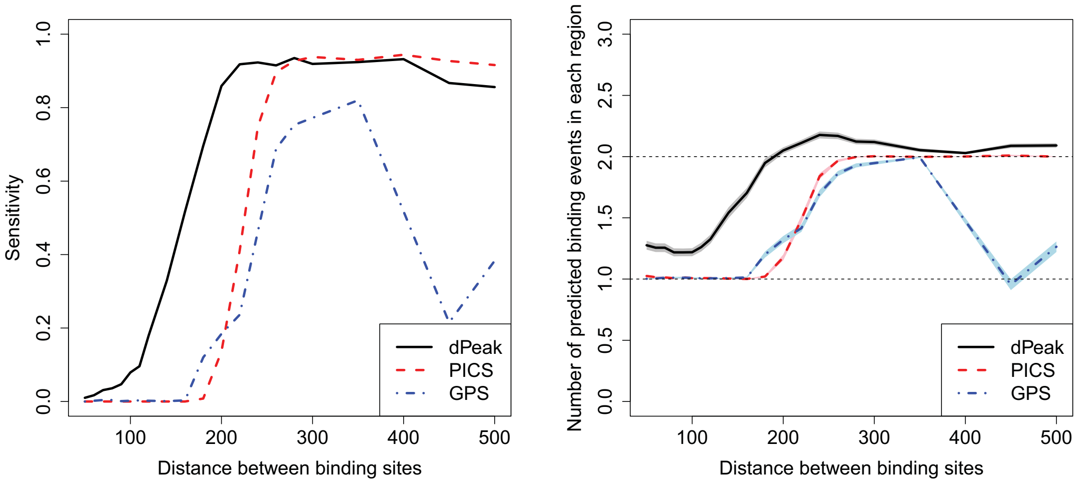
3.3.1 Performance comparison for SET ChIP-Seq data

We compared the dPeak algorithm with the two competing algorithms, GPS (Guo et al., 2010) and PICS (Zhang et al., 2011), for the SET ChIP-Seq data. We did not include the CSDeconv algorithm (Lun et al., 2009) in this comparison because it is computationally several orders of magnitude slower than the algorithms considered here. We compared the sensitivity and the number of predictions among these algorithms, where sensitivity is the proportion of true binding events identified by each algorithm. A binding event is considered 'identified' if the distance between the binding event and the predicted position is less than $20bp$. Note that we chose a more stringent criteria for defining true positives because $100bp$ used by GPS is not high enough resolution for prokaryotic genomes. For the PICS algorithm, we used the R package `PICS` version 1.10, which is available from Bioconductor. For the GPS algorithm, we used its Java implementation version 1.1 from `http://cgs.csail.mit.edu/gps/`.

We utilized the synthetic ChIP-Seq data previously used to evaluate deconvolution algorithms (Guo et al., 2010). Synthetic data used for the method comparison is available at http://cgs.csail.mit.edu/gps/ and its description is provided in Supplementary information of the GPS paper (Guo et al., 2010). In this synthetic data, binding events were generated by placing binding events from actual CTCF data at predefined intervals (Guo et al., 2010). Data consisted of 1,000 joint, i.e., close proximity, binding events, each with two events, and 20,000 single binding events. We assessed performances on these two sets separately so that we could assess sensitivity and specificity for each of these cases. The candidate regions for the dPeak algorithm were identified using the conditional binomial test (Ji et al., 2008) with the false discovery rate 0.05 by applying a Benjamini-Hochberg correction (Benjamini and Hochberg, 1995). These regions from the dPeak algorithm were also explicitly provided to the GPS algorithm as candidate regions. Candidate regions for PICS were identified using the function segmentReads() in the PICS R package (default parameters). For all methods, default tuning parameters were used for model fitting.

Fig. 3.4a shows the sensitivity of each algorithm at different distances between the joint binding events. dPeak outperforms other methods across all considered distances between the joint binding events and especially for closely located binding events with distance less than $250bp$. When the distance between the joint binding events is about $200bp$, dPeak is able to identify 80% of the events whereas neither PICS nor GPS can detect more than 20%. Further investigation indicates that the PICS algorithm merges closely spaced binding events into one event too often. We also found that, in the GPS algorithm, peak shape is incorrectly estimated when ChIP-Seq data harbors many closely located binding events. Specifically, when two binding sites are in close proximity, GPS models them as one event during peak shape estimation (Figure 3.5). Interestingly, we found that the sensitivity of GPS also decreases significantly when the distance between joint binding events increases. A closer look at the results reveals that GPS filters out too many predictions for joint binding events.

To evaluate specificity, we plotted the number of binding events predicted by each algorithm at different distances between the joint binding events in Fig. 3.4b. Since there are two true binding events in each region, two predictions at every distance would correspond to perfect specificity. dPeak on average generates more than one prediction and does not over-estimate the number of



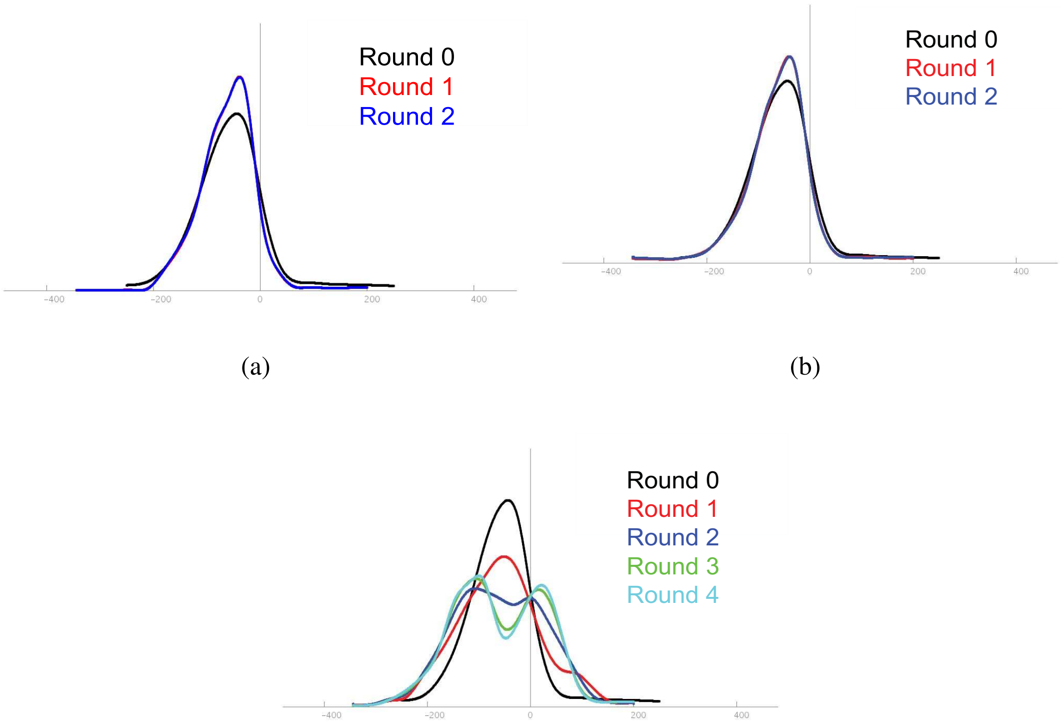
(a)

(b)

Figure 3.4 **Sensitivity and specificity comparisons of high resolution binding site identification algorithms.** (a) Comparison of dPeak with PICS and GPS in computational experiments designed for the GPS algorithm. dPeak has higher sensitivity than both PICS and GPS for SET ChIP-Seq data, especially when the distance between binding events is less than the library size. (b) When there are two true binding events in each region, dPeak on average generates more than one prediction and results in a notably higher specificity compared to PICS and GPS. PICS and GPS on average generate only one prediction when the distance between binding events is less than the library size. Shaded areas around each line indicate confidence intervals.

binding events when the distance between joint events is less than the average library size. In contrast, PICS and GPS generate on average only one prediction for closely located binding events, which recapitulates the conclusions from Fig. 3.4a. In summary, dPeak outperforms the state of the art deconvolution methods across different distances between joint binding events, especially when the distance between the binding events is less than the average library size.

Next, we evaluated the sensitivity and specificity of the three methods on $20,000$ candidate regions with a single binding event using the additional synthetic data from Guo et al. (2010) (Table 3.4). Average number of predictions per region with at least one predicted binding event and the corresponding standard errors are as follows: dPeak 1.16 (0.42), PICS 1.02 (0.16), GPS 2.72 (1.69). Overall, dPeak slightly over-estimates the number of binding events for regions with a single binding event, and hence PICS is slightly better than dPeak in specificity for these regions.



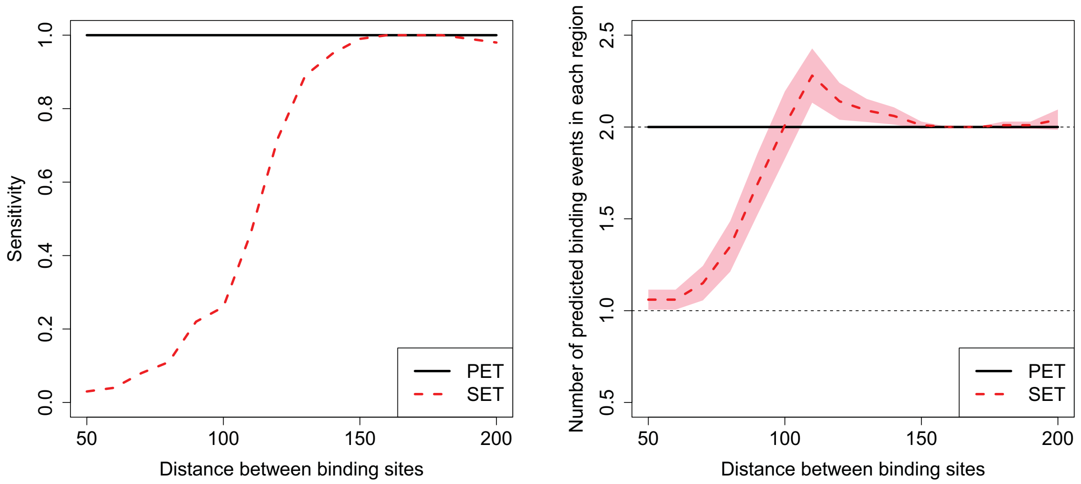
(a)

(b)

(c)

Figure 3.5 **Peak shapes estimated by the GPS algorithm for synthetic ChIP-Seq data** when there is a single binding event (a) and when the distance between joint binding sites set to 450 (b) and 140 (c).

However, as shown in our joint event analysis, this conservative approach of PICS severely under-estimates the number of binding events when multiple events reside closely. In contrast, GPS significantly under-estimates the number of binding events for the regions with a single binding event since it filters out too many predictions and does not result in a prediction for 82% of the regions. Comparisons in these two scenarios with and without joint binding events indicate that dPeak strikes a good balance between sensitivity and specificity for both cases.



(a) (b)

Figure 3.6 **Performance of the dPeak algorithm on PET vs. SET data.** (a) For SET ChIP-Seq data, the sensitivity of binding event detection significantly decreases as the distance between the locations of the events decreases. In contrast, sensitivity from PET ChIP-Seq data is robust to the distance between closely located binding events. (b) PET ChIP-Seq data on average predicts two binding events at any distance between the two joint binding events and results in excellent specificity. SET ChIP-Seq data predicts significantly fewer number of binding events as the distance between binding sites decreases. Shaded areas around each line indicate confidence intervals. dPeak is used for both (a) and (b).

3.3.2 Comparison of PET and SET data in resolution

Once we developed dPeak as a high resolution peak detection method for both SET and PET data, we implemented simulation studies to evaluate the PET and SET assays for resolving closely spaced binding events in an unbiased manner. We generated 100 simulated PET and SET ChIP-Seq data with two closely spaced binding events and evaluated binding event predictions of the two data types with the dPeak algorithm. Specifically, we considered distances between binding sites ranging from $50bp$ to $200bp$ which characterize the typical binding event spacing in *E. coli*. We generated and assigned 300 DNA fragments to each of two binding events as follows. For each DNA fragment, we drew the length (L_i) from the distribution of library size, $P(L)$, estimated empirically from the actual σ^{70} PET ChIP-Seq data and group index (Z_i) from multinomial distribution with parameters (0.5, 0.5). Then, for given library size and group index ($Z_i = g$), leftmost position of the paired reads (S_i) were generated from Uniform distribution between $\mu_g - L_i + 1$ and μ_g, where μ_g is the position of g-th binding event. Rightmost position of the paired reads was calculated as $E_i = S_i + L_i - 1$. These paired reads constructed PET data. SET data was generated by randomly sampling one of two ends from each of these paired reads. For the SET analysis, average library size was assumed to be $150bp$. In addition, we randomly assigned 10 DNA fragments to arbitrary positions within the candidate region to generate non-specific binding (background) reads. We generated 100 PET and SET data with this procedure. The sensitivity and the number of predictions (specificity) were summarized over these 100 simulations. A binding event was considered as 'identified' if the distance between the binding site and the predicted position was less than $20bp$.

Fig. 3.6a plots the sensitivity of dPeak at different distances between the joint binding events for both of the PET and SET settings. When the distance between the events is at least as large as the average library size ($\geq 150bp$), the sensitivity using PET and SET data are comparable. However, as the distance between joint binding events decreases, the sensitivity using SET data decreases significantly. In contrast, PET ChIP-Seq retains its high sensitivity even for binding events that are located as close as $50bp$. Fig. 3.6b displays the number of binding events predicted by dPeak at different distances between joint binding events and evaluates specificity. With PET ChIP-Seq, dPeak accurately chooses the number of binding events by BIC out of a maximum of five binding events at any distance between the joint binding events. In contrast, SET ChIP-Seq predicts less than two binding events when the distance between the events is less than $100bp$ and slightly over-estimates the number of binding events for distances between $100bp$ and $150bp$.

Figure 3.7 further reveals that even for cases with only single binding event, PET has a slight advantage over SET because it predicts the location of the binding event more accurately. Specifically, for any number of DNA fragments, PET data always provides higher resolutions, compared to SET data. When there are 300 DNA fragments, the averaged distance between predicted and

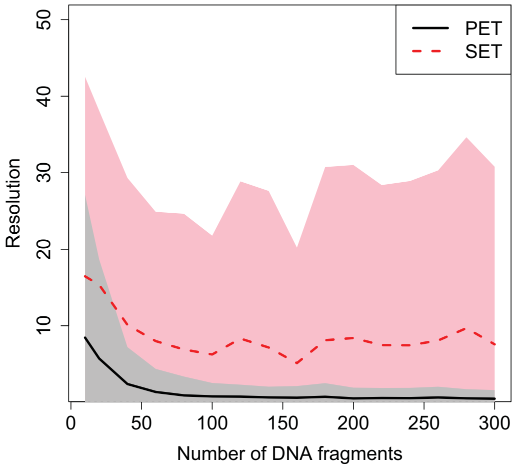
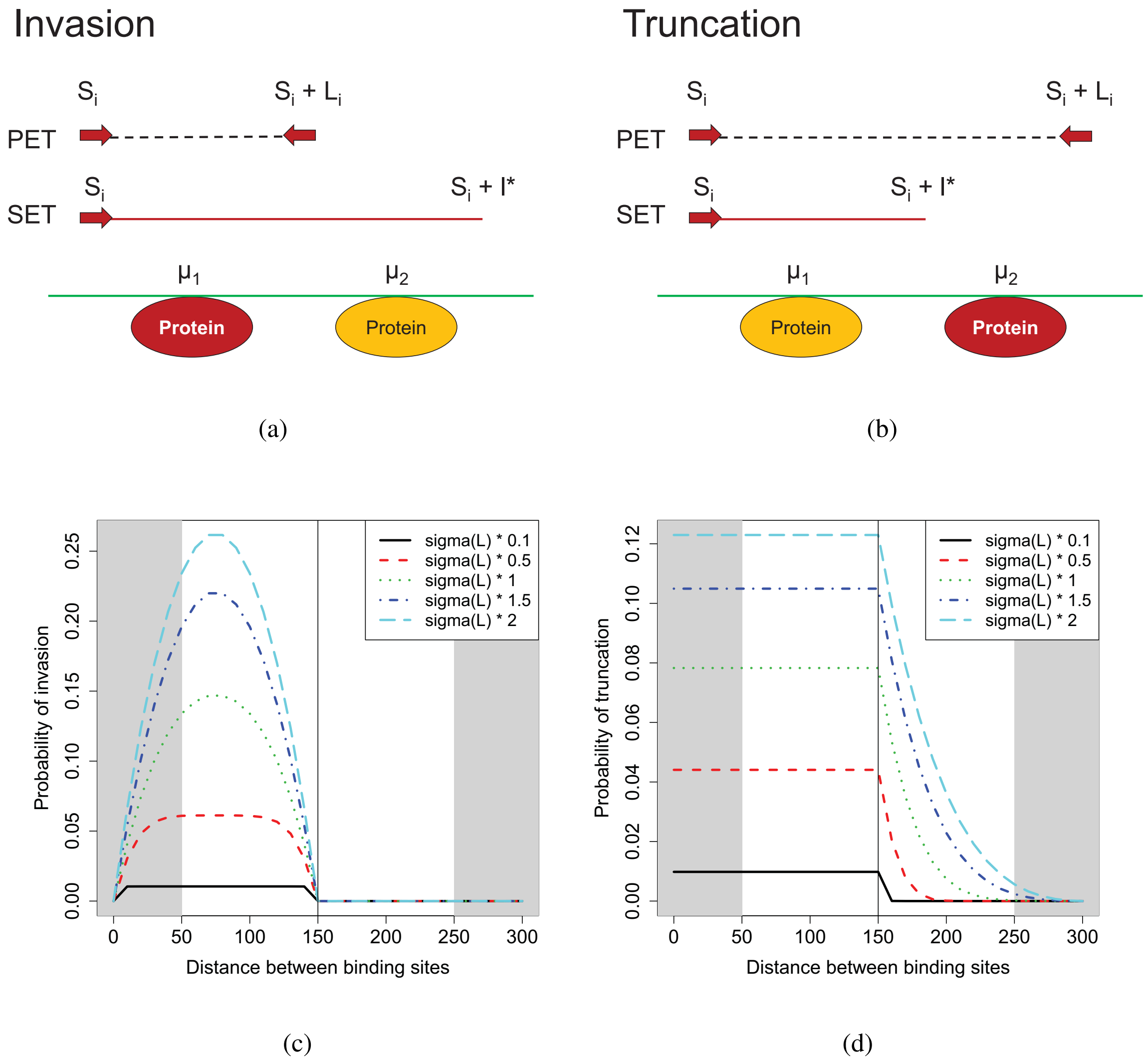


Figure 3.7 **Plot of resolution as a function of number of DNA fragments, in PET and SET simulation data, respectively, with a single binding event.** Black solid and red dotted curves indicate averaged resolutions for each number of DNA fragments in PET and SET data, respectively. Gray and pink shades indicate their confidence intervals in PET and SET data, respectively.

true binding events is $0.5bp$ (standard deviation = $0.6bp$) in the PET data while it is $7.6bp$ (standard deviation = $11.8bp$) in the SET data.

3.3.3 Analytical investigation of PET and SET data

Lower sensitivity of the SET compared to PET data is mainly driven by the loss of information due to unknown library size. We describe this information loss by two concepts named *invasion* (Figure 3.8a) and *truncation* (Figure 3.8b). Figure 3.8a depicts two closely spaced binding events and a DNA fragment that is informative for the first binding event (in red) in the PET data. *Invasion* refers to over-estimation of the library size and extending the read to a longer than the true length. Equivalently, in the shifting procedure, this corresponds to shifting the read more than necessary. As a result, the read extended to the estimated library size covers both of the two closely spaced binding events in the SET data and becomes uninformative or less informative for the binding event it corresponds to. Figure 3.8b also depicts two closely spaced binding events. In this case, even though the displayed DNA fragment corresponds to the second binding event (in red), it is long and it spans both of the binding events. Therefore, it is not informative for the second binding



(a)

(b)

(c)

(d)

Figure 3.8 **Illustration of loss of information in SET assay compared to PET assay.** Concepts of (a) *invasion* and (b) *truncation*. In each of (a) and (b), the first and second lines indicate PET and SET ChIP-Seq data, respectively. Red horizontal line depicts estimated library size (l^*) in the SET data. Red circles denote the protein binding event that the read corresponds to. In the case of invasion, this read becomes uninformative regarding the protein binding event whereas with truncation, the read provides incorrect information about the protein binding event. (c) Probability of invasion as a function of distance between binding sites based on the dPeak generative model. (d) Probability of truncation as a function of distance between binding sites based on the dPeak generative model. In (c) and (d), *sigma(L)* refers to the variance of library size distribution in σ^{70} PET ChIP-Seq data and non-shaded areas depict typical range of library sizes.

event in the PET data. *Truncation* refers to underestimation of the library size. As a result, the read extended to estimated library size only covers the first binding event in the SET data and provides incorrect information about the binding event it corresponds to.

We used the dPeak generative model and calculated the probability of invasion and truncation as follows. As in Section 3.2, let S and L be start position of DNA fragment and its length, respectively, in PET ChIP-Seq data. Let l^* be the extension we use for SET ChIP-Seq data and it is assumed to be fixed. Z indicates group index of the DNA fragment and $Z = 1$ and $Z = 2$ mean that this DNA fragment belongs to the first and second binding events, respectively. Let μ_1 and μ_2 be positions of first and second binding events, respectively, and assume that $\mu_1 < \mu_2$. Probability of invasion (Figure 3.8a) is obtained as:

$$\begin{aligned} P(Invasion) &= E_L[P(S < \mu_1 < S + L < \mu_2 < S + l^* | Z = 1)] \\ &= \sum_{L=l} P(L = l) P(S < \mu_1 < S + l < \mu_2 < S + l^* | Z = 1, L = l) \\ &= \sum_{L=l} P(L = l) min \left\{ l, \mu_2 - \mu_1, l^* - l, l^* - (\mu_2 - \mu_1) \right\} / l \end{aligned}$$

Probability of truncation (Figure 3.8b) is obtained as:

$$\begin{aligned} P(Truncation) &= E_L[P(S + l^* < \mu_2, S < \mu_1 < \mu_2 < S + L | Z = 2)] \\ &= \sum_{L=l} P(L = l) P(S + l^* < \mu_2, S < \mu_1 < \mu_2 < S + l | Z = 2, L = l) \\ &= \sum_{L=l} P(L = l) min \left\{ l - l^*, l - (\mu_2 - \mu_1) \right\} / l \end{aligned}$$

We evaluated the frequency by which fragments with invasion and truncation arise in SET data with a simulation study. Consider a region with two closely located binding events. Processing of DNA fragments generated from this region will lead to classification of the fragments in one of the following four categories:

Only fragments in category I are truly informative. Fragments in category II is less informative than fragments in category I. They could potentially contribute to both binding events, possibly through proportional allocation based on relative distances from each binding event. However, ambiguity in prediction increases as the number of fragments in category II increases. Fragments in category III introduce noise to binding event estimation since they are associated with the wrong binding event.

Category I Fragments overlapping a single true binding event.

Category II Fragments overlapping both binding events.

Category III Fragments overlapping only the false binding event.

Category IV Fragments not overlapping any binding events.

Fragments in category IV is uninformative. In summary, invasion refers to increased number of category II fragments in SET data compared to PET data and truncation refers to increased number of category III and IV fragments in SET data compared to PET data.

Table 3.5 displays the number of fragments in each category from one of simulation data where we set the distance between the two binding events to $50bp$. Average library size is $139bp$ in the PET data. The estimated library size used with SET analysis are reported in parentheses in the first column. In the corresponding SET data, even when library size is estimated relatively accurately (estimated library size = $150bp$), number of fragments in categories II to IV increases significantly compared to PET data. When the library size is under-estimated as $100bp$, we have significantly more fragments in categories III and IV (truncation; Figure 3.8b). In contrast, when it is over-estimated as $200bp$, we have significantly more fragments in category II (invasion; Figure 3.8a). In other words, our results indicate that as high as 76.8% and 25.5% of the fragments for a typical peak region can be subject to invasion and truncation with the SET assay.

Figures 3.8c,d display the probability of invasion and truncation of a fragment, respectively, as a function of the distance between closely spaced binding events and the variance of the library size. The analytical calculations are based on the dPeak generative model. Probabilities of invasion and truncation are higher for closely spaced binding events, especially when the library size is shorter than the estimated library size ($150bp$ in this case). In Figure 3.8c, the probability of invasion decreases for very closely spaced binding events, i.e., when the distance between two binding events is less than $75bp$. When two binding events are very closely spaced, most DNA fragments cover both binding events and the configuration in the first diagram of Figure 3.8a is hard to occur. In this case, there is already insufficient information to predict two binding events even in PET data and hence, relative loss of information (i.e., invasion) in SET data is not remarkable. These

Category	I	II	III	IV
	Only overlapping true binding events	Overlapping both binding events	Overlapping only false binding event	Not overlapping any binding event
PET	225	375	0	0
SET (150)	174	391	19	16
SET (100)	232	215	89	64
SET (200)	133	461	3	3

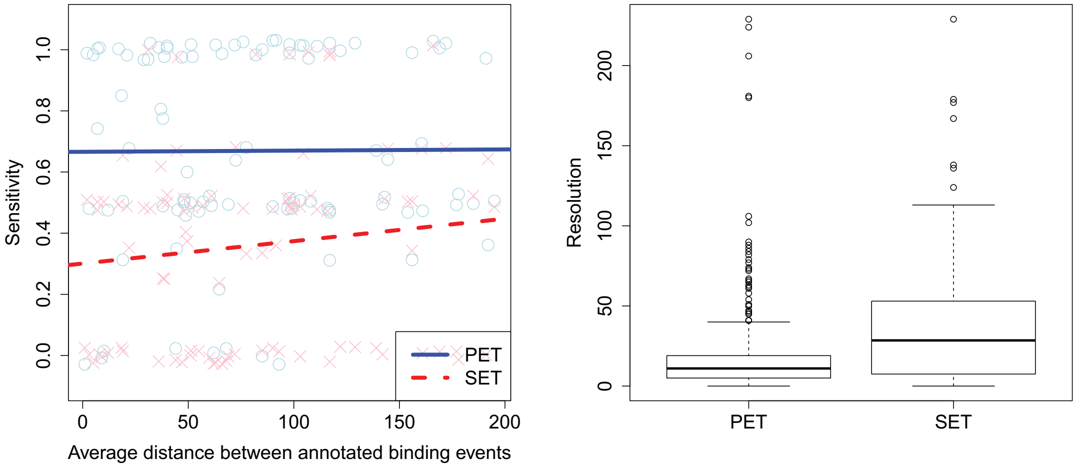
Table 3.5 **Classification of 600 DNA fragments from one simulation data with two binding events separated by 50bp.** Average library size is $139bp$ in the PET data. The estimated library size used with SET analysis are reported in parentheses in the first column.

concepts describe how information on binding events can be lost or distorted by the incorrect estimation of library size in the SET data. In summary, analytical calculations based on the dPeak generative model show that invasion and truncation influence closely located binding events the most especially when the library size is not tightly controlled, i.e., exhibit large variation (Figs 3.8c,d).

3.4 Case study: *E. coli* σ^{70} ChIP-Seq Data

3.4.1 The dPeak analysis of σ^{70} PET ChIP-Seq data

We compared the performance of PET and SET sequencing for the aerobic condition by generating a ‘quasi-SET data’ by randomly sampling one of the two ends of each paired reads in PET data and comparing binding events identified from both sets. Comparison with the quasi-SET data controlled for the differences in the sequencing depths of the original PET and SET samples, in addition to the biological variation of the replicates. We then evaluated the dPeak predictions from the PET and SET analyses using the σ^{70} factor binding site annotations in the RegulonDB database as a gold standard. Because a significant number of promoter regions lack any regulonDB annotations, we evaluated the sensitivity based on the regions that contain at least one annotated binding site. This corresponds to 539 binding sites in 363 regions. Of these 363 regions, 240 harbor only



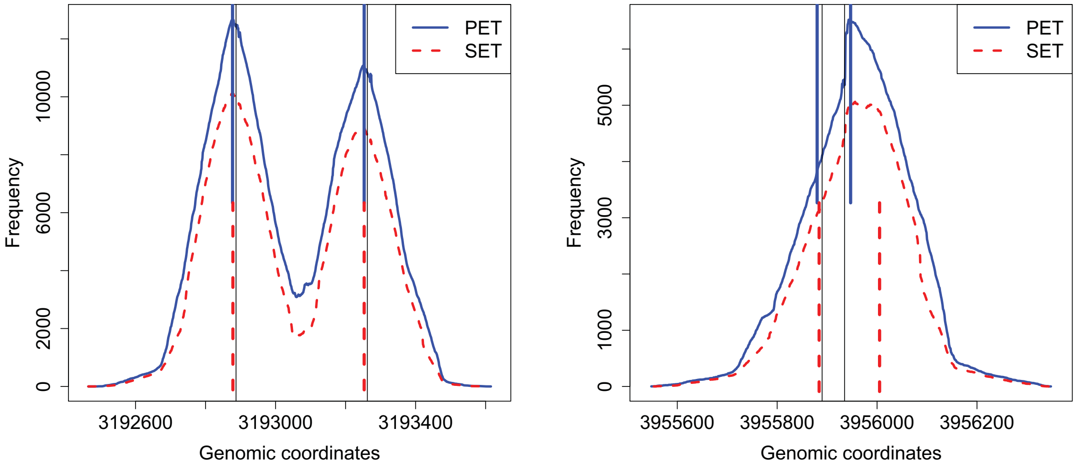
(a)

(b)

Figure 3.9 **Analysis of σ^{70} PET and SET ChIP-Seq data.** The dPeak algorithm is used to generate (a) and (b). Annotated σ^{70} factor binding sites from the RegulonDB database are used as a gold standard to evaluate predictions. (a) Proportion of correctly identified binding sites are plotted as a function of the distances between the RegulonDB reported binding events. Linear lines (solid for PET, dashed for SET) through the data points depict general trends. SET data can only identify 38% of the the annotated RegulonDB binding sites. In contrast, PET data identifies 66%. (b) Box plot of resolutions of PET and SET data, respectively, for the regions which harbor a single annotated binding event.

a single annotated binding event. For the regions with more than one annotated binding events, average distance between binding events is $126bp$. dPeak analysis of the SET data identifies only 38% of the the annotated binding events. In contrast, analysis of the PET data with dPeak detects 66% of the annotated binding sites. Fig. 3.9a displays average sensitivity as a function of the average distance between annotated binding events for the regions with at least two regulonDB annotations. A linear line is superimposed to capture the trend for both data types. Notably, the lower sensitivity of SET compared to PET is mainly due to closely located binding events.

We also compared prediction accuracies of the PET and SET assays by comparing them on the 240 regions which harbor a single annotated binding event. Fig. 3.9b shows resolutions of each of the PET and SET data, where resolution is defined as the minimum of distances between predicted and annotated positions of binding events. Medians of resolutions of the PET and SET



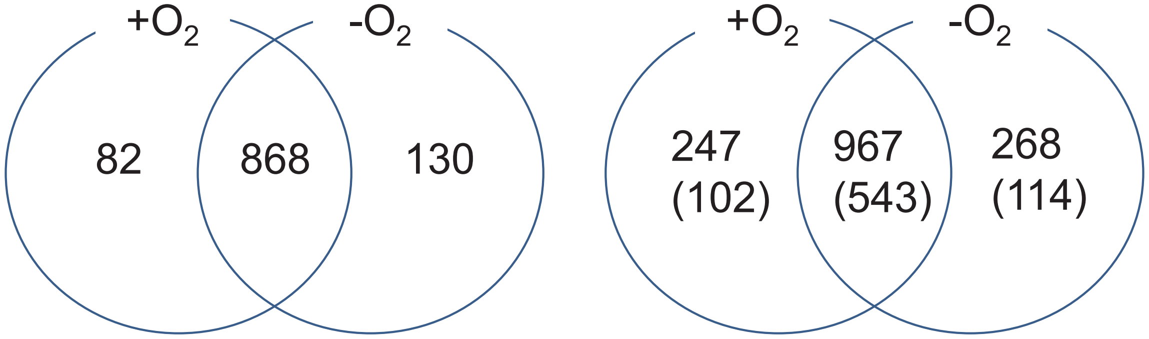
(a)

(b)

Figure 3.10 **Coverage plots for representative examples of identified σ^{70} binding sites from PET (blue solid lines) and SET (red dotted lines) data.** The dPeak algorithm is used to generate (a) and (b). Blue and red dotted vertical lines indicate predictions using the PET and SET data, respectively. Black solid vertical lines indicate the annotated binding sites.

data are $11bp$ (IQR = $14bp$) and $28.5bp$ (IQR = $45.25bp$), respectively, for these regions. This result indicates that positions of binding events can be more accurately predicted when PET data is used, compared to the case that SET data is used, even for the regions with a single binding event.

Figs. 3.10a and 3.10b display two representative examples from this analysis. Fig. 3.10a illustrates two binding events in the promoter regions of sibD and sibE genes separated by $375bp$. In this case, two peaks are easily distinguishable just by visual inspection and the predictions using both PET and SET data are comparably accurate. Note that although these two binding events are visually distinguishable, standard applications of MACS and MOSAiCS identify this region as a single peak. Widths of MOSAiCS and MACS peaks are $900bp$ and $2,042bp$, respectively, and in MACS, the position of right binding event is identified as the "summit" of this peak region (summit = $3,193,216$). Fig. 3.10b displays the promoter regions of ilvC and ilvY genes, where the distance between the two annotated binding events is only $45bp$. In this case, both PET and SET correctly predict the number of binding events as two; however, PET prediction for the right



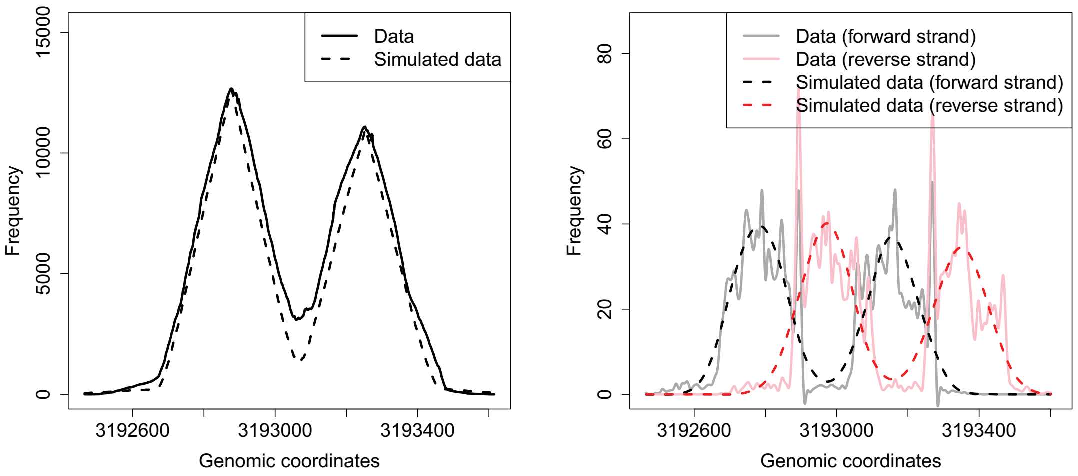
(a) (b)

Figure 3.11 **Summary of the analysis of $+O_2$ and $-O_2$ PET ChIP-Seq data in (a) candidate region level and (b) binding event level.** Each of 82, 868, and 130 candidate region level peaks (the first diagram) corresponds to 1%, 11%, and 1% of the *E. coli* genome, respectively.

hand side binding event is significantly more accurate than the SET prediction (distances to the annotated binding site are $69bp$ and $11bp$ for SET and PET, respectively).

3.4.2 Differential binding analysis using the dPeak algorithm

High resolution identification of binding sites is especially important for differential occupancy analysis where factor of interest is profiled under different conditions. In our study, we set out to identify differential promoter usage in *E. coli* between aerobic and anearobic conditions by profiling the σ^{70} factor. Results for dPeak analysis of the aerobic and anaerobic PET data are summarized in Figure 3.11, both at the region (i.e., peak) and binding event levels. We utilized top 50% of the predicted binding events from each condition. We used the analogues SET ChIP-Seq from biological replicates under both conditions as further validation of the results. We identified 868 peaks corresponding to 967 binding events that were common between the $+O_2$ and $-O_2$ conditions. 56.1% of these binding events were also independently identified by SET ChIP-Seq analysis of the two conditions. 82 peaks corresponding to 247 binding events were unique to $+O_2$ condition. SET ChIP-Seq analysis recovered 41.3% of these binding events as unique to $+O_2$ condition. A total of 130 peaks corresponding to 268 binding events were unique to $-O_2$ and 42.5% of these were identified by the SET analysis as well.



(a)

(b)

Figure 3.12 **Goodness of fit (GOF) plots** of Figure 3.10a for (a) the PET ChIP-Seq data and (b) the quasi-SET ChIP-Seq data, respectively.

3.4.3 Evaluation of the dPeak model

We further consider σ^{70} PET and SET ChIP-Seq data to evalate the dPeak model. Specifically, we investigated goodness of fit of the dPeak model and performance of the local estimation of library size in the dPeak algorithm.

Figure 3.12a and Figure 3.12b show the goodness of fit (GOF) plots of Figure 3.10a for the PET and quasi-SET ChIP-Seq data, respectively. GOF plots compare the empirical distribution of the read positions with that obtained by simulating from estimated model parameters. The GOF plots indicate that the dPeak models fit the data well. These GOF plots are representative of the GOF plots for other candidate regions.

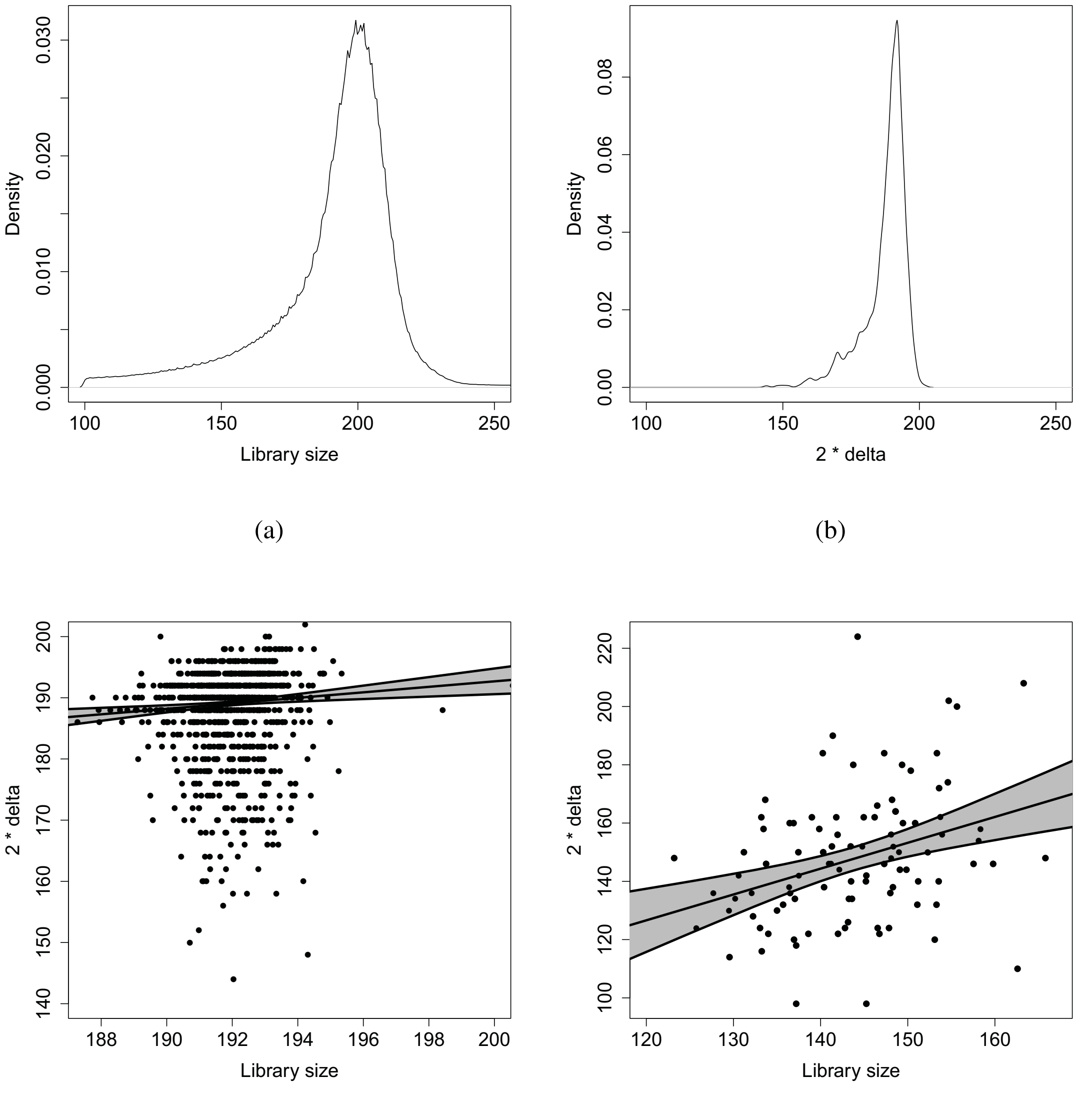
Figure 3.13 shows results for the performance of the local estimation of library size in the dPeak algorithm. Figure 3.13a shows the density of library size in the σ^{70} PET ChIP-Seq data. The corresponding mean and standard deviations are $192.01bp$ and $26.90bp$, respectively. Figure 3.13b shows the density of $2\hat{\delta}$ in the σ^{70} quasi-SET ChIP-Seq data. Mean and standard deviation of $2\hat{\delta}$ are $187.36bp$ and $9.04bp$, respectively. Figure 3.13c shows the scatter plot of library size vs. $2\hat{\delta}$

(Figure 3.13d shows the scatter plot for the SET simulation data with a single binding event). This plot indicates that, overall, we have larger $2\hat{\delta}$ values for the candidate regions with larger average library size.

3.5 Discussion

High resolution identification of binding sites with ChIP-Seq has profound effects for studying protein-DNA interactions in prokaryotic genomes and differential occupancy. We evaluated PET and SET ChIP-Seq assays and illustrated that PET has considerably more power for deciphering locations of closely spaced binding events. For regions with only a single binding event or binding events separated by a distance larger than that of the library size, SET and PET have comparable accuracy. However, when the distance between binding events in close proximity of each other gets smaller than the average library size, SET analysis have notably less power than the PET analysis. We developed and evaluated the dPeak algorithm, a model-based approach to identify protein binding sites in high resolution, with data-driven computational experiments. dPeak is currently the only algorithm that can utilize both PET and SET ChIP-Seq data and can accommodate high levels of non-specific binding apparent in deeply sequenced ChIP samples (Table 3.3). Application of dPeak to σ^{70} ChIP-Seq data from *E. coli* under aerobic and anaerobic conditions revealed that although many peaks (i.e., enriched regions) identified by standard application of popular peak finders might appear as common between the two conditions, a considerable proportion corresponds to condition-specific binding events.

The advantages of using the dPeak algorithm are not limited to the study of prokaryotic genomes. Applications in eukaryotic genomes include identification of the exact locations of binding motifs when multiple closely located consensus sequences reside in a peak region, studies of *cis* regulatory modules (CRM), and refining consensus sequences. Figure 3.14 displays an example application of dPeak for differentiating among closely located GATA1 binding sites with the [AT]GATA[AG] consensus within a peak region critical for erythroid differentiation in mouse embryonic stem cells (data from Wu et al. (2011)). CRM studies investigate relationships between spatial configurations



(a)

(b)

(c)

(d)

Figure 3.13 **Estimation of library size in SET ChIP-Seq data.** (a) Density of library size in the σ^{70} PET ChIP-Seq data. (b) Density of $2\hat{\delta}$ in the σ^{70} quasi-SET ChIP-Seq data. (c) Scatter plot of library size versus $2\hat{\delta}$ for the σ^{70} quasi-SET ChIP-Seq data. (d) Scatter plot of library size vs. $2\hat{\delta}$, for SET simulation data with a single binding event. In (c) and (d), the solid line and shades indicate the fitted robust linear model (RLM) fit and the corresponding confidence intervals, respectively.

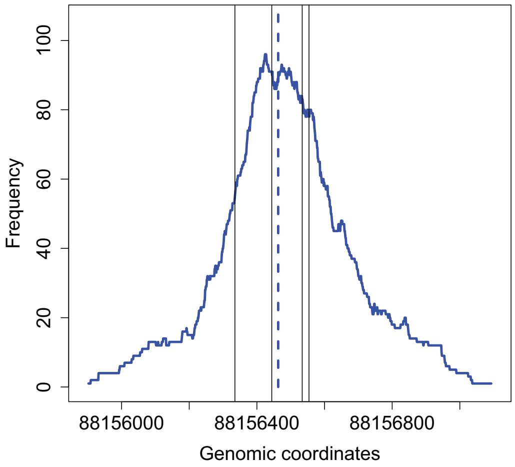


Figure 3.14 **A representative example from GATA1 data.** Blue curve and blue dotted vertical line indicate the GATA1 SET ChIP-Seq data and the prediction using the dPeak algorithm, respectively. Black solid vertical lines indicate positions of the GATA1 binding motif, [AT]GATA[AG].

of binding sites of multiple transcription factors and the expression of genes. Relative orders, positions, and distances of binding sites of multiple factors and their relative strengths are key factors in CRM studies (Noto and Craven, 2007). Because dPeak facilitates identification of the binding sites of transcription factors in high resolution from ChIP-Seq data, it can enable construction of complex interaction networks among diverse factors in multiple growth conditions.

Chapter 4

Software for ChIP-Seq Data Analysis

4.1 Introduction

In ChIP-Seq literature, even though many ChIP-Seq data analysis methods have been proposed, some important issues still have not fully investigated yet. These issues include identification of binding events in repetitive regions, consideration of important sequence biases in ChIP-Seq data, such as mappability and GC content, and identification of closely spaced binding events. In order to address these problems, we developed novel statistical methods, CSEM (Chapter 2) (Chung et al., 2011), MOSAiCS (Kuan et al., 2011), and dPeak (Chapter 3), respectively. In order to further facilitate application of these algorithms, we developed computationally efficient and user friendly software implementing the proposed methods. Figure 4.1 shows the analysis workflow using `csem`, `mosaics`, and `dpeak` software. Table 4.1 summarizes main features of these software. For notational consistency, throughout this chapter, we will denote the algorithms as CSEM, MOSAiCS, and dPeak while we denote the corresponding software as `csem`, `mosaics`, and `dpeak`.

One of main complications in the genomic data analysis is integration of multiple tools and efficient data sharing among these tools. For example, in the case of ChIP-Seq data, one usually needs to preprocess original read files, align reads to a reference genome, identify binding sites using a peak caller, and do the downstream analysis. This requires users run programs in diverse software environments such as unix system, R, perl, and Java, some of which users might not be familiar with. Furthermore, in each step, users also need to do laborious jobs to prepare data in the format that each software can handle. Galaxy (Goecks et al., 2010; Blankenberg

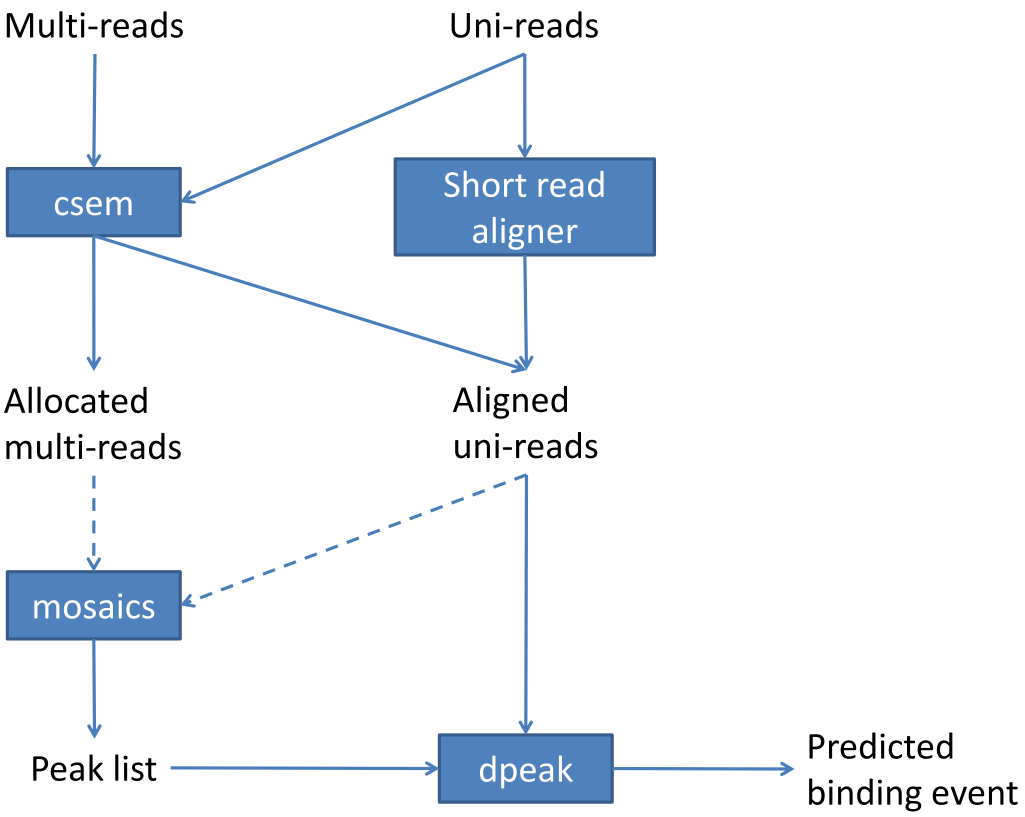


Figure 4.1 **Analysis workflow using the `csem`, `mosaics`, and `dpeak` software.** Dashed arrows indicate that `mosaics` software can accept either only aligned uni-reads or both aligned uni-reads and allocated multi-reads.

	csem	mosaics	dpeak
Objective	Multi-read allocation	Peak calling	Peak deconvolution
Software interface	C++	R	R
Galaxy tool	Yes	Yes	(under development)
Input	Genomic sequences	aligned reads	aligned reads
Input file format	FASTA, FASTQ	aligned read files *	aligned read files *
Output	allocated multi-reads	peak list	binding event prediction
Output file format	BED, GFF	BED, GFF	BED, GFF
Diagnostics tools	No	Yes	Yes
Support parallel computing	(under development)	Yes	Yes

Table 4.1 **Summary of ChIP-Seq data analysis software.** * Both mosaics and dpeak software support Eland result, Eland extended, Eland export, default Bowtie, SAM, and BED file formats for single-end tag (SET) data containing only uni-reads. mosaics software also supports CSEM BED file format for SET data incorporating multi-reads. Both mosaics and dpeak software support Eland result and SAM file formats for paired-end tag (PET) data containing only uni-reads.

et al., 2010; Giardine et al., 2005) (`http://galaxyproject.org/`), an open web-based platform for genomic research, addresses these issues and provides user-friendly interface for integration of multiple bioinformatics tools. Researchers also easily share data and analysis workflow with others using Galaxy system. Galaxy provides free public service at its main Galaxy server (`https://main.g2.bx.psu.edu/`) and anyone can also install their own instance of Galaxy in their local system. Galaxy Tool Shed (`http://toolshed.g2.bx.psu.edu/`) is a public repository for Galaxy tools and it provides even further flexibility to Galaxy system. Bioinformatics researchers can upload their software and Galaxy interface for each of their software to Galaxy Tool Shed and anyone who is interested in these software can download and extend their local Galaxy instance by installing these software.

Motivated by such exciting opportunities provided by Galaxy, we developed Galaxy tools for our ChIP-Seq data analysis software, `csem`, `mosaics`, and `dpeak`, which correspond to different steps in ChIP-Seq data analysis workflow (Galaxy tool for `dpeak` is currently under development). When these three tools are added to the Galaxy system, users do not need to consider differences in the software environment (e.g., `csem` is C++ binary while `mosaics` and `dpeak` are R packages) and can use these software as if they are one integrative software. Furthermore, by being integrated into Galaxy system, `csem`, `mosaics`, and `dpeak` can be painlessly connected with other Galaxy tools, such as quality control and manipulation tools, short read aligners, file format converters, and downstream analysis tools. In addition to Galaxy tools, we also provide and maintain each of our software in public repositories such as Bioconductor (`http://www.bioconductor.org/`) and this will provide further flexibility to users in their analysis. Because our software is developed for popular software environments such as R, our software will be able to be easily integrated with other tools, depending on users' need.

This chapter is organized as follows. In Section 4.2, we describe statistical frameworks for `csem`, `mosaics`, and `dpeak` and their implementations. In Section 4.3, using human STAT1, mouse GATA1, and *Escherichia coli (E. coli)* σ^{70} data, we illustrate application of our software. In Section 4.4, we discuss contributions of our software to genomic studies.

4.2 Methods

4.2.1 Utilizing multi-reads using csem

The state of the art for analyzing ChIP-Seq data relies only on using reads that map uniquely to a relevant reference genome (*uni-reads*). This can lead to the omission of up to 30% of alignable reads. In order to address this problem, we developed the CSEM algorithm to utilize reads that map to multiple locations on the reference genome (*multi-reads*) (Chung et al., 2011). Specifically, we cast the multi-read problem as a nonparametric estimation problem of a mixing density and derive an Expectation-Maximization-Smoothing (EMS) algorithm. Using human STAT1 and mouse GATA1 ChIP-Seq datasets, we illustrate that incorporation of multi-reads using the CSEM algorithm leads to detection of novel peaks that are not otherwise identifiable with only uni-reads and the majority of these novel peaks reside in segmental duplications. Our computational and experimental results establish that multi-reads can be of critical importance for studying transcription factor binding in highly repetitive regions of genomes with ChIP-Seq experiments.

csem is implemented as a C++ binary for computational efficiency. Stand-alone version of csem and its Galaxy tool are publicly available from the csem software website (http://www.stat.wisc.edu/∼keles/Software/multi-reads/) and Galaxy Tool Shed, respectively. The csem software utilizes the Bowtie aligner (Langmead et al., 2009) to align reads against an appropriate reference genome (e.g., mouse MM9). It takes single-end reads, in FASTA or FASTQ formats, as input (quality scores of reads are ignored) and generates aligned uni-reads and allocated multi-reads as output. This output can be used as input for the mosaics software (Figure 4.1).

4.2.2 Unbiased peak calling using mosaics

Despite its increasing and well-deserved popularity, there is little research that investigates and accounts for sources of biases in the ChIP-Seq technology. These biases typically arise from both the standard pre-processing protocol and the underlying DNA sequence of the generated data. We studied data from a naked DNA sequencing experiment, which sequences non-cross-linked DNA after deproteinizing and shearing, to understand factors affecting background distribution of data generated in a ChIP-Seq experiment. Based on our investigation, we developed the MOSAiCS algorithm, a flexible mixture model to detect peaks in both one-sample analysis (using only ChIP sample) and two-sample analysis (using both ChIP and matched control samples) of ChIP-Seq data (Kuan et al., 2011). In the MOSAiCS algorithm, we introduce a background model that accounts for apparent sources of biases such as mappability and GC content. Using human STAT1 and mouse GATA1 data, we illustrated that our model fits observed ChIP-Seq data well and further demonstrated advantages of MOSAiCS over commonly used tools for ChIP-Seq data analysis with several case studies (Kuan et al., 2011).

`mosaics` is implemented as an R package. `mosaics` R package and its Galaxy tool are publicly available from Bioconductor and Galaxy Tool Shed, respectively. `mosaics` R package allows the one- and two-sample analysis utilizing mappability and GC content and the two-sample analysis without utilizing mappability and GC content. `mosaics` R package also provides various diagnostics tools for in-depth investigation of ChIP-Seq data. `mosaics` Galaxy tool currently supports only two sample analysis without utilizing mappability and GC content and other analysis approaches will also be supported in the future.

For aligned uni-reads from single-end tag (SET) experiments, `mosaics` software accepts most popular aligned read file formats, such as Eland result, Eland extended, Eland export, default Bowtie, SAM, and BED, as input. When multi-reads are incorporated using `csem`, `mosaics` accepts them in CSEM BED file format, which is output from the `csem` Galaxy tool. If input file format is neither in BED nor CSEM BED formats, it filters out all the multi-reads in the file by default. For aligned uni-reads from paired-end tag (PET) experiments, `mosaics` supports Eland result and SAM file formats. For output, `mosaics` generates peak list, which can be used as input for the `dpeak` software (Figure 4.1). `mosaics` software supports parallel computing using either `multicore` or `parallel` packages.

4.2.3 High resolution binding site identification using `dpeak`

Because the compact prokaryotic genomes harbor binding sites some of which are separated by only few base pairs, applications of ChIP-Seq in this domain have not reached their full potential. ChIP-Seq applications in prokaryotic genomes are further hampered by the fact that well studied data analysis methods for ChIP-Seq do not result in a resolution required for deciphering the locations of nearby binding events. Through comparison between single-end tag (SET) and paired-end tag (PET) ChIP-Seq data for σ^{70} factor in *E. coli*, we found that although PET assay enables higher resolution identification of binding events, standard ChIP-Seq analysis methods are not equipped to utilize PET-specific features of the data. To address this problem, we developed dPeak, a high resolution binding site identification (deconvolution) algorithm. The dPeak algorithm implements a probabilistic model that accurately describes ChIP-Seq data generation process for each of SET and PET assays. It is currently the only algorithm that is applicable with both SET and PET data while utilizing features specific to these data types. For SET data, dPeak outperforms the state of the art high-resolution ChIP-Seq peak deconvolution algorithms such as PICS and GPS. When coupled with PET data, the dPeak algorithm can identify closely spaced binding sites with high accuracy. Applications of the dPeak algorithm to σ^{70} ChIP-Seq data in *E. coli* under aerobic and anaerobic conditions reveal closely located promoters that are differentially occupied between the two conditions.

`dpeak` is implemented as an R package and its Galaxy tool is currently under development. Upon completion, `dpeak` R package and its Galaxy tool will be contributed to bioconductor and Galaxy Tool Shed, respectively. As input for SET data, `dpeak` accepts most popular aligned read file formats, such as Eland result, Eland extended, Eland export, default Bowtie, SAM, and BED. If input file format is not in BED format, it filters out all the multi-reads in the file. As input for PET data, `dpeak` accepts aligned reads in Eland result and SAM file formats. For output, `dpeak` generates list of predicted binding events and their strengths (number of reads corresponding to each binding event). `dpeak` can also provide various diagnostics tools and it supports parallel computing using `multicore` package.

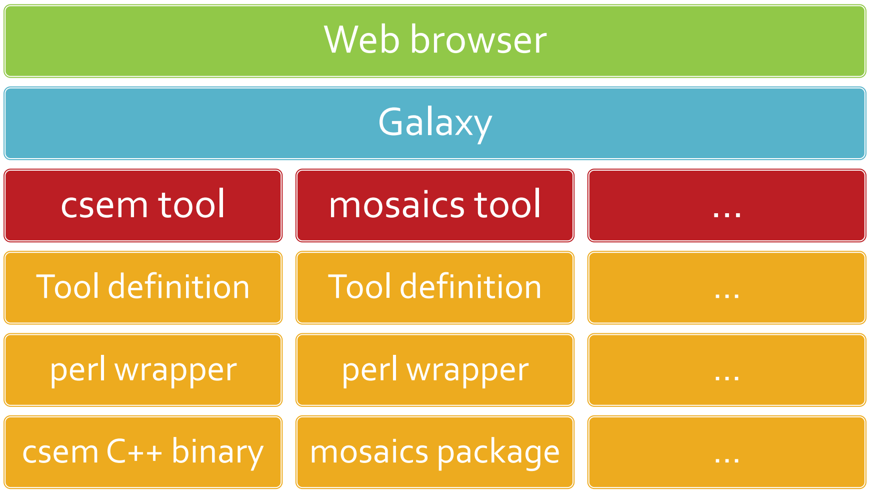


Figure 4.2 **Overview of the structure of the Galaxy system.**

4.2.4 Development of Galaxy tools

Figure 4.2 shows basic structure of the Galaxy system. Users communicate with Galaxy system using their web browsers such as Firefox and Chrome. Galaxy system links user interaction with underlying Galaxy tools. Because of such structure of Galaxy system, users do not need to know implementation details of each Galaxy tool and developers do not need to worry about developing complicated user interface for their software.

In order to add software to Galaxy, developers write a tool definition file for each of their software. Tool definition file defines user interface for Galaxy tool and specifies how Galaxy system communicates with underlying software. More details about Galaxy tool definition can be found in Galaxy Wiki (`http://wiki.g2.bx.psu.edu/Admin`). For many Galaxy tools, a perl or python wrapper is also written for easier communication between the tool definition and the underlying software. As examples, the tool definition and perl wrapper for the `mosaics` Galaxy tool are provided in Appendices C and D, respectively.

4.3 Results

4.3.1 Utilizing multi-reads using csem

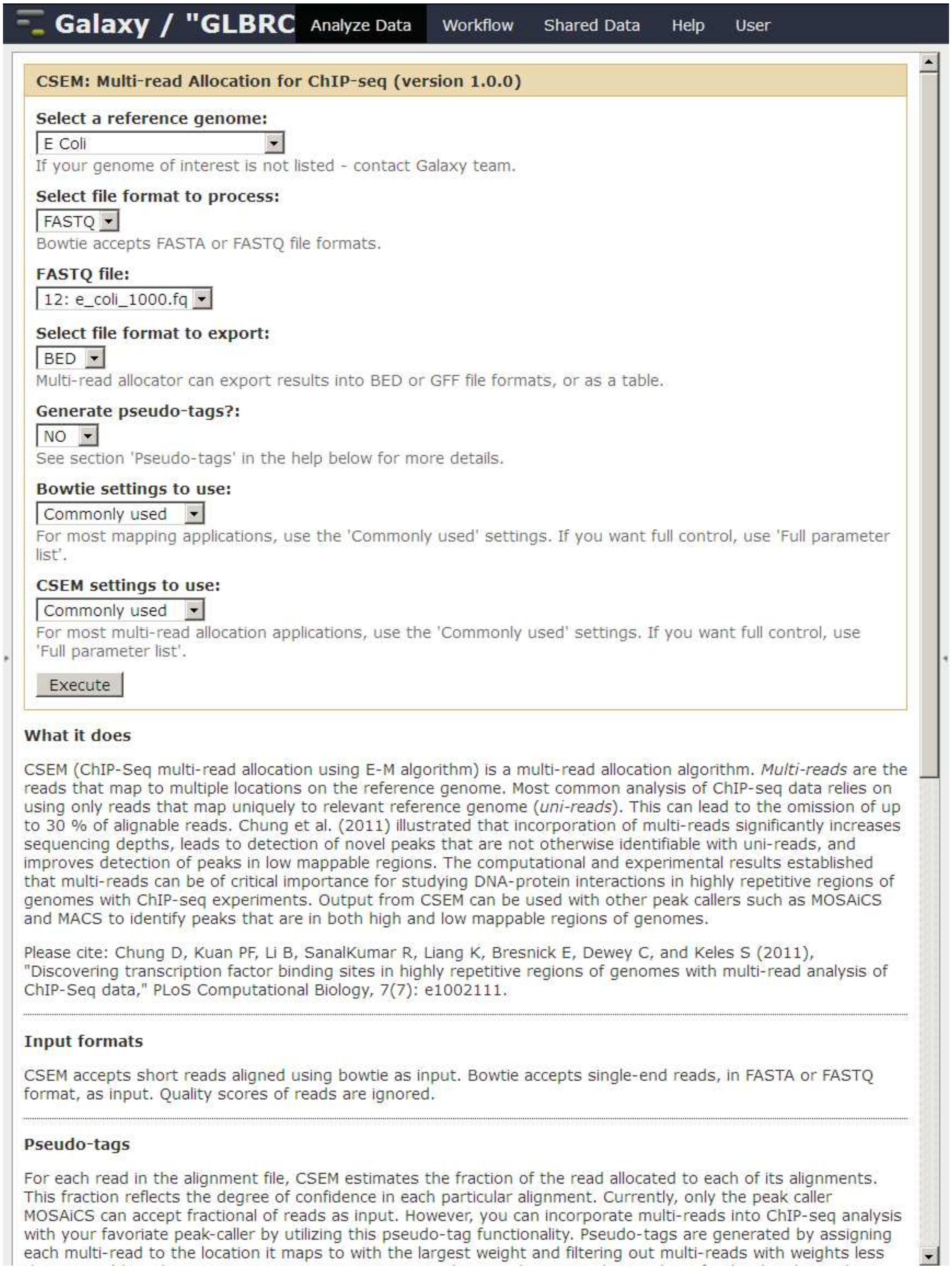
Figure 4.3 shows Galaxy interface for the csem software. By taking genomic sequences as input in either FASTA or FASTQ file formats, csem generates aligned uni-reads and allocated multi-reads as output in either BED or GFF formats. csem Galaxy tool provides well-tuned default parameters for both the csem software and the Bowtie aligner generating alignments to the csem software.

Based on the fitted model, csem allocates proportions of each multi-read to its possible positions. mosaics software can handle such proportionally allocated multi-reads input. However, many of existing software start the analysis with aligned or raw tag files and they do not allow multi-reads input. To accommodate some of the existing software that rely on aligned read files (or alignment results in the bed format), csem can also generate "pseudo-tags". Pseudo-tags are generated by rounding the multi-read weights to the nearest integer and adding the ones that round up to one to the original alignment files as pseudo reads so that they can be utilized. Although this implementation decreases the number of utilized multi-reads by about a half (for GATA1 data), it still leads to a significant increase in the sequencing depth compared to using uni-reads alone. An application of this strategy with the MACS algorithm (Zhang et al., 2008) was able to identify 37% of the MR-only peaks identified by the MOSAiCS MR analysis.

Table 4.2 shows multi-read allocation results for human STAT1 (Rozowsky et al., 2009) and mouse GATA1 (Cheng et al., 2009) using csem software. The results indicate that by utilizing csem software, we can now utilize significant number of multi-reads corresponding to about 20% of number of uni-reads.

4.3.2 Unbiased peak calling using mosaics

Figure 4.4 shows Galaxy interface for mosaics software. By taking aligned uni-reads or allocated multi-reads as input in popular file formats, mosaics generates peak list as output in either BED or GFF formats. Users can set parameters for false discovery rate (FDR), average fragment



CSEM: Multi-read Allocation for ChIP-seq (version 1.0.0)

Select a reference genome:
E Coli
If your genome of interest is not listed - contact Galaxy team.

Select file format to process:
FASTQ
Bowtie accepts FASTA or FASTQ file formats.

FASTQ file:
12: e_coli_1000.fq

Select file format to export:
BED
Multi-read allocator can export results into BED or GFF file formats, or as a table.

Generate pseudo-tags?:
NO
See section 'Pseudo-tags' in the help below for more details.

Bowtie settings to use:
Commonly used
For most mapping applications, use the 'Commonly used' settings. If you want full control, use 'Full parameter list'.

CSEM settings to use:
Commonly used
For most multi-read allocation applications, use the 'Commonly used' settings. If you want full control, use 'Full parameter list'.

What it does

CSEM (ChIP-Seq multi-read allocation using E-M algorithm) is a multi-read allocation algorithm. *Multi-reads* are the reads that map to multiple locations on the reference genome. Most common analysis of ChIP-seq data relies on using only reads that map uniquely to relevant reference genome (*uni-reads*). This can lead to the omission of up to 30 % of alignable reads. Chung et al. (2011) illustrated that incorporation of multi-reads significantly increases sequencing depths, leads to detection of novel peaks that are not otherwise identifiable with uni-reads, and improves detection of peaks in low mappable regions. The computational and experimental results established that multi-reads can be of critical importance for studying DNA-protein interactions in highly repetitive regions of genomes with ChIP-seq experiments. Output from CSEM can be used with other peak callers such as MOSAiCS and MACS to identify peaks that are in both high and low mappable regions of genomes.

Please cite: Chung D, Kuan PF, Li B, SanalKumar R, Liang K, Bresnick E, Dewey C, and Keles S (2011), "Discovering transcription factor binding sites in highly repetitive regions of genomes with multi-read analysis of ChIP-Seq data," PLoS Computational Biology, 7(7): e1002111.

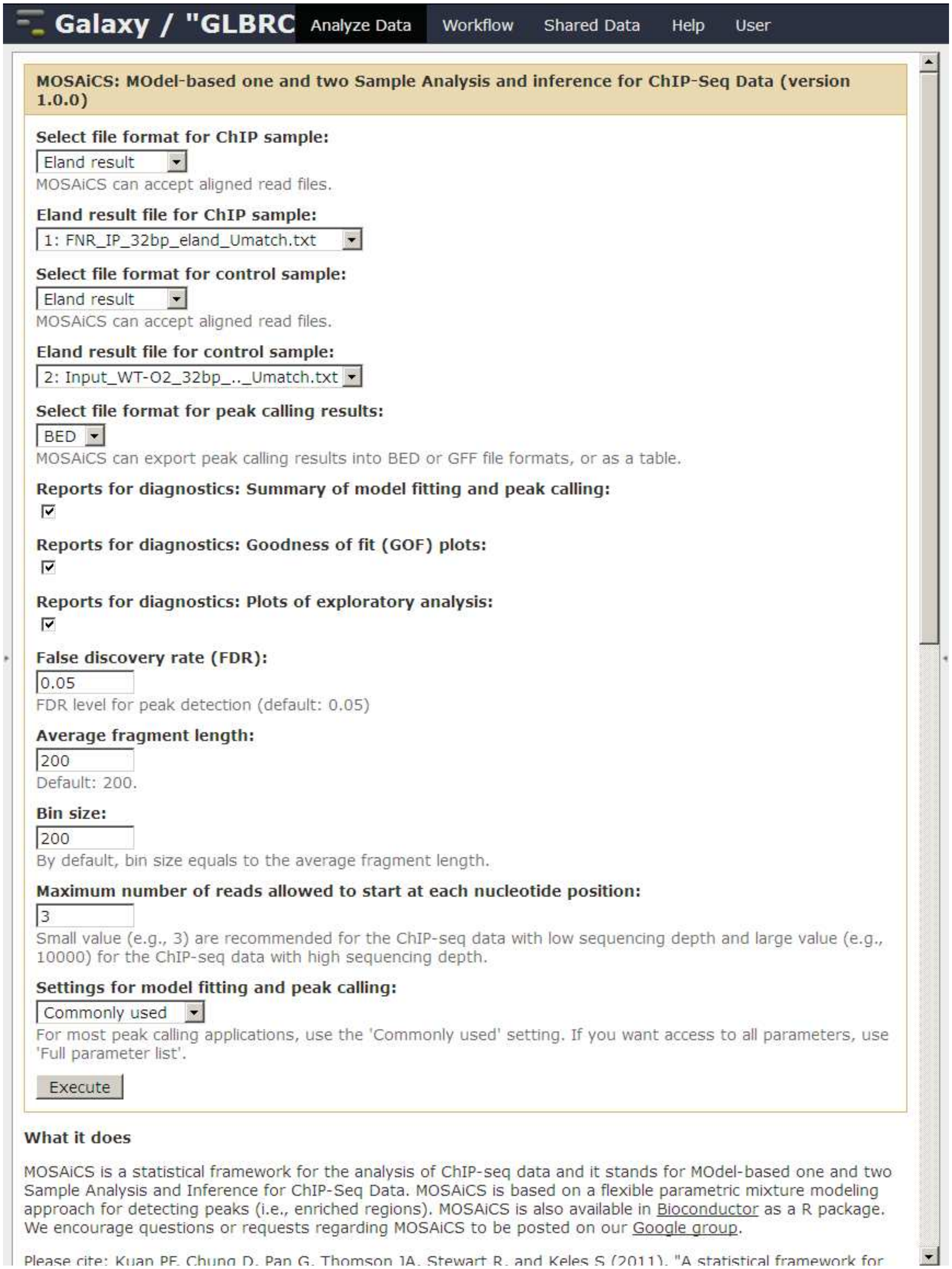
Input formats

CSEM accepts short reads aligned using bowtie as input. Bowtie accepts single-end reads, in FASTA or FASTQ format, as input. Quality scores of reads are ignored.

Pseudo-tags

For each read in the alignment file, CSEM estimates the fraction of the read allocated to each of its alignments. This fraction reflects the degree of confidence in each particular alignment. Currently, only the peak caller MOSAiCS can accept fractional of reads as input. However, you can incorporate multi-reads into ChIP-seq analysis with your favoriate peak-caller by utilizing this pseudo-tag functionality. Pseudo-tags are generated by assigning each multi-read to the location it maps to with the largest weight and filtering out multi-reads with weights less

Figure 4.3 **Galaxy interface for the** `csem` **software.**



MOSAiCS: MOdel-based one and two Sample Analysis and inference for ChIP-Seq Data (version 1.0.0)

Select file format for ChIP sample:
Eland result
MOSAiCS can accept aligned read files.

Eland result file for ChIP sample:
1: FNR_IP_32bp_eland_Umatch.txt

Select file format for control sample:
Eland result
MOSAiCS can accept aligned read files.

Eland result file for control sample:
2: Input_WT-O2_32bp_..._Umatch.txt

Select file format for peak calling results:
BED
MOSAiCS can export peak calling results into BED or GFF file formats, or as a table.

Reports for diagnostics: Summary of model fitting and peak calling:
☑

Reports for diagnostics: Goodness of fit (GOF) plots:
☑

Reports for diagnostics: Plots of exploratory analysis:
☑

False discovery rate (FDR):
0.05
FDR level for peak detection (default: 0.05)

Average fragment length:
200
Default: 200.

Bin size:
200
By default, bin size equals to the average fragment length.

Maximum number of reads allowed to start at each nucleotide position:
3
Small value (e.g., 3) are recommended for the ChIP-seq data with low sequencing depth and large value (e.g., 10000) for the ChIP-seq data with high sequencing depth.

Settings for model fitting and peak calling:
Commonly used
For most peak calling applications, use the 'Commonly used' setting. If you want access to all parameters, use 'Full parameter list'.

What it does

MOSAiCS is a statistical framework for the analysis of ChIP-seq data and it stands for MOdel-based one and two Sample Analysis and Inference for ChIP-Seq Data. MOSAiCS is based on a flexible parametric mixture modeling approach for detecting peaks (i.e., enriched regions). MOSAiCS is also available in Bioconductor as a R package. We encourage questions or requests regarding MOSAiCS to be posted on our Google group.

Please cite: Kuan PF, Chung D, Pan G, Thomson JA, Stewart R, and Keles S (2011). "A statistical framework for

Figure 4.4 **Galaxy interface for the `mosaics` software.**

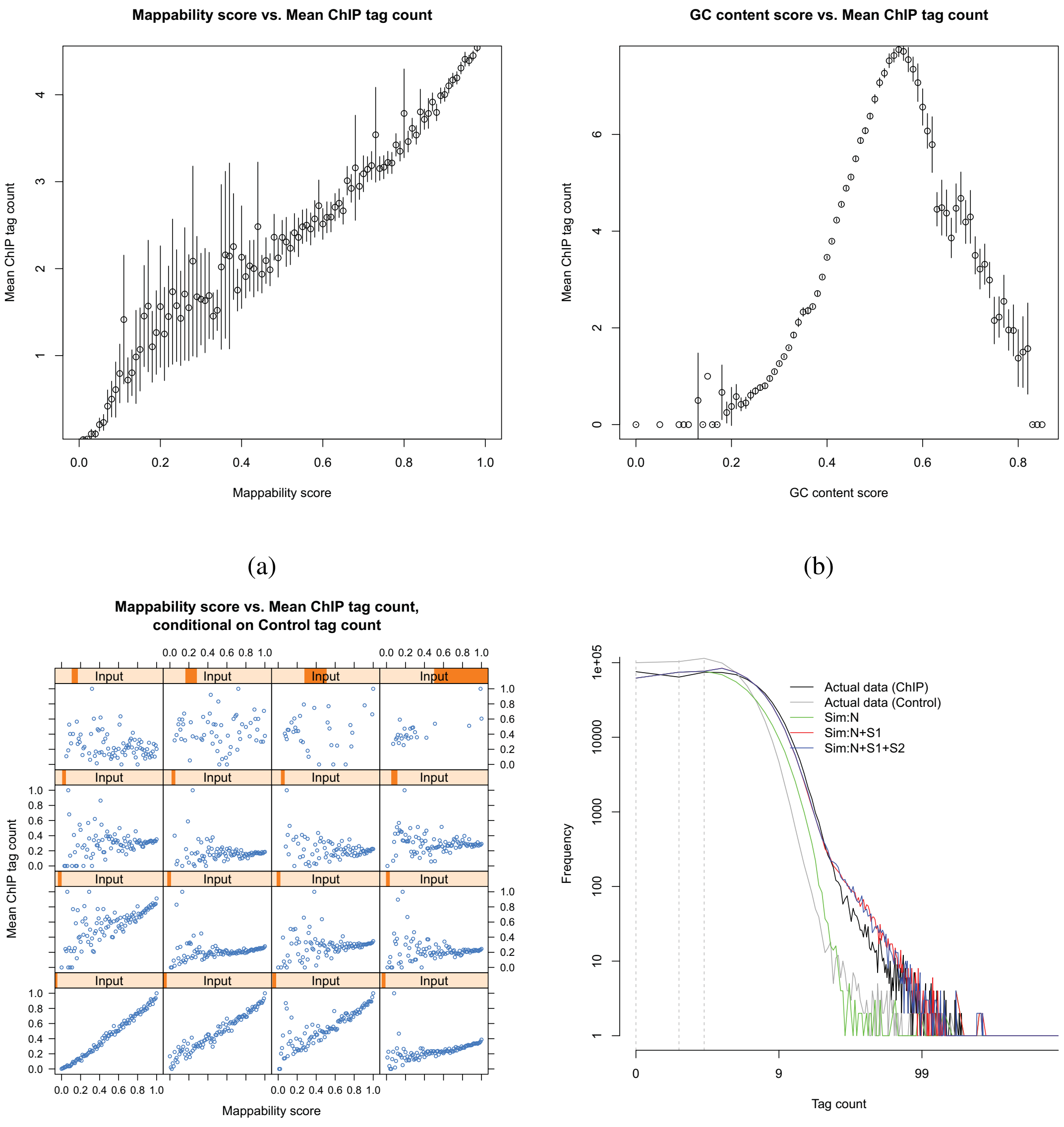
Table 4.2 **Multi-read allocation using the csem software.** In the first column, "(C)" and "(I)" refer to ChIP and input samples, respectively. Percentages in the third to fifth columns are calculated with respect to the total number of reads (the second column). "% Rescued" in the last column is obtained as the number of multi-reads divided by the number of uni-reads and it indicates the gain in sequencing depth due to multi-reads.

Dataset	# of reads	% Alignable	% Uni-reads	% Multi-reads	% Rescued
STAT1(C)	76,913,219	36.64	29.92	6.72	22.46
STAT1(I)	49,771,625	47.90	38.31	9.59	25.03
GATA1(C)	33,124,216	79.27	67.81	11.46	16.90
GATA1(I)	20,711,007	82.37	69.38	12.99	18.73

length, bin size, and maximum number of reads allowed to start at each nucleotide position. By checking radio boxes for reports, users can also obtain various reports for diagnostics, including summary of model fitting and peak calling, goodness of fit (GOF) plots, and plots of exploratory analysis. mosaics Galaxy tool provides well-tuned default tuning parameters for model fitting and peak calling.

mosaics R package consists of four main functions, constructBins(), readBins(), mosaicsFit(), and mosaicsPeak(). mosaicsRunAll() is a wrapper of these functions and it runs all the analysis workflow with a single command line. constructBins() converts the aligned read files into bin-level files. readBins() loads bin-level files and constructs a BinData class object. The BinData class object contains information about ChIP sample and subsets of matched control sample, mappability score, and/or GC content, depending on the analysis type. mosaicsFit() takes BinData class object as input and constructs a MosaicsFit class object, which is the fitted MOSAiCS model. Finally, mosaicsPeak() takes MosaicsFit class object as input and constructs a MosaicsPeak class object. The MosaicsPeak class object contains the peak list and related information.

Figure 4.5 shows diagnostics plots for GATA1 MR sample, generated by the mosaics software. Figure 4.5a,b,c are obtained by applying plot() method to BinData class object. Figures 4.5a and 4.5b show the plots of mean ChIP tag counts against mappability and GC content, respectively,



(a) (b)

(c) (d)

Figure 4.5 **Diagnostics plots and data exploratory plots for for the GATA1 MR data from the `mosaics` software.** (a) Mean ChIP tag counts against mappability. (b) Mean ChIP tag counts against GC content. (c) Mean ChIP tag counts against mappability, conditional on input tag counts. (d) Goodness of fit. Both axes are in log10 scale.

Table 4.3 **Peak calling for STAT1 and GATA1 data, using the `mosaics` software.** UR peaks are obtained by using uni-reads alone. MR peaks are obtained by using both uni-reads and multi-reads.

Dataset	# of UR-only peaks	# of common peaks	# of MR-only peaks
STAT1	0	23175	2546
GATA1	0	6038	2146

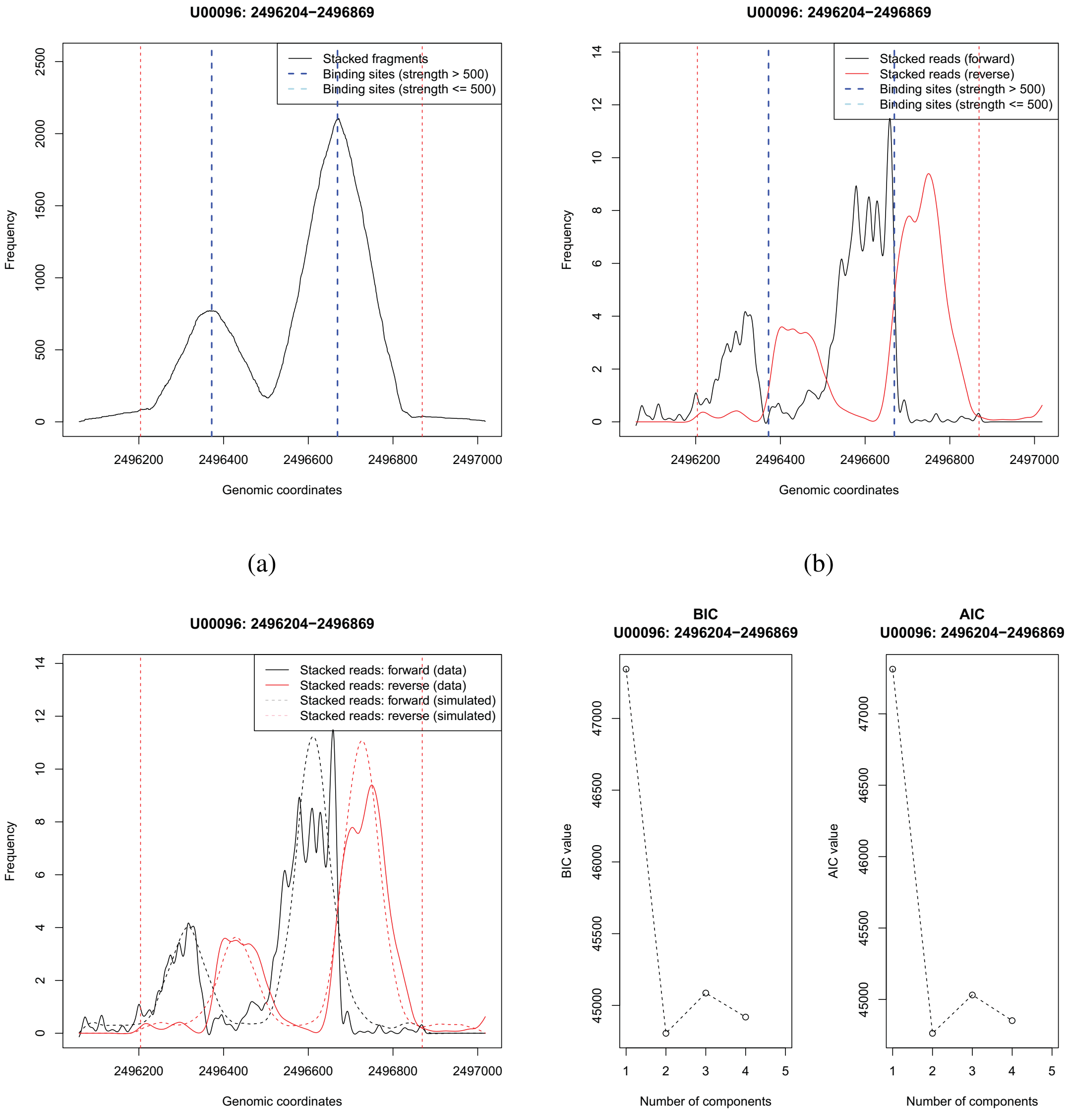
which are especially useful exploratory plots for the one-sample analysis. Figure 4.5c shows the plot of mean ChIP tag counts against mappability, conditional on input tag counts, and it indicates that adjustment using mappability is still needed for this data when input tag count is low.

Figure 4.5d is generated by applying `plot()` method to `MosaicsFit` class object and it indicates that the MOSAiCS model with two signal components fits the data well. `show()` and `print()` methods can be applied to `MosaicsPeak` class object and generate a brief summary of peak calling and return peak list (in data frame format), respectively. `export()` method applied to the `MosaicsPeak` class object will generate an output peak list file in either BED or GFF file formats.

For each of STAT1 and GATA1 data, we created two bin-level datasets using (1) uni-reads only (UR sample) and (2) both uni-reads and multi-reads (MR sample). Peaks obtained from each of these two samples are referred as UR peaks and MR peaks. Table 4.3 shows peak calling results using the `mosaics` software for each of STAT1 and GATA1 data. There are no UR-only peaks and `mosaics` identifies 11% and 36% more high quality peaks for STAT1 and GATA1, respectively, by utilizing multi-reads.

4.3.3 High resolution binding site identification using `dpeak`

By taking both a list of candidate regions and aligned reads in popular file formats as input, `dpeak` generates binding event prediction as output in either BED or GFF formats. This list of candidate regions can be provided by users or obtained from peak callers. Peak list file generated by `mosaics` can be directly utilized as input for `dpeak`. In the list of candidate regions, the first three columns are assumed to be chromosome ID, peak start position, and peak end position. The `dpeak`



(a)

(b)

(c)

(d)

Figure 4.6 **Prediction and diagnostics plots for *E. coli* σ^{70} data from the `dpeak` software.** (a, b) Data plots with predicted binding events. In (a), height at each position indicates the number of reads mapping to the position, where each read is extended to its 3' direction by average library size. In (b), heights of black solid and red dotted lines at each position indicate the number of 5' end of reads mapping to the position in the forward and reverse strands, respectively. In (a) and (b), blue vertical dashed lines indicate predicted positions of binding events. (c) Goodness of fit (GOF). In (a)-(c), red vertical dotted lines indicate start and end positions of the candidate region. (d) Plot of Bayesian information criterion (BIC) as a function of number of binding event components. Lower BIC value indicates better model fit.

R package consists of two main functions, `dpeakRead()` and `dpeakFit()`. `dpeakRead()` loads both the list of candidate regions and the aligned read file. It construct a `DpeakData` class object and this object contains information about reads mapping to each candidate region. `dpeakFit()` takes `DpeakData` class object as input and construct a `DpeakFit` class object. The `DpeakFit` class object contains binding event predictions for each candidate region.

We consider *E. coli* σ^{70} SET ChIP-Seq data (unpublished data provide by Professor Robert Landick, University of Wisconsin-Madison) for the illustration. We applied the `mosaics` software to this dataset and obtained the candidate regions. Here, we consider a candidate region, $(2496204, 2496869)$, for the illustration. Figure 4.6a,b show the data in this candidate region. These plots are generated by applying `plot()` method to `DpeakFit` class object with argument `plotType="fit"`. If `plot()` method is applied to `DpeakData` class object, similar plots are obtained but without predictions. Using these plots, we see that two peaks are visually identifiable in this region. However, due to short distance between these two peaks, they were identified as one region in the peak calling using `mosaics` software. The regions identified using MACS (Zhang et al., 2008), a popular peak caller, were even wider than ones identified using the `mosaics` software (data not shown).

The dPeak algorithm deconvolves this region and successfully predicts two binding events in this region (Figure 4.6a, b). Figure 4.6c and Figure 4.6d show dignostics plots for this region and they are generated by applying `plot()` method to `DpeakFit` class object with arguments `plotType="GOF"` and `plotType="BIC"`, respectively. Figure 4.6c indicates that the dPeak model fits the data well and Figure 4.6d shows that the model with two binding events are optimal (based the BIC score) in this region. `export()` method applied to the resulting `DpeakFit` class object generates the following BED file, where score indicates number of reads belonging to each binding event:

```
track name=dpeak description="Dpeak binding sites" useScore=1
U00096  2496362 2496382 U00096:2496204-2496869  705.228812549497
U00096  2496659 2496679 U00096:2496204-2496869  2137.71406388651
```

4.4 Conclusion

In spite of development of various statistical methods and their software for ChIP-Seq data analysis, there are still tremendous needs for more elaborated methods and software to address important biological problems in an unbiased way. User-friendly and computationally efficient software for `csem`, `mosaics`, and `dpeak` will help researchers investigate their biological questions of interest using ChIP-Seq data more accurately. Their Galaxy tools will provide even more easy-to-use interface and allow convenient integration of various bioinformatics tools and analysis of multiple related data.

Bibliography

Auerbach, R. K., Euskirchen, G., Rozowsky, J., Lamarre-Vincent, N., Moqtaderi, Z., Lefrançois, P., Struhl, K., Gerstein, M., and Snyder, M. (2009), "Mapping accessible chromatin regions using Sono-Seq," *Proceedings of the National Academy of Sciences of the United States of America*, 106, 14926–14931.

Bailey, J. and Eichler, E. (2006), "Primate segmental duplications: crucibles of evolution, diversity and disease," *Nature Reviews Genetics*, 7, 552–564.

Bailey, J., Gu, Z., Clark, R., Reinert, K., Samonte, R., Schwartz, S., Adams, M., Myers, E., Li, P., and Eichler, E. (2002), "Recent segmental duplications in the human genome," *Science*, 297, 1003–1007.

Bailey, T. L., Boden, M., Buske, F. A., Frith, M., Grant, C. E., Clementi, L., Ren, J., Li, W. W., and Noble, W. S. (2009), "MEME Suite: tools for motif discovery and searching," *Nucleic Acids Research*, 37, W202–W208.

Barski, A., Cuddapah, S., Cui, K., Roh, T., Schones, D., Wang, Z., Wei, G., Chepelev, I., and Zhao, K. (2007), "High-resolution profiling of histone methylations in the human genome," *Cell*, 129, 823–837.

Becker, N. and Marschner, I. (1993), "A method for estimating the age-specific relative risk of HIV infection from AIDS incidence data," *Biometrika*, 80, 165–178.

Becker, N., Watson, L., and Carlin, J. (1991), "A method of non-parametric back-projection and its application to AIDS data," *Statistics in Medicine*, 10, 1527–1542.

Benjamini, Y. and Hochberg, Y. (1995), “Controlling the false discovery rate: a practical and powerful approach to multiple testing,” *Journal of the Royal Statistical Society: Series B*, 57, 289–300.

Blahnik, K., Dou, L., O’Geen, H., McPhillips, T., Xu, X., Cao, A., Iyengar, S., Nicolet, C., Ludascher, B., Korf, I., and Farnham, P. (2009), “Sole-Search: an integrated analysis program for peak detection and functional annotation using ChIP-seq data,” *Nucleic Acids Research*, 38, e13.

Blankenberg, D., Kuster, G. V., Coraor, N., Ananda, G., Lazarus, R., Mangan, M., Nekrutenko, A., and Taylor, J. (2010), “Galaxy: a web-based genome analysis tool for experimentalists,” *Current Protocols in Molecular Biology*, 89, 19.10.1–19.10.21.

Bourque, G., Leong, B., Vega, V. B., Chen, X., Lee, Y. L., Srinivasan, K. G., Chew, J.-L., Ruan, Y., Wei, C.-L., Ng, H. H., and Liu, E. T. (2008), “Evolution of the mammalian transcription factor binding repertoire via transposable elements,” *Genome Research*, 18, 1752–1762.

Bulyk, M., McGuire, A., Masuda, N., and Church, G. (2004), “A motif co-occurrence approach for genome-wide prediction of transcription-factor-binding sites in *Escherichia coli*,” *Genome Research*, 14, 201–208.

Celeux, G. and Diebolt, J. (1985), “The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem,” *Computational Statistics Quarterly*, 2, 73–82.

Chen, Y., Negre, N., Li, Q., Mieczkowska, J. O., Slattery, M., Liu, T., Zhang, Y., Kim, T.-K., He, H. H., Zieba, J., Ruan, Y., Bickel, P. J., Myers, R. M., Wold, B. J., White, K. P., Lieb, J. D., and Liu, X. S. (2012), “Systematic evaluation of factors influencing ChIP-seq fidelity,” *Nature Methods*, 9, 609–614.

Cheng, Y., Wu, W., Kumar, S., Yu, D., Deng, W., Tripic, T., King, D., Chen, K., Zhang, Y., Drautz, D., Giardine, B., Schuster, S., Miller, W., Chiaromonte, F., Zhang, Y., Blobel, G., Weiss, M., and Hardison, R. (2009), “Erythroid GATA1 function revealed by genome-wide analysis of transcription factor occupancy, histone modifications, and mRNA expression,” *Genome Research*, 19, 2172–2184.

Chung, D., Kuan, P. F., Li, B., Sanalkumar, R., Liang, K., Bresnick, E. H., Dewey, C., and Keleş, S. (2011), “Discovering transcription factor binding sites in highly repetitive regions of genomes with multi-read analysis of ChIP-Seq data,” *PLoS Computational Biology*, 7, e1002111.

Crawford, S. (1994), “An application of Laplace method to finite mixture distributions,” *Journal of the American Statistical Association*, 89, 259–267.

Day, D., Luquette, L., Park, P., and Kharchenko, P. (2010), “Estimating enrichment of repetitive elements from high-throughput sequence data,” *Genome Biology*, 11, R69.

Dempster, A., Laird, N., and Rubin, D. (1977), “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society: Series B*, 39, 1–38.

Dennis, G., Sherman, B., Hosack, D., Yang, J., Gao, W., Lane, H., and Lempicki, R. (2003), “DAVID: Database for Annotation, Visualization, and Integrated Discovery,” *Genome Biology*, 4, P3.

Dohm, J., Lottaz, C., Borodina, T., and Himmelbauer, H. (2008), “Substantial biases in ultra-short read data sets from high-throughput DNA sequencing,” *Nucleic Acids Research*, 36, e105.

Evans, T., Reitman, M., and Felsenfeld, G. (1988), “An erythrocyte-specific DNA-binding factor recognizes a regulatory sequence common to all chicken globin genes,” *Proceedings of the National Academy of Sciences of the United States of America*, 85, 5976–5980.

Fan, J. and Gijbels, I. (1996), *Local Polynomial Modeling and its Applications*, Chapman and Hall.

Faulkner, G. J., Forrest, A. R. R., M.Chalk, A., Schroder, K., Hayashizaki, Y., Carninci, P., Hume, D., and Grimmond, S. (2008), “A rescue strategy for multimapping short sequence tags refines surveys of transcriptional activity by CAGE,” *Genomics*, 91, 281–288.

Feschotte, C. (2008), “Transposable elements and the evolution of regulatory networks,” *Nature Reviews Genetics*, 9, 397–405.

Fraley, C. and Raftery, A. (1998), “How many clusters? Which clustering method? Answers via model-based cluster analysis,” *The Computer Journal*, 41, 578–588.

Fraley, C. and Raftery, A. E. (2000), “Model-based clustering, discriminant analysis, and density estimation,” *Journal of the American Statistical Association*, 97, 611–631.

Fujiwara, T., O’Green, H., Keleş, S., Blahnik, K., Linneman, A. K., Kang, Y.-A., Choi, K., Farnham, P. J., and Bresnick, E. H. (2009), “Discovering hematopoietic mechanisms through genomewide analysis of GATA factor chromatin occupancy,” *Molecular Cell*, 36, 667–681.

Fullwood, M., Wei, C.-L., Liu, E., and Ruan, Y. (2009), “Next-generation DNA sequencing of paired-end tags (PET) for transcriptome and genome analyses,” *Genome Research*, 19, 521–532.

Gama-Castro, S., Salgado, H., Peralta-Gil, M., Santos-Zavaleta, A., Muniz-Rascado, L., Solano-Lira, H., et al. (2011), “RegulonDB version 7.0: transcriptional regulation of *Escherichia coli* K-12 integrated within genetic sensory response units (Gensor Units),” *Nucleic Acids Research*, 39, D98–D105.

Giardine, B., Riemer, C., Hardison, R. C., Burhans, R., Elnitski, L., Shah, P., Zhang, Y., Blankenberg, D., Albert, I., Taylor, J., Miller, W., Kent, W. J., and Nekrutenko, A. (2005), “Galaxy: A platform for interactive large-scale genome analysis,” *Genome Research*, 15, 1451–1455.

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

Goedert, J., Chen, B., Preiss, L., Aledort, L., and Rosenberg, P. (2007), “Reconstruction of the hepatitis C virus epidemic in the US hemophilia population, 1940-1990,” *American Journal of Epidemiology*, 165, 1443–1453.

Gonzalez, E., Kulkarni, H., Bolivar, H., Mangano, A., Sanchez, R., Catano, G., Nibbs, R., Freedman, B., Quinones, M., Bamshad, M., Murthy, K., Rovin, B., Bradley, W., Clark, R., Anderson, S., O'Connell, R., Agan, B., Ahuja, S., Bologna, R., nd M. Dolan, L. S., and Ahuja, S. (2005), "The influence of CCL3L1 gene-containing segmental duplications on HIV-1/AIDS susceptibility," *Science*, 307, 1434–1440.

Guo, Y., Papachristoudis, G., Altshuler, R. C., Gerber, G. K., Jaakkola, T. S., Gifford, D. K., and Mahony, S. (2010), "Discovering homotypic binding events at high spatial resolution." *Bioinformatics*, 26, 3028–34.

Hastie, T. and Tibshirani, R. (1990), *Generalized Additive Models*, Chapman and Hall.

Hedgepeth, J., Gallucci, V., O'Sullivan, F., and Thorne, R. (1999), "An expectation maximization and smoothing approach for indirect acoustic estimation of fish size and density," *ICES Journal of Marine Science*, 56, 36–50.

Huang, D., Sherman, B., and Lempicki, R. (2009), "Systematic and integrative analysis of large gene lists using DAVID Bioinformatics Resources," *Nature Protocols*, 4, 44–57.

Hurles, M. (2004), "Gene duplication: the genomic trade in spare parts," *PLoS Biology*, 2, e206.

Ji, H., Jiang, H., Ma, W., Johnson, D. S., Myers, R. M., and Wong, W. H. (2008), "An integrated software system for analyzing ChIP-chip and ChIP-seq data," *Nature Biotechnology*, 26, 1293–1300.

Johnson, D., Mortazavi, A., Myers, R., and Wold, B. (2007), "Genome-wide mapping of *in vivo* protein-DNA interactions," *Science*, 316, 1497–1502.

Jurka, J., Kapitonov, V., Pavlicek, A., Klonowski, P., Kohany, O., and Walichiewicz, J. (2005), "Repbase Update, a database of eukaryotic repetitive elements," *Cytogenetic and Genome Research*, 110, 462–467.

Kharchenko, P. V., Tolstorukov, M., and Park, P. J. (2008), "Design and analysis of ChIP-seq experiments for DNA-binding proteins," *Nature Biotechnology*, 26, 1351–1359.

Kuan, P., Chun, H., and Keles, S. (2008), "CMARRT: A tool for the analysis of ChIP-chip data from tiling arrays by incorporating the correlation structure," *Proceedings of the Pacific Symposium of Biocomputing*, 13, 515–526.

Kuan, P. F., Chung, D., Pan, G., Thomson, J. A., Stewart, R., and Keles, S. (2011), "A Statistical Framework for the Analysis of ChIP-Seq Data," *Journal of the American Statistical Association*, 106, 891–903.

Laird, N. (1978), "Nonparametric maximum likelihood estimation of a mixing distribution," *Journal of the American Statistical Association*, 73, 805–811.

Langmead, B., Trapnell, C., Pop, M., and Salzberg, S. L. (2009), "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome," *Genome Biology*, 10, R25.

Li, B., Ruotti, V., Stewart, R., Thomson, J., and Dewey, C. (2010), "RNA-Seq gene expression estimation with read mapping uncertainty," *Bioinformatics*, 26, 493–500.

Liu, L., Levine, M., and Zhu, Y. (2009), "A functional EM algorithm for mixing density estimation via nonparametric penalized likelihood maximization," *Journal of Computational and Graphical Statistics*, 18, 481–504.

Lun, D. S., Sherrid, A., Weiner, B., Sherman, D. R., and Galagan, J. E. (2009), "A blind deconvolution approach to high-resolution mapping of transcription factor binding sites from ChIP-seq data." *Genome Biology*, 10, R142.

Marques-Bonet, T., Girirajan, S., and Eichler, E. (2009), "The origins and impact of primate segmental duplications," *Trends in Genetics*, 25, 443–454.

McLachlan, G. and Krishnan, T. (2008), *The EM Algorithm and Extensions*, Wiley, New York, 2nd ed.

McLachlan, G. and Peel, D. (2000), *Finite Mixture Models*, Wiley, New York.

Mendoza-Vargas, A., Olvera, L., Olvera, M., Grande, R., Vega-Alvarado, L., Taboada, B., Jimenez-Jacinto, V., Salgado, H., Juarez, K., Contreras-Moreira, B., Huerta, A. M., Collado-Vides, J., and Morett, E. (2009), "Genome-wide identification of transcription start sites, promoters and transcription factor binding sites in *E. coli*," *PLoS ONE*, 4, e7526.

Meng, X.-L. and Rubin, D. B. (1993), "Maximum likelihood estimation via the ECM algorithm: a general framework," *Biometrika*, 80, 267–278.

Mikkelsen, T., Ku, M., Jaffe, D., Issac, B., Lieberman, E., Giannoukos, G., Alvarez, P., Brockman, W., Kim, T., Koche, R. P., Lee, W., Mendenhall, E., O'Donovan, A., Presser, A., Russ, C., Xie, X., Meissner, A., Wernig, M., Jaenisch, R., Nusbaum, C., Lander, E., and Bernstein, B. (2007), "Genome-wide maps of chromatin state in pluripotent and lineage-committed cells," *Nature*, 448, 653–560.

Mortazavi, A., Williams, B., McCue, K., Schaeffer, L., and Wold, B. (2008), "Mapping and quantifying mammalian transcriptomes by RNA-Seq," *Nature Methods*, 5, 621–628.

Nadaraya, E. (1964), "On estimating regression," *Theory of Probability and its Applications*, 9, 141–142.

Nicholas, T., Cheng, Z., Ventura, M., Mealey, K., Eichler, E., and Akey, J. (2009), "The genomic architecture of segmental duplications and associated copy number variants in dogs," *Genome Research*, 19, 491–499.

Noto, K. and Craven, M. (2007), "Learning probabilistic models of *cis*-regulatory modules that represent logical and spatial aspects," *Bioinformatics*, 23, e156–e162.

Nychka, D. (1990), "Some properties of adding a smoothing step to the EM algorithm," *Statistics and Probability Letters*, 9, 187–193.

Park, P. (2009), "ChIP-seq: advantages and challenges of a maturing technology," *Nature Reviews Genetics*, 10, 669–680.

Polak, P. and Domany, E. (2006), “Alu elements contain many binding sites for transcription factors and may play a role in regulation of developmental processes,” *BMC Genomics*, 7, 133.

Portales-Casamar, E., Thongjuea, S., Kwon, A. T., Arenillas, D., Zhao, X., Valen, E., Yusuf, D., Lenhard, B., Wasserman, W. W., and Sandelin, A. (2010), “JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles,” *Nucleic Acids Research*, 38, D105–D110.

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

Reznikoff, W. S., Siegele, D. A., Cowing, D. W., and Gross, C. A. (1985), “The regulation of transcription initiation in bacteria,” *Annual Review of Genetics*, 19, 355–387.

Rhead, B., Karolchik, D., Kuhn, R., Hinrichs, A., Zweig, A., Fujita, P., Diekhans, M., Smith, K., Rosenbloom, K., Raney, B., Pohl, A., Pheasant, M., Meyer, L., Learned, K., Hillman-Jackson, J., Harte, R., Giardine, B., Dreszer, T., Clawson, H., Barber, G., Haussler, D., and Kent, W. (2010), “The UCSC Genome Browser database: update 2010,” *Nucleic Acids Research*, 38, D613–D619.

Roman, A. C., Benitez, D. A., Carvajal-Gonzalez, J. M., and Fernandez-Salguero, P. M. (2008), “Genome-wide B1 retrotransposon binds the transcription factors dioxin receptor and Slug and regulates gene expression *in vivo*,” *Proceedings of the National Academy of Sciences of the United States of America*, 105, 1632–1637.

Rowen, L., Williams, E., Glusman, G., Linardopoulou, E., Friedman, C., Ahearn, M., J, S., Boysen, C., Qin, S., Wang, K., Kaur, A., Bloom, S., Hood, L., and Trask., B. (2005), “Interchromosomal segmental duplications explain the unusual structure of PRSS3, the gene for an inhibitor-resistant trypsinogen,” *Molecular Biology and Evolution*, 22, 1712–1720.

Rozowsky, J., Euskirchen, G., Auerbach, R., Zhang, D., Gibson, T., Bjornson, R., Carriero, N., Snyder, M., and Gerstein, M. (2009), "PeakSeq enables systematic scoring of ChIP-Seq experiments relative to controls," *Nature Biotechnology*, 27, 66–75.

Schwarz, G. (1978), "Estimating the dimension of a model," *The Annals of Statistics*, 6, 461–464.

Seo, Y., Chong, H., Infante, A., In, S., Xie, X., and Osborne, T. (2009), "Genome-wide analysis of SREBP-1 binding in mouse liver chromatin reveals a preference for promoter proximal binding to a new motif," *Proceedings of the National Academy of Sciences of the United States of America*, 106, 13765–13769.

Silverman, B., Jones, M., Wilson, J., and Nychka, D. (1990), "A smoothed EM approach to indirect estimation problems, with particular reference to stereology and estimation tomography (with discussion)," *Journal of the Royal Statistical Society: Series B*, 52, 271–324.

Smit, A. F. A., Hubley, R., and Green, P. (2010), "RepeatMasker Open-3.0. 1996-2010," URL: `http://www.repeatmasker.org`.

Taub, M., Lipson, D., and Speed, T. (2010), "Methods for allocating ambiguous short-reads," *Communications in Information and Systems*, 10, 69–82.

Valouev, A., Johnson, D., Sundquist, A., Medina, C., Anton, E., Batzoglou, S., Myers, R., and Sidow, A. (2008), "Genome-wide analysis of transcription factor binding sites based on ChIP-Seq data," *Nature Methods*, 5, 829–834.

Wang, J., Huda, A., Lunyak, V. V., and Jordan, I. K. (2010), "A Gibbs sampling strategy applied to the mapping of ambiguous short-sequence tags," *Bioinformatics*, 26, 2501–2508.

Wang, T., Zeng, J., Lowe, C. B., Sellers, R. G., Salama, S. R., Yang, M., Burgess, S. M., Brachmann, R. K., and Haussler, D. (2007), "Species-specific endogenous retroviruses shape the transcriptional network of the human tumor suppressor protein p53," *Proceedings of the National Academy of Sciences of the United States of America*, 104, 18613–18618.

Watson, G. (1964), "Smooth regression analysis," *Sankhya: The Indian Journal of Statistics, Series A*, 26, 359–372.

Wilbanks, E. G. and Facciotti, M. T. (2010), "Evaluation of algorithm performance in ChIP-Seq peak detection," *PLoS ONE*, 5, e11471.

Wu, C., Orozco, C., Boyer, J., Leglise, M., Goodale, J., Batalov, S., Hodge, C., Haase, J., Janes, J., Huss, J., and Su, A. (2009), "BioGPS: an extensible and customizable portal for querying and organizing gene annotation resources," *Genome Biology*, 10, R130.

Wu, W., Cheng, Y., Keller, C. A., Ernst, J., Kumar, S. A., Mishra, T., Morrissey, C., Dorman, C. M., Chen, K.-B., Drautz, D., Giardine, B., Shibata, Y., Song, L., Pimkin, M., Crawford, G. E., Furey, T. S., Kellis, M., Miller, W., Taylor, J., Schuster, S. C., Zhang, Y., Chiaromonte, F., Blobel, G. A., Weiss, M. J., and Hardison, R. C. (2011), "Dynamics of the epigenetic landscape during erythroid differentiation after GATA1 restoration," *Genome Research*, 21, 1659–1671.

Zhang, X., Robertson, G., Krzywinski, M., Ning, K., Droit, A., Jones, S., and Gottardo, R. (2011), "PICS: probabilistic inference for ChIP-seq." *Biometrics*, 67, 151–63.

Zhang, Y., Liu, T., Meyer, C., Eeckhoute, J., Johnson, D., Bernstein, B., Nussbaum, C., Myers, R., Brown, M., Li, W., and Liu, X. (2008), "Model-based Analysis of ChIP-Seq (MACS)," *Genome Biology*, 9, R137.

DISCARD THIS PAGE

Appendix A: Derivation of the EM algorithm for dPeak PET model

The complete likelihood is obtained as

$$\begin{aligned}
L_c &= \prod_{i=1}^{n} p(s_i, l_i, z_i) = \prod_{i=1}^{n} p(s_i|l_i, z_i)p(l_i)p(z_i) \\
&= \prod_{i=1}^{n} p(l_i) \prod_{g=0}^{g*} \{P(Z_i = g)p(s_i|l_i, Z_i = g)\}^{z_i g} \\
&= \prod_{i=1}^{n} p(l_i) \prod_{g=0}^{g*} \{\pi_g p(s_i|l_i, Z_i = g)\}^{z_i g}
\end{aligned}$$

and hence, the complete log likelihood is obtained as

$$\begin{aligned}
\log L_c = \sum_{i=1}^{n}[\log p(l_i) &+ z_{i0}\log \pi_0 + z_{i0}\log p(s_i|l_i, Z_i = 0) \\
&+ \sum_{g=1}^{g*} z_{ig}\log \pi_g + \sum_{g=1}^{g*} z_{ig}\log p(s_i|l_i, Z_i = g)] \\
= \sum_{i=1}^{n}[\log p(l_i) &+ z_{i0}\log \pi_0 - z_{i0}\log(m + l_i - 1) + \sum_{g=1}^{g*} z_{ig}\log \pi_g \\
&+ \sum_{g=1}^{g*} z_{ig} 1\{s_i \in B_g\} \log \frac{1-\gamma}{l_i} \\
&+ \sum_{g=1}^{g*} z_{ig} 1\{s_i \in C \backslash B_g\} \log \frac{\gamma}{m-1}].
\end{aligned}$$

The EM algorithm for the PET data can be summarized as follows:

E-step:

Let $\Theta = (\pi_0, \pi_1, \pi_2, \cdots, \pi_{g*}, \mu_1, \mu_2, \cdots, \mu_{g*}, \gamma)$. For $g = 1, 2, \cdots, g^*$,

$$z_{ig}^{(t)} = P(Z_{ig} = 1|S_1, S_2, \cdots, S_n, L_1, L_2, \cdots, L_n; \Theta^{(t)})$$

$$
\begin{aligned}
&= P(Z_{ig} = 1 | S_i, L_i; \Theta^{(t)}) \\
&= \frac{P(Z_{ig} = 1; \Theta^{(t)}) P(S_i | L_i, Z_{ig} = 1; \Theta^{(t)})}{\sum_{g'=0}^{g*} P(Z_{ig'} = 1; \Theta^{(t)}) P(S_i | L_i, Z_{ig'} = 1; \Theta^{(t)})} \\
&= \frac{\pi_g^{(t)}}{A} \left[\frac{(1-\gamma^{(t)})}{L_i} \right]^{1\left\{S_i \in B_g^{(t)}\right\}} \left[\frac{\gamma^{(t)}}{m-1} \right]^{1\left\{S_i \in C \backslash B_g^{(t)}\right\}},
\end{aligned}
$$

and similarly, for $g = 0$,

$$
z_{i0}^{(t)} = \frac{\pi_0^{(t)}}{A(m + L_i - 1)},
$$

if we define $A \equiv \sum_{g'=0}^{g*} P(Z_{ig'} = 1; \Theta^{(t)}) P(S_i | L_i, Z_{ig'} = 1; \Theta^{(t)})$.

M-step:

If we fix $\mu'_g, \forall g' \neq g$ and consider only terms related to only μ_g in $\log L_c$, then we have

$$
\mu_g^{(t+1)} = argmax_{\mu_g} \sum_{i=1}^{n} z_{ig}^{(t)} \left[1\left\{S_i \in B_g\right\} \log \frac{(1-\gamma^{(t)})}{L_i} + 1\left\{S_i \in C \backslash B_g\right\} \log \frac{\gamma^{(t)}}{m-1} \right].
$$

If we differentiate $\log L_c$ with respect to $\pi_g, g = 0, 1, 2, \cdots, g^*$ under the sum constraint $\sum_{g=0}^{g*} \pi_g = 1$, then for $g = 0, 1, 2, \cdots, g^*$,

$$
\pi_g^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} z_{ig}^{(t)}
$$

Finally, if we differentiate $\log L_c$ with respect to γ, then we have

$$
\gamma^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} \sum_{g=1}^{g^*} z_{ig}^{(t)} 1\left\{S_i \in C \backslash B_g^{(t+1)}\right\}.
$$

Appendix B: Derivation of the EM algorithm for dPeak SET model

The complete likelihood is obtained as

$$\begin{aligned}
L_c &= \prod_{i=1}^{n} p(r_i, d_i, z_i) = \prod_{i=1}^{n} p(r_i|d_i, z_i)p(d_i)p(z_i) \\
&= \prod_{i=1}^{n}\prod_{g=0}^{g*}[P(Z_i = g)\left\{p_D p(r_i|D_i = 1, Z_i = g)\right\}^{d_i} \\
&\quad \left\{(1-p_D)p(r_i|D_i = 0, Z_i = g)\right\}^{(1-d_i)}]^{z_i g} \\
&= \prod_{i=1}^{n}\prod_{g=0}^{g*}[\pi_g\left\{p_D p(r_i|D_i = 1, Z_i = g)\right\}^{d_i} \\
&\quad \left\{(1-p_D)p(r_i|D_i = 0, Z_i = g)\right\}^{(1-d_i)}]^{z_i g}
\end{aligned}$$

and hence, the complete log likelihood is obtained as

$$\begin{aligned}
\log L_c &= \sum_{i=1}^{n}[z_{i0}\left\{\log\pi_0 + d_i\log p_D + (1-d_i)\log(1-p_D)\right\} \\
&+ z_{i0}\left\{d_i\log p(r_i|D_i = 1, Z_i = 0) + (1-d_i)\log p(r_i|D_i = 0, Z_i = 0)\right\} \\
&+ \sum_{g=1}^{g*} z_{ig}\left\{\log\pi_g + d_i\log p_D + (1-d_i)\log(1-p_D)\right\} \\
&+ \sum_{g=1}^{g*} z_{ig}\left\{d_i\log p(r_i|D_i = 1, Z_i = g) + (1-d_i)\log p(r_i|D_i = 0, Z_i = g)\right\}] \\
&= \sum_{i=1}^{n}[z_{i0}\left\{\log\pi_0 + d_i\log p_D + (1-d_i)\log(1-p_D)\right\} - z_{i0}\log(m+\beta-1) \\
&+ \sum_{g=1}^{g*} z_{ig}\left\{\log\pi_g + d_i\log p_D + (1-d_i)\log(1-p_D)\right\} \\
&+ \sum_{g=1}^{g*} z_{ig}\left\{-\frac{1}{2}\log(2\pi\sigma^2) - \frac{d_i}{2\sigma^2}(r_i - (\mu_g - \delta))^2 - \frac{(1-d_i)}{2\sigma^2}(r_i - (\mu_g + \delta))^2\right\}].
\end{aligned}$$

The EM algorithm for the SET data can be summarized as follows:

E-step:

Let $\Theta = (\pi_0, \pi_1, \pi_2, \cdots, \pi_{g*}, \mu_1, \mu_2, \cdots, \mu_{g*}, \delta, \sigma^2)$. For $g = 1, 2, \cdots, g^*$,

$$\begin{aligned}
z_{ig}^{(t)} &= P(Z_{ig} = 1 | R_1, R_2, \cdots, R_n, D_1, D_2, \cdots, D_n; \Theta^{(t)}) \\
&= P(Z_{ig} = 1 | R_i, D_i; \Theta^{(t)}) \\
&= \frac{P(Z_{ig} = 1; \Theta^{(t)}) P(R_i | D_i, Z_{ig} = 1; \Theta^{(t)})}{\sum_{g'=0}^{g*} P(Z_{ig'} = 1; \Theta^{(t)}) P(R_i | D_i, Z_{ig'} = 1; \Theta^{(t)})} \\
&= \frac{\pi_g^{(t)}}{A\sqrt{2\pi(\sigma^2)^{(t)}}} \left[p_D \exp\left\{ -\frac{1}{2(\sigma^2)^{(t)}} (R_i - (\mu_g^{(t)} - \delta^{(t)}))^2 \right\} \right]^{1\{D_i=1\}} \\
&\quad \left[(1 - p_D) \exp\left\{ -\frac{1}{2(\sigma^2)^{(t)}} (R_i - (\mu_g^{(t)} + \delta^{(t)}))^2 \right\} \right]^{1\{D_i=0\}},
\end{aligned}$$

and for $g = 0$,

$$z_{i0}^{(t)} = \frac{\pi_0^{(t)} p_D^{1\{D_i=1\}} (1 - p_D)^{1\{D_i=0\}}}{A(m + \beta - 1)},$$

if we define $A \equiv \sum_{g'=0}^{g*} P(Z_{ig'} = 1; \Theta^{(t)}) P(R_i | D_i, Z_{ig'} = 1; \Theta^{(t)})$.

M-step:

If we differentiate $\log L_c$ with respect to $\mu_g, g = 1, 2, \cdots, g^*$, we obtain

$$\mu_g^{(t+1)} = \frac{1}{\sum_{i=1}^n z_{ig}^{(t)}} \sum_{i=1}^n z_{ig}^{(t)} \left[(R_i + \delta^{(t)}) 1\{D_i = 1\} + (R_i - \delta^{(t)}) 1\{D_i = 0\} \right].$$

If we differentiate $\log L_c$ with respect to $\pi_g, g = 0, 1, 2, \cdots, g^*$ under the sum constraint $\sum_{g=0}^{g*} \pi_g = 1$, then for $g = 0, 1, 2, \cdots, g^*$,

$$\pi_g^{(t+1)} = \frac{1}{n} \sum_{i=1}^n z_{ig}^{(t)}.$$

Finally, if we differentiate $\log L_c$ with respect to each of δ and σ^2, then we have

$$\delta^{(t+1)} = \frac{1}{n}\sum_{i=1}^{n}\sum_{g=1}^{g^*} z_{ig}^{(t)} \left[(\mu_g^{(t+1)} - R_i)1\left\{D_i = 1\right\} + (R_i - \mu_g^{(t+1)})1\left\{D_i = 0\right\}\right],$$

and

$$\begin{aligned}(\sigma^2)^{(t+1)} = \frac{1}{n}\sum_{i=1}^{n}\sum_{g=1}^{g^*} \quad z_{ig}^{(t)} \quad & [\left\{R_i - (\mu_g^{(t+1)} - \delta^{(t+1)})\right\}^2 1\left\{D_i = 1\right\} + \\ & \left\{R_i - (\mu_g^{(t+1)} + \delta^{(t+1)})\right\}^2 1\left\{D_i = 0\right\}],\end{aligned}$$

respectively.

Appendix C: `mosaics` Galaxy Tool: Tool Definition File

```
<tool id="MOSAiCS" name="MOSAiCS: MOdel-based one and two Sample Analysis
  and inference for ChIP-Seq Data" version="1.0.0">

  <description></description>

  <parallelism method="basic"></parallelism>

  <requirements>
    <requirement type="binary">R</requirement>
  </requirements>

  <command interpreter="perl">
    mosaics_wrapper.pl
      ## input file name (chip and control)
      $chipParams.chip
      $controlParams.control
      ## input file format (chip and control)
      $chipParams.chipFileFormat
      $controlParams.controlFileFormat
      ## peak file name
      $out_peak
      ## peak file format
      $OutfileFormat
      ## analysis type
      IO
      ## optional output
      $report_summary
      $report_gof
      $report_exploratory
      ## settings for model fitting and peak calling:
      ## required (0.05, 200, 50)
      $fdrLevel
      $fragLen
      $binSize
      $capping
      ## settings for model fitting and peak calling: optional
      #if $fitParams.fSettingsType == "preSet"
  BIC
  0.25
```

```
200
50
10
    #else
$fitParams.signalModel
$fitParams.d
$fitParams.maxgap
$fitParams.minsize
$fitParams.thres
    #end if
    ## Number of cores to use
    8
</command>

<inputs>
<conditional name="chipParams">
  <param name="chipFileFormat" type="select"
  label="Select file format for ChIP sample"
  help="MOSAiCS can accept aligned read files.">
    <option value="eland_result">Eland result</option>
    <option value="eland_extended">Eland extended</option>
    <option value="eland_export">Eland export</option>
    <option value="bowtie">Bowtie default</option>
    <option value="sam">SAM</option>
  </param>
  <when value="eland_result">
    <param name="chip" type="data" format="eland"
    label="Eland result file for ChIP sample"/>
  </when>
  <when value="eland_extended">
    <param name="chip" type="data" format="eland"
    label="Eland extended file for ChIP sample"/>
  </when>
  <when value="eland_export">
    <param name="chip" type="data" format="eland"
    label="Eland export file for ChIP sample"/>
  </when>
  <when value="bowtie">
    <param name="chip" type="data"
    label="Bowtie default file for ChIP sample"/>
  </when>
  <when value="sam">
```

```
    <param name="chip" type="data"
    format="sam" label="SAM file for ChIP sample"/>
  </when>
</conditional> <!-- chipParams -->
<conditional name="controlParams">
  <param name="controlFileFormat" type="select"
  label="Select file format for control sample"
  help="MOSAiCS can accept aligned read files.">
    <option value="eland_result">Eland result</option>
    <option value="eland_extended">Eland extended</option>
    <option value="eland_export">Eland export</option>
    <option value="bowtie">Bowtie default</option>
    <option value="sam">SAM</option>
  </param>
  <when value="eland_result">
    <param name="control" type="data" format="eland"
    label="Eland result file for control sample"/>
  </when>
  <when value="eland_extended">
    <param name="control" type="data" format="eland"
    label="Eland extended file for control sample"/>
  </when>
  <when value="eland_export">
    <param name="control" type="data" format="eland"
    label="Eland export file for control sample"/>
  </when>
  <when value="bowtie">
    <param name="control" type="data"
    label="Bowtie default file for control sample"/>
  </when>
  <when value="sam">
    <param name="control" type="data" format="sam"
    label="SAM file for control sample"/>
  </when>
</conditional> <!-- inputParams -->

<param name="OutfileFormat" type="select"
label="Select file format for peak calling results"
help="MOSAiCS can export peak calling results
into BED or GFF file formats, or as a table.">
  <option value="bed">BED</option>
  <option value="gff">GFF</option>
  <option value="txt">table</option>
</param>
<param name="summary" type="boolean" truevalue="1" falsevalue="0"
display="checkboxes"
label="Reports for diagnostics:
Summary of model fitting and peak calling" />
<param name="gof" type="boolean" truevalue="1" falsevalue="0"
display="checkboxes"
label="Reports for diagnostics: Goodness of fit (GOF) plots" />
<param name="exploratory" type="boolean" truevalue="1" falsevalue="0"
display="checkboxes"
label="Reports for diagnostics: Plots of exploratory analysis" />

<param name="fdrLevel" type="float" value="0.05" min="0" max="1"
label="False discovery rate (FDR)"
help="FDR level for peak detection (default: 0.05)" />
<param name="fragLen" type="integer" value="200"
label="Average fragment length" help="Default: 200." />
<param name="binSize" type="integer" value="200"
label="Bin size" h
elp="By default, bin size equals to the average fragment length." />
<param name="capping" type="integer" value="3"
label="Maximum number of reads allowed
to start at each nucleotide position"
help="Small value (e.g., 3) are recommended for the ChIP-seq data
with low sequencing depth and large value (e.g., 10000)
for the ChIP-seq data with high sequencing depth." />

<conditional name="fitParams">
  <param name="fSettingsType" type="select" l
  abel="Settings for model fitting and peak calling"
  help="For most peak calling applications,
  use the 'Commonly used' setting.
  If you want access to all parameters, use 'Full parameter list'.">
    <option value="preSet">Commonly used</option>
    <option value="full">Full parameter list</option>
  </param>
  <when value="preSet" />
  <when value="full">
    <param name="signalModel" type="select" label="Signal model"
    help="By default, signal model is chosen using BIC.
    Instead, user can specify signal model among one or
    two signal component models.">
      <option value="BIC">Automatic model selection based on BIC</option>
      <option value="1S">One-signal-component model</option>
      <option value="2S">Two-signal-component model</option>
    </param>
    <param name="d" type="float" value="0.25" label="d"
    help="Parameter for estimating background distribution.
    Default is 0.25." />
    <param name="maxgap" type="integer" value="200"
    label="maxgap" help="Initial nearby peaks are merged
    if the distance (in bp) between them is less than ’maxgap’.
    Default is 200." />
    <param name="minsize" type="integer" value="50"
    label="minsize" help="An initial peak is removed
    if its width is narrower than ’minsize’. Default is 50." />
    <param name="thres" type="integer" value="10"
    label="thres" help="A bin within initial peak is removed
    if its ChIP tag counts are less than ’thres’. Default is 10." />
  </when> <!-- full -->
</conditional> <!-- fitParams -->
</inputs>

<outputs>
<data format="tabular" name="out_peak">
  <change_format>
    <when input="OutfileFormat" value="bed" format="bed" />
    <when input="OutfileFormat" value="gff" format="gff" />
  </change_format>
</data>
<data format="txt" name="report_summary">
  <filter>summary == 1</filter>
</data>
<data format="pdf" name="report_gof">
  <filter>gof == 1</filter>
</data>
<data format="pdf" name="report_exploratory">
  <filter>exploratory == 1</filter>
</data>
</outputs>

<help>
```

```
**What it does**

MOSAiCS is a statistical framework for the analysis of ChIP-seq data and
it stands for MOdel-based one and two Sample Analysis and
 Inference for ChIP-Seq Data. MOSAiCS is based on a flexible parametric
 mixture modeling approach for detecting peaks (i.e., enriched regions).
MOSAiCS is also available in Bioconductor_ as a R package.
We encourage questions or requests regarding MOSAiCS to be posted
on our ‘Google group‘_.

Please cite: Kuan PF, Chung D, Pan G, Thomson JA, Stewart R, and
Keles S (2011), "‘A statistical framework for the analysis of
ChIP-Seq data‘_," To appear in the *Journal of the American Statistical
Association*.

.. _Bioconductor: http://www.bioconductor.org/help/bioc-views/
2.8/bioc/html/mosaics.html
.. _Google group: http://groups.google.com/group/mosaics_user_group
.. _A statistical framework for the analysis of ChIP-Seq data:
http://pubs.amstat.org/doi/abs/10.1198/jasa.2011.ap09706

------

**Input formats**

MOSAiCS accepts aligned read files of ChIP and control samples as input.
Currently, MOSAiCS accepts single-end reads, in Eland result,
Eland extended, Eland export, Bowtie default, and SAM formats.

------

**Outputs**

Peak calling results of MOSAiCS can be exported into BED or GFF file
formats, or as a table.
Each line of the output file specifies a single peak.

If the output is a table, it has the following columns::

   Column    Description
   --------  --------------------------------------------------------
     1       Chromosome of the peak
     2       Start position of the peak
     3       End position of the peak
     4       Width of the peak
     5       Averaged posterior probability of the peak
     6       Minimum posterior probability of the peak
     7       Averaged ChIP tag counts of the peak
     8       Maximum ChIP tag counts of the peak
     9       Averaged control tag counts of the peak
    10       Averaged control tag counts of the peak,
             scaled by sequencing depth
    11       Averaged log base 2 ratio of ChIP over input tag counts

If the output is in BED format, it has the following columns::

    Column        Description
    ------------  --------------------------------------------------------
    1 chrom       Chromosome of the peak
    2 chromStart  Start position of the peak
    3 chromEnd    End position of the peak
    4 name        Always "MOSAiCS_peak"
    5 score       Averaged ChIP tag counts of the peak

If the output is in GFF format, it has the following columns::

    Column     Description
    ---------  --------------------------------------------------------
    1 seqname  Chromosome of the peak
    2 source   Always "MOSAiCS"
    3 feature  Always "MOSAiCS_peak"
    4 start    Start position of the peak
    5 end      End position of the peak
    6 score    Averaged ChIP tag counts of the peak
    7 strand   Always "."
    8 frame    Always "."
    9 group    Always "."

------

**Reports for diagnostics**

*Summary of model fitting and peak calling*: This report provides
information about input and output files, parameter settings
```

```
used for model fitting and peak calling,
and brief summary of peak calling results.

*Goodness of fit (GOF) plots*: This report allows visual comparisons
of the fits of the background, one-signal-component,
and two-signal-component models with the actual data.

*Plots of exploratory analysis*: This report provides the histograms
of ChIP and control samples and
the scatter plots of ChIP versus control tag counts.

More details regarding these reports can be found here_:

------

**Settings for model fitting and peak calling**

More details about the tuning of these parameters can be found here_:

.. _here: http://www.bioconductor.org/packages/2.8/bioc/vignettes/
mosaics/inst/doc/mosaics-example.pdf

  </help>
</tool>
```

Appendix D: `mosaics` Galaxy Tool: Perl Wrapper for `mosaics` Package

```
# Wrapper for MOSAiCS
# Written by Dongjun Chung, Sep. 15, 2011

#!/usr/bin/env perl;
use warnings;
use strict;
use File::Temp qw/tempfile/;
use File::Temp qw/tempdir/;
use File::Basename;

# parse command arguments

die "Usage: perl mosaics_wrapper.pl [chip_path] [control_path]
    [chip_file_format] [control_file_format] [peak_path] [peak_file_format]
    [analysis_type] [report_summary_path] [report_gof_path]
    [report_exploratory_path] [fdr_level] [frag_len] [bin_size] [capping]
    [signal_model] [d] [maxgap] [minsize] [thres] [n_core]"
   unless @ARGV == 20;

my ( $chip_path, $control_path, $chip_file_format, $control_file_format,
   $peak_path, $peak_file_format, $analysis_type, $report_summary_path,
   $report_gof_path, $report_exploratory_path, $fdr_level,
   $frag_len, $bin_size, $capping, $signal_model,
   $d, $maxgap, $minsize, $thres, $n_core ) = @ARGV;

# parse options: analysis type

if ( $analysis_type ne "IO" ) {
   print "Only 'IO' is supported for analysis type!\n";
   exit 1;
}

# parse options: ChIP, control, peak

my ($chip_filename, $chip_dir) = fileparse($chip_path);
my ($control_filename, $control_dir) = fileparse($control_path);
my ($peak_filename, $peak_dir) = fileparse($peak_path);
```

```
# parse options: report summary

my $report_summary = "FALSE";
my $summary_dir = "NULL";
my $summary_filename = "NULL";
if ( $report_summary_path ne "None" ) {
   $report_summary = "TRUE";
   ($summary_filename, $summary_dir) = fileparse($report_summary_path);
}

# parse options: report GOF

my $report_gof = "FALSE";
my $gof_dir = "NULL";
my $gof_filename = "NULL";
if ( $report_gof_path ne "None" ) {
   $report_gof = "TRUE";
   ($gof_filename, $gof_dir) = fileparse($report_gof_path);
}

# parse options: report exploratory analysis

my $report_exploratory = "FALSE";
my $exploratory_dir = "NULL";
my $exploratory_filename = "NULL";
if ( $report_exploratory_path ne "None" ) {
   $report_exploratory = "TRUE";
   ($exploratory_filename, $exploratory_dir) =
      fileparse($report_exploratory_path);
}

# write a R scrip to run

my $tempdir_bin = tempdir();

my $cmd = qq|
   suppressPackageStartupMessages(library(mosaics))
   try( suppressPackageStartupMessages(library(rparallel)), silent=TRUE )

   mosaicsRunAll(
      chipDir="$chip_dir",
      chipFileName="$chip_filename",
      chipFileFormat="$chip_file_format",
      controlDir="$control_dir",
      controlFileName="$control_filename",
      controlFileFormat="$control_file_format",
      binfileDir="$tempdir_bin",
      peakDir="$peak_dir",
      peakFileName="$peak_filename",
      peakFileFormat="$peak_file_format",
      reportSummary=$report_summary,
      summaryDir="$summary_dir",
      summaryFileName="$summary_filename",
      reportExploratory=$report_exploratory,
      exploratoryDir="$exploratory_dir",
      exploratoryFileName="$exploratory_filename",
      reportGOF=$report_gof,
      gofDir="$gof_dir",
      gofFileName="$gof_filename",
      FDR=$fdr_level,
      fragLen=$frag_len,
      binSize=$bin_size,
      capping=$capping,
      analysisType="$analysis_type",
      d=$d,
      signalModel="$signal_model",
      maxgap=$maxgap,
      minsize=$minsize,
      thres=$thres,
      nCore=$n_core )

   q()
   |;

# run R

open( FT, "| R --slave --vanilla >& /dev/null" )
   or die "Couldn't call R!\n";
print FT $cmd, "\n";
close FT or die "Couldn't finish R!\n";

exit;
```