MODEL-BASED APPROACHES FOR THE DETECTION OF BIOLOGICALLY ACTIVE GENOMIC REGIONS FROM NEXT GENERATION SEQUENCING DATA

Naim Rashid

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Biostatistics.

Chapel Hill
2013

Approved by:

Joseph Ibrahim

Wei Sun

Terry Furey

Hongtu Zhu

Pei-Fen Kuan

ABSTRACT

NAIM RASHID: Model-based approaches for the detection of biologically active genomic regions from next generation sequencing data (Under the direction of Joseph Ibrahim and Wei Sun)

Next Generation Sequencing (NGS) technologies are quickly gaining popularity in biomedical research. A popular application of NGS is to detect potential gene regulatory elements that are captured or enriched by certain experimental procedures, for example, Chromatin Immunoprecipitation (ChIP-seq), DNase hypersensitive site mapping (DNase-seq), and Formaldehyde-Assisted Isolation of Regulatory Elements (FAIRE-seq), among others. While ChIP-seq can be use to identify protein-DNA interaction sites, both DNase-seq and FAIRE-seq can be used to identify open chromatin regions, which are more likely to contain elements involved in gene expression regulation. We collectively refer to these types of sequencing data as DAE-seq, where DAE stands for DNA After Enrichment. DAE-seq data can provide important insight into gene regulation, which is crucial to understanding the molecular mechanism of phenotypic outcomes, such as complex diseases.

Here we address several practical issues facing biomedical researchers in the analysis of DAE-seq data through the development of several new and relevant statistical methods. We first introduce a three-component mixture regression model to discover "enriched regions", i.e., the genomic regions with more DAE-seq signal than expected in relation to background regions. We demonstrate its practical utility and accuracy in detecting regions of active regulatory elements across a wide range of commonly used DAE-seq datasets and experimental conditions. We then develop a novel Autoregressive Hidden Markov Model (AR-HMM) to account for often-ignored spatial dependence in

DAE-seq data, and demonstrate that accounting for such dependence leads to increased performance in identifying biologically active genomic regions in both simulated and real datasets. We also introduce an efficient and novel variable selection procedure in the context of Hidden Markov Models when the means of the emission distributions of each state are modelled with covariates. We study the asymptotic properties of the proposed variable selection procedure and apply this approach to simulated and real DAE-seq data. Lastly, we introduce a new method for the joint analysis of total and allele-specific read counts from DAE-seq data and RNA-seq data. In all, we develop several statistical procedures for the analysis of DAE-seq data that are highly relevant to biomedical researchers and have broader applicability to other problems in statistics.

This thesis is dedicated to my loving wife Ashmita, and to my family (old and new), whose support throughout these years I could not have lived without

ACKNOWLEDGMENTS

I am deeply grateful to my advisors Dr. Joseph Ibrahim and Dr. Wei Sun who have mentored me throughout the years. It was truly a blessing to have their support and encouragement throughout the ups and downs of graduate school, both academic and non-academic. I am indebted to them for all that they have taught me and the opportunities that they have given me. I am lucky to have been able to work with them.

I also owe a lot of thanks to Dr. Jason Lieb for his tireless energy and time throughout our projects together. The members of his lab were truly a pleasure to work with, and I learned much about being a good collaborative statistician through working with them. I am very thankful for his constructive comments on my research and for his exacting attention to detail.

I would also like to thank my committee members Dr. Hongtu Zhu and Dr. Pei-Fen Kuan for serving on my committee and for their helpful comments on my research. A special thanks goes to Dr. Terry Furey who agreed to be on my committee in a pinch, and I am very grateful for his time.

To my friends in the department, its been a crazy ride so far and it wouldn't have been the same without you there. Another big thanks goes out to all of the faculty and staff in the department for all their help and hard work. And of course my biggest thanks go out to my wife and family, I couldn't have done this without them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Literature Review

We first review how DAE-seq data is generated and the various ways it is characterized numerically. We then describe current major challenges in the analysis of DAE-seq data not accounted for by existing methods. Lastly, we discuss several classes of existing methods and compare the relative advantages and disadvantages of methods belonging to each class.

1.1 DAE-seq Data: Data Generation

All DAE-seq assays share a common goal of isolating regions of a sample's genome harboring a particular biological activity of interest. The types of samples analyzed vary and may include biopsied tumors, healthy tissue, or cells grown under certain experimental conditions. The accurate determination of these regions allow researchers to relate the biological activity of interest with other genomic features or clinical phenotypes, such as downstream gene expression or individuals' disease statuses.

Chromatin Immunoprecipitation is one example of such assays, where regions of the genome containing sites of protein-DNA interaction are isolated (61). Examples include the isolation of genomic locations where a particular protein of interest has bound to DNA (Transcription Factor Binding Sites) or locations where a bound protein of interest has been chemically modified (such as a histone modification). The FAIRE assay (Figure 1.1) in contrast isolates "open" regions of the genome not bound to large

proteins involved in DNA packaging (nucleosomes) (25). These "open" regions have been suggested to harbor active regulatory elements associated with gene expression regulation and other processes, and are thus of great interest to determine. Regardless of the assay used, the final step of each procedure is to collect "fragments" of the genome that contain the specific biological activity of interest. Such a sample is termed to be "enriched" for genomic DNA pertaining to these regions.

Following fragment isolation, the genomic locations of the collected sequence fragments are determined through NGS. In general, strong local aggregations of fragments provide evidence of the activity of interest occurring in that region of the genome. As shown in Figure 1.2, these regions typically manifest themselves as "peaks", defined as regions of high density sequence fragments relative to the surrounding regions (background). In order to to determine the location of each isolated fragment, x of these fragments are sequenced on a NGS platform such as the Illumina Genome Analyzer, where x is typically a predetermined number based on cost. The first n base pairs of each fragment are sequenced ("reads"), where n is typically 36-72 base pairs in length. In other applications, both ends of a fragment are sequenced (paired-end sequencing), allowing for more accurate determination of the total fragment length. Then, short read aligners are utilized to determine each sequence's most likely location in the genome based on a sequence alignment with a standard reference genome pertaining to the sample ("read mapping"). Some reads can match more than one place in the genome, and in certain cases these reads are deemed as uninformative and are discarded. In other cases, reads matching up to a certain number of genomic locations are kept and others are thrown away (61). Discarding such reads helps to reduce potential signal amplifications in regions of the genome with repetitive elements or commonly found sequences, however at the cost of removing reads that may belong to true enriched regions.

At the end of the sequencing and read mapping process, the data of one sample is reduced to a large set of genomic coordinates corresponding to each sequenced fragment. This coordinate, in its most basic form, is simply the chromosome identifier, read start position on the chromosome (in base pairs), read stop position on the chromosome (in base pairs), and the strand of the DNA that the read aligns to (forward or reverse strand). Information such as the read mapping score and the read sequence may be included as well, among other information. This coordinate forms the most basic and raw form of DAE-seq data. All downstream analyses of this data summarize this information in some way to quantify and determine local aggregations of reads across the genome.

1.2 Statistical Representations of DAE-seq data

We now discuss several common statistical representations of DAE-seq data and their modeling implications, including Single Base Pair Coverage (SBPC), window read counts, and kernel density estimates. Regardless of the summary used, the goal of each these representations is to provide a measure of the read density in each location across the genome.

For example, SBPC is defined as the number of overlapping reads at each base in the genome. In DAE-seq data, typically only the first n base pairs from one end each fragment are sequenced. To mitigate this, typically the length of a read is often extended m base pairs downstream from its start point, where m is the average fragment length of the library used (often between 150 and 300 base pairs). The number of overlapping extended reads at each base pair is then calculated and plotted, as in Figure 1.2. Denote l_i to be the start position of read i, $i = 1, ..., x$ on a particular chromosome, and S_i as the strand the read belongs to then the SBPC at a given position b on the chromosome

is simply

$$SBPC_b = \sum_{j=b-m}^{b} \sum_{i=1}^{x} I[j < l_i \leq b, S_i = +] + \sum_{j=b+1}^{b+m} \sum_{i=1}^{x} I[j < l_i \leq b, S_i = -].$$

Since each read is extended to m base pairs, it is easy to see that the SBPC at base b is correlated with those within the range of $(b - m, b + m)$ bases from it. It is this long-distance serial correlation that gives rise to well-defined "peak" shapes such as in Figure 1.2 and provides a resolution at the single base pair level to define signal. Because of this, the SBPC representation has been typically used to visualize DAE-seq data and is a representation of commonly used to display DAE-seq data in the UCSC genome browser (68). However, this long-range correlation also creates difficulty in deriving a theoretical distribution of the SBPC at a given base. As a result, simulation is typically used to determine an empirical measure for chance that one would observe a particular SBPC value within a fixed region, given that you are drawing from a set of n reads belonging to that region. This approach can be computationally intensive and is not easily amenable to adjusting for the effects of multiple biases that may affect local read density.

Another approach to characterize local aggregations of DAE-seq reads is by utilizing a smoothing algorithm such as a Gaussian Kernel Density Estimator (KDE). Given a certain bandwidth h, the kernel density estimate at base b is given as

$$KDE_b = \sum_{i=1}^{n} \frac{1}{nh} K \left(\frac{\sum_{i=1}^{x} I[l_i = b] - \sum_{i=1}^{x} I[l_i = i]}{h} \right) \tag{1.1}$$

where h is the bandwidth, K is the Gaussian kernel density function, and $\sum_{i=1}^{x} I[l_i = b]$ is the number of read tags that start at base pair b. However for computational expediency, the region of KDE computation can be limited to w bases on each side of the central base b. The resulting estimates when plotted along the genome generate smoothed

shapes whose heights are relative to the underlying read density (Figure 1.3C,D). However, this smoothness is dependent on the choice of h and w, which is not often known *apriori* and may results in oversmoothing of the data (and missed peaks) if chosen incorrectly. In addition, the theoretical distribution of kernel density estimates is not well known, thus it is difficult to model outside of simulation or permutation-based approaches.

The last common way to characterize local read density is by using window read counts (Figure 1.3A,B), defined as the number of reads falling into fixed-length, non-overlapping regions spanning the genome ("windows"). We can define the window read count (WRC) at a window starting at base b of length w as

$$WRC_b = \sum_{i=1}^{x} I[b \leq l_i < b + w].$$

Other methods may use the center of the extended read to determine the window membership of a particular read. While the resolution of window read counts in characterizing local read density is dependent on the chosen length of each window, they are comparably easier to model given that each unit of observation is essentially a count. There is a large amount of statistical literature related to the analysis of count data, which provides a relatively more natural basis for statistical modelling, such as with the Negative Binomial or Poisson distributions (Figure 1.4). However, given the fixed, non-overlapping nature of these windows, it is possible that the boundaries of these windows may bisect peak regions and not fully contain a true peak.

1.3 DAE-seq Modelling Challenges in the literature

Here we review several issues discussed in the literature that are problematic in the interpretation of DAE-seq data and may complicate peak calling with existing methods.

1.3.1 Biases Affecting DAE-seq Data

Several biases have been described in the literature to artificially inflate or deflate the number of DAE-seq reads in a given region of the genome. These biases may stem from the assay used to generate biological sample itself or originate from certain technical aspects the DAE-seq sequencing and read mapping process.

Mappability is one such bias, a concept that was developed in the peak-calling algorithm Peakseq (71) and quantifies the ability of a short read aligner to map a read to a particular location in the genome. For a given base pair in the genome, mappability is defined as the number of times that the downstream sequence of fixed length starting at that base occurs throughout the genome. This downstream sequence length is equal to the length of the reads utilized in ones experiment. If this downstream sequence is unique in the genome, then a read mapper can uniquely place a read matching this sequence to this location in the genome. However, if this downstream sequence is non-unique, a read mapper may randomly place a read matching to this sequence to one of other similar sequences located in other regions of the genome. Typically, reads matching more than a certain number of locations in the genome are removed from ones data, where these reads are assumed to be uninformative.

It is shown in (71) that there are local differences in mappability throughout the human genome. As a result, removing reads from one's experiment matching more than one or more places in the genome artificially reduces the read density in low-mappability regions relative to those found in higher mappability regions. Because of this read density bias, it is difficult to compare local read density across different regions of the genome. Therefore, it is important to account for mappability to accurately determine which regions of the genome are enriched, and the effect of mappability may vary depending on the read filtering threshold used.

Depending on the DAE-seq experiment, the filtering threshold used to remove reads

mapping to low mappability regions varies. For example, in ChIP-seq experiments for Transcription Factor Binding Sites (TFBS), usually reads matching to only one place in the genome are retained. In this situation the sequences these reads align to are unique and have a mappability score of 1. In FAIRE-seq experiments reads matching up to 4 locations in the genome are typically retained, resulting in fewer reads removed from the data set.

The percentage of G and C nucleotides in a particular region of the genome has also been shown to artificially affect local DAE-seq read density. In early ChIP-seq studies it has been shown that an increasing G/C nucleotide composition in a particular region was positively related with the read count in that region, however the magnitude of this effect was found to vary between experiments (15). In DAE-seq data it had been postulated that G/C content bias is related to PCR amplification bias during the preparation of the sample (64, 32) and G/C-related sequencing errors (15). Previous technologies such as microarrays have similarly observed substantial bias in signal related to the G/C content of probes, where the G/C-effect had to be corrected prior to downstream analysis.

Copy Number Variation (CNV) or Aberration is another DAE-seq read density bias, where extra copies of certain regions of a sample's genome can significantly inflate the read density in that particular region (27). CNVs may either originate from from chromosomal duplications or from regions that have strong homologies with a region on another chromosome, and are best identified by broad amplifications in signal. Copy number aberrations are most pronounced and common in tumor samples, where chromosomal abnormalities are common (27). From SBPC maps these regions are often easily identifiable as they are many times higher in counts than their surroundings, with the degree of amplification may changing within different segments of the CNV. Ideally, a method should adjust for the local amplification in background due to CNVs and, if

possible, call peaks within these regions with respect to this amplified background.

1.3.2 Diversity in Genomic Distribution of DAE-seq data and Signal-to-noise Ratios

In the context of DAE-seq data, the term "signal-to-noise ratio" can be defined as the relative level of signal typically found in genomic regions enriched for DAE-seq reads versus levels in those typically found in regions not enriched for DAE-seq reads. Therefore, a high signal-to-noise ratio would correspond to a dataset where enrichment regions are easily distinguishable over signal in surrounding background regions. A wide diversity in signal-to-noise ratios and lengths of enriched regions can be seen among DAE-seq datasets (Figure 1.2). These differences in enrichment patterns are a reflection of the type of biological activity that each DAE-seq experiment seeks to capture (61). For example, in Figure 1.2 the transcription factor CTCF tends to bind tightly to DNA only at very specific locations of the genome (typically where a CTCF binding motif is located), and therefore its signal is characteristically high signal-to-noise with very little background. FAIRE-seq data however is characteristically low signal-to-noise and has a very high amount of background. This is a reflection of the transient and ubiquitous nature of open chromatin regions, where enriched regions reflect open chromatin that is open slightly more stably than others. Histone modification data can vary widely in terms of length of enriched regions (58). Histone H3 Trimethylation data (H3K36me3) for example have regions of enrichment that tend to be distributed broadly across gene bodies (Figure 1.2). This diversity in signal poses great difficulty to existing methods, as many are tailored for a specific type of DAE-seq data. The modelling assumptions suitable for one type of DAE-seq data may not work well in another (43, 85, 33).

1.3.3 Role of Input Control

An Input control sample is genomic DNA that has been stripped of all proteins and fragmented either through physical sonication or enzymatic digestion ("naked DNA"). As a result, these fragments may come from any part of the genome and are not localized to regions harboring a type of biological activity, serving as a negative control to a DAE-seq experiment. Local read density from such a sample can be interpreted as the local read density that one may expect in the absence of enrichment (83).

It is generally thought that input control is able to capture the biases described in the previous section, where variation in input control signal due to such factors can similarly explain variability in signal in background regions of DAE-seq data (95). Therefore, to determine whether the DAE-seq read density in a particular regions is "significant", many methods compare this read density with the read density in a matching region of a sequence input control sample. That is, if the DAE-seq signal in a particular region of the genome is much greater than the read density found in a matching region of the input control dataset, the region is judged to have more signal than one would expect in the absence of enrichment. The way in which this comparison is done varies between methods and the type of DAE-seq representation chosen, which we will detail in the next section. However, input control is not always available since it must be sequenced separately than the DAE-seq sample and provides additional cost to researchers to sequence. Furthermore, it is unknown whether biases affect input control similarly in a similar manner in which they affect background regions of DAE-seq data.

1.3.4 Excess Zero Signal Regions

The ability to identify enrichment regions over background is generally related to the number of reads sequenced in their sample (95), referred to as "sequencing depth". Increasing sequencing depth tends to increase the signal-to-noise ratio in one's data

relative to lower sequencing levels, and leads to the discovery of more enriched regions in one's data (95). In earlier studies, the cost to deeply sequence one's sample was financially prohibitive, therefore fewer sequencing reads were purchased by current standards resulting in an over-abundance of zero read regions across the genome. When utilizing the window read count representation, many windows will in turn have a window read count of zero at a frequency much more than expected by the Poisson or Negative Binomial distributions. Not accounting for these excess zeros can lead to overdispersion and reduced model fit, and also tends to bias estimated parameters (44, 31) and in turn affect peak calling accuracy. These zero read regions in low-read-depth samples may belong to true enrichment regions as the read depth increases. Low mappability may also further exacerbate the frequency of zero read regions in certain parts of the genome.

1.4 Current Approaches to DAE-seq Analysis

Many algorithms have been proposed to determine enrichment regions in DAE-seq data. However, none are able to account for multiple factors that may bias DAE-seq data or are designed to handle the wide range of enrichment patterns found across DAE-seq datasets. We review the relative merits of several categories of methods and then summarize their general limitations. In general, methods not utilizing window read counts are limited to permutation/simulation approaches to test whether a region is significantly enriched for DAE-seq reads, which limit their generalizability to handle the issues described in the previous section.

1.4.1 Algorithms Utilizing SBPC

Several computational and statistical algorithms have been developed to detect enriched regions in DAE-seq data utilizing the SBPC representation, include Peakseq

(71), FindPeaks (21), the method used in (58), and GLITR (81). Enriched regions are typically defined as those with SBPC values above a particular SBPC height threshold corresponding to some empirically derived criteria of significance. These significance criteria are determined through simulations, where typically the DAE-seq read positions from a sample are randomly permuted across the genome or subregion of the genome, and the SBPC is computed at each base for each permutation. This permutation is performed k times, where k is large and to get an empirical distribution of of SBCP under a random assortment of reads.

For example, in (58) SBPC values were calculated for each of histone modification ChIP-seq datasets utilized in their experiment, and enrichment regions were determined by selecting regions that exceeded a chosen SBPC height cutoff. To determine the SBPC height threshold used to determine enrichment regions, a permutation approach was employed. If we let $(l_1, ..., l_N)$ be the set of randomly permuted start positions of each of the N sequenced reads in a sample then for the b'th base pair in the kth permutation the SBPC is calculated as defined earlier. For each of the k permutations, the set of SBPCs $(S_{k1}, ..., S_{kB})$ are recorded. Using these results we can determine an empirical SBPC distribution across all k and b, where the frequency of each SBPC value is tallied and can be used to compute an empirical p-value associated with observing a particular SBPC value by random chance. In this method the SBPC threshold corresponding to an empirical p-value of 10^{-5} was chosen to determine significance.

Peakseq (71) is another method that utilizes the SBPC representation. In the first of its processing steps, the number of reads falling into non-overlapping large windows are counted (default 1 Megabase). Then for each window, the window length is then scaled in proportion to the fraction of "mappable" bases contained in that window. For example, if 80 percent of the bases in a window are mappable, then the length of the region used in the simulation is reduced from 1 Megabase to 0.8 Megabase.

Then, similar to (58) above, the start positions of each read in the original window are randomly permuted within this smaller region, and the read overlap profiles for each permuted set are calculated. For a given SBPC threshold value, the number of regions exceeding this height threshold in each of the permuted datasets within the window are counted and is used to determine an empirical False Discovery Rate corresponding to this threshold. In this manner, the SBPC height thresholds are adjusted for low mappability by permuting the original set of reads in a smaller space prior to the FDR calculation.

While SBPC data can graphically represent local enrichment with high resolution, simulation based methods have been mostly used for the estimation of an empirical background distribution. In addition, this approach tacitly assumes that the null distribution of reads is random, and is also computationally expensive. These simulations assume that the distribution of reads in non-enriched (null-signal) regions is random, an assumption that does not hold when biases due to G/C content, mappability, and copy number variation are present. Incorporating factors other than mappability into a simulation framework is difficult, and thus the generalizability of such methods is limited. Methods utilizing SBPC tend to one of those earliest introduced in the literature.

1.4.2 Algorithms Utilizing KDEs

As described earlier, methods using KDEs such as F-seq (7), QUEST (82), and CSDeconv (53) all utilize a Gaussian kernel similar to (1.1) with a predetermined bandwidth and window length. For example, after partitioning the genome into windows of fixed length, F-seq calculates the KDE at the center of each window for a given bandwidth. To determine a KDE cutoff for declaring enriched regions. After 1000 of these permutations, an empirical null distribution is calculated and the KDE corresponding to 4 standard deviations above from the mean sample KDE is utilized as the default

threshold.

In QUEST, an "unnormalized "version of the Gaussian kernel in (1.1) is utilized, where

$$KDE_{b,QUEST} = \sum_{i=b-3h}^{b+3h} K\left(\frac{\sum_{i=1}^{x} I[l_i = b] - \sum_{i=1}^{x} I[l_i = i]}{h}\right) \times C_b,$$

where again x is the number of sequenced reads in one's sample, h is the chosen bandwidth, and C_b is the total number of reads falling into the region $(b-3h, b+3h)$. These values were calculated separately for the forward and reverse strands. After correcting for shifts in the read density profiles in each strand, the estimated from each strand are summed together for a total score at each position in the genome. Peaks were then called using a post-hoc procedure where ultimately peak significance was determined via ratio of signal found in the sample versus those found in input. To determine the best ratio cutoff for significance, a simulation-based procedure was utilized. The input control sample read is first randomly split into a "background" sample and a "psuedo-ChIP" sample. The number of peaks called in the ChIP relative to those in the pseudo-ChIP datasets using the "background" sample to calculate a false discovery rate, which they define as the ratio between the number of peaks called in the pseudo-ChIP dataset and the ChIP dataset. The threshold chosen is one that corresponds to the threshold that produces only one peak in the pseudo-ChIP sample relative to background.

However, the statistical properties of working directly with KDE smoothed data are not well known, which is further complicated when adjusting for multiple biases affecting DAE-seq data. This drawback is reflected in the fact that many methods utilizing KDE's use permutation or cross-validation based methods to determine significance. KDE's are also are susceptible to being influenced highly by the surrounding signal, which may cause problems with peak calling, especially in noisy data sets.

1.4.3 Algorithms Utilizing Window Read Counts

Methods utilizing this representation of the data can be split into two general categories: fixed-window and sliding window methods. Fixed window methods, such as CisGenome (36), Bayespeak (74), and HPeak (63) typically assume the window read counts are distributed by some parametric count distribution such as the Poisson or Negative Binomial distributions, the latter being able to account for overdispersion in counts where the variance is not necessarily equal to the mean.

For example in Cisgenome,the genome is divided into non-overlapping windows with length w (typically 100 base pairs). The read count for each window is tabulated. In background regions it is assumed that the window read count

$$Y \sim Po(\lambda)$$

where $\lambda \sim Gamma(\alpha, \beta)$. This implies that marginally $Y \sim NB(\mu, \phi)$ in background regions. To estimate these parameters, a truncated Negative Binomial distribution is fitted to windows with $Y_i \leq 2$, $i = 1, ..., B$, where B denotes the total number of windows spanning the genome. The fitting method assumes that most windows with small read counts represent noise. The assumption usually holds true with sufficient depth of sequencing, however may not hold for lowly sequenced datasets or those with broader mode of enrichment such as with ChIP-seq of histone modifications. When an input control sample is also available, then for a given window i the number of reads in the ChIP sample Y_{i1}, the number of reads in the input control sample Y_{i2} and the total read number $Y_{i1} + Y_{i2}$ are counted. In background regions, it is assumed that

$$Y_{i1}|Y_{i1} + Y_{i2} \sim Bin(Y_{i1} + Y_{i2}, p_0)$$

distribution, where p_0 is estimated based on windows with small total counts and is

used to estimate the FDR associated with each level of $Y_{i1} + Y_{i2}$ and $Y_{i1}|Y_{i1} + Y_{i2}$. Given the fitted background distribution for each, a p-value can be calculated for the observed read counts belonging to this background model, and those p-values meeting the FDR-level cutoff are selected as enriched.

In BayesPeak and HPeak, an HMM approach is used to model window read counts. In BayesPeak, the genome is divided into non-overlapping windows (default 100 bp) and the read counts falling into each window from the forward and reverse strands are tabulated. It is assumed that the underlying state at window i, denoted by S_i is such that $S_i = 0, 1$, where 0 corresponds to a non-enriched region and 1 corresponding to an enriched region. It is also assumed that the emission probability of observing window read count

$$Y_i|Z_i = 0 \sim Po(\lambda_0 \gamma^{w_t})$$

and

$$Y_i|Z_i = 1, 2, 3 \sim Po(\lambda_0 + \lambda_1)w^t),$$

where $Z_0 = (S_t = 0, S_{t+1} = 0)$, $Z_1 = (S_t = 0, S_{t+1} = 1)$, $Z_2 = (S_t = 1, S_{t+1} = 0)$, $Z_3 = (S_t = 1, S_{t+1} = 1)$ and $\lambda_0 \sim Gamma(\alpha_0, \beta_0)$ and $\lambda_1 \sim Gamma(\alpha_1, \beta_1)$. The model is fit using an approach similar to the EM algorithm where the states are simulated from the joint posterior mass function of all the states given $\hat{\phi}^{(s)}$, the current estimate of the model parameters at step s. Given the complete data set (observed read counts and simulated states), each parameter is updated conditionally on the values of the remaining parameters using Gibbs updates. For most of them the form of the likelihood and the conjugate priors lead to closed-form posterior distributions. For all others, they use Metropolis-Hastings updates with symmetric (Normal) proposals centered at their accepted values.

The advantage of such fixed window approaches is that parametric distributions

for count data can be utilized. However, such methods cannot account for overdispersion due to excess zero regions in lowly sequenced datasets. Furthermore, one general drawback is that while smaller window sizes may provide greater resolution of local read density, it also increases the computational burden on each algorithm as more observations are generated per dataset. Also, smaller window sizes increases the local correlation between adjacent windows, violating assumptions of statistical independence. As we will see later, these HMM still cannot account for all of the dependence seen in the data and currently do not adjust for biases in signal.

Sliding window approaches are also very popular, for example in the popular tool MACS (95). Other methods using this approach include SPP (39), Sole-Search (6), SiSSRs (37), and USeq (59). This is a variation on the non-overlapping window approach where only a single window is moved across genome in short increments, where at each position the window read counts are tabulated and are either compared to the read counts in the surrounding region or those in a matching input control sample. In MACS, significance of an enriched region is determined by computing the Poisson probability under the null assumption that the the counts falling into a window from the experimental sample and those from the input control sample are similar, where this probability is given as

$$1 - \sum_{j=0}^{WRC_{i,sample}} \frac{e^{-\lambda_{i,input}} \lambda_{i,input}^{j}}{j!},$$

where $\lambda_{i,input} = \max(\lambda_{BG}, \lambda_{1K}, \lambda_{5K}, \lambda_{10K})$, each defined as the number of reads falling into the entire genome, 1000 bp window, 5000 bp window, and a 10000 bp window centered at window i divided by the end of each region respectively. This is done to be robust to situations where no input control reads are available in the local vicinity of window i. Because the input control is typically has less number of reads than

the experimental sample, λ_i is scaled by a constant $c = N_{sample}/N_{input}$ to account for this disparity. Windows that meet the default p-value of 10^{-5} are determined to be significant, and adjacent significant windows are merged into contiguous regions.

1.4.4 Summary

Currently there are many algorithms available for the identification of genomic regions enriched by a given experiment. Although each method may be well suited for the analysis of a particular intended data type, the underlying assumptions are not always suitable for the multitude of possible enrichment patterns found in DAE-seq datasets. For example, the majority of existing algorithms perform optimally for the identification of transcription factor binding sites (TFBS) from ChIP-seq data (85, 43). However, as the proportion of the genome that is enriched increases and/or the signal-to-noise ratio decreases compared with TFBS data (7, 89, 33, 87) the performance of many existing tools declines (85, 33, 49, 43, 29). Unfortunately, researchers interested in analysis of several types of data for a given experiment must often combine results from different algorithms. A statistical approach capable of robust detection of enrichment across a multitude of enrichment patterns, with performance comparable to the existing set of algorithms specific to each data type, would have high utility.

In addition, most of the methods introduced have not directly addressed the need to adjust for multiple local effects that may artificially increase or decrease local read density. Those that have local biases (71) been the only so far to adjust for mappability, however their way to deal with it has been to scale the expected counts falling into that window proportional to the mappability of that window. Given that there may be other biases that may potentially influence window read counts, attempting to adjust for any other set of biases with mappability simultaneously through scaling is would require some sort of systematic way to determine the relationship with counts and their

relative influence before knowing the direction and magnitude of adjustment needed. MACS address local fluctuation in background by looking at mean counts in several fixed regions around the area of interest, but the effects of these covariates were never explored directly. Zero Inflation is also a problem when modelling window read count data, none so far take it into account.

Furthermore, methods that are dependent on scoring the significance of peaks in the sample by directly comparing counts to those found in control (after compensating for sequencing depth disparity through scaling factor) are especially susceptible to problems associated with insufficiently sequenced controls. KDE-based methods to smooth the data also are susceptible to problems of choosing the appropriate bandwidth, and nearby points are prey to either inflating or deflating estimates based on their relative enrichments. Callers that use genome-wide cutoffs cannot adjust for local effects, and thus suffer from high FDR or low sensitivities in variable data. Lastly, few or none of these methods account for local correlation between observed window counts, which violate assumptions of statistical independence, complicating FDR and parameter estimation.

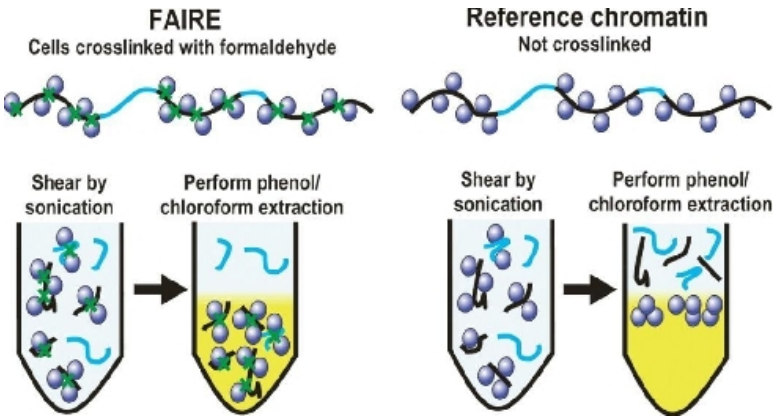

Figure 1.1: Formaldehyde Assisted Isolation of Regulatory Elements (FAIRE) procedure. Regions of the genome not packaged by nucleosomes are isolated by a chemical gradient. A reference (control) sample is sometimes obtained to provide an estimate for null background signal. (right)

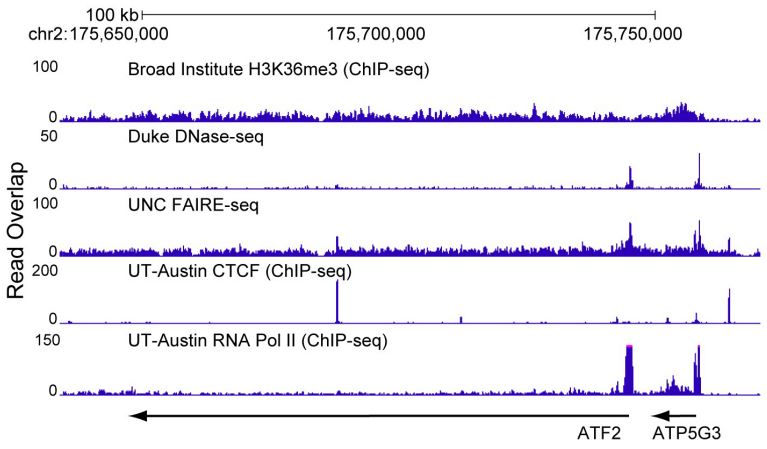

Figure 1.2: Example of SBPC representation for several DAE-seq data types (rows) in a given region of Human Chromosome 22.

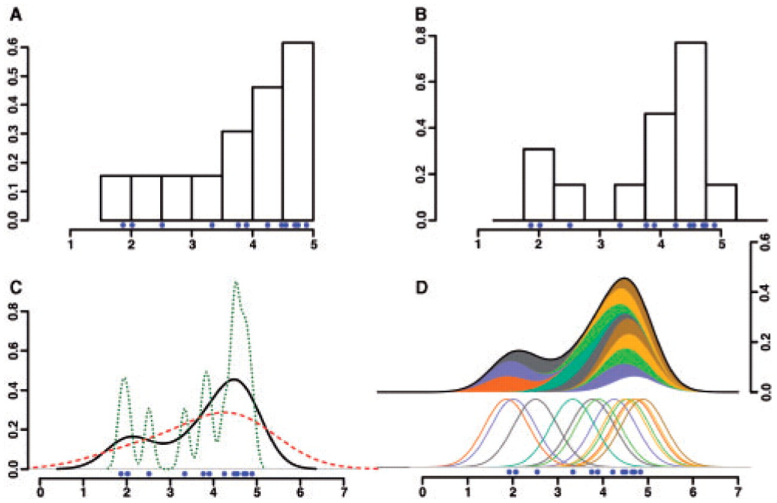

Figure 1.3: Examples of Window Read Count (A,B) and KDE (C,D) representations of DAE-seq data in a small region of the genome

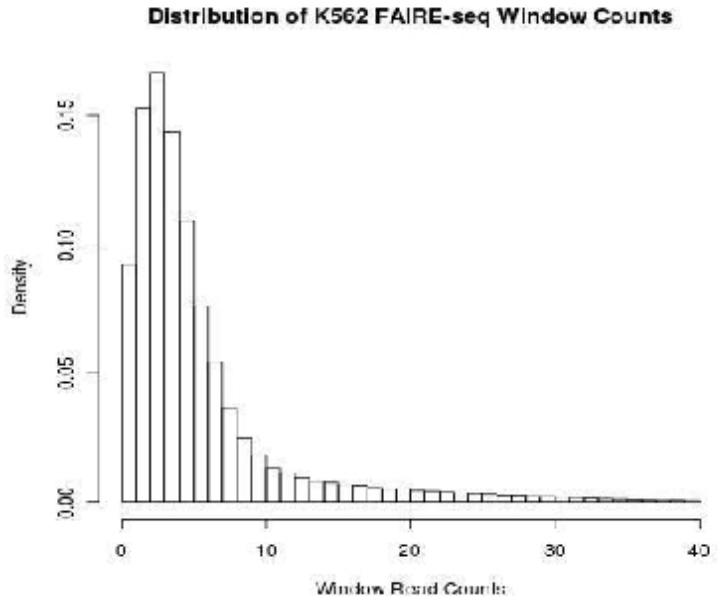

Figure 1.4: Distribution of window read counts from a K562 FAIRE-seq dataset. Distributional assumptions including the Poisson and Negative Binomail Distribution are appropriate for this type of data

Chapter 2

Zero Inflated Negative Binomial Algorithm

2.1 Introduction

To address these issues, we introduce a flexible statistical framework called ZINBA (Zero-Inflated Negative Binomial Algorithm) that identifies genomic regions enriched for sequenced reads across a wide spectrum of signal patterns and experimental conditions (Figure 2.1A). ZINBA implements a mixture regression approach, which probabilistically classifies genomic regions into three general components: background, enrichment, and an artificial zero count. The regression framework allows each of the components to be modelled separately using a set of covariates, which leads to better characterization of each component and subsequent classification outcomes. In addition, the mixture-modelling approach affords ZINBA the flexibility to determine the set of genomic regions comprising background without relying on any prior assumptions of the proportion of the genome that is enriched. Following classification, neighboring regions classified as enriched are merged and boundaries of punctate signal within enriched regions are determined, allowing the isolation of both broad and narrow elements.

ZINBA performs three steps: data preprocessing, determination of significantly enriched regions, and an optional boundary refinement for more narrow sites (Figure 2.1B). The first step involves tabulating the number of reads falling into contiguous

non-overlapping windows (default 250 bp) tiled across each chromosome and scoring corresponding covariate information. Covariates can consist of any quantity that may co-vary with signal in a given region, including, for example, G/C content, a smoothed average of local background, read counts for an input control sample, or the proportion of mappable (71) bases, which we define as the mappability score. Optionally, additional sets of contiguous windows with offset starting positions can be tabulated for increased resolution. Each set of offset windows is analyzed independently in the next step.

In the second step, a novel mixture regression model is used to probabilistically classify each window into one of three components: background, enrichment, or zero-inflated. In this context, and throughout this document, the term 'enrichment' will refer to genomic DNA sequences that were captured specifically as the result of the biological experiment under consideration. The term 'background' includes genomic DNA sequences that appear due to experimental noise, noise that arises in the sequencing process, or noise that arises in the computational processing of the data. The term 'zero-inflated' refers to those genomic locations at which we might expect coverage by a sequencing read derived from either the background or enrichment signal components, but that are not represented in the real data. Zero-inflation typically occurs due to a lack of sequencing depth and is common in many NGS datasets. Regions containing higher proportions of non-mappable bases are also more likely to be zero-inflated, as it is more difficult to assign reads to these regions during the mapping process. We describe this procedure on more detail below.

Finite mixtures of regression models (FMRs) have been utilized in an array of fields such as economics, public health, and genetics (56). Central to the application of FMRs is the desire to simultaneously classify and profile observations into clusters through component-specific covariates. This is advantageous in situations where signal within components is heterogeneous and is known *apriori* to be associated with multiple factors

in potentially component-specific ways.

2.2 Model

Let us assume that $Y = (Y_1, \ldots, \ldots Y_n)$ is a vector of n consecutive window read counts from a particular chromosome. We assume Y_i follows a three component mixture distribution consisting of a point mass at zero (corresponding to zero-inflated regions of signal), a negative binomially distributed component (corresponding to background windows), and another negative binomially distributed component (corresponding to enrichment windows). This is an extension of the zero-inflated negative binomial distribution, where we add an additional component to account for stronger signal in enriched windows relative to background windows. With this mixture assumption, we define the following mixture distribution for Y_i:

$$p(Y_i = y_i \mid \mu_i, \phi, \pi_i) = \begin{cases} \pi_{i0} + (1-\pi_{i0})\pi_1 \left(\frac{\phi_1}{\mu_{i1}+\phi_1}\right)^{\phi_1} + (1-\pi_{i0})\pi_2 \left(\frac{\phi_2}{\mu_{i2}+\phi_2}\right)^{\phi_2} & y_i = 0 \\ (1-\pi_{i0})\pi_1 \frac{\Gamma(y_i+\phi_1)}{y_i!\Gamma(\phi_1)} \left(\frac{\phi_1}{\mu_{i1}+\phi_1}\right)^{\phi_1} \left(\frac{\mu_{i1}}{\mu_{i1}+\phi_1}\right)^{y_i} & \\ +(1-\pi_{i0})\pi_2 \frac{\Gamma(y_i+\phi_2)}{y_i!\Gamma(\phi_2)} \left(\frac{\phi_2}{\mu_{i2}+\phi_2}\right)^{\phi_2} \left(\frac{\mu_{i2}}{\mu_2+\phi_2}\right)^{y_i} & y_i > 0 \end{cases}$$

where $\mu_i = (\mu_{i1}, \mu_{i2})$ corresponds to the means of the negative binomially distributed background and enrichment components respectively for window i, and $\phi = (\phi_1, \phi_2)$ are the corresponding dispersion parameters for each component. Also, $\pi_i = (\pi_{i0}, \pi_1, \pi_2)$ are the corresponding mixture proportions for the zero-inflated, background and enrichment components, respectively. π_{i0} corresponds to the prior probability that window i is zero-inflated, where $\pi_0 = (\pi_{10}, \ldots, \pi_{n0})$ is the $n \times 1$ vector of zero inflated prior probabilities for each window. We set π_1 and π_2 as scalars where $\pi_1 + \pi_2 = 1$. In the next section, we set up an EM algorithm to estimate the maximum likelihood estimates of the model parameters and obtain posterior probabilities of component membership

for each window given these parameter estimates.

Because of the role of biases such as G/C content and mappability in modelling DAE-seq data, we allow the means of each component distribution to be modelled by sets of covariates. The observed data for a chromosome is given as (Y, X_0, X_1, X_2) where

- $Y = n \times 1$ vector of observed window read counts counts
- $X_1 = n \times (p + 1)$ covariate matrix pertaining to the background component
- $X_2 = n \times (q + 1)$ covariate matrix pertaining to the enrichment component
- $X_0 = n \times (r + 1)$ covariate matrix pertaining to the zero-inflation component

Here, p, q, and r are the number of covariates for the background, enrichment, and zero-inflation components, respectively, and n is the number of windows in that chromosome. For each component we assume an intercept, represented by a column of ones in the first column of each covariate matrix. In the ZINBA data preprocessing step we obtain Y_i and corresponding values of several factors, including window G/C content, proportion of mappable bases, read counts from a matching input control (if included) and a local background estimate. We use these factors to construct each of the covariate matrices above, including main effects of each factor and interaction terms between them if desired (pair-wise and three-way).

The mean values of the negative binomially distributed background and enrichment components are modelled as a function of a set of covariates through the log link, such that $\log(\mu_1) = X_1\beta_1$ and $\log(\mu_2) = X_2\beta_2$, μ_1 and μ_2 are $n \times 1$ vectors and X_0, X_1, and X_2 are the covariate matrices pertaining to each parameter. $\beta_1 = (\beta_{01}, \beta_{11}, \ldots, \beta_{p1})$ and $\beta_2 = (\beta_{02}, \beta_{12}, \ldots, \beta_{q2})$ are vector of regression parameters corresponding to the background and enrichment components, respectively. The parameter β_{01} and β_{02} represent the intercept parameter for each component, interpreted as the average level

of signal in each component when all component-specific covariates are equal to zero. We also model the vector of prior probabilities of zero-inflation π_0 as a function of a set of covariates through the logit link $\pi_0 = \frac{e^{X_0\gamma}}{1+e^{X_0\gamma}}$, where $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_r)$ is the vector of regression parameters corresponding to the zero-inflated component. Note that we do not directly model the probabilities of enrichment and background for the sake of robustness of the algorithm, although technically it is straight forward to do so.

2.2.1 Complete Data Log Likelihood

The missing data in this framework is the true component membership of each window. Let z_{i1} be the indicator function for when window i truly belongs to background, z_{i2} the indicator function for when window i truly belongs to enrichment, and $z_{i0} = 1 - z_{i1} - z_{i2}$ be the indicator function for when window i truly belongs to the zero-inflated component. We consider $z_i = (z_{i0}, z_{i1}, z_{i2})$ to be a draw from the Multinomial distribution such that $z_i \sim \text{Multinomial}(1, (\pi_{i0}, (1-\pi_{i0})\pi_1, (1-\pi_{i0})\pi_2))$. Then the complete data log likelihood is given as

$$\begin{aligned}
L_c(\Psi|X, y, z) &= \sum_{i=1}^{n} z_{i0}\log(\pi_{i0})I[y_i = 0] + (1 - z_{i0})\log(1 - \pi_{i0}) \\
&+ z_{i1}\left[\log(\pi_1) + \log\left(\frac{\Gamma(y+\phi_1)}{y_i!\Gamma(\phi_1)}\left(\frac{\phi_1}{\mu_{i1}+\phi_1}\right)^{\phi_1}\left(\frac{\mu_{i1}}{\mu_{i1}+\phi_1}\right)^{y_i}\right)\right] \\
&+ z_{i2}\left[\log(\pi_2) + \log\left(\frac{\Gamma(y+\phi_2)}{y_i!\Gamma(\phi_2)}\left(\frac{\phi_2}{\mu_{i2}+\phi_2}\right)^{\phi_2}\left(\frac{\mu_{i2}}{\mu_{i2}+\phi_2}\right)^{y_i}\right)\right] \\
&= L_c(\gamma \mid X_0, y, z) + L_c(\beta_1, \phi_1 \mid X_1, y, z) + L_c(\beta_2, \phi_2 \mid X_2, y, z)
\end{aligned}$$

where $\Psi = (\gamma, \beta_1, \beta_2, \phi_1, \phi_2, \pi_0, \pi_1, \pi_2)$, $\pi_0 = (\pi_{10}, ..., \pi_n) = \frac{e^{X_0\gamma}}{1+e^{X_0\gamma}}$, $X = (X_0, X_1, X_2)$, $\mu_1 = (\mu_{11}, ..., \mu_{n1}) = \exp(X_1\beta_1)$, and $\mu_2 = (\mu_{12}, ..., \mu_{n2}) = \exp(X_2\beta_2)$ are defined as before.

It is easy to see that we can separate out the complete data log likelihood with respect to each component and their set of parameters. Thus, we can seek maximize each likelihood separately in the M-step (44, 56)).

2.2.2 E-Step

The Q-function for the E-step at iteration k is given as the expectation of the complete likelihood with respect to z_i, given the estimates of the model parameters from the M-step.

$$\begin{aligned} Q(\Psi \mid \Psi^{(s)}) &= \sum_{i=1}^{n} \tau_{i0}^{(s)}(y_i, X_i, \Psi^{(s)}) \log\left(\pi_{i0}^{(s)}\right) + \log(1-\pi_{i0}) \\ &+ \tau_{i1}^{(s)}(y_i, X_i, \Psi^{(s)}) \left[\log(\pi_1) + \log(f_1(y_i \mid \mu_{i1}^{(s)}, \phi_1^{(s)}))\right] \\ &+ \tau_{i2}^{(s)}(y_i, X_i, \Psi^{(s)}) \left[\log(\pi_2) + \log(f_2(y_i \mid \mu_{i2}^{(s)}, \phi_2^{(s)}))\right] \end{aligned}$$

and

$$\begin{aligned} E[z_{i0}|y_i, X_i, \Psi^{(s)}] &= \tau_{i0}^{(s)}(y_i, X_i, \Psi^{(s)}) = \frac{\pi_{i0}^{(s)} f_0\left(y_i\right)}{T_i}, \\ E[z_{ik}|y_i, X_i, \Psi^{(s)}] &= \tau_{ik}^{(s)}(y_i, X_i, \Psi^{(s)}) = \frac{\left(1-\pi_{i0}^{(s)}\right)\pi_k^{(s)} f_k\left(y_i \mid \mu_{ik}^{(s)}, \phi_k^{(s)}\right)}{T_i}, \end{aligned}$$

where $k = 1, 2$, $\pi_{i0}^{(s)} = \frac{e^{X_{i0}\gamma^{(s)}}}{1+e^{X_{i0}\gamma^{(s)}}}$, $\mu_{ik}^{(s)} = \exp\left(X_{ik}\beta_k^{(s)}\right)$,and

$$T_i = \pi_{i0}^{(s)} f_0(y_i) + (1-\pi_{i0})\left[\pi_1^{(s)} f_1\left(y_i \mid \mu_{i1}^{(s)}, \phi_1^{(s)}\right) + \pi_2^{(s)} f_2\left(y_i \mid \mu_{i2}^{(s)}, \phi_2^{(s)}\right)\right].$$

Here,

$$f_0(y_i) = \begin{cases} 1 & y_i = 0 \\ 0 & y_i > 0 \end{cases}$$

pertains to whether the observed window read count y_i is zero and

$$\begin{aligned} f_k(y_i \mid \mu_{ik}^{(s)}, \phi_k^{(s)}) &= \frac{\Gamma(y+\phi_k^{(s)})}{y_i!\Gamma(\phi_k^{(s)})}\left(\frac{\phi_k^{(s)}}{\mu_k^{(s)}+\phi_k^{(s)}}\right)^{\phi_k^{(s)}}\left(\frac{\mu_k^{(s)}}{\mu_k^{(s)}+\phi_k^{(s)}}\right)^{y_i} \\ &= \frac{\Gamma(y+\phi_k^{(s)})}{y_i!\Gamma(\phi_k^{(s)})}\left(\frac{\phi_k^{(s)}}{e^{X_k\beta_k^{(s)}}+\phi_k^{(s)}}\right)^{\phi_k^{(s)}}\left(\frac{e^{X_k\beta_k^{(s)}}}{e^{X_k\beta_k^{(s)}}+\phi_k^{(s)}}\right)^{y_i}, \end{aligned}$$

The posterior probabilities of component membership are adjusted for each window's set of covariates, their estimated effects in each component, and the estimated baseline effect of each component.

2.2.3 M-Step

Because the Q function with respect to each set of regression parameters is distinct, we can maximize each separately using weighted Generalized Linear Models. We obtain the parameter model estimates in the manner as follows:

For $\gamma^{(s+1)}$: maximize

$$L_c(\gamma \mid X_0, y, z) = \sum_{i=1}^{n} \tau_{i0}^{(s)} I[y_i = 0] X_{i0}\gamma - \sum_{i=1}^{n} \log\left(1 + e^{X_{i0}\gamma}\right)$$

Now, suppose n_0 of the y_i's are 0 such that $y_{i1}, ..., y_{in_0}$ are zero and $y_{i(n_0+1)}, ..., y_{in}$ are greater than zero. Then, specify a matrix $W^{(s)}$ with diagonal $w^{(s)} = (w_{n_0}^{(s)}, w_n^{(s)}) = (\tau_{i0}^{(s)}, ..., \tau_{n_0 0}^{(s)}, 1 - \tau_{(n_0+1)0}^{(s)}, ..., 1 - \tau_{n0}^{(s)})$, where $\tau_{i0}^{(s)}$ is the posterior probability of the ith observation belonging to the zero inflated component at iteration k. Then $\gamma^{(s+1)}$ can be calculated by weighted logistic regression for the response y for $y = 0$ vs. $y > 0$, where weight matrix $W^{(s)}$ reduces the maximization of the zero-inflated likelihood to weighted logistic regression (Lambert 1992).

For $\beta_k^{(s+1)}$: maximize

$$L_c(\beta_k, \phi_k \mid X_k, y, z) = \sum_{i=1}^{n} \tau_{i1}^{(s)} \left[\log(\pi_k) + \log \left(\frac{\Gamma(y+\phi_k)}{y!\Gamma(\phi_k)} \left(\frac{\phi_k}{e^{X_{ik}\beta_k} + \phi_k} \right)^{\phi_k} \left(\frac{e^{X_{ik}\beta_k}}{e^{X_{ik}\beta_k} + \phi_k} \right)^{y} \right) \right].$$

Then, $\beta_k^{(s+1)}$ can be calculated by running a weighted negative binomial regression for the response y with prior weights $\tau_k^{(s)}$ (Lambert 1992, McLachlan 2007). Weighted negative binomial regression maximizes the above likelihood also for ϕ_k similar to the iterative method described in (31).

Lastly,

$$\pi_0^{(s+1)} = \frac{e^{X_0\gamma^{(s)}}}{1 + e^{X_0\gamma^{(s)}}}, \tag{2.1}$$

$$\pi_1^{(s+1)} = \frac{\sum_{i=1}^{n} \tau_{i1}}{\sum_{i=1}^{n} \tau_{i1} + \tau_{i2}}, \text{ and} \tag{2.2}$$

$$\pi_2^{(s+1)} = 1 - \pi_1^{(s+1)} \tag{2.3}$$

For identifiability reasons, we place a constraint on π_1 such that

$$\pi_1^{(s+1)*} = \max\left(\pi_{1,min}, \pi_1^{(s+1)}\right)$$

where $\pi_{1,min}$ is chosen to be 0.5.

We set the convergence criterion as when the relative change in the complete model log-likelihood at iteration k compared to $k - 10$ is less than 10^{-5}.

2.3 Simulation Results and Real Data Application

We first demonstrate through simulation the utility of the mixture regression approach in the detection of enriched regions in DAE-seq data. We then applied ZINBA to FAIRE-seq and ChIP-seq of CTCF, RNA polymerase II (RNA Pol II), and histone H3 lysine 36 tri-methylation (H3K36me3) (Figure 2.1A). These datasets represent a

diversity of signal patterns ranging from narrow peaks with high signal-to-noise ratios (CTCF) to broad enrichment regions with low signal-to-noise ratios (H3K36me3). In addition to identifying biologically relevant signals in each of these datasets, ZINBA is capable of estimating the contribution of component-specific covariates to signal in each component. Incorporation of covariates into the model improved peak detection in difficult modelling situations, such as in amplified genomic regions. In the absence of input control, we show that other covariates allow for comparable performance as when input control is utilized. Lastly, we demonstrate that ZINBAs ability to isolate broad and narrow enrichment regions reveals functional differences in RNA Pol II elongation status. We conclude that ZINBA provides a general and flexible framework for the analysis of a diverse set of DAE-seq datasets.

2.3.1 Simulation

To evaluate the utility of incorporating covariate information for the detection of enriched regions, we constructed simulated datasets, and used G/C content as one example of such a covariate. Simulated datasets were constructed to artificially control the relationship between G/C content and the enrichment, background, and zero-inflated components. Window count data were simulated to represent three types of common NGS signal patterns, ranging from TFBS (high signal-to-noise ratio, 1% of genome belongs to enrichment component), FAIRE (moderate signal-to-noise ratio, 5% of genome belongs to enrichment component), to some histone modifications (low signal-to-noise ratio, 10% of genome belongs to enrichment component). For each data type, three sets of data were simulated, hence nine datasets in total. In each data set, G/C content always had a positive relationship with signal in the background component and a positive relationship with the probability of being zero-inflated. However, G/C content was simulated to have either a positive, neutral or negative relationship with enrichment.

For each of the nine datasets, 100,000 windows were simulated. These consisted of 250-bp windows from human chromosome 22. G/C content was simulated from these windows as well.

For each of the nine simulated datasets, three different uses of the covariate were employed to model the simulated data: (a) model 1, no covariates; (b) model 2, G/C content is incorporated in modelling the zero-inflated and background components only; (c) model 3, G/C content is incorporated in modelling all three components.

Our results show that models that properly accounted for the underlying simulated relationships with G/C content in each component resulted in the best classification outcomes. For example, when enrichment had an inverse relationship with G/C content (Figure 2.2A, B), model 3 consistently led to higher sensitivity and specificity relative to models 1 and 2 (Figure 2.2C, D). Simulated component-specific relationships between G/C content and signal were also correctly captured in model 3 (Figure 2.2E, F), with average enrichment signal decreasing and average background signal increasing with respect to G/C content. Ignoring the role of G/C content completely (model 1) resulted in classification based purely on signal, which misses informative trends in the data. We find similar results for the simulated condition of positive and neutral relationships between G/C content and enrichment. Thus, including relevant covariates to model each component provides a more informed assessment of enrichment versus background.

2.3.2 All methods perform similarly in High Signal-to-Noise Ratio DAE-seq data

For the CTCF ChIP-seq data set, the set of ranked peaks for each algorithm was compared to the occurrence of the CTCF motif (JASPAR motif MA0139.1). The genome-wide set of motifs was identified using FIMO, part of the MEME suite (2),

with default parameters. All of the algorithms were able to identify a high proportion of sites containing the CTCF motif (Figure 2.3A) and had comparable peak lengths (Figure 2.3C). Positioning of peaks called by ZINBA was slightly closer to the CTCF motifs (Figure 2.3B). These results are consistent with other comparisons of ChIP-seq peak calling algorithms (85), which revealed few differences in sensitivity and specificity when applied to high signal-to-noise ChIP-seq data. Of the 50,228 refined peaks called by ZINBA, 95.2% were in common with MACS (60,135 peaks) and 99.9% were in common with F-seq (276,879 peaks).

2.3.3 ZINBA captures both Broad and Short regions of enrichment

One unique feature of RNA Pol II ChIP-seq data is that enrichment consists of both punctate high signal-to-noise ratio peaks at transcription start sites (TSSs) and broader, low signal-to-noise peaks into the body of genes (61). All of the algorithms were able to capture a large proportion of annotated TSSs (Figure 2.3D, E). However, the set of refined peaks called by the shape detection algorithm within ZINBA resulted in a set of narrower peaks much more closely associated with the TSSs of genes (Figure 2.3E, F) compared with MACS, F-Seq, and unrefined ZINBA peak calls. A relatively high degree of overlap can be seen between each of the peak sets, although the overlap is not as strong compared to those observed for the CTCF dataset.

The ability to produce both a refined (punctate) and unrefined (broad) set of peak calls using ZINBA provides an opportunity to infer elongating versus stalled RNA Pol II. For the case of stalled RNA Pol II, one would expect a punctate peak at the TSS, but no broad peak within the body of the gene (86). Under this expectation, we computed a 'stalling score', where smaller values correspond to a broad high-amplitude signal across the gene, and larger values to a punctate signal near the 5' end of the gene

and lower-amplitude signal along the gene body. Previous computations of RNA Pol II stalling scores utilized a height ratio between the punctate peak at the TSS and the median height of the broader region (92) (Figure 2.4A). Using ZINBA, our stalling score further incorporates the lengths of the broad and punctate enriched regions found in the experimental sample. The stalling index had a strong negative relationship (P-value $< 10^{-10}$) to the expression of the nearby gene (Figure 2.4B) and explained more of the variance in measured gene expression (R2 = 3.5%) than a score utilizing only the ratio of punctate to broad signal height (R2 = 0.04%). The ability to calculate this metric reflects one potential use of the peak boundary refinement module within the ZINBA framework.

2.3.4 ZINBA performs well in Low signal-to-noise DAE-seq Data

FAIRE-seq (25, 26) differs from ChIP-seq in that it is an antibody-free method that recovers DNA fragments that are relatively resistant to formaldehyde crosslinking to proteins. The crosslinking profile of chromatin is likely dominated by histone-DNA interactions, and therefore the sites preferentially recovered by FAIRE correspond to sites of nucleosome depletion. On average the size of each FAIRE site corresponds to the loss of approximately one nucleosome (200 to 300 bp). Compared to the binding events identified for TFBSs by ChIP-seq, the FAIRE-seq sites tend to have much lower signal-to-noise, have a slightly broader pattern of enrichment, and encompass a larger proportion (1 to 2%) of the genome. In addition, input control is often not available. Therefore, many of the assumptions utilized by existing algorithms, especially for the analysis of TFBS ChIP-seq, are not well-suited to the analysis of this data type (43).

We analyzed a K562 FAIRE-seq dataset lacking a matching input control sample with each algorithm, and compared the resulting set of peaks from each algorithm to

a set of DNase I hypersensitivity sites (DHSs) (68) isolated from the exact same set of cells. The DHSs were called by F-seq, and were selected as a standard because of the longstanding use of DNase as a method for identification of open chromatin sites. Both ZINBA and MACS called a high proportion of FAIRE sites that overlapped a DHS, but a low proportion of FAIRE sites called by F-seq were localized to a DHS (Figure 2.3G). The set of sites called by both MACS and F-Seq tended to be longer and more errant in K562 CNV regions, where approximately 37% of MACS and 27% of F-seq peaks were localized to a DHS, compared to 50% of ZINBA peaks. Overlap between called peak sets from ZINBA, MACS, and F-seq for FAIRE were more disparate than those found in high signal-to noise CTCF data.

Open chromatin regions tend to have strong correspondence to active regulatory elements and promoter regions of expressed genes (25). Comparison of the set of ZINBA RNA Pol II and FAIRE-seq refined peak calls yielded a significantly higher degree of overlap compared to the other algorithms (Figure 2.5A), indicating consistency in ZINBA peak calls across data types.

2.3.5 ZINBA captures broad patterns of enrichment

The deposition of H3K36me3 is mediated by enzymes that travel along with RNA Pol II during transcriptional elongation, and therefore this histone modification typically occurs in broad segments encompassing a large proportion of gene bodies (66). Utilizing the 'broad' ZINBA setting, the H3K36me3-enriched regions identified by ZINBA correspond to the broad patterns of enrichment covering actively transcribed gene bodies, as expected.

On average, 80% of the lengths of the top N most active UCSC gene bodies were covered by the set of H3K36me3 ZINBA peaks (Figure 2.5B). A lower level of gene body coverage was found from other methods. Of the 40,180 H3k36me3 merged ZINBA

peaks, 71% overlap a gene body, compared with only 59% of F-seq peaks merged in a similar fashion, suggesting higher specificity of these broad ZINBA regions to gene bodies. Of the set of ZINBA merged peak calls that overlapped a gene body, the median and 75th percentile of peak lengths was 5,374 and 18,370 bp respectively, indicative of the broader set of features that are being called.

Within the set of H3K36me3 enrichment regions identified by ZINBA, those that overlap ZINBA RNA Pol II broad regions also contain significantly higher levels of RNA expression compared to those that do not overlap broad RNA Pol II regions (Figure 2.5C). Approximately 85% of ZINBA H3K36me3 broad regions that overlap a ZINBA RNA Pol II broad region contain non-zero RNA-seq signal (7,585 out of 8,873 overlapping regions), compared to only 58% of those that do not (18,134 out of 31,312 non-overlapping regions). Furthermore, of ZINBA H3K36me3 regions with non-zero RNA-seq signal, those that overlapped a ZINBA RNA Pol II broad region had three-fold higher median RNA expression. The relationships we observe among our ZINBA calls recapitulates the biology of H3K36me3, where higher levels RNA Pol II activity correspond to higher levels of RNA transcription and histone modification (Figure 2.5D).

2.3.6 ZINBA performs comparably with or without input control data

Comparison of ZINBA peak calls from BIC-selected models considering input as a covariate versus those that do not reveal similar performance in isolating relevant enriched regions. For example, 94% of the CTCF ChIP-seq peaks discovered using a model that included input were held in common with a model considering only G/C content, mappability score, and the local background estimate as starting covariates. Recovery of sites overlapping a CTCF motif was also very similar (Figure 2.6A). This

similarity in performance with and without input extended to the lower signal-to-noise H3K36me3 ChIP-seq data (Figure 2.6B). Because of the broad nature of H3K36me3 enrichment, we only considered G/C content and the mappability score as potential covariates in the no-input model. These results demonstrate the ability of ZINBA to distinguish regions of enrichment from background in the absence of input control.

2.4 Conclusions

Two major challenges in the analysis of DAE-seq data are the diversity in signal patterns that exist across the wide range of possible experiments, and sample-specific issues such as CNV that may further complicate analysis. ZINBA is a flexible statistical framework capable of identifying regions of enrichment across a wide variety of DAE-seq data types, enrichment patterns, and experimental conditions. ZINBA's flexibility in modelling background and enrichment regions with sets of covariates allows for the identification of enriched regions in difficult modelling conditions, such as in datasets with complex local CNVs or lacking a matching input control sample. ZINBA can identify both broad and sharp regions of enrichment, and we demonstrate this capability in differentiating RNA Pol II elongation status. In addition, the statistical framework used is applicable to both high signal-to-noise data such as from CTCF ChIP-seq, as well as to low signal-to-noise data such as from FAIRE-seq. ZINBA produces peak calls that are consistent with known biological patterns, and performs favorably relative to existing specialized methods over a broad range of signal patterns and data types. ZINBA is implemented as a freely available R package.

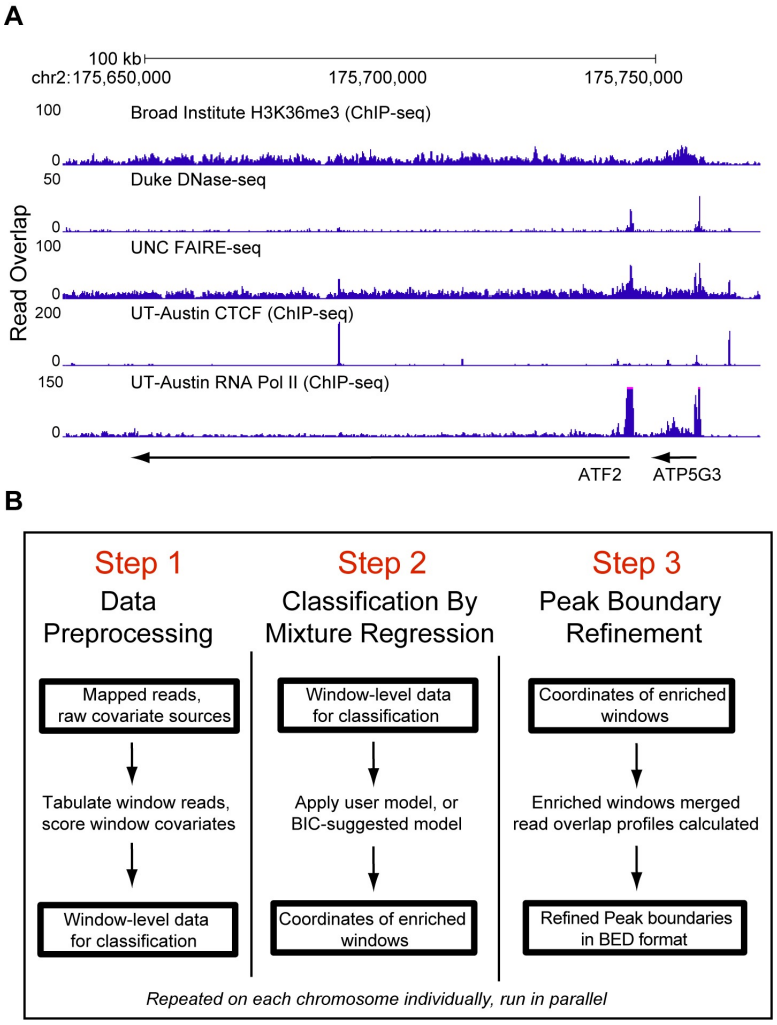

Figure 2.1: A) A variety of signal patterns exist across different types of DAE-seq data. B) ZINBA overview

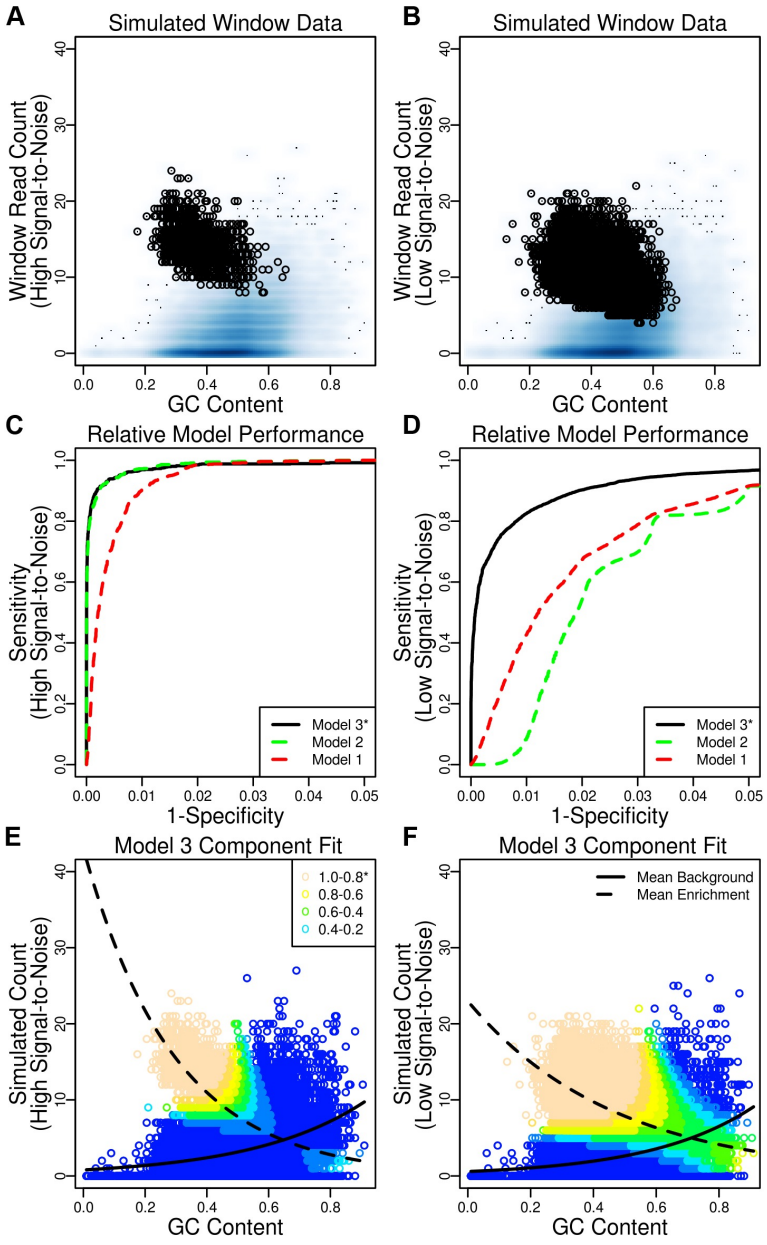

Figure 2.2: (A, B) Density plots showing the distribution of background (blue shading) and enriched (black circles) simulated counts (y-axis) versus G/C content (x-axis). Window counts were simulated with either (A) a low proportion of high signal-to-noise sites or (B) a high proportion of low signal-to-noise sites. (C, D) ROC curves for the performance of three different component-specific covariate model formulations, including no covariates (model 1, red dashed line), G/C content modelling the background and zero-inflated components (model 2, green dashed line) and G/C content modelling the background, zero-inflated and enriched components (model 3, black solid line). Classification results for the simulated (C) low proportion of high signal-to-noise sites and (D) high proportion of low signal-to-noise sites. (E, F) Scatter plot of G/C content (x-axis) versus simulated window counts (y-axis) using model 3 to estimate the posterior probability of a window being enriched, which is depicted as a color gradient.

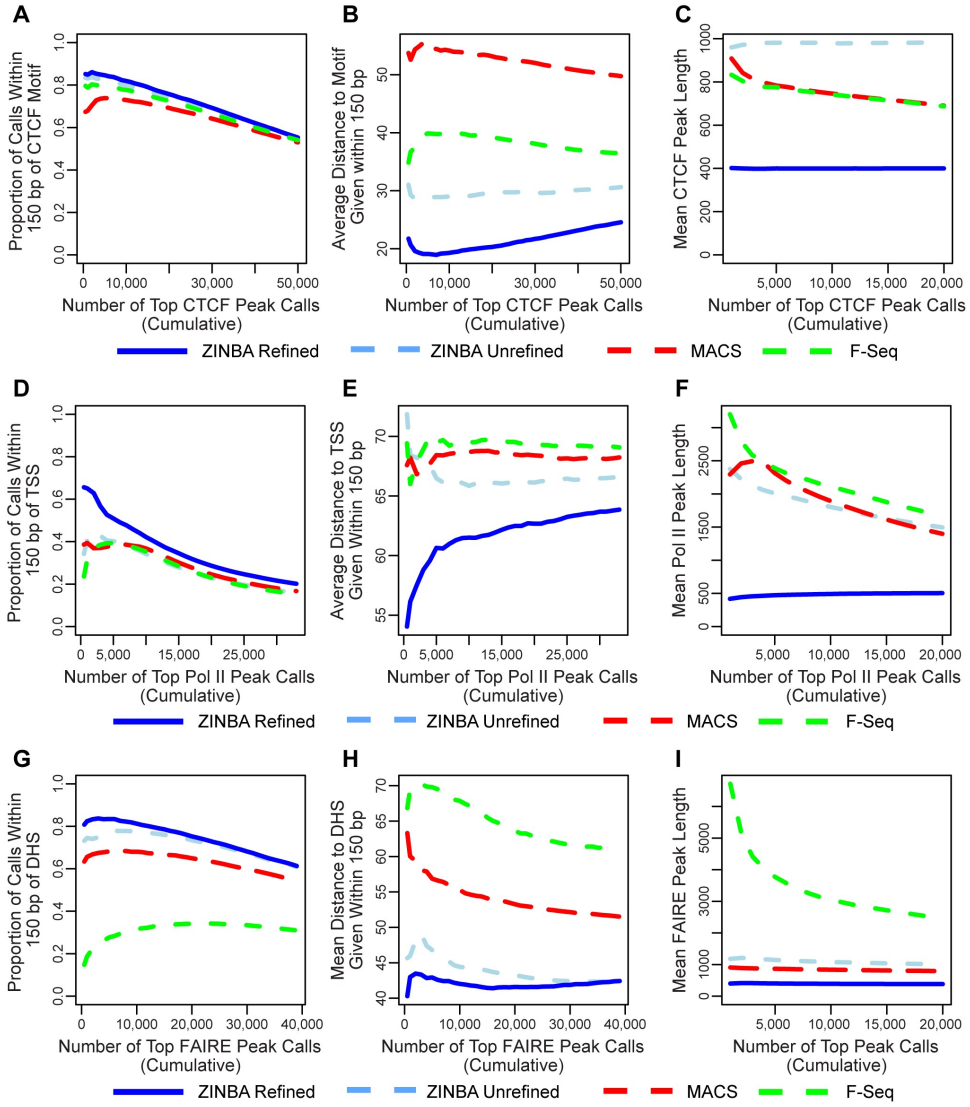

Figure 2.3: (a-i) For CTCF ChIP-seq (A-C), RNA Pol II ChIP-seq (D-F) and FAIRE-seq (G-I) data, the top N ranked peaks from MACS (red dashed line), F-Seq (green dashed line) and ZINBA unrefined regions (light blue dashed line), and ZINBA refined regions (blue solid line) were compared based on the proportion overlapping a biologically relevant set of features (A, D, G), average distance to the biologically relevant set of features (B, E, H) and average length of peaks (C, F, I). The biologically relevant set of features included the CTCF motif (A), transcription start sites (TSSs) for RNA Pol II (D) and DNase hypersensitive sites (DHSs) for FAIRE (G).

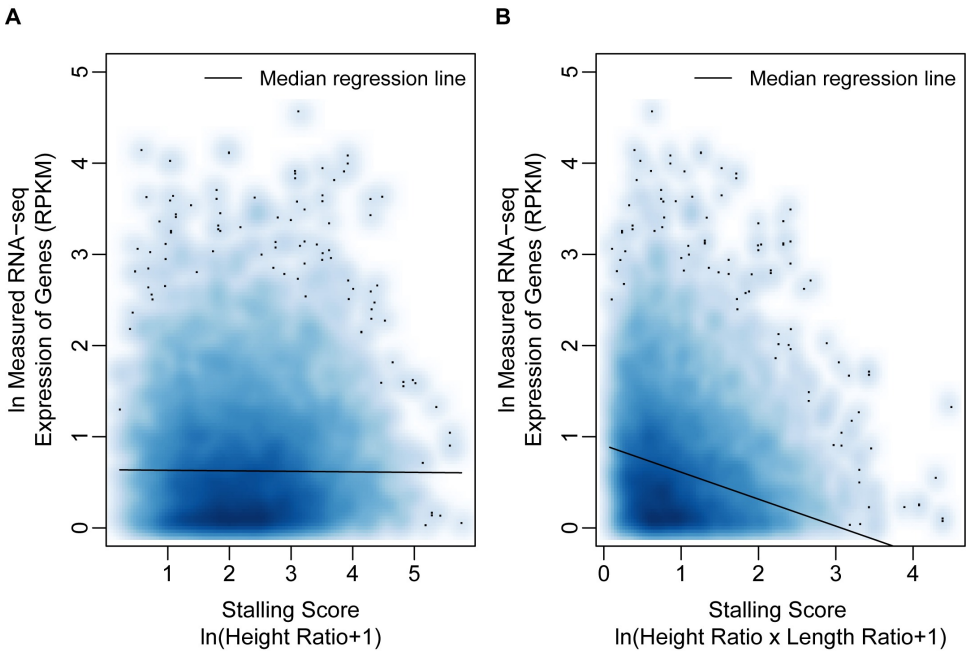

Figure 2.4: (A) Scatter plot of gene expression versus stalling score, considering a stalling metric based only on the height ratio between the punctate peak and the broader region. A median regression line modeling the natural log of nearby gene expression as a function of this stalling score is overlain. (B) Scatter plot of gene expression versus the ZINBA stalling score additionally accounting for the ratio of RNA Pol II punctate peak length to broad peak length. A strong negative association can be seen between our stalling score and corresponding expression (p-value $< 10^{-10}$), where genes having likely stalled polymerase (higher scores) have much lower levels of gene expression. Higher scores are indicative of regions with less elongation but contain a punctate peak near the transcription start site. The score considering only the height ratios of punctate to broad regions explained much less of the variance in measured gene expression ($R^2 = 0.0004$) versus the ZINBA stalling score ($R^2 = 0.035$), suggesting that the incorporation of punctate to broad peak lengths ratios into the ZINBA score represents a marked improvement.

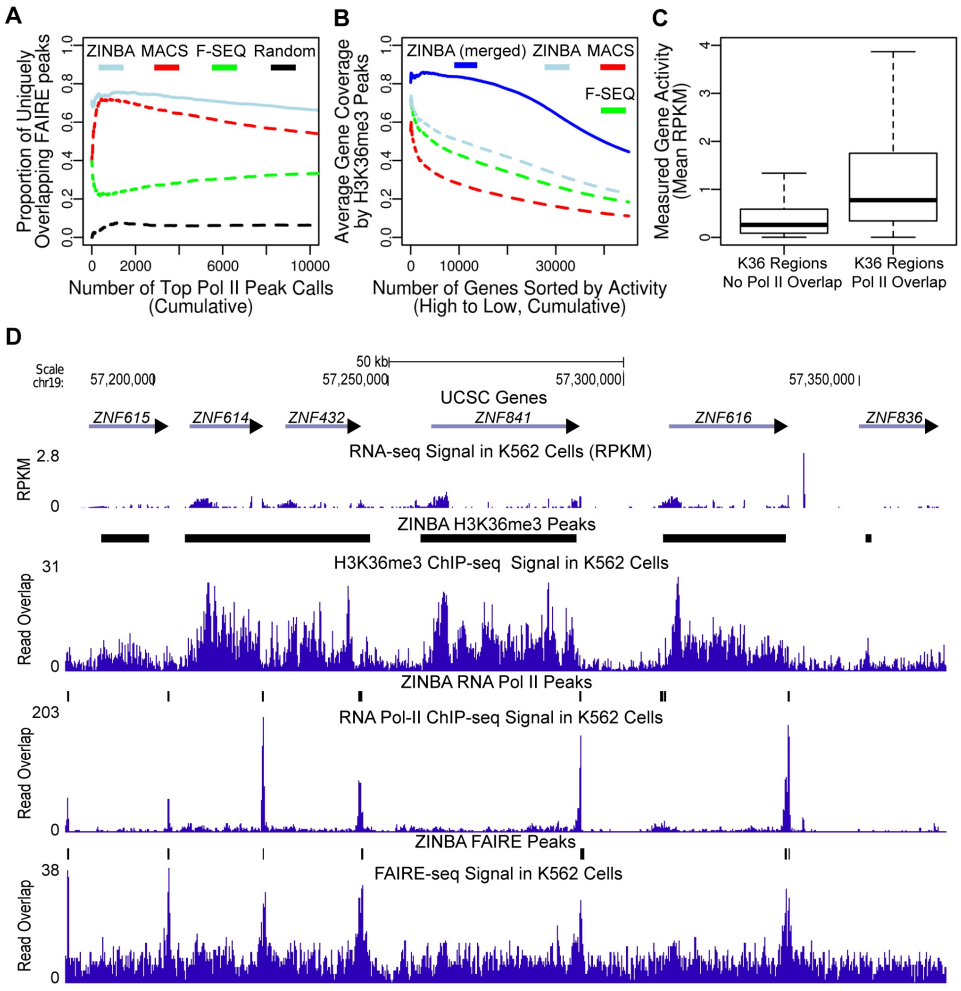

Figure 2.5: (A) The proportion of the top cumulative sets of MACS (red dashed line), F-Seq (green dashed line) and ZINBA refined (light blue line) RNA Pol II peaks that uniquely overlap a FAIRE-seq peak called by the respective method. For comparison, overlap was also compared using randomly permuted RNA Pol II and FAIRE-seq ZINBA peak calls (black dashed line). (B) The average coverage of the cumulative sets of the top N ranked genes (expression, high to low) by H3K36me3 regions called by MACS (red dashed line), F-Seq (green dashed line) and ZINBA unrefined regions (light blue dashed line). The set of unrefined ZINBA H3K36me3 regions were further clustered throughout the genome to merge nearby peaks (blue solid line) and compared to the ranked list of genes in terms of gene body coverage. (C) Comparison of measured gene expression levels for the set of ZINBA H3K36me3 broad regions that either did or did not overlap a ZINBA RNA Pol II broad region. Those overlapping a ZINBA RNA Pol II broad region had three-fold higher median levels of measured gene expression than H3K36me3 regions that did not have any overlap. (D) Representative view of the set of H3K36me3 broad, FAIRE-seq refined and RNA Pol II refined ZINBA peak calls displayed in the UCSC Genome Browser along with the respective read overlap data.

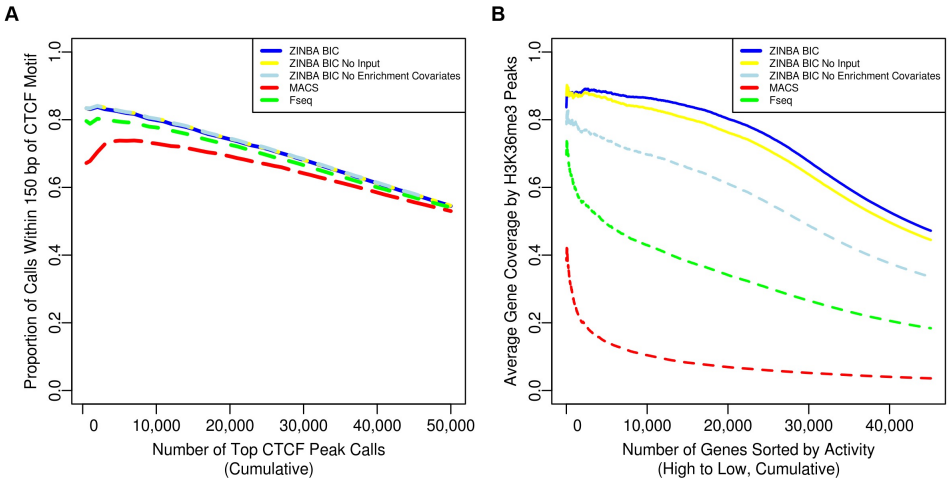

Figure 2.6: (A) In CTCF ChIP-seq data, BIC selected models not considering input control as a starting covariate (using G/C-content, mappability score, local background estimate) perform similarly to BIC selected models considering input control (using input control, G/C-content, mappability score). In addition, we find that not modeling enrichment covariates has little impact on eventual classification performance (light blue). (B) In contrast, not modeling enrichment in low signal to-noise H3K36me3 ChIP-seq data has a large impact on ZINBAs ability to recover enriched regions spanning gene bodies (light blue). Similar to CTCF, not considering input control (G/C content, mappability score) results in similar performance as when input control is considered (yellow).

Chapter 3

Some Statistical Strategies for DAE-seq Data Analysis: Variable Selection and Modeling Dependencies among Observations

As mentioned in the previous chapters, there are several challenges in the analysis of DAE-seq data. For example, several confounding factors may influence DAE-seq read density across the genome. Therefore, it is important to adjust for the effects of these factors, especially when a matching control dataset is not available (67). Examples of these factors include the local percentage of G and C nucleotides ("G/C content"), the ability to accurately assign reads to a particular region of the genome ("mappability"), and the presence of local DNA copy number alterations (71, 42, 67). When the number of factors (covariates) is large, it is challenging to choose the best subset of them to model DAE-seq data since the relevant set of covariates may be different for background and enriched regions.

In addition, window read counts from DAE-seq data are often serially correlated. This correlation may simply be due to the dependence of underlying states of adjacent windows. For example, an enriched region may cover several consecutive windows in certain data types. However, we have noticed that given the underlying states, nearby windows' read counts may still exhibit moderate to strong autocorrelation, even if they are from non-overlapping windows (Figure 3.1). This autocorrelation may be due to other covariates that are either unmeasurable or not included in the analysis, for example, DNA characteristics other than GC content or some bias due to the sequencing

technique. Explicitly modeling this autocorrelation may explain a greater proportion of the variation in observed window read counts and may lead to more accurate estimates of the effects of other covariates as well as more accurate detection of enriched regions.

Several methods have been introduced to utilize Hidden Markov Models (HMMs) to account for the dependence between underlying states and identify enriched regions in DAE-seq data (88, 74, 63), where the transitions between latent states are explicitly modeled and the window read counts are assumed to be conditionally independent given the underlying states. One drawback of these approaches is that the confounding covariates, such as GC content and mappability, have not been incorporated into the HMM. In addition, potential autocorrelation of adjacent windows given the underlying states is ignored. A few methods utilizing Finite Mixtures of Regression Models (42, 67) have been proposed to incorporate the effects of multiple covariates to identify enriched regions. Unfortunately, these methods ignore any dependence between adjacent windows' read counts. Most notably, when the number of covariates is large, no computationally efficient method exists to automatically select state-specific covariates for HMMs where the observations are non-independent.

To address these challenges, we develop an Autoregressive Hidden Markov Model (AR-HMM) with covariates for DAE-seq data analysis. We derive a novel EM algorithm to estimate model parameters and we show that our method achieves better performance in the detection of enriched regions in simulated and real DAE-seq datasets. We also introduce a computationally efficient penalized maximum likelihood estimation procedure to perform state-specific variable selection, and establish the conditions for the existence, sparsity, and asymptotic normality of the penalized maximum likelihood estimates for a general class of penalty functions. We demonstrate the performance of this procedure in simulation studies, and apply it to discover a subset of 40 transcription factors whose protein-DNA interaction profiles are associated with a well-studied

histone modification mark. In summary, we provide several practical solutions to challenges in DAE-seq analysis with broader applicability to other areas of statistics.

3.1 Background

3.1.1 DAE-seq data analysis using Finite Mixtures of Regression Models

Consider a random sample of n responses $Y_1, \ldots, Y_n$ from a Finite Mixture of Regressions Model (FMR) such that for each realization y_i

$$p(y_i|\mathbf{X}, \Psi_F) = \sum_{k=1}^{K} \pi_k f_k(y_i|X_{ik}, \beta_k, \phi_k), \tag{3.1}$$

where K is the number of mixture components, $\mathbf{X}$ is an $n \times p$ matrix that includes the values of p covariates, $X_k \in \mathbb{R}_{n\times p_k}$ contains p_k columns of $\mathbf{X}$ that correspond to the p_k covariates pertaining to component k, $X_{ik} \in \mathbb{R}_{1\times p_k}$ is the i^{th} row of X_k, $\Psi_F = (\beta^T, \phi^T, \pi^T)^T$, $\beta = (\beta_1^T, \ldots, \beta_K^T)^T$ where β_k is a $p_k \times 1$ vector of regression coefficients for component k, $\phi = (\phi_1, .., \phi_K)^T$ where ϕ_k is the dispersion parameters for the k-th component, and $\pi = (\pi_1, \ldots, \pi_K)^T$ is the set of prior probabilities of component membership such that $\sum_{k=1}^{K} \pi_k = 1$ and $\pi_k > 0$. Also, $f_k(y_i|X_{ik}, \beta_k, \phi_k)$ is the conditional density that y_i is generated from mixture component k with mean μ_{ik} and link function $h(\cdot)$ such that $h(\mu_{ik}) = X_{ik}\beta_k$. Denote the underlying mixture component for window i by Z_i where $Z_i = 1, \ldots, K$.

Under the assumptions of the FMR, we have $Z_i \perp Z_j$ and $y_i|Z_i \perp y_j|Z_j$ for $1 \leq i \neq j \leq n$. Given $\mathbf{X}$ and $\hat{\Psi}_F$, the posterior probability that window i belongs to component k can be computed and utilized for classification purposes (56). In DAE-seq data analysis, each chromosome is typically modeled separately. Therefore the sample size of this problem is the number of windows spanning a chromosome, which may range from 100,000 to almost a million depending on the chosen window length (typically

50-500 bp) and chromosome size.

FMR-based methods such as (42) and (67) utilize $K = 2$ Negative Binomial mixture components pertaining to the background and enriched regions of DAE-seq data. In addition, (67) assumed an additional component to account for potential zero-inflation in window read counts, whereas (42) modeled zero-inflation through a binary latent variable in the background component. These FMR-based approaches can flexibly account for the effects of multiple covariates that influence the window read counts in background and/or enriched regions. However, they ignore the dependence that may exist between adjacent windows, which may be due to dependence of underlying components or dependence of observations given underlying components. As a result, *ad-hoc* approaches were required to detect broader enriched regions for epigenetic marks (67).

3.1.2 Variable Selection via Penalized Likelihood for FMR

In previous work involving FMRs and their applications to DAE-seq data analysis, (67) employed all-subset selection coupled with BIC (72) to select the best set of covariates for each mixture component. This approach is not computationally feasible when the number of covariates p is large, especially in the mixture distribution case where the number of possible models is 2^{pK}(38).

An enormous amount of statistical literature has been devoted to variable selection by penalized regression or penalized likelihood, and different types of penalty functions have been developed including the LASSO (79), SCAD (18), adaptive LASSO (97), MCP (93), Log penalty (23) among many others. (38) have introduced variable selection via penalized likelihood in FMRs. They developed an EM algorithm to maximize the penalized FMR likelihood and showed that the Penalized Maximum Likelihood Estimate (PMLE) in the M-step of the EM algorithm can achieve the "oracle property",

where the zero coefficients are estimated to be zero with probability approaching to one and the non-zero coefficients are unbiasedly and efficiently estimated as if the "true" submodel is known (18).

We extend the results of (38) to establish an efficient variable selection procedure (EM + coordinate descent algorithm) in the context of Hidden Markov Models (HMMs) where the emission probability of each state is modeled by a set of covariates. We derive the asymptotic properties of the PMLE for the M-step of the algorithm and evaluate this algorithm using both simulations and real data analysis.

3.1.3 Accounting for Serial Dependence in Generalized Linear Models

Generalized linear models that account for serial correlation in observations fall into two categories: parameter-driven and observation-driven (10). Parameter-driven models assume that the dependence between subsequent observations is controlled by a latent process that induces the correlation. For example, (90) modeled a time series of counts, denoted by y_t, by a log-linear model conditioning on a latent process ϵ_t, such that $u_t = E(y_t|\epsilon_t) = \exp(x_t'\beta)\epsilon_t$ and $\text{var}(y_t|\epsilon_t) = u_t$. The correlations among y_t's are induced by the correlations among ϵ_t's . In contrast, observation-driven models specify the conditional distribution of y_t as a function of past observations y_{t-1}, ..., y_1. For example, an autoregressive (AR) model is an example of observation-driven model. (91) introduced a Poisson generalized linear AR model, which, in the case of AR(1), has the following link function

$$\log(\mu_i) = X_i\beta + \nu\left\{\log(y_{i-1} + c) - \log[\exp(X_{i-1}\beta) + c]\right\}, \tag{3.2}$$

where X_i is the i^{th} row of $\mathbf{X}$, i.e., the covariates' values for the i^{th} sample, β is a $p \times 1$ vector of regression coefficients, ν is the auto-correlation coefficient, and $0 < c < 1$ is

used to avoid taking log of a zero.

Estimation for parameter-driven models is computationally difficult, especially in longer time series (12), making them less desirable choices in DAE-seq data analysis. Therefore, we utilize an observation-driven approach. Denote the data from the prior observation as $F_{i-1} = (X_{i-1}, y_{i-1})$. The model of (91) assumes $\mu_i = E[Y_i|F_{i-1}]$ and $h(\mu_i) = X_i\beta + \nu g(F_{i-1})$, where $h()$ is a link function. We generalize the model of (91) to an observation-driven autoregressive-HMM (AR-HMM) with K states. We assume an AR(1) dependence, which is reasonable for DAE-seq data. Let $Z_i = 1, \ldots, K$ be a random variable of the underlying state of the i-th observation, and thus $Z = (Z_1, \ldots, Z_n)$ are the random variables for the state path. Given a particular instance of state path, denoted by $z = (z_1, \ldots, z_n)$, we have $g(F_{i-1}, z) = \log(y_{i-1}+c) - \log[\exp(X_{i-1,z_{i-1}}\beta_{z_{i-1}}) + c]$, where $X_{i-1,z_{i-1}}$ are the $(i-1)$-th observations of the covariates for state z_{i-1}. However, when the state path is unknown, such a generalization is non-trivial. To the best of our knowledge, an AR-HMM that allows the autoregressive term to be dependent on state path and state-specific covariates has not been introduced in the literature. We develop such a model in this paper.

3.2 Methods

3.2.1 Penalized MLE for HMMs with covariates

In a Hidden Markov Model with covariates, the observations $Y_1, \ldots, Y_n$ have a natural order (e.g., observations along time points) and the transitions between latent states along the ordered observations are explicitly modeled. We again denote the random variable for state path by $Z = (Z_1, \ldots, Z_n)$ and $z = (z_1, \ldots, z_n)$ denotes an observed state path. Let K be the number of states, and let S be the set of K^n possible state paths of length n. We assume a stationary Markov chain with state-to-state transition probabilities $\gamma = (\gamma_{11}, \ldots, \gamma_{KK})^T$, where $\gamma_{jk} = p(Z_i = k|Z_{i-1} = j)$ for $i = 2, \ldots, n$,

$\sum_{k=1}^{K} \gamma_{jk} = 1$, $\gamma_{jk} > 0$ for all $j, k = 1, \ldots, K$, and $Y_{i-1} \perp Y_i | (Z_{i-1}, Z_i)$. Then the likelihood of the observed data is

$$\begin{aligned} L_n(\Psi_H|\mathbf{X}, y) &= \sum_{z \in S} \left\{ \prod_{k=1}^{K} \left[\pi_k f_k(y_1|X_{1k}, \beta_k, \phi_k)\right]^{I[z_1=k]} \right. \\ &\times \left. \prod_{i=2}^{n} \prod_{k=1}^{K} \left[f_k(y_i|X_{ik}, \beta_k, \phi_k)^{I[z_i=k]} \prod_{j=1}^{K} \gamma_{jk}^{I[z_{i-1}=j, z_i=k]} \right] \right\}, \end{aligned}$$

where $\Psi_H = (\beta^T, \phi^T, \gamma^T, \pi^T)^T$, $\beta = (\beta_1^T, \ldots, \beta_K^T)^T$, and $\mathbf{X}_{n \times p}$ are the set of p covariates that may be related with the mean value of each state distribution, while the relevant covariates for each state may be a subset of the p covariates. In contrast to the notation used for the FMR, $\pi = (\pi_1, \ldots, \pi_K)^T$ is now known as the set of prior probabilities of state membership for the first observation. Conditional density $f_k(y_i|X_{ik}, \beta_k, \phi_k)$, which belongs to exponential family, is now defined as the state-specific emission density. The remaining variables are defined similarly as those for the FMR, which have been introduced in Section 3.1.1.

Let $l_n(\Psi_H|\mathbf{X}, y) = \log L_n(\Psi_H|\mathbf{X}, y)$ be the log likelihood. To achieve our goal in variable selection, which is to select relevant covariates pertaining to each state, we maximize the following penalized log likelihood

$$pl_n(\Psi|\mathbf{X}, y) = l_n(\Psi|\mathbf{X}, y) - \mathcal{P}(\Psi), \tag{3.3}$$

where Ψ is defined as $\Psi = (\beta^T, \phi^T, \gamma^T, \pi^T, \eta^T)^T = (\Psi_H^T, \eta^T)^T$, $\eta = (\eta_1, \ldots, \eta_k)^T$, and η_k is the proportion of the observations belonging to state k. $\mathcal{P}(\Psi) = \sum_{k=1}^{K} \eta_k \sum_{l=1}^{p} \rho_{\omega_k}(\beta_{lk})$ is the total penalty to the likelihood, and $\rho_{\omega_k}(\beta_{lk})$ denotes a penalty function with tuning parameter(s) ω_k, which could be a function of the sample size n. Give the stationarity assumptions, the parameter η can be obtained from transition probability γ, however we keep η for notational simplicity.

Maximization of the penalized likelihood in (3.3) with respect to β balances the overall model fit, $l_n(\Psi|\mathbf{X}, y)$, and the cost of model complexity, controlled by $\mathcal{P}(\Psi)$. In this paper, we employed three penalties that represent a broad class of the available penalties.

- LASSO: $\rho_{\lambda_k}(\beta_{lk}) = \lambda_k|\beta_{lk}|$, for $\lambda_k > 0$,
- SCAD: $\rho'_{\lambda_k}(\beta_{lk}) = \lambda_k \left\{ I(|\beta_{lk}| \leq \lambda_k) + \frac{I(|\beta_{lk}|>\lambda_k)(a\lambda_k - |\beta_{lk}|)_+}{\lambda_k(a-1)} \right\}$, for $a > 2$ and $\lambda_k > 0$, where $x_+ = x$ if $x \geq 0$ and $x_+ = 0$ otherwise.
- Log Penalty: $\rho_{\lambda_k,\tau_k}(\beta_{lk}) = \lambda_k \log(|\beta_{lk}| + \tau_k)$, for $\lambda_k > 0$ and $\tau_k > 0$.

The LASSO (i.e., L_1 penalty) is a convex penalty, while both SCAD and Log penalty belongs to a class of folded concave penalties (19). The Log penalty can be interpreted as an Iterative Adaptive LASSO (IAL) penalty (76), which represents a class of penalties that bridge the L_0 penalty ($\rho_{\lambda_k}(\beta_{lk}) = \lambda_k I[\beta_{lk} \neq 0]$) and the L_1 penalty. The LASSO penalty has only one tuning parameter λ_k. SCAD (18) has two regularization parameters λ_k and a. Following (18), we set $a = 3.7$ for all states $k = 1, \ldots, K$, and only treat λ_k as a tuning parameter. The Log penalty has two tuning parameters λ_k and τ_k.

(38) have studied the theoretical properties of the PMLE in the content of the FMR. Specifically, they establish the conditions on penalty $p_{w_k}(\cdot)$ such that the Oracle Property can be achieved for the PMLE, which is estimated by penalized weighted least squares in the M-step of their algorithm. We extend the results of (38) to the HMM with covariates, which requires some additional regularity conditions from (5). Partition $\beta_k^T = (\beta_{k1}^T, \beta_{k2}^T)$ such that β_{k2} pertains to the zero effects. In addition, we partition $\Psi^T = (\Psi_1^T, \Psi_2^T)$ such that Ψ_2 contains zero parameters in the model, namely β_{k2}, $k = 1, \ldots, K$. Let Ψ_0 be the true values of Ψ and $\beta_{lk,0}$ be the true regression coefficients corresponding to the l^{th} covariate in the k^{th} state. Define $a_n = \max_{l,k} \{\rho_{\omega_k}(\beta_{lk,0})/\sqrt{n} : \beta_{lk,0} \neq 0\}$,

$b_n = \max_{l,k}\left\{|\rho'_{\omega_k}(\beta_{lk,0})|/\sqrt{n} : \beta_{lk,0} \neq 0\right\}$, and $c_n = \max_{l,k}\left\{|\rho''_{\omega_k}(\beta_{lk,0})|/n : \beta_{lk,0} \neq 0\right\}$, where $\rho'_{\omega_k}(\beta_{lk,0})$ and $\rho''_{\omega_k}(\beta_{lk,0})$ represent the first and second derivatives of $\rho_{\omega_k}(\beta_{lk})$ with respect to β_{lk}, respectively. We place the following conditions on the penalty $\rho_{\omega_k}(\beta_{lk})$:

P0: The penalty $\rho_{\omega_k}(\beta_{lk})$ is symmetric around 0, nondecreasing for β_{lk} in $(0, \infty)$ and is twice differentiable for all β_{lk} in $(0, \infty)$. $\rho_{\omega_k}(\beta_{lk})$ attains its minimum at $\beta_{lk} = 0$.

P1: As $n \to \infty$, $a_n = o_p(1 + b_n)$ and $c_n = o_p(1)$.

P2: For $N_n = \left\{\beta_{lk} : 0 < \beta_{lk} \leq n^{-\frac{1}{2}} \log(n)\right\}$, $\lim_{n\to\infty} \inf_{\beta_{lk}\in N_n} \rho'_{\omega_k}(\beta_{lk})/\sqrt{n} = \infty$.

Corollary 1: Assume the regularity conditions apply (see Appendix Section A.1). We assume that $(Y_1, Z_1), \ldots, (Y_n, Z_n)$ is a discrete-time stochastic process corresponding to the HMM with covariates such that $(Y_1, Z_1)|\mathbf{X}_1, \ldots, (Y_n, Z_n)|\mathbf{X}_n$ is stationary conditional on $\mathbf{X}_i$. Then, given conditions P0-P2 and assuming the number of states K is known, we have the following conclusions:

1. Consistency: There exists a local maximizer $\hat{\Psi}$ of $pl_n(\Psi|\mathbf{X}, y)$ such that $||\hat{\Psi}-\Psi|| = O_p(n^{-\frac{1}{2}}(1 + b_n))$. where $||.||$ represents the euclidean norm.

2. Sparsity: $p(\hat{\Psi}_2 = 0) \to 1$ as $n \to \infty$

3. Asymptotic Normality: $\sqrt{n}\left\{[I(\Psi_{01}) - \mathcal{P}''(\Psi_{01})/n]\,(\hat{\Psi}_1 - \Psi_{01}) + \mathcal{P}'(\Psi_{01})/n\right\} \to N(0, I(\Psi_{01}))$, where $I(\Psi_{01})$ is the subset of Fisher Information matrix for the non-zero effects, and $\mathcal{P}'(\Psi_{01})$ and $\mathcal{P}''(\Psi_{01})$ are the first and second derivatives of penalty function $\mathcal{P}(\Psi_{01})$ with respect to Ψ_{01}.

Therefore under the regularity conditions and given conditions P0-P2, the PMLE corresponding to penalty $p_{w_k}(\beta_{lk})$ can achieve the Oracle Property. The proof is similar to the proofs of Theorems 2 and 3 in (38), and we briefly described the proof in Appendix

Section A.1. We note that the above theoretical properties are for the M-step estimates of the EM algorithm instead of the final estimates from the EM algorithm.

3.2.2 An EM + coordinate descent algorithm

In this section, we provide the details of our EM algorithm that maximizes the Penalized likelihood of a HMM. Recall that the random variable $Z = (Z_1, \ldots, Z_n)$ denotes the state path. In the s-th step of the EM algorithm, the Q-function of penalized likelihood (3.3) is

$$\begin{aligned}
& Q(\Psi|\Psi^{(s)}) = E_Z\left[pl_n(\Psi|y, \mathbf{X}, Z)|y, \mathbf{X}, \Psi^{(s)}\right] \\
= & \sum_{k=1}^{K} p(Z_1 = k|y, \mathbf{X}, \Psi^{(s)}) \log(\pi_k) + \sum_{i=2}^{n}\sum_{k=1}^{K}\sum_{j=1}^{K} p(Z_{i-1} = j, Z_i = k|y, \mathbf{X}, \Psi^{(s)}) \log(\gamma_{jk}) \\
+ & \sum_{i=1}^{n}\sum_{k=1}^{K} p(Z_i = k|y, \mathbf{X}, \Psi^{(s)}) \log\left[f_k(y_i|X_{ik}, \beta_k, \phi_k)\right] - \mathcal{P}(\Psi) \\
= & Q(\pi|\Psi^{(s)}) + Q(\gamma|\Psi^{(s)}) + \left[Q(\beta, \phi|\Psi^{(s)}) - \mathcal{P}(\Psi)\right]. \qquad (3.4)
\end{aligned}$$

In the E-step, $p(Z_i = k|y, \mathbf{X}, \Psi^{(s)})$ and $p(Z_{i-1} = j, Z_i = k|y, \mathbf{X}, \Psi^{(s)})$ can be computed by the standard forward-backward algorithm, detailed in Appendix A.2. Similar to the FMR, the posterior probability $p(Z_i = k|y, \mathbf{X}, \Psi^{(s)})$ is utilized in the classification of observations. In the M-step, the Q function is separable for π, γ, and (β, ϕ), and only β is penalized. Therefore, π and γ can be estimated from the unpenalized likelihood such that $\gamma_{jk}^{(s+1)} = \left[\sum_{i=2}^{n} p(Z_{i-1} = j, Z_i = k|y, \mathbf{X}, \Psi^{(s)})\right] / \left[\sum_{i=2}^{n} p(Z_{i-1} = j|y, \mathbf{X}, \Psi^{(s)})\right]$, and $\pi_k^{(s+1)} = p(Z_1 = k|y, \mathbf{X}, \Psi^{(s)})$. Under the assumptions of stationarity we can derive $\eta_k^{(s+1)}$ as the solution to $\eta_k^{(s+1)}\Pi\left(\gamma^{(s+1)}\right) = \eta_k^{(s+1)}$, where $\Pi\left(\gamma^{(s+1)}\right)$ is the $K \times K$ transition probability matrix based on $\gamma^{(s+1)}$. For simplicity we estimate η_k such that $\eta_k^{(s+1)} = \sum_{i=1}^{n} p(Z_i = k|y, \mathbf{X}, \Psi^{(s)})/n$. This estimate works well in our simulations.

$Q(\beta, \phi|\Psi^{(s)})$ can be decomposed into K components, one for each state. Therefore

we can maximize the last term of (3.4) with respect to β_k and ϕ_k separately for each state k. One approach is to alternately estimate β_k and ϕ_k until convergence (31). However, this approach is computationally intensive and we adopt a one-step update in our algorithm. Specifically, we perform a conditional maximization to obtain $\beta_k^{(s+1)}$ given $\phi_k^{(s)}$ using penalized Iteratively Reweighted Least Squares (IRLS) followed by an conditional maximization to obtain $\phi_k^{(s+1)}$ given $\beta_k^{(s+1)}$. Our algorithm can be considered as an ECM algorithm (57) where we perform conditional maximization of β_k and ϕ_k. In contrast to alternately estimating β_k and ϕ_k until convergence for each M-step, our one-step update of β_k and ϕ_k leads to more iterations in the ECM algorithm, but overall less computational time. The details of this ECM algorithm are presented in the Appendix, Section AI.3.

Here we briefly describe a key part of this algorithm, the penalized IRLS to estimate $\beta_k^{(s+1)}$. Employing a canonical link function, we can derive the following objective function of the penalized IRLS:

$$\mathcal{Q}_k(\beta_k|\Psi^{(s)}) = \frac{1}{2}\sum_{i=1}^{n}\zeta_{ik}^{(s)}\left[v_{ik}^{(s)}(q_{ik} - X_{ik}\beta_k)^2\right] + \eta_k^{(s)}\sum_{l=1}^{p_k}\rho_{\omega_k}(\beta_{lk}), \tag{3.5}$$

where $\zeta_{ik}^{(s)} = p(Z_i = k|y, \mathbf{X}, \Psi^{(s)})$, $\mu_{ik}^{(s)} = E(y_i|Z_i = k, X_{ik}, \Psi^{(s)})$, $v_{ik}^{(s)} = \text{Var}(y_i|Z_i = k, X_{ik}, \Psi^{(s)})$, and $q_{ik}^{(s)} = X_{ik}\beta_k^{(s)} + (y_i - \mu_{ik}^{(s)})/v_{ik}^{(s)}$. We minimize the above objective function by a coordinate descent algorithm. Prior to minimization, we standardize the columns of $\mathbf{X}$ to be mean 0 and variance 1, and we transform the final estimates of $\hat{\beta}_k$ back to their unstandardized values following convergence of the coordinate descent algorithm.

To select tuning parameters, we follow the procedure similar to (38) where we first obtain the MLE under the full model $\hat{\Psi}_{\texttt{full}}$ on the data. We then select the optimal set of tuning parameters for each state individually, while fixing the parameters of all other

states at their full model MLEs. This procedure significantly reduces the computational cost in tuning parameter selection when a large number of states exist. For each state, we select tuning parameters by minimizing BIC.

3.2.3 Autoregressive Hidden Markov Model with Covariates (AR-HMM)

We extend the HMM with covariates described in the Section 3.2.1 to allow dependence between the observations conditional on the hidden states. Given underlying states, we assume that there is AR(1) dependence between Y_i and Y_{i-1} conditional on (Z_{i-1}, Z_i) such that $f(y_i|y_{i-1}, \ldots, y_1, Z_{i-1} = j, Z_i = k, \mathbf{X}, \beta, \phi) =$
$f_{jk}(y_i|X_{ik}, \beta_k, \phi_k, \nu_k, r_{i-1,j})$, where the subscript $_{jk}$ in f_{jk} indicates $Z_{i-1} = j, Z_i = k$ and

$$r_{i-1,j} = \begin{cases} 0 & \text{if } i = 1 \\ \log(y_{i-1} + 1) - \log[\exp(X_{i-1,j}\beta_j) + 1] & \text{if } i > 1. \end{cases} \tag{3.6}$$

If the underlying state is k for the i-th observation, then
$E(y_i|Z_i = k, Z_{i-1} = j, X_{ik}, \beta_k, \phi_k, \nu_k, r_{i-1,j}) = \mu_{ikj}$ with link function $h(\mu_{ikj}) = X_{ik}\beta_k +$
$\nu_k r_{i-1,j}$ where ν_k is the set of AR coefficient for state k. Then the AR-HMM complete data likelihood given some state path z is

$$\begin{aligned} L(\Psi_A|\mathbf{X}, y, z) &= \prod_{k=1}^{K} [\pi_k f_k(y_1|X_{ik}, \beta_k, \phi_k)]^{I[z_1=k]} \\ &\times \prod_{i=2}^{n} \prod_{k=1}^{K} \prod_{j=1}^{K} [\gamma_{jk} f_{jk}(y_i|X_{ik}, \beta_k, \phi_k, \nu_k, r_{i-1,j})]^{I[z_{i-1}=j, z_i=k]}, \end{aligned}$$

where $\Psi_A = (\beta^T, \phi^T, \gamma^T, \pi^T, \nu^T)^T$ and $\beta = (\beta_1^T, \ldots, \beta_K^T)^T$.

We develop an EM algorithm inspired by (34) to obtain the MLE of the AR-HMM.

We can show that the Q-function is

$$
\begin{aligned}
Q(\Psi_A|\Psi_A^{(s)}) &= E_Z\left\{\log\left[L(\Psi_A|y,\mathbf{X},z)\right]|\Psi_A^{(s)},y,\mathbf{X}\right\} \\
&= \sum_{k=1}^{K} p(Z_1=k|y,\mathbf{X},\Psi_A^{(s)})\log(\pi_k) \qquad (3.7)\\
&+ \sum_{i=2}^{n}\sum_{k=1}^{K}\sum_{j=1}^{K} p(Z_{i-1}=j,Z_i=k|y,\mathbf{X},\Psi_A^{(s)})\log(\gamma_{jk}) \\
&+ \sum_{i=1}^{n}\sum_{k=1}^{K}\sum_{j=1}^{K} p(Z_{i-1}=j,Z_i=k|y,\mathbf{X},\Psi_A^{(s)})\log[f_{jk}(y_i|X_{ik},\beta_k,\phi_k,\nu_k,r_{i-1,j})].
\end{aligned}
$$

Direct maximization of the above Q function is computationally difficult because β is also present in $r_{i-1,j}$. We adopt an approximation to fix $r_{i-1,j}$ at $r_{i-1,j}^{(s)}$, which is the value of $r_{i-1,j}$ at step s given $\beta^{(s)}$. This approximation significantly improves the computational efficiency of our algorithm, which is very important for the analysis of DAE-seq data with tens of thousands of observations. Later simulation results show that this approximation does not lead to any bias of MLE. At time point $i = 1$, the likelihood that the current state is k is weighted by $p(Z_1 = k|y,\mathbf{X},\Psi_A^{(s)})$, and for $i > 1$ the likelihood that the current state is k and previous state is j is weighted by $p(Z_{i-1} = j, Z_i = k|y,\mathbf{X},\Psi_A^{(s)})$. We can derive these quantities from the forward and backward probabilities, see the Appendix Section A.2 for details.

Given the weights from the E-step, we obtain the MLE of Ψ_A in the M-step. Since the Q-function can be separated into three sets of parameters π, γ, and $(\beta^T,\phi^T,\nu^T)^T$, we can estimate each set of parameters separately. First, $\pi_k^{(s+1)} = p(Z_1 = k|y,\mathbf{X},\Psi_A^{(s)})$, $\gamma_{jk}^{(s+1)} = \sum_{i=2}^{n}\tau_{ijk}/\left[\sum_{i=2}^{n}\sum_{k=1}^{K}\tau_{ijk}\right]$ where $\tau_{ijk} = p(Z_{i-1} = j, Z_i = k\ |y,\mathbf{X},\Psi_A^{(s)})$. We estimate β_k, ϕ_k, and ν_k for each state k using the following augmented regression to account for missing data due to the AR component in the model. Following (34), let $\tilde{y}$ be the augmented version of y by repeating each y_i K times. In other words,

$\tilde{y} = (\underbrace{y_1, \ldots, y_1}_{\text{K}}, \underbrace{y_2, \ldots, y_2}_{\text{K}}, \ldots, \underbrace{y_n, \ldots, y_n}_{\text{K}})^T$. Let $\tilde{\mathbf{X}}$ be the augmented version of $\mathbf{X}$. The dimension of $\tilde{\mathbf{X}}$ is $nK \times (p+1)$. The first p columns of $\tilde{\mathbf{X}}$ is constructed by repeating each row of $\mathbf{X}$ K times. Let $\mathbf{r}_i = (r^{(s)}_{i-1,1}, \ldots, r^{(s)}_{i-1,K})^T \in \mathbb{R}_{K\times 1}$, where $r^{(s)}_{i-1,j}$ is defined in (3.6) given $\beta^{(s)}$, and we set the $(p+1)$th column of $\tilde{\mathbf{X}}$ as $\mathbf{r} = (\mathbf{r}_1^T, \ldots, \mathbf{r}_n^T)^T \in \mathbb{R}_{nK\times 1}$. We construct $\tilde{\mathbf{X}}_k \in \mathbb{R}_{nK\times(p_k+1)}$ by extracting the p_k columns of $\tilde{\mathbf{X}}$ corresponding to the p_k covariates for state k and the $(p+1)$th column of $\tilde{\mathbf{X}}$. Then the parameters β_k, ϕ_k and ν_k can be estimated by a weighted generalized linear regression of $\tilde{y}$ on $\tilde{\mathbf{X}}_k$ with the weights $\mathbf{w}_k = (\mathbf{w}_{1k}^T, \ldots, \mathbf{w}_{nk}^T)^T$, $\mathbf{w}_{ik} = (w_{i1k}, .., w_{iKk})^T$ for $i = 1, \ldots, n$, and $w_{ijk} = p(Z_{i-1} = j, Z_i = k \mid y, \mathbf{X}, \Psi_A^{(s)})$, for $j = 1, \ldots, K$. This is equivalent to complete data maximum likelihood estimation where the missing data is "filled in" with a set of weighted values spanning the range of the discrete missing covariate, in this case $r_{i-1,j}$. The approach in (54) allows for computationally efficient and exact computation of the observed information matrix using a modified forward-backward algorithm.

The penalization procedure described in Section 3.2.1 extends to the AR-HMM by simply replacing $\mathbf{X}$ with $\tilde{\mathbf{X}}$, y with $\tilde{y}$, and utilizing weights w_{ijk} from the AR-HMM E-step. This procedure is similar to penalized estimation with missing data (24), but the AR part complicates the asymptotic theory. We expect the oracle properties similar to Corollary 1 to hold, but a careful theoretical study is beyond the scope of this paper.

3.3 Simulation Studies

In this section, we evaluate the performance of the proposed variable selection procedure in simulated data. We also compare the classification performance and accuracy of the AR-HMM relative to other methods in simulated DAE-seq datasets.

3.3.1 Simulation Setup

Simulated datasets of a two-state Poisson AR-HMM with first order dependence in states and AR(1) dependence in observations were generated in the following manner. We simulated data to mimic window read counts from a CTCF (CCCTC- binding factor) ChIP-seq dataset and a H3K36me3 (Trimethylation of Lys36 in histone H3) histone modification ChIP-seq data set. In the CTCF dataset, enrichment regions are short, rare, and contain much higher signals relative to background. This enrichment pattern is typical for most transcription factor binding sites. In contrast, enrichment regions from H3K36me3 dataset are typically broader, more abundant, and contain weaker signals. We simulated window read counts corresponding to these two DAE-seq data types to represent a wide range of enrichment patterns found in real data analysis.

For both simulated data types, we set $\gamma_{11} = 0.9$ and $\gamma_{12} = 0.1$, which correspond to the background-to-background and background-to-enrichment state transition probability, respectively. For simulated CTCF ChIP-seq data, we set $\gamma_{21} = 0.9$ and $\gamma_{22} = 0.1$, corresponding to the enrichment-to-background and enrichment-to-enrichment transition probability, respectively. For simulated histone modification data we set $\gamma_{21} = 0.1$ and $\gamma_{22} = 0.9$. This simulation setup results in rare transitions from background to enriched regions in both data types, shorter regions of enrichment in the CTCF data, and broader regions of enrichment in the histone modification data.

The underlying state path z was simulated using the transition probabilities corresponding to each simulated data type. For each window i ($i = 1, \ldots, n$), we simulated a set of p covariates $X_i = (x_{i1}, \ldots, x_{ip})$ as uniform (0,1) random variables to generate covariate matrix $\mathbf{X}_{n \times p}$. We utilized the same $\mathbf{X}$ to simulate window read counts corresponding to either of the $K = 2$ states. In each data type, the relative strength of the signal in each state can be tuned by modifying β_1 and β_2 appropriately. Then, given X, z and the selected model parameters, we recursively simulated window read counts

$y_1, \ldots, y_n$ for each simulation case in the AR-HMM using the following procedure:

1. for $i = 1$, $y_1 \sim \text{Pois}(\exp\{X_1\beta_k\})$ if $z_1 = k$,

2. for $i > 1$, $y_i \sim \text{Pois}\left(\exp\left\{X_i\beta_k + \nu_k\left[\log(y_{i-1}+1) - \log(\exp(X_{i-1}\beta_j)+1)\right]\right\}\right)$ if $z_i = k$ and $z_{i-1} = j$.

To simulate window read counts from HMM with covariates, we simply followed the above procedure except we set $\nu_1 = \nu_2 = 0$.

3.3.2 Variable Selection in Hidden Markov Models with Covariates

In the following simulation studies, we evaluate our variable selection method in the context of the HMM with covariates using small ($n = 200$) or large ($n = 10000$) sample size and low ($p = 5$) or high ($p = 100$) dimension. We employ three penalties: the LASSO, SCAD, and Log penalties. The true parameter values corresponding to the first 4 covariates per component are listed in Table 3.1, and all the other coefficients are set to be 0. We utilize the same set of regression parameters to simulate both the CTCF and histone modification-style datasets so that we can directly compare the effect of relative state frequencies on variable selection. Variable selection performance is measured by the number of true discoveries (TDs) and the number of false discoveries (FDs). Specifically, among all the covariates selected by a variable selection method, a TD is a covariate that has (true) non-zero coefficient and a FD is a covariate that has (true) zero coefficient. These numbers of TDs/FDs are averaged across 100 simulations for each simulation situation.

Overall, the number of TDs increases and the number of FDs decreases as the sample size n increases. We observe that of all the penalties, the LASSO has the worst variable selection performance and greatest bias in the estimated values for the true non-zero parameters. This is in line with the results from (18, 38), since the

LASSO cannot satisfy all of the penalty conditions P0-P2 for the Oracle Property. As a result, it cannot simultaneously achieve sparsity and unbiased estimation of the true non-zero coefficients as $n \to \infty$. The Log and SCAD penalties, however, satisfy these conditions, and have substantially better performance than the LASSO. These results provide empirical support for the Oracle Property of Corollary 1.

In this simulation, approximately 10% of simulated windows in the simulated CTCF-style data are from the enrichment state, in contrast, ∼50% of simulated windows in the histone modification-style data are from enrichment state. Comparing the performance across the two simulated data types, the variable selection performance and parameter estimation accuracies decrease when the relative state frequencies are unbalanced, such as in the simulated CTCF-style data. However given the large sample sizes that are typical in DAE-seq datasets, the effect of this imbalance will be limited.

3.3.3 AR-HMM

For each simulated data type (e.g., CTCF or histone modification), we simulated 1000 datasets of $n = 10,000$ observations each from a two-state Poisson AR-HMM with first order dependence in states and AR(1) dependence in observations. The mean value of each state-specific emission distribution is a function of two covariates plus an intercept. To simulate CTCF-style data with higher levels of signal in the enrichment state relative to background, we set $\beta_1 = (\beta_{01}, \beta_{11}, \beta_{21}) = (0, 1, 1)$ and $\beta_2 = (\beta_{02}, \beta_{12}, \beta_{22}) = (1.5, 2, 2)$. In the histone modification-style data, we set $\beta_1 = (\beta_{01}, \beta_{11}, \beta_{21}) = (0, 1, 1)$ and $\beta_2 = (\beta_{02}, \beta_{12}, \beta_{22}) = (0.5, 2, 2)$ to simulate weaker signals in the enrichment state. Within each simulated data type, we allowed ν_2 to be either 0.2 or 0.8 (weak or strong auto-correlation) and we fixed ν_1 to be 0.2 to mimic the observed low dependence between windows in background (Figure 3.1). For each simulation case, we compared the parameter estimates and classification performance of the AR-HMM

with those from the FMR and the HMM with covariates. The AR-HMM estimates are accurate regardless of the values for ν_2 or simulated data type (Table 3.3), suggesting that the AR-HMM estimation procedure is robust over a range of conditions.

In contrast, the estimates from the HMM and FMR tend to be biased in each simulation setting. The magnitude of the bias increases as the value of ν_2 increases. This bias however is larger in the simulated histone modification-style data. In the simulated CTCF-style data, the parameter estimates for the FMR and HMM are very similar (differences are on the order of 10^{-5}). This is due to the fact that the majority of transitions in the CTCF-style data are background-to-background, and that the enrichment regions are relatively easy to discern by each method due to their strong signals. Therefore, accounting for dependence in states alone in the HMM does not yield better accuracy in parameter estimates relative to the FMR.

Our main interest however is the performance of each method to distinguish enriched and background regions. We evaluated such classification performance by ROC curves (Figure 3.2). In the simulated CTCF ChIP-seq data with $\nu_2 = 0.2$ (Figure 3.2A), all methods perform similarly. This is expected, as in CTCF ChIP-seq the strong and sharp signals in enrichment regions allow for adequate detection of enrichment even in the absence of any covariate information (67). When the dependence between observations in the enrichment state increases from $\nu_2 = 0.2$ to $\nu_2 = 0.8$, the AR-HMM performs slightly better than other methods (Figure 3.2B).

However, in the simulated histone modification-style data, the AR-HMM performs much better relative to other methods. When the dependence between observations from the enrichment state is low ($\nu_2 = 0.2$), both the HMM and AR-HMM perform much better than the FMR (Figure 3.2C). This is because the FMR cannot account for the more prevalent enrichment-enrichment transitions between windows, which can aid the detection of regions containing weaker enrichment signals. When the dependence

between observations from the enrichment state is high ($\nu_2 = 0.8$), the AR-HMM performs much better than both the HMM and FMR (Figure 3.2D).

We also observe that under model misspecification, where there is no correlation in the underlying states and observations given the states, the AR-HMM performs similarly to the correct model: the FMR. For example, using CTCF style data simulated under FMR assumptions, we find that the parameter estimates for the AR-HMM, HMM, and FMR are almost the same, and the estimates for autoregressive parameters ν_1 and ν_2 are close to 0 (Table 3.8). Therefore these methods have the same performance to identify enriched regions.

3.3.4 Variable Selection in the AR-HMM

Next we demonstrate the performance of variable selection method in AR-HMM. We generated simulated data sets similar to Section 3.3.2, except that we allow for dependence between simulated observations given the states by setting $\nu_1 = \nu_2 = 0.4$. Similar to the results from Section 3.3.2, variable selection performance improves and estimation bias drops as the sample size increases (Table 3.4).

The estimation accuracy of ν_k, which we do not penalize, also increases with sample size. For CTCF style data, the variable selection performance is worse in the high-dimensional low sample size case (p=100 and n=200), owing to the small number of samples in the enrichment state in the simulation (approximately 20). In real data analyses we typically observe sample sizes much larger than $n = 200$ so we do not expect this to be an issue. Other conclusions with respect to data type and penalties are similar to what are observed in the case of HMM with covariates. Empirically, these results demonstrate that variable selection performance is adequate in the AR-HMM and the PMLEs in this context share similar properties to those in the HMM with covariates.

3.4 Application to Human GM12878 CTCF and H3K36me3 ChIP-seq datasets

3.4.1 Data preparation and model selection

We benchmarked the performance of the FMR, HMM, and the AR-HMM in two ChIP-seq datasets in terms of their ability to identify biologically relevant signals. These datasets were obtained from the ENCODE project (4) and included a human GM12878 CTCF ChIP-seq dataset (UT-Austin, Replicate 3) and a human GM12878 H3K36me3 ChIP-seq dataset (Broad Histone, Replicates 1 and 2). In the CTCF ChIP-seq data, we checked whether the significant regions called by each method overlapped with CTCF binding motifs, which are conserved DNA sequences that the CTCF transcription factor preferentially binds to (40). H3K36me3 histone modifications are deposited broadly across gene bodies during transcription (3), and thus we benchmarked the enriched regions of H3K36me3 histone modifications by their overlap with gene bodies.

In each dataset, non-overlapping 250bp windows from Human Chromosome 22 were utilized to tabulate window read counts. Covariate information and read counts for each window were tabulated in the manner detailed in (67). In this analysis we considered the covariates GC-content, mappability, and window read counts from a matching input control. We then applied the two-state Negative Binomial AR-HMM, two-state Negative Binomial HMM, and two-component Negative Binomial FMR model to each dataset. For each method, the mean value of each state distribution was modeled with some covariates using a log link function.

Each of these methods can calculate the posterior probability for each window belonging to background. Denote the posterior probability that the i-th window belongs to background by κ_i, $(i = 1, ..., n)$. Such κ_i's are also referred to as local FDRs (16) for detecting enriched regions. For a cutoff of posterior probability α, the total FDR

is $\sum_{i=1}^{n}[\kappa_i I(\kappa_i \leq \alpha)]/\sum_{i=1}^{n}[I(\kappa_i \leq \alpha)]$. We chose a posterior probability cutoff by controlling FDR. Adjacent windows meeting a given FDR threshold were merged together into a single region, and multiple performance metrics were calculated for the set of enriched regions identified by each method.

It is not known *a priori* which set of covariates should be used to model the mean of each state-specific emission distribution. Therefore, we employed the proposed variable selection procedure to determine the best model for HMM and AR-HMM in each dataset. The full model includes an intercept (fixed), the main effects of mappability, GC content, and input control, as well as their two-way and three-way interactions. In the AR-HMM model, we included the autoregressive covariate from (3.2) but did not subject it to penalization. Given the simulation results from Tables 3.2 and 3.4, we used the SCAD penalty in our real data application. Including the main effects and interactions, there are 7 covariates for the mean model of each state, hence 128 possible models per state and 16384 models for two states. Therefore all-subset selection is infeasible even in this relatively simple situation. In regression studies involving interactions, a reasonable constraint is that higher order interactions are included in the model if and only if all the corresponding main effects and lower order interactions are also included in the model. We did not implement this constraint because of computational challenge and because these covariates and their interactions were not of biological interest. The benefit of variable selection of these covariates was to provide an automatic procedure for model fitting. An example of selecting biological meaningful factors is presented in the next section.

We find that in each dataset, the model selected for the AR-HMM has much better fit than the model selected for the HMM in terms of BIC (Table 3.5). In addition, the AR-HMM estimates for ν_2 in both datasets are large, suggesting that strong dependence exists between window read counts in enrichment regions. In background regions, this

dependence is much weaker, where the estimate for ν_1 are 0.283 and 0.163 for the CTCF and H3K36me3 ChIP-seq datasets, respectively. We also observe that each selected model of background state includes the three-way interaction of GC content, mappability, and input control ($\beta_{123,1}$), suggesting a strong synergistic relationship in their effects on background signals.

3.4.2 Performance comparison for CTCF ChIP-seq

Given these selected models, we first examined the classification performance of the FMR, HMM, and the AR-HMM in the CTCF ChIP-seq data across FDR cutoffs. For the FMR, we utilized the model selected for the HMM for all of the subsequent analyses. In the CTCF ChIP-seq data, both the AR-HMM and HMM call less enriched regions than the FMR (Figure 3.3A). A slightly higher proportion of the enriched regions called by the AR-HMM or the HMM overlap with CTCF binding sites (Figure 3.3C), which is partly due to the enriched regions called by both methods tending to be longer (Figure 3.3B, F).

Next we compared the performances of different methods using ROC curves while defining a true discovery as the window/region that overlaps a CTCF motif. The three methods perform similarly (Figure 3.5), and the FMR performs slightly better in terms of number of windows overlapping CTCF sites. This is because AR-HMM and HMM tends to call longer regions that cover more windows than FMR (Figure 3.3F). The majority of significant regions called by each method overlap those called by other methods (Table 3.6), and the maximum signals of the significant regions that are called uniquely by each method are much greater than the background (Figure 3.3D), which suggest that none of methods call many false positives. Therefore we conclude that the three methods perform similarly in the CTCF data. This is expected given the simulation

results and the fact that CTCF data have strong and easily discernible enriched regions. We notice, however, that the enrichment-enrichment transition probability (γ_{22}) of the HMM and AR-HMM (Table 3.7) is relatively large, suggesting that some regions of CTCF ChIP-seq enrichment may span more than a single window in real data. Examples are shown in Figure 3.3E,F. Finally, we also examined the performance of two popular existing methods F-seq (7) and MACS (94). Similar to results from (67), we found all methods perform similarly for CTCF ChIP-seq data (Figure 3.6A).

3.4.3 Performance comparison for the H3K36me3 ChIP-seq data

In contrast to the CTCF ChIP-seq data, the enrichment regions in H3K36me3 ChIP-seq are much broader and have relatively weaker signals. We benchmark the enriched region calls by their coverage of gene bodies rather than whether an enriched region has overlap with any portion of a gene body. We observe a significant improvement in performance of the AR-HMM relative to the HMM or FMR. For example, enriched regions called by the AR-HMM ("AR-HMM calls" for short) generally span greater lengths of gene bodies (Figure 3.4A) and each AR-HMM call tend to be longer than HMM calls or FMR calls (Figure 3.4B). Although the FMR calls overlap more genes than the AR-HMM calls (Figure 3.4C), the AR-HMM calls cover a greater average proportion of the overlapped gene bodies (Figure 3.4D). ROC curves confirm that the AR-HMM and HMM have acceptable specificity (Figure 3.5C). The enriched windows identified by AR-HMM and HMM cover ∼20% of Chr22. About 90% of these enriched regions overlap with a gene body, which is much higher than expected by chance considering less than 40% of Chr22 are covered by gene bodies (including both intronic and exonic regions). The performance difference is most apparent in regions where the enrichment signal is relatively weak. For example, in the regions shown in Figure

3.4E-F, the AR-HMM and HMM tends to classify consecutive windows into enriched regions while the FMR calls are much more sporadic. As a result, the number of regions called by FMR is much greater than those from AR-HMM and HMM (Figure 3.5D), but overall these regions covered fewer potions of gene bodies (Figure 3.5C).

In addition, we applied MACS and F-seq to detect enriched regions in the H3K36me3 ChIP-seq data. We find that our AR-HMM and HMM perform significantly better in terms of sensitivity and specificity of gene body coverage relative to these methods (Figure 3.6B). Therefore, based on the above results of real data analysis and simulations, we conclude that accounting for multiple sources of dependence in the observations may significantly improve the performance of detecting enriched regions in epigenetic datasets.

All the previous results are based on non-overlapping windows. Many analyses utilize overlapping windows to account for possible window "boundary effects", where regions of elevated signal may be split by a window's boundary (96, 36). In such situations, adjacent windows have stronger AR correlations because they are partially overlapped, and thus we would expect the AR-HMM to have a greater advantage over the HMM. To illustrate this point, we performed real data analysis in our H3K36me3 dataset using overlapping windows (250bp windows with 125bp overlap). Figure 3.7A confirms that the advantage of AR-HMM is larger when using overlapping windows than non-overlapping windows. In fact, HMM calls include more false positives when studying read counts of overlapping windows, as it cannot distinguish correlation due to underlying states dependence or due to AR dependence (Figure 3.7B-D).

3.5 The Relation Between Histone Modification H3K36me3 and Transcription Factor Occupancy

The functional role of histone modification H3K36me3 has attracted a great amount of research interest. It has been shown that H3K36me3 is involved in the elongation

phase of transcription (45), leukaemogenesis (84), mRNA splicing (41), and DNA mismatch repair (46). In earlier sections of this paper we sought to classify genomic regions as either H3K36me3-enriched or background. However, the magnitude of ChIP-seq signals within enriched or background regions itself is also biologically meaningful since it reflects the proportion of cells having a H3K36me3 mark at that location, among a large population of cells. Furthermore, the genome-wide variability of these signals in enriched/background regions may be associated with a subset of biological factors. However, no method currently exists to efficiently discover *state-specific* relationships between DAE-seq signals and a large number of biological factors.

In this section, we use our variable selection procedure and the ChIP-seq data from the ENCODE (Encyclopedia of DNA Elements) project (78) to study the state-specific relationship between H3K36me3 signals and the DNA binding signals of 40 transcription factors (TFs) in either H3K36me3-enriched regions or background regions. We study this relationship in two ways. First, we study the relationship between H3K36me3 signal and TF binding within the same window. Then, we assess the association between TF binding in promoter regions and H3K36me3 signals in downstream genes. The former study examines state-specific relationships between H3K36me3 and local TF binding, while the later directly examines promoter-driven regulation of H3K36me3 signal across gene bodies.

The ENCODE ChIP-seq data utilized in this study were all generated from the K562 cell line, which is a myelogenous leukemia line derived from a 53 year old female CML (chronic myelogenous leukemia) patient (52). We downloaded ChIP-seq data of H3K36me3 and 40 transcription factors including RNA polymerase II (Pol2) from the UCSC Genome Browser (See Table 3.9 for the list of bam files). All downloaded files correspond to untreated samples with reads mapped to human genome build hg19. The H3K36me3 data have $\sim$25 million reads. To normalize for read-depth differences, we

randomly down-sampled each TF dataset to approximately 10 million reads. Then for each dataset we counted the number of reads in 250 bp non-overlapping windows in chromosome 19 similar to (67), resulting in approximately $n = 220,000$ windows per sample. For each window i ($i = 1, ..., n$), we had window read counts y_i corresponding to the H3K36me3 data and window read counts X_{il}, $l = 1, .., 40$ from each of the $p = 40$ TF ChIP-seq datasets.

Utilizing our penalized AR-HMM with log penalty, we first seek to select covariates related with y_i from 47 variables that include variables for the 40 TFs and 7 possible confounding effects: mappability, GC content, input control, and their 2-way or 3-way interactions. In the following discussion, we omit the variable selection results for the confounding effects since they are not of biological interest. To avoid the over-dispersed nature of count data, we test three transformations of the TF count data: $\log(X_{il}+1)$, $I(X_{il} > q_{X_{il},90})$, and $I(X_{il} > q_{X_{il},95})$, where $I()$ is an indicator function and $q_{X_{il},\alpha}$ indicates the α percentile of X_{il}. The thresholding of the window read counts of a TF serves to be a binary approximation of a TF binding event. The variable selection and model parameter estimation results for three transformations are similar (Figure 3.8, Table 3.10) and thus we only summarize the results from transformation $I(X_{il} > q_{X_{il},95})$.

As shown in Figure 3.9, the TFs selected in the background state and the enriched state have some similarities. In both states, the TF with strongest association with H3K36me3 is RNA Polymerase II (Pol2), which is expected given the involvement of H3K36me3 in transcriptional elongation. The TF with the next strongest association with H3K36me3 in both states is ZBTB7, which has been shown to interact with histone deacetylase-1 (48), and thus our results imply the possibility of interplay between histone methylation and acetylation. ZBTB7 is also related with leukemia, where it is

also known as Leukemia/lymphoma-related factor. Given H3K36me3's known association with leukemia, it would be interesting to study whether the association between H3K36me3 and ZBTB7 is specific in leukemia cell lines. In addition, BRG1 binding is positively related with H3K36me3 signals in the background but not the enrichment state, in line with its known role of the selective remodeling of chromatin structure outside of genes to aid in the recruitment of transcription factors (69). Other factors exhibit weaker effects which may suggest less frequent interactions associated with local H3K36me3 deposition or background signal.

We would like to clarify that the above analysis is different from more commonly used analyses to examine the marginal correlation between H3K36me3 and an individual TF in three aspects. First, we assess associations within H3K36me3-enriched and background regions separately, instead of performing genome-wide association. Second, these associations are conditioned on the presence of all other TFs present in the model, which may be different from marginal associations. For example, a TF may modify H3K36me3 through Pol2 regulation, and thus marginally associated with H3K36me3. However such association may be attenuated given Pol2 signals. Third, we examine the association of H3K36me3 signals and TF bindings within the same window whereas previous studies often examine TF binding at gene promoters.

While it is not unreasonable to expect that certain DNA-protein binding events may directly affect local H3K36me3 deposition or background signal, another biologically interesting situation is to examine the association between TF binding at promoters and H3K36me3 at downstream genes. Since H3K36me3 typically covers gene bodies of actively transcribed genes, in this setup H3K36me3-enriched and background regions would arise from those genes with high transcriptional activity vs. those with no or low transcriptional activity. We have conducted such an analysis to focus on H3K36me3 signals along gene bodies, adjusting for confounding factors as in the previous study

but now defining the TF covariate for an entire gene as a binary variable indicating promoter region binding of that particular TF for that particular gene (See Appendix, Section AII.1) for the details of data preparation).

We applied our penalized AR-HMM with log penalty to this data. Since two adjacent genes may be far apart in the genome, we reset the autoregressive covariate to be 0 at the beginning of each new gene in the data matrix to avoid unjustified autoregressive effects. Similar to previous study, we found H3K36me3 is negatively associated with ZBTB7 binding, and positively associated with Pol2 in both H3K36me3-enriched and background regions (Figure 3.10). In contrast, some TFs show different effects in these two analyses. For example, in previous analysis, cMYC binding is not associated with H3K36me3 in enriched regions and is negatively correlated with H3K36me3 in background regions (Figure 3.9). However cMYC binding in promoters show strong positive effects on H3K36me3 in both H3K36me3-enriched and background regions, in line with its role as a transcriptional activator (22).

In summary, we find the occupancy of multiple TFs are associated with H3K36me3 signatures and such associations may vary between H3K36me3 -enriched regions and background regions. The functions of these TFs, together with the involvement of H3K36me3 in cancer-related processes imply interesting connections between chromatin modification and tumorigenesis, a theme that is attracting increasing interest recently (77).

3.6 Conclusion

We have proposed and implemented two novel strategies for DAE-seq data analysis: to account for dependency of DAE-seq data from adjacent genomic loci using HMM/AR-HMM with covariates, and to conduct variable selection in the setup of HMM or AR-HMM. Our simulation and real data analysis results suggest an existing approach of Finite Mixture Regression (FMR) model is sufficient for DAE-seq data

where signal-to-noise ratio is high and the enriched regions are short. When the enriched regions are longer, HMM and AR-HMM show advantages. When there are autocorrelations between adjacent windows (which is a natural consequence of using overlapping sliding windows) given hidden state, AR-HMM performs better than the other methods. We show that even if the true model is FMR, both HMM and AR-HMM perform well. In addition, some DAE-seq data may have a mixture of two types of patterns: sharp peaks and segmental low-signal enrichments. Therefore applying AR-HMM is much more convenient for real data analysis. We applied our variable selection method in a chromosome-wide analysis and a gene-centered analysis. This type of study can be conducted in genome-wide scale or more focused regions such as the genes belonging to the same pathway. The response variable can be other quantitative features such as open chromatin regions captured by DNase-seq (78).

We have implemented our methods in an R package that can be downloaded from `http://code.google.com/p/hmmcov/`. Our software implementation is computationally efficient. For example, in our real data analysis for CTCF or H3K36me3, to analyze $\sim$140,000 non-overlapping windows spanning Chr22, it takes less than 120/180 seconds for HMM and AR-HMM, respectively; and to analyze $\sim$280,000 overlapping windows spanning Chr22, it takes less than 220/540 seconds for HMM and AR-HMM, respectively. For the real data analysis of H3K36me3 signals versus 47 covariates (40 TFs + 7 confounding factors) at Chr19, the total computational time is less than 4 hours with 25 tuning parameter combinations.

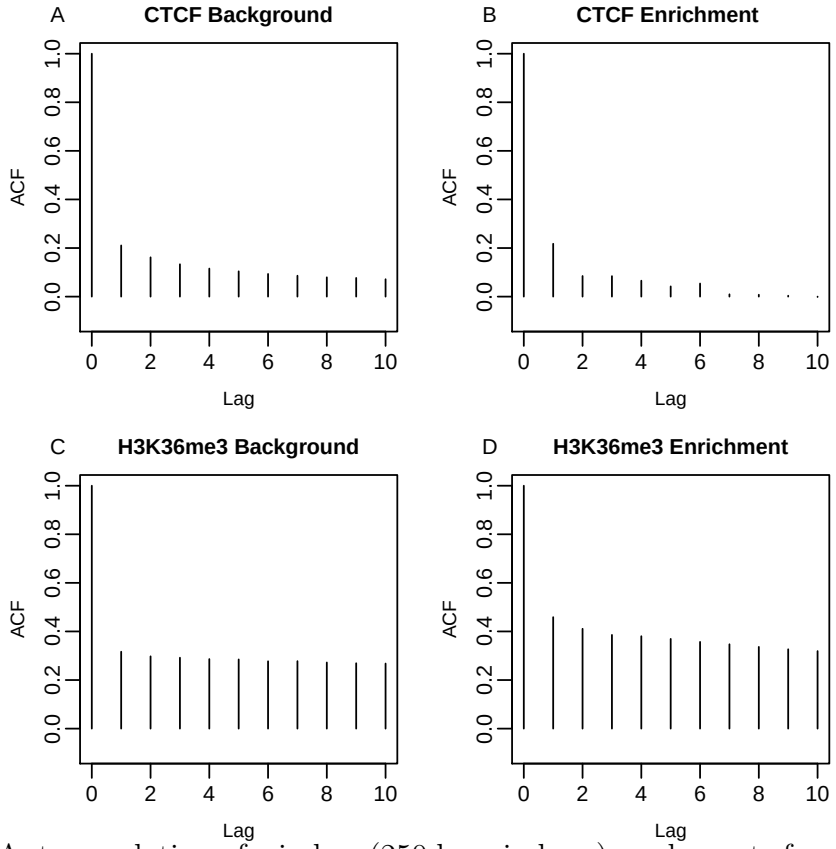

Figure 3.1: Autocorrelation of window (250 bp windows) read counts from background (A,C) and enriched regions (B,D) of CTCF (CCCTC- binding factor) ChIP-seq (A,B) and H3K36me3 (Trimethylation of Lys36 in histone H3) ChIP-seq (C,D) dataset measured in Chr22 of a human cell line (GM12878). Window read counts were log transformed. Likely enriched regions were determined by fitting a two-component Negative Binomial Finite Mixture Model and regions classified to be enriched at an FDR threshold of 0.05.

Table 3.1: Two-state Poisson HMM variable selection simulation setup for regression coefficients corresponding to states 1 and 2 (background vs. enriched).

	β_1	β_2	β_3	β_4	β_5
State 1	2	0	0	0	0
State 2	0	2	2	2	0

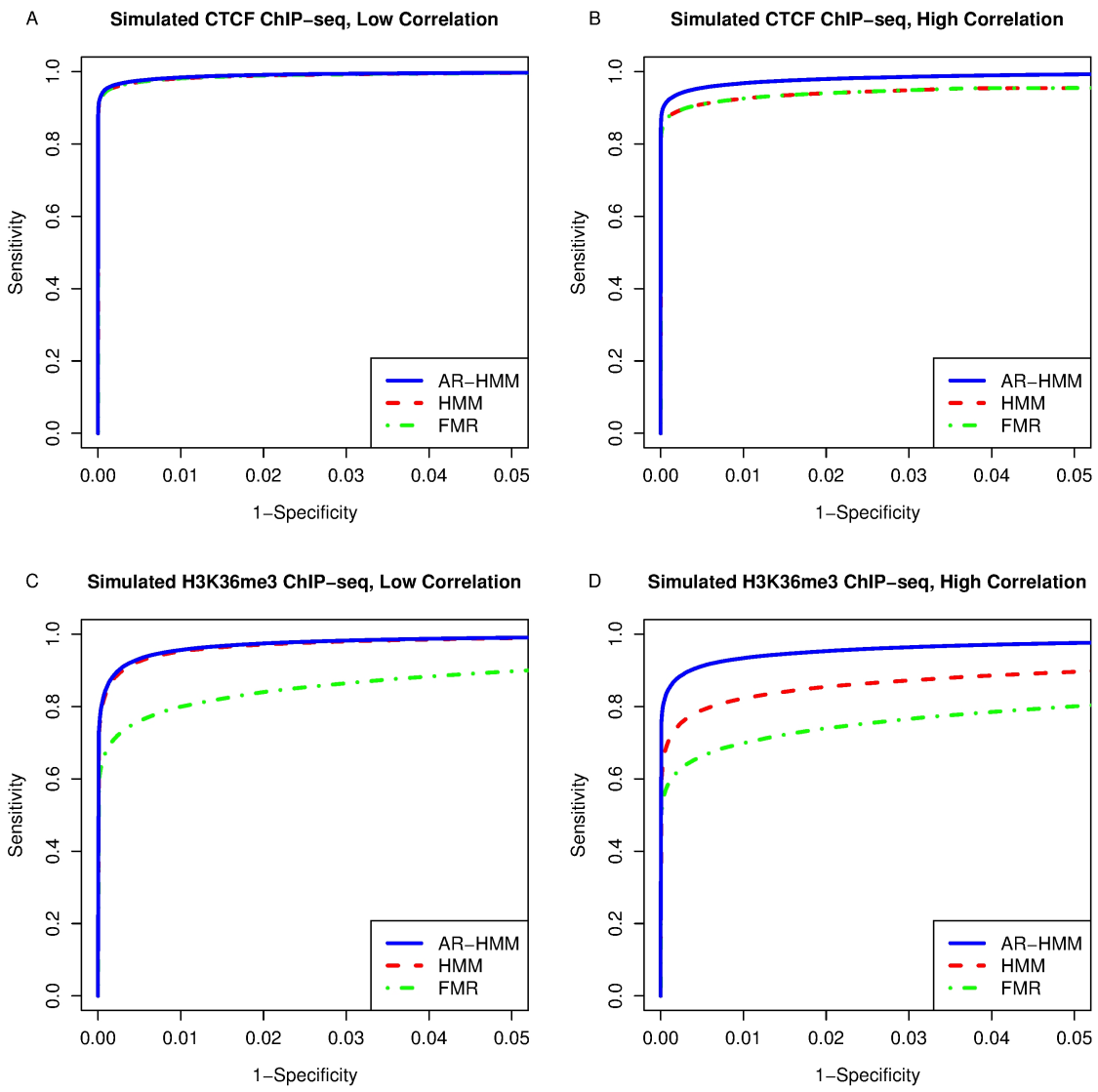

Figure 3.2: c Comparison of classification performance (at window level) of the AR-HMM, HMM, and FMR for $\nu_1 = 0.2$ and $\nu_2 = 0.2$ (first column, low auto-correlation in enriched regions) and $\nu_1 = 0.2$ and $\nu_2 = 0.8$ (second column, high auto-correlation in enriched regions). The first row are the results for CTCF-style ChIP-seq data with short enriched regions, and the second row are the results for H3K36me3 histone modification-style data with longer regions of enrichment.

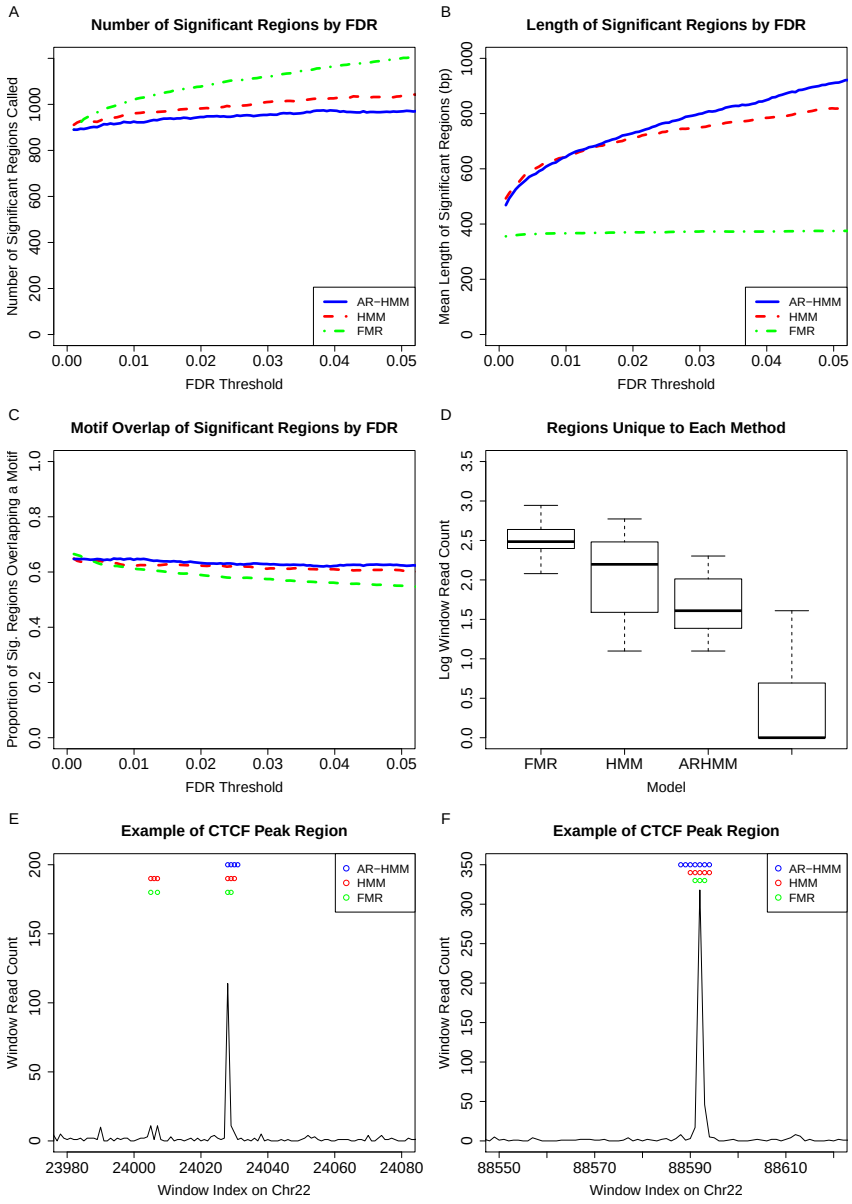

Figure 3.3: Classification performance comparison of the FMR, HMM, and AR-HMM models in GM12878 CTCF ChIP-seq. A) Number of significantly enriched regions (which are generated by collapsing adjacent significant windows) called by each method across FDR thresholds. B) Average length of significant regions across FDR thresholds. C) Classification performance relative to FDR threshold, where regions overlapping a CTCF binding motif are classified as "correct". D) Box plots of maximum window read counts from significant regions called uniquely by each method (the first three box plots) and box plot of maximum window read count from background called by all three methods (the last box plot). E-F) Examples of enriched regions called by each method at FDR level 0.05.

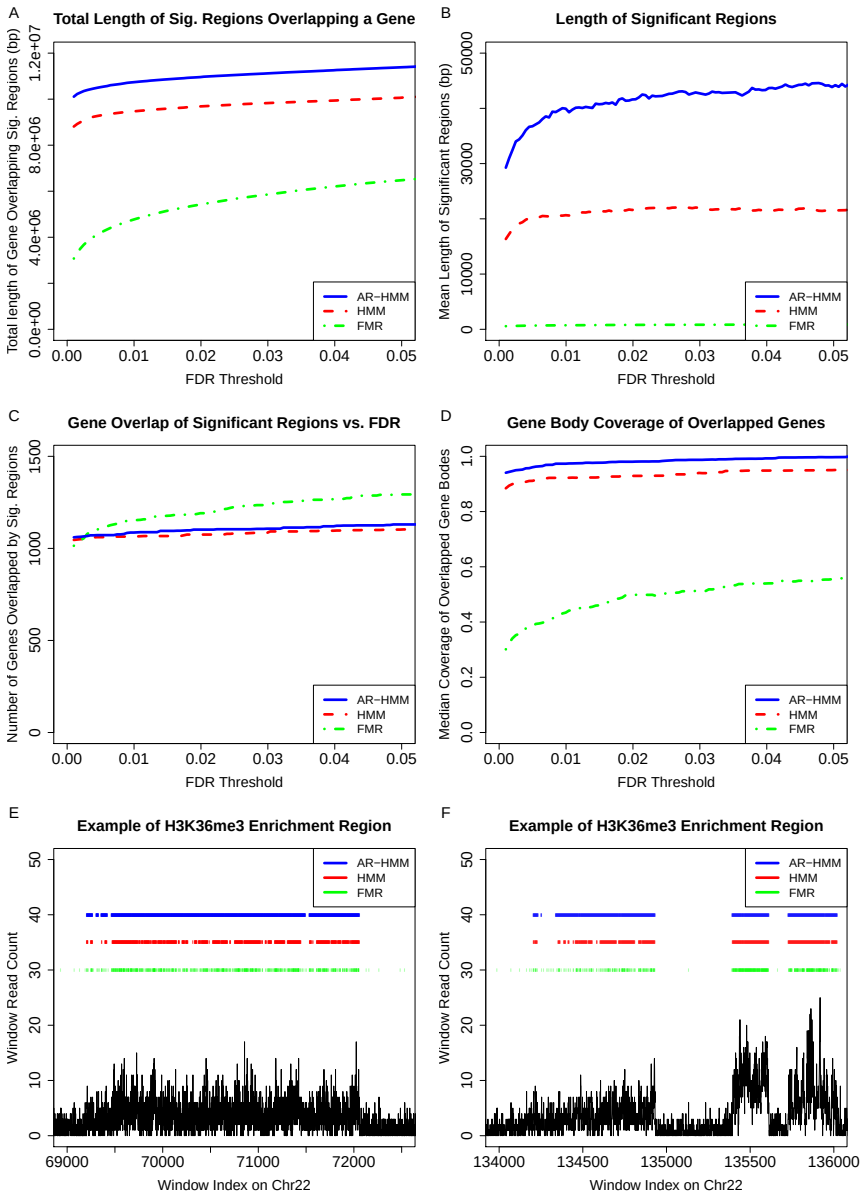

Figure 3.4: Classification performance comparison of the FMR, HMM, and AR-HMM models in GM12878 H3K36me3 ChIP-seq data. A) The total length of significantly enriched regions (which are generated by collapsing adjacent significant windows) that overlap a gene body across FDR thresholds. B) Average lengths of significant regions across FDR thresholds. C) Number of genes overlapped with significant regions across FDR thresholds. D) Median proportion of gene bodies covered by significantly enriched regions. E-F) Examples of regions called as enriched by each method.

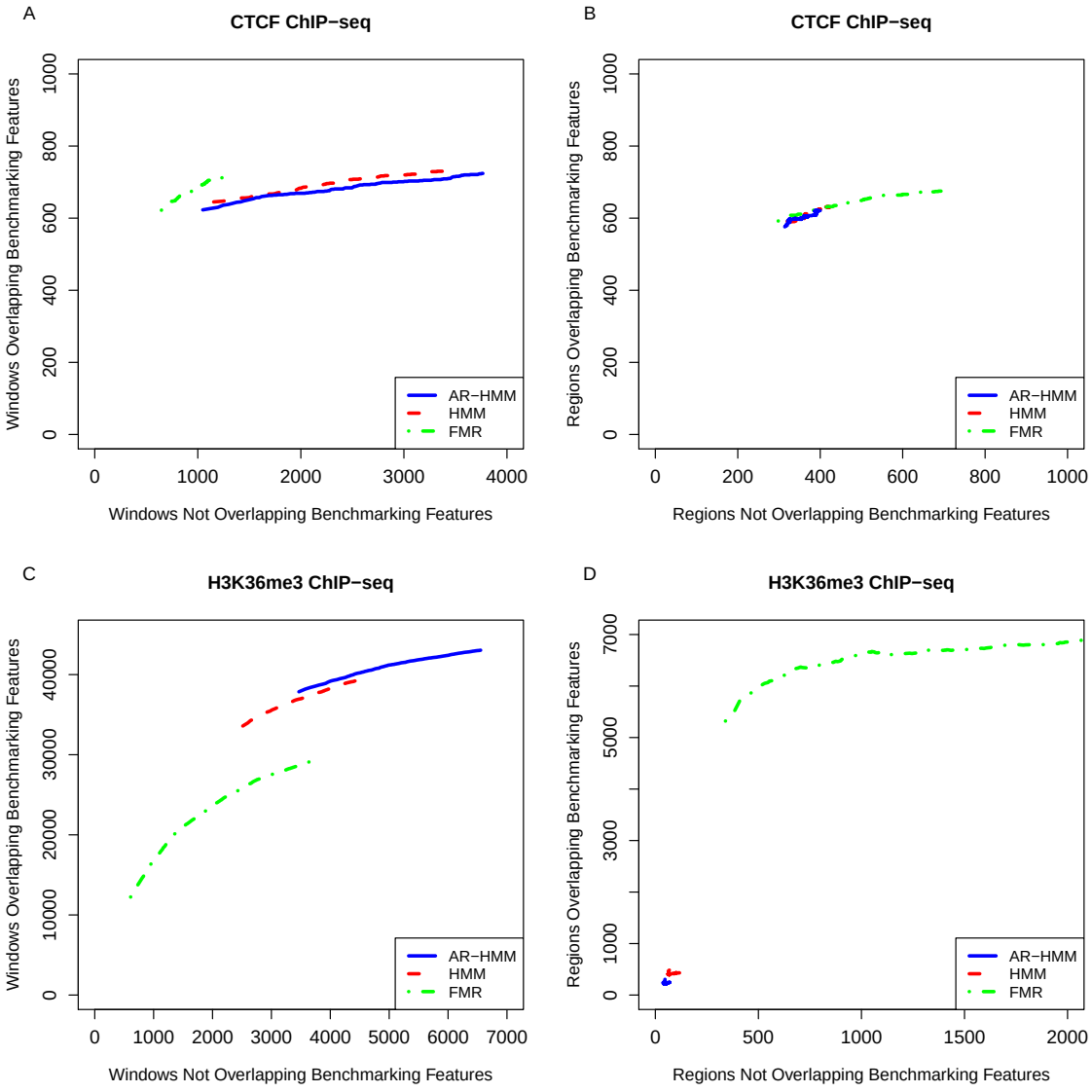

Figure 3.5: Comparison of the performances of the FMR, HMM, and AR-HMM in GM12878 CTCF ChIP-seq and H3K36me3 ChIP-seq. X-axes are the number of significant windows/regions that do not overlap with benchmarking features at various FDR thresholds (binding motif for CTCF and gene bodies for H3K36me3). Y-axes are the number of windows/regions that do overlap with benchmarking features.

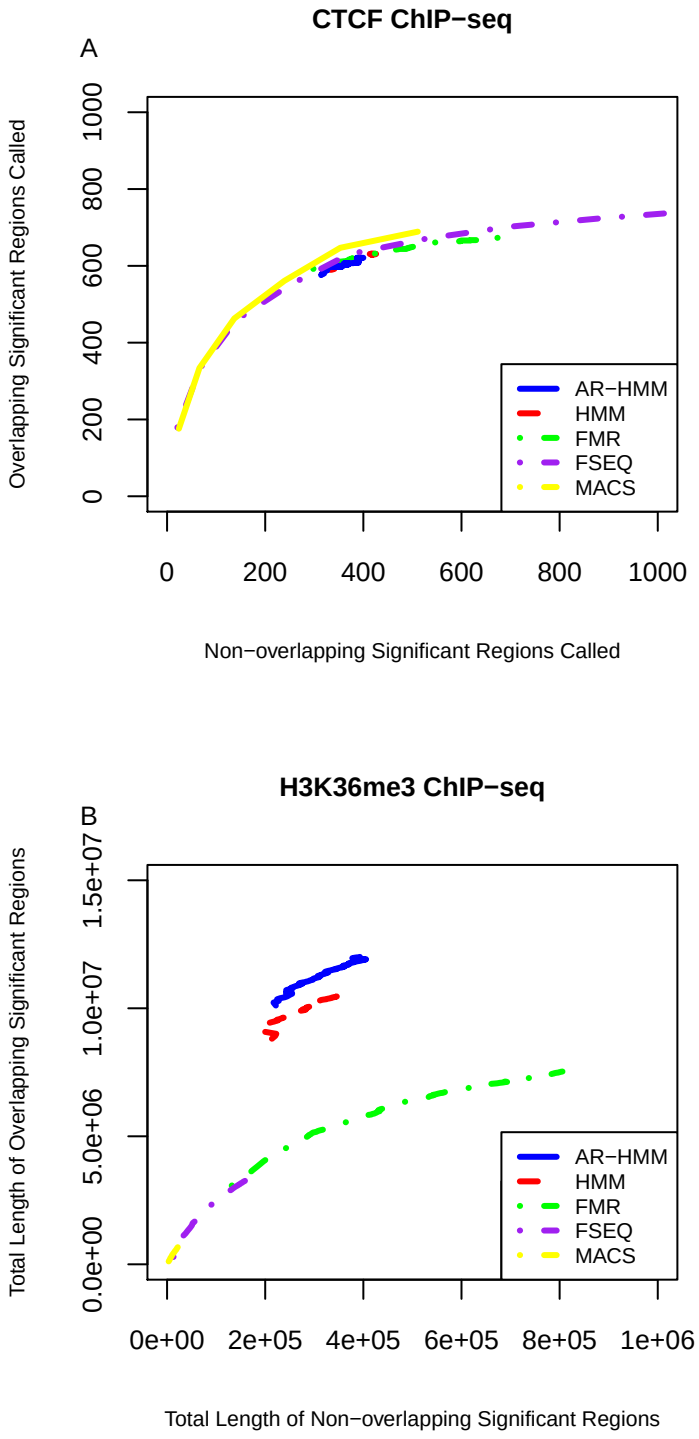

Figure 3.6: Comparing the HMM and AR-HMM relative performance with common Peak Callers MACS and FSEQ. All algorithms perform similarly in high signal to noise CTCF ChIP-seq data. The HMM and AR-HMM perform significantly better in broader H3K36me3 ChIP-seq data.

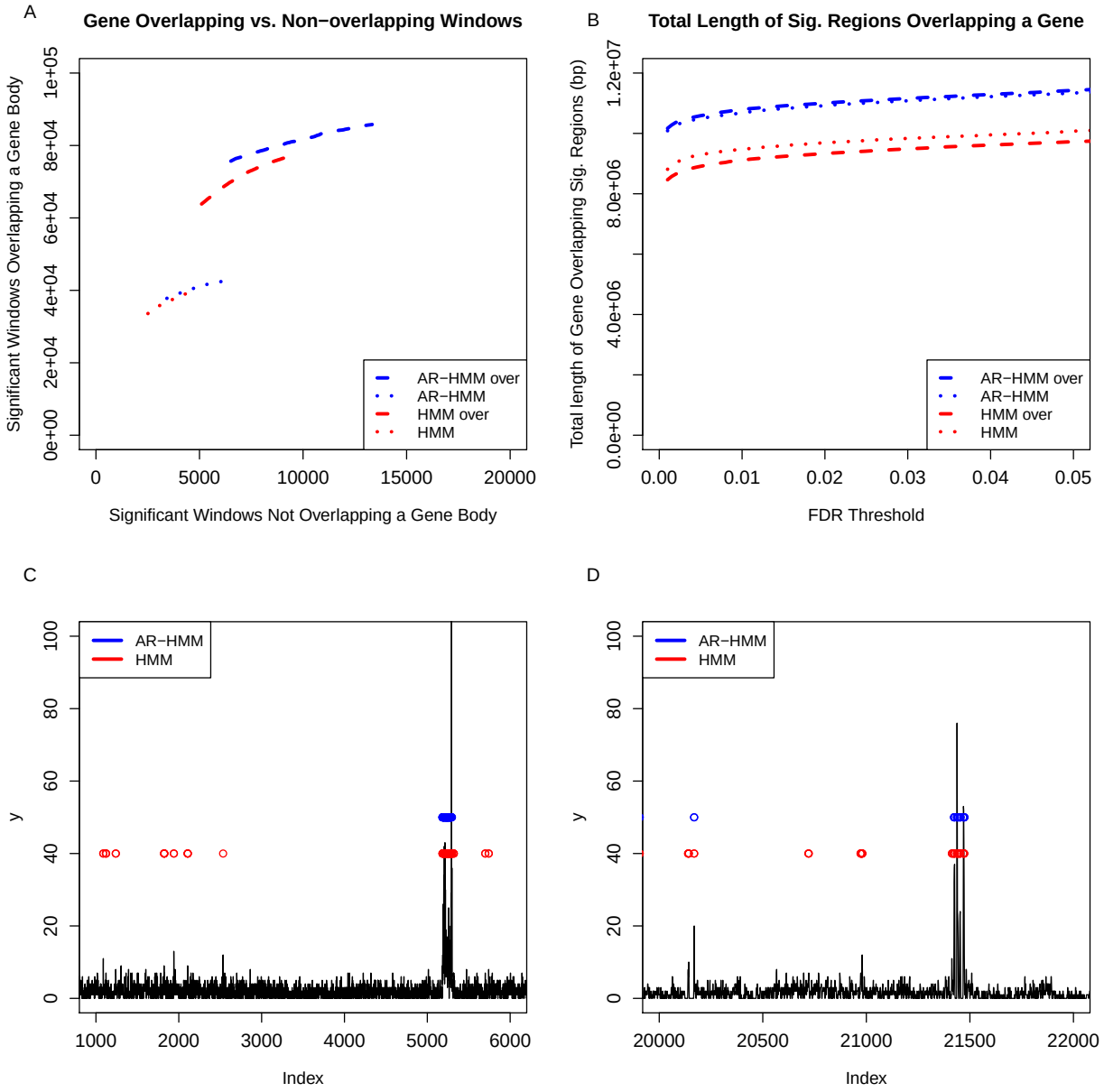

Figure 3.7: Comparing the performance of the HMM and AR-HMM for read counts data from overlapping and non-overlapping windows in the H3K36me3 ChIP-seq dataset. (A) The sensitivity/specificity gains of the AR-HMM over the HMM are larger when analyzing data of overlapping windows. The legend "method over" indicates the method applied to overlapping windows. (B) The performance of the AR-HMM is similar for overlapping or non-overlapping windows, however the the HMM performs worse for overlapping windows. (C), (D) examples where the HMM calls are likely false positives on data from overlapping windows.

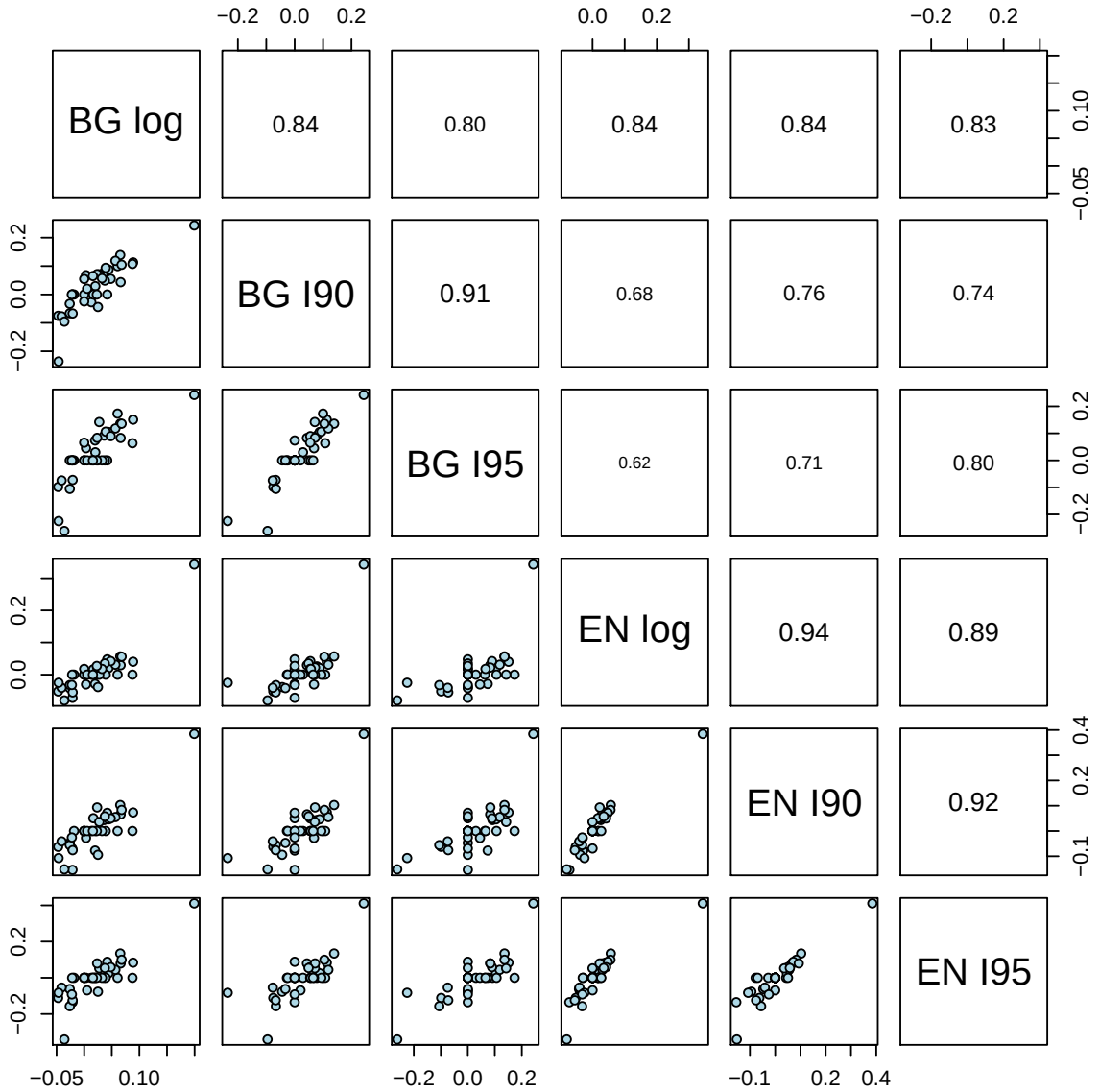

Figure 3.8: The penalized coefficient estimates for background (BG) state or enrichment (EN) state. The upper panel shows correlations and the lower panel shows scatter plots. For reach state, we have three sets of results corresponding to three transformations of the covariates: $\log(X_{il}+1)$ (log), $I(X_{il} > q_{X_{il},90})$ (I90), and $I(X_{il} > q_{X_{il},95})$ (I95), where $I()$ is an indicator function and $q_{X_{il},\alpha}$ indicates the α percentile of X_{il}.

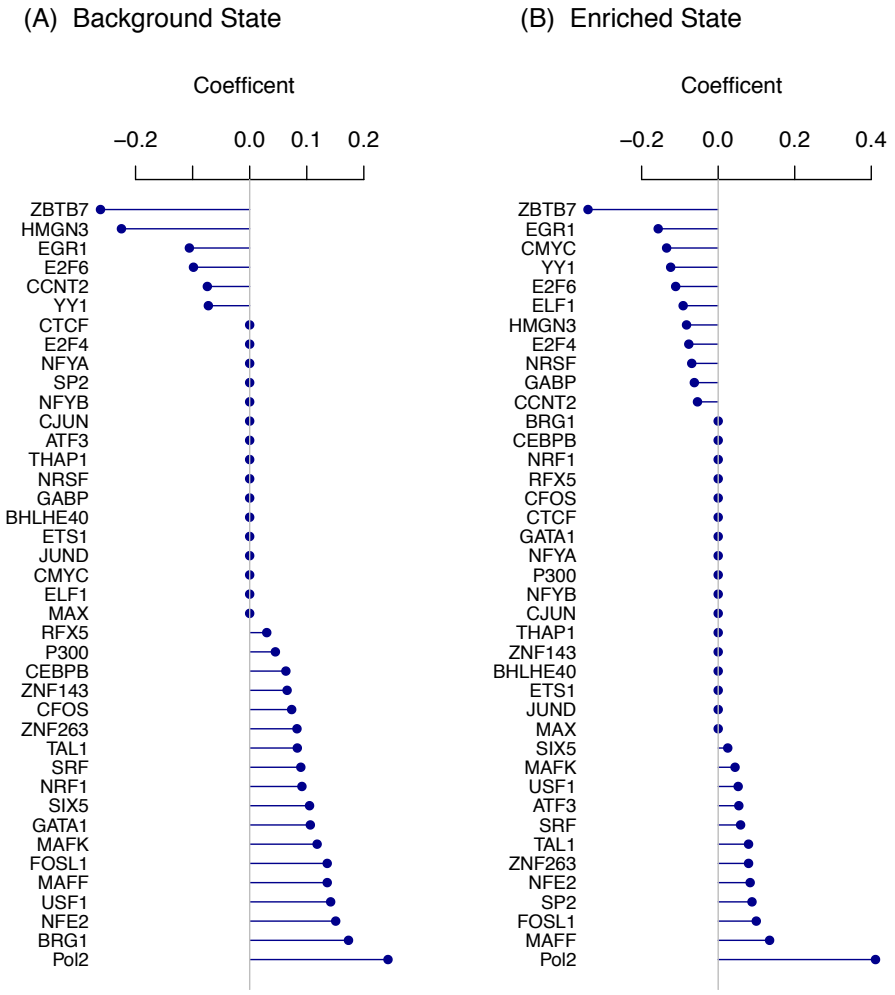

Figure 3.9: The variable selection results for background state and enriched state of H3K36me3. Each variable represent the binding signals of a transcription factor (TF).

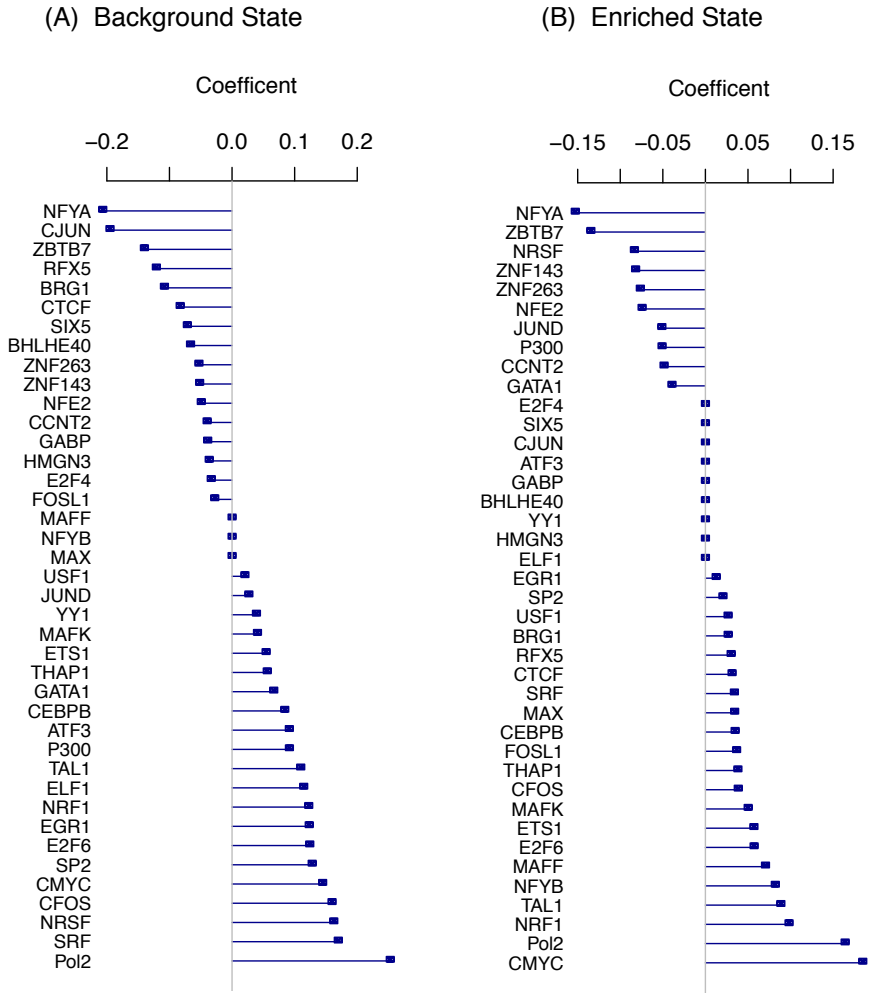

Figure 3.10: The variable selection results for background state and enriched state of H3K36me3. Each variable represent the present of binding a transcription factor (TF) in the promoter regions of UCSC genes.

Table 3.2: Variable selection performance in simulated two-state Poisson HMM with covariates over a range of conditions. p: simulated number of covariates per state; TD: average number of "True Discoveries" (equal to 4); FD: average number of "False Discoveries"; the remaining columns are the mean estimates of the true non-zero coefficients in the model from Table 3.1.

Type	p	n	Pen	TD(4)	FD(0)	β_{11}	β_{22}	β_{23}	β_{24}	γ_{11}	γ_{12}	γ_{21}	γ_{22}
CTCF	5	200	lasso	3.99	1.81	1.79	1.88	1.93	1.91	0.894	0.106	0.891	0.109
			log	3.91	0.33	1.93	1.93	1.96	1.93	0.896	0.104	0.904	0.0956
			scad	4	0.43	2	2.01	2	1.98	0.899	0.101	0.903	0.0968
		10000	lasso	4	0.99	1.96	1.98	1.97	1.98	0.901	0.0994	0.901	0.0994
			log	4	0.14	1.99	2	2	2	0.899	0.101	0.9	0.0997
			scad	4	0.18	2	2	2	2	0.9	0.0998	0.901	0.0989
CTCF	100	200	lasso	2.94	14.00	1.46	0.705	0.585	0.572	0.911	0.0887	0.913	0.0873
			log	3.01	7.53	1.9	1.13	0.937	0.836	0.909	0.0905	0.917	0.0829
			scad	3.1	8.44	1.92	1.55	1.24	1.05	0.89	0.11	0.9	0.0998
		10000	lasso	4	1.78	1.92	1.92	1.93	1.92	0.903	0.0973	0.902	0.0977
			log	4	0.43	2	2	2	2	0.9	0.0999	0.9	0.0997
			scad	4	0.18	2	2	2	2	0.899	0.101	0.901	0.0988
Histone	5	200	lasso	4	1.55	1.71	1.95	1.96	1.94	0.897	0.103	0.117	0.883
			log	3.97	0.15	1.92	1.99	1.99	1.99	0.894	0.106	0.117	0.883
			scad	4	0.14	1.97	1.99	1.99	2.01	0.9	0.0999	0.113	0.887
		10000	lasso	4	0.56	1.95	1.99	1.99	1.99	0.9	0.0998	0.0999	0.9
			log	4	0.11	2	2	2	2	0.9	0.1	0.101	0.899
			scad	4	0.03	2	2	2	2	0.9	0.0998	0.101	0.899
Histone	100	200	lasso	4	7.05	1.38	1.78	1.77	1.78	0.894	0.106	0.106	0.894
			log	3.98	2.98	1.95	1.98	1.99	1.95	0.9	0.0995	0.111	0.889
			scad	4	4.94	1.99	2.01	1.99	2	0.903	0.0967	0.106	0.894
		10000	lasso	4	0.95	1.91	1.97	1.97	1.97	0.9	0.1	0.1	0.9
			log	4	0.45	2	2	2	2	0.9	0.0998	0.101	0.899
			scad	4	0.04	2.01	2	2	2	0.9	0.1	0.1	0.9

Table 3.3: Mean parameter estimates (of 1000 simulations) for the AR-HMM, HMM, and FMR models applied to simulated two-state Poisson AR-HMM ChIP-seq data of CTCF binding sites (CTCF) and H3K36me3 histone modifications (Histone) over various conditions. The number of windows $n = 10,000$.

Type	$\nu_{1,r}$	$\nu_{2,r}$	Model	β_{10}	β_{11}	β_{21}	ν_1	β_{02}	β_{12}	β_{22}	ν_2	γ_{11}	γ_{12}	γ_{21}	γ_{22}
CTCF	0.2	0.2	ARH	0	1	0.999	0.2	1.5	2	2	0.2	0.9	0.1	0.9	0.1
	0.2	0.2	HMM	-0.017	0.999	0.999		1.484	1.998	1.999		0.9	0.1	0.9	0.1
	0.2	0.2	FMR	-0.017	0.999	0.999		1.484	1.998	1.999					
	0.2	0.8	ARH	-0.001	1	1.001	0.2	1.5	2	2	0.8	0.9	0.1	0.9	0.1
	0.2	0.8	HMM	0.011	0.985	0.985		1.573	1.939	1.942		0.907	0.093	0.903	0.097
	0.2	0.8	FMR	0.011	0.985	0.985		1.573	1.939	1.942					
Histone	0.2	0.2	ARH	0.001	0.999	0.999	0.199	0.5	1.999	2	0.2	0.9	0.1	0.101	0.899
	0.2	0.2	HMM	-0.026	1.006	1.006		0.497	1.995	1.996		0.9	0.1	0.101	0.899
	0.2	0.2	FMR	-0.021	1.004	1.002		0.495	1.996	1.997					
	0.2	0.8	ARH	-0.002	1.001	1.002	0.199	0.5	2	2	0.8	0.9	0.1	0.1	0.9
	0.2	0.8	HMM	-0.052	1.102	1.104		0.629	1.921	1.922		0.902	0.098	0.129	0.871
	0.2	0.8	FMR	0.021	1.051	1.053		0.655	1.904	1.905					

Table 3.4: Variable selection performance and estimation accuracy for true non-zero coefficients in simulated two-state Poisson AR-HMM with covariates data over a range of conditions and $\nu_1 = \nu_2 = 0.4$

Type	p	n	Pen	TD(4)	FD(0)	β_{11}	ν_1	β_{22}	β_{23}	β_{24}	ν_2	γ_{11}	γ_{21}	γ_{12}	γ_{22}
CTCF	5	200	lasso	3.97	2.06	1.79	0.367	1.84	1.88	1.81	0.189	0.909	0.0909	0.92	0.0799
			log	3.86	0.43	1.96	0.367	1.89	1.85	1.91	0.227	0.902	0.0982	0.927	0.0729
			scad	3.99	0.58	1.97	0.386	1.97	2.06	1.95	0.11	0.902	0.0979	0.911	0.0894
		10000	lasso	4	1.18	1.96	0.396	1.98	1.98	1.98	0.388	0.901	0.0994	0.898	0.102
			log	4	0.05	2	0.393	2	2	2	0.386	0.9	0.1	0.901	0.0989
			scad	4	0.03	2	0.391	2	2	2	0.388	0.9	0.0997	0.899	0.101
	100	200	lasso	2.59	17.11	1.194	0.352	0.468	0.47	0.418	0.218	0.865	0.135	0.877	0.123
			log	2.69	9.7	1.657	0.355	0.936	0.832	0.917	0.034	0.864	0.136	0.876	0.124
			scad	3.32	12.11	1.315	0.342	1.519	1.525	1.315	-0.049	0.85	0.15	0.861	0.139
		10000	lasso	4	2.49	1.931	0.403	1.929	1.928	1.932	0.402	0.902	0.098	0.9	0.1
			log	4	0	1.976	0.402	1.996	1.996	1.995	0.401	0.9	0.1	0.9	0.1
			scad	4	0.23	1.994	0.401	2	2	2.003	0.401	0.9	0.1	0.9	0.1
Histone	5	200	lasso	4	1.8	1.72	0.387	1.97	1.96	1.94	0.395	0.893	0.107	0.113	0.887
			log	3.97	0.31	1.93	0.379	2	2	1.99	0.39	0.898	0.102	0.108	0.892
			scad	4	0.2	1.95	0.384	2.01	1.99	1.98	0.383	0.901	0.0987	0.103	0.897
		10000	lasso	4	0.72	1.96	0.401	1.99	1.99	1.99	0.401	0.9	0.0996	0.101	0.899
			log	4	0.1	2	0.401	2	2	2	0.401	0.9	0.1	0.101	0.899
			scad	4	0.03	2	0.4	2	2	2	0.4	0.901	0.0995	0.1	0.9
	100	200	lasso	4	19.07	1.33	0.367	1.812	1.829	1.799	0.386	0.896	0.104	0.114	0.886
			log	4	5.57	1.856	0.382	1.978	1.957	1.973	0.389	0.908	0.092	0.107	0.893
			scad	4	16.48	1.847	0.344	1.991	1.98	1.986	0.337	0.897	0.103	0.117	0.883
		10000	lasso	4	1.93	1.909	0.407	1.974	1.973	1.974	0.404	0.901	0.099	0.101	0.899
			log	4	0.03	1.986	0.4	1.985	1.985	1.985	0.401	0.901	0.099	0.101	0.899
			scad	4	0.2	1.999	0.397	2.001	2.001	2.001	0.402	0.9	0.1	0.1	0.9

Table 3.5: GM12878 CTCF and H3K36me3 ChIP-seq Chr22 two-state Negative Binomial HMM and AR-HMM real data variable selection results. $\beta_{0,k}$ is the intercept in state k, $\beta_{1,k}$ corresponds to the G/C content main effect, $\beta_{2,k}$ corresponds to the mappability main effect, $\beta_{3,k}$ corresponds to the input control main effect. Interaction terms are denoted with combination of indices, for example $\beta_{12,k}$ corresponds to G/C content-mappability interaction term.

Data	CTCF	CTCF	Histone	Histone
Model	HMM	ARH	HMM	ARH
BIC	386889	384512	545036	532506
$\beta_{0,1}$	-2.273	-2.318	0.270	0.179
$\beta_{1,1}$	1.611	1.613	1.900	2.204
$\beta_{2,1}$	1.277	1.387	0.241	0.325
$\beta_{3,1}$			0.072	0.0617
$\beta_{12,1}$			-2.707	-3.039
$\beta_{13,1}$				
$\beta_{23,1}$				
$\beta_{123,1}$	0.628	0.57	0.256	0.245
ν_1		0.283		0.161
$\beta_{0,2}$	-2.557	-3.171	1.315	1.071
$\beta_{1,2}$	5.91	6.987		0.521
$\beta_{2,2}$	2.123	2.201	0.545	0.734
$\beta_{3,2}$	0.597	0.557	0.235	0.193
$\beta_{12,2}$			0.377	
$\beta_{13,2}$	0.002			
$\beta_{23,2}$				
$\beta_{123,2}$			-0.297	-0.258
ν_2		0.749		0.583

Table 3.6: Proportion of significant regions from each method (columns) that overlap with peaks from other methods (FDR=0.05). Cells corresponding to the same method are those that unique only to that method. For example, 92% and 88% of the 1180 significant FMR regions (Column 1, Rows 2 and 3) overlap with the HMM and AR-HMM, respectively.

	FMR	HMM	ARHMM
FMR	1	0.95	0.97
HMM	0.92	1	0.96
ARH	0.88	0.93	1
Regions	1180	1039	962

Table 3.7: Chr22 GM12878 CTCF and H3K36me3 ChIP-seq penalized model estimates - transition probabilities and dispersion parameters

Data	Type	γ_{11}	γ_{12}	γ_{21}	γ_{22}	ϕ_1	ϕ_2
CTCF	ARH	0.988	0.012	0.183	0.817	1.095	0.637
CTCF	HMM	0.989	0.011	0.255	0.745	1.061	0.563
CTCF	FMR					1.136	0.105
H3K36me3	ARH	0.997	0.003	0.006	0.994	5.567	6.748
H3K36me3	HMM	0.995	0.005	0.012	0.988	4.752	4.874
H3K36me3	FMR					4.656	3.779

Table 3.8: Mean parameter estimates for the AR-HMM, HMM, and FMR models applied to simulated two-component Poisson FMR ChIP-seq data of CTCF binding sites (CTCF) (1000 simulations, $n = 10,000$).

Model	β_{10}	β_{11}	β_{21}	ν_1	β_{02}	β_{12}	β_{22}	ν_2	γ_{11}	γ_{12}	γ_{21}	γ_{22}
FMR	0	1	0.999		1.499	2	2.001					
HMM	0	1	0.999		1.499	2	2.001		0.9	0.1	0.9	0.1
ARHMM	0	1	1	0.001	1.499	2	2.001	0	0.9	0.1	0.9	0.1

Table 3.9: The bam files used in the study of "The Relation Between Histone Modification H3K36me3 and Transcription Factor Occupancy". All the bam files were downloaded from `http://hgdownload.cse.ucsc.edu/goldenPath/hg19/encodeDCC/`.

File Name
H3K36me3
wgEncodeBroadHistone/wgEncodeBroadHistoneK562H3k36me3StdAlnRep1.bam
wgEncodeBroadHistone/wgEncodeBroadHistoneK562H3k36me3StdAlnRep2.bam
Input Control
wgEncodeBroadHistone/wgEncodeBroadHistoneK562ControlStdAlnRep1.bam
Pol2
wgEncodeBroadHistone/wgEncodeBroadHistoneK562Pol2bStdAlnRep1.bam
wgEncodeBroadHistone/wgEncodeBroadHistoneK562Pol2bStdAlnRep2.bam
Transcription Factors
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Atf3V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Atf3V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Cebpbsc150V0422111AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Cebpbsc150V0422111AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562CtcfcPcr1xAlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562CtcfcPcr1xAlnRep1V2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562CtcfcPcr1xAlnRep2V2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562E2f6sc22823V0416102AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562E2f6sc22823V0416102AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Egr1V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Egr1V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Elf1sc631V0416102AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Elf1sc631V0416102AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Ets1V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Ets1V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Fosl1sc183V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Fosl1sc183V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562GabpV0416101AlnRep1.bam

Continued on next page

Table 3.9 – *Continued from previous page*

File Name
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562GabpV0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562MaxV0416102AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562MaxV0416102AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562NrsfV0416102AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562NrsfV0416102AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Six5V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Six5V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Sp2sc643V0416102AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Sp2sc643V0416102AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562SrfV0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562SrfV0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Thap1sc98174V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Thap1sc98174V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Usf1V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Usf1V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Yy1sc281V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Yy1sc281V0416101AlnRep2.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Zbtb7asc34508V0416101AlnRep1.bam
wgEncodeHaibTfbs/wgEncodeHaibTfbsK562Zbtb7asc34508V0416101AlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Bhlhe40nb100IggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Bhlhe40nb100IggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Brg1IggmusAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Brg1IggmusAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Ccnt2StdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Ccnt2StdAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562CfosStdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562CfosStdAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562CfosStdAlnRep3.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562CjunIggrabAlnRep1.bam

Continued on next page

Table 3.9 – *Continued from previous page*

File Name
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562CjunIggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562CmycIggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562CmycIggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562E2f4UcdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562E2f4UcdAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Gata1bIggmusAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Gata1bIggmusAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Hmgn3StdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Hmgn3StdAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562JundIggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562JundIggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562MaffIggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562MaffIggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Mafkab50322IggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Mafkab50322IggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Nfe2StdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Nfe2StdAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562NfyaStdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562NfyaStdAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562NfybStdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562NfybStdAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Nrf1IggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Nrf1IggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562P300IggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562P300IggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Rfx5IggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Rfx5IggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Tal1sc12984IggmusAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Tal1sc12984IggmusAlnRep2.bam

Continued on next page

Table 3.9 – *Continued from previous page*

File Name
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Znf143IggrabAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Znf143IggrabAlnRep2.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Znf263UcdAlnRep1.bam
wgEncodeSydhTfbs/wgEncodeSydhTfbsK562Znf263UcdAlnRep2.bam

Table 3.10: Parameter estimates of the penalized AR-HMM for three types of covariates transformations: $\log(X_{il}+1)$, $I(X_{il} > q_{X_{il},90})$, and $I(X_{il} > q_{X_{il},95})$, where $I()$ is an indicator function and $q_{X_{il},\alpha}$ indicates the α percentile of X_{il}. γ_{ij} $(1 \leq i, j \leq 2)$ are transition probabilities, ϕ_1 and ϕ_2 are dispersion parameters, and η_2 is the proportion of windows belonging to enriched state.

Covariate	γ_{11}	γ_{12}	γ_{21}	γ_{22}	ϕ_1	ϕ_2	η_2
$\log(X_{il}+1)$	0.9955	0.0045	0.0104	0.9896	3.2851	4.4308	0.2995
$I(X_{il} > q_{X_{il},90})$	0.9955	0.0045	0.0099	0.9901	3.1106	3.7457	0.3137
$I(X_{il} > q_{X_{il},95})$	0.9956	0.0044	0.0094	0.9906	3.0512	3.6682	0.3178

Chapter 4

An Integrative Study of Standard and Allele-specific Associations of DNA Polymorphisms, Gene Expression, and Epigenetic Features from High Throughput Sequencing Data

4.1 Introduction

Gene expression regulation is an essential biological process by which static genetic information gives rise to dynamic organismal phenotypes (35). Multiple epigenetic marks are involved in gene expression regulation, including DNase I hypersensitive sites (DHSs) (73), DNA methylation (20), and histone modifications (30). DHSs, which delineate open chromatin regions, are among the most well-studied epigenetic marks. DHSs often harbor regulatory DNA elements that can influence gene expression abundance (78), and as a result the presence of DHSs is often associated with variations in gene expression (14). Both gene expression abundance and DHSs are inheritable (55), and previous studies have found their variations are often associated with DNA variants such as single nucleotide polymorphisms (SNPs) (62, 13).

Gene expression and epigenetic marks are being routinely assessed by high-throughput sequencing solutions, where the resulting data are the number of sequenced reads within a certain genomic region. For example, the number of RNA-seq reads within a gene provides a measure of gene expression abundance, which can be further normalized by read depth (the total number of sequencing reads sampled per individual) and gene length to facilitate the comparison across individuals and across genes. Sequencing data not only provide more comprehensive and more accurate assessments of genomic activity, but also reveal novel information that are not available from traditional microarrays, such as allele-specific signals. In a diploid genome, two copies of DNA are inherited from the maternal and paternal genomes, respectively. Each copy of the DNA sequence at a genetic locus is referred to as an allele. Traditional microarrays typically measure the aggregate signals from both alleles, while sequencing data allow for the delineation of allele-specific signals. Recently, allele-specific signals have

been studied in various sequencing studies, including gene expression (62), DNA methylation (20), transcription factor binding (70), and chromatin accessibility (13). Such allele-specific signals can be used to distinguish *cis*-acting and *trans*-acting genetic effects (75). A *cis*-acting DNA polymorphism only modifies the gene expression or epigenetic marks that are located on the same haploid genome as the DNA polymorphism. In contrast, a *trans*-acting DNA polymorphism has the same effect on both alleles of its target. Therefore, an imbalance of allele-specific read counts (ASReCs) of the two alleles within one individual implies the presence of a *cis*-acting regulatory element, and the variation of the total read count (TReC) (summation of read count from either allele) across individuals can be due to either *cis*-acting or *trans*-acting regulations.

Previous studies have demonstrated the association between gene expression and epigenetic marks using either TReC or ASReC and their associations with DNA polymorphisms. However, no study has systematically assessed the associations between gene expression and epigenetic marks using both TReC and ASReC while accounting for possible shared genetic effects. To address this issue, we develop a novel statistical method, which we refer to as BASeG (Bivariate Aassociation studies using Sequencing data, while accounting for shared Genetic effects). Specifically, we study the association of TReC and ASReC using Bivariate Poisson-Log-Normal (BPLN) regression, and Bivariate Binomial-Logistic-Normal (BBLN) regression, respectively. We demonstrate BASeG's utility in simulations and a study of the association between gene expression (measured by RNA-seq) and DHSs (measured by DNase-seq). BASeG is general enough to be applied to study the associations between any two types of sequencing data, such as gene expression (by RNA-seq) vs. DNA methylation measured by bisulfite sequencing or histone modifications measured by ChIP-seq (Chromatin Immunoprecipitation followed by sequencing).

4.2 Methods

4.2.1 Bivariate Poisson-Lognormal Regression for Total Read Count

We first consider the statistical model for Total Read Count (TReC). Assume we are interested in the RNA-seq TReC of a particular gene, denoted by T_R, and the DNase-seq TReC within a particular genomic region (e.g., a 250-bp window in the promoter of the gene of interest), denoted by T_C. We assume the expected value of T_R is associated with a genetic variable Z_R and some other covariates X_R, and similarly, the expected value of T_C is associated with a genetic variable Z_C and some other covariates X_C. In this paper, we assume the effect of a genetic variable is additive and is defined

in terms of the number of copies of the non-reference allele. In other words, Z_R or Z_C equals 0, 1, or 2, which is the number of non-reference (alternative) alleles of the SNP. The determination of the reference versus alternative allele is with respect to the SNP annotation utilized in a study. In this paper, the reference allele of a SNP is defined by the 1000 Genomes Project SNP annotation file and this definition is applied consistently across samples. It is straightforward to define other types of genetic effects if desired. We model the joint distribution of T_R and T_C by a bivariate Poisson-log-normal (BPLN) distribution:

$$f_{\texttt{BPLN}}(T_R, T_C) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{\texttt{P}}(T_R;\mu_R) f_{\texttt{P}}(T_C;\mu_C)\phi(\epsilon_R,\epsilon_C;\Sigma_1)\, d\epsilon_{Ri} d\epsilon_{Ci}, \tag{4.1}$$

where $f_{\texttt{P}}(;\mu)$ denotes a Poisson distribution with mean value μ. For RNA-seq and DNase-seq data, we assume $\log(\mu_R) = X_R\beta_R + Z_R b_R + \epsilon_R$ and $\log(\mu_C) = X_C\beta_C + Z_C b_C + \epsilon_C$, respectively, where ϵ_R and ϵ_C are two random variables following a bivariate normal distribution with mean 0 and covariance Σ_1. We denote the joint density function of ϵ_R and ϵ_C by $\phi(\epsilon_R, \epsilon_C; \Sigma_1)$, where

$$\Sigma_1 = \begin{pmatrix} \sigma_R^2 & \rho_1\sigma_R\sigma_C \\ \rho_1\sigma_R\sigma_C & \sigma_C^2 \end{pmatrix}$$

and $-1 \le \rho_1 \le 1$ is a correlation parameter. Therefore in this BPLN distribution, the correlation between T_R and T_C is induced by the correlation ρ_1 between ϵ_R and ϵ_C.

The probability density function of (T_R, T_C) is obtained by integrating out the random effects ϵ_R and ϵ_C. To efficiently approximate this integral computationally, we utilize a multivariate form of adaptive Gauss-Hermite quadrature (51):

$$f_{\texttt{BPLN}}(T_R, T_C) \approx \sum_{j=1}^{s}\sum_{k=1}^{s} w_j^* w_k^* f_{\texttt{P}}(T_R;\mu_R^*) f_{\texttt{P}}(T_C;\mu_C^*)\phi\left(\epsilon_j^*, \epsilon_k^*; \Sigma_1\right), \tag{4.2}$$

where the s quadrature nodes ϵ_j^* and ϵ_k^* are chosen with respect to the mode of the integrand and the weights w_j^* and w_k^* are scaled according to the estimated curvature at the mode (28). Here $\log(\mu_R^*) = X_R\beta_R + Z_R b_R + \epsilon_j^*$ and $\log(\mu_C^*) = X_C\beta_C + Z_C b_C + \epsilon_k^*$. Adaptive quadrature approaches are typically utilized to increase the accuracy of an integral approximation while utilizing fewer quadrature points to control computational cost. Details regarding the adaptive quadrature procedure is given in the

Appendix, Section AII.2. The log likelihood can then be expressed as

$$l_{\texttt{BPLN}}\left(T_R, T_C\right) = \sum_{i=1}^{n} \log\left[f_{\texttt{BPLN}}\left(T_{Ri}, T_{Ci}\right)\right].$$

The derivatives of this log likelihood can be factored into the form of (4.2), and thus maximization with respect to the parameters $\beta_R, \beta_C, b_R, b_C, \sigma_R, \sigma_C$, and ρ_1 can be performed via quasi-newton methods such as L-BFGS-B. We provide further details of the maximization procedure in the Appendix, Section AII.3.

4.2.2 Bivariate Binomial Logistic-Normal Model for Allele-specific Read Counts

Next we consider the statistical model for allele-specific read counts (ASReC). Similar to the previous section, we wish to assess conditional correlations after accounting for genetic effects. For a gene of interest, we assume its two haplotypes are known, and denote them by h_1 and h_2, respectively. Let N_{R1} and N_{R2} be the number of allele-specific RNA-seq reads from haplotype h_1 and h_2 respectively, and let $N_R = N_{R1} + N_{R2}$. Analogously, we define N_{C1}, N_{C2}, and N_C for the DNase-seq data. We exclude those samples with $N_C < u$ and $N_R < u$ for ASReC studies because allelic imbalance cannot be reliably estimated when there are few allele-specific reads. In the following simulation and real data studies, we set $u = 10$. For the remaining samples, we model the joint distribution of N_{R1} and N_{C1} by a Bivariate Binomial-Logistic-Normal regression model (BBLN), denoted by $f_{\texttt{BBLN}}$:

$$f_{\texttt{BBLN}}(N_{C1}, N_{R1}) \quad = \quad \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{\texttt{B}}(N_{R1}; N_R, \pi_R) f_{\texttt{B}}(N_{C1}; N_C, \pi_C) \phi\left(\xi_R, \xi_C; \Sigma_2\right) d\xi_R d\xi_C,$$

where $f_{\texttt{B}}(; N, \pi)$ denotes a binomial distribution with N trials and probability of success (in this case, success means alignment to haplotype h_1) being π, and π_R and π_C are such success probabilities in the RNA-seq and DNase-seq data, respectively. We model π_R and π_C such that $\log[\pi_R/(1-\pi_R)] = v_R E_R + \xi_R$ and $\log[\pi_C/(1-\pi_C)] = v_C E_C + \xi_C$, where E_R or E_C describes the effect of a SNP:

$$E_R \text{ (or } E_C) = \begin{cases} 0 & \text{if the SNP is homozygous} \\ -1 & \text{if the SNP is heterozygous and its reference allele is on } h_1 \\ 1 & \text{if the SNP is heterozygous and its reference allele is on } h_2. \end{cases}$$

When the SNP is homozygous, it has the same allele in both haplotypes, and thus cannot lead to any allelic imbalance of gene expression. Therefore E_R (or E_C) $= 0$ if the SNP is homozygous. When the SNP is heterozygous and it is responsible for allelic imbalance of gene expression, the higher expression haplotype should have one of the two SNP alleles (the reference allele or alternative allele). Because of this, the definition of genetic effect relies on which haplotype has the reference allele. Recall that in this paper, the reference allele of a SNP is defined by the 1000 Genomes Project SNP annotation file. The confounding covariates X_R or X_C used for TReC model are ignored because such covariates' effects are often cancelled out when we compare the expression of one allele vs. the other allele. It is straightforward to add such effects back into the model if needed. Similarly to the model for TReC data, we assume ξ_C and ξ_R follow a bivariate normal distribution: $\phi(\xi_C, \xi_R; \Sigma_2) \sim \mathcal{N}(0, \Sigma_2)$, where

$$\Sigma_2 = \begin{pmatrix} \kappa_R^2 & \rho_2 \kappa_R \kappa_C \\ \rho_2 \kappa_R \kappa_C & \kappa_C^2 \end{pmatrix},$$

and $-1 \leq \rho_2 \leq 1$ is the correlation parameter. Therefore the dependence between the observed allele-specific read counts (N_{R1} and N_{C1}) is induced by the correlation parameter ρ_2 between ξ_C and ξ_R.

Finally, the joint log likelihood of ASReC for n individuals is

$$l_{\texttt{BBLN}}(N_{Ci1}, N_{Ri1}) = \sum_{i=1}^{n} I(N_{Ri} \geq u \text{ and } N_{Ci} \geq u) \log\left[f_{\texttt{BBLN}}(N_{R1i}, N_{C1i})\right],$$

where $I()$ is an indicator function. We obtain the MLE (Maximum Likelihood Estimate) of the parameters similarly to the BPLN model for TReC data. See the Appendix, Section AII.3 for details.

4.2.3 Testing Framework using TReC or ASReC

Utilizing the MLE of the above models, we employ likelihood ratio tests (LRTs) with degree of freedom 1 to assess the the correlation between gene expression and DHS site. Specifically, we will conduct the following four tests.

1. *Assess the correlation between RNA-seq and DNase-seq TReC in the presence of genetic effects.* Conduct the LRT using the TReC likelihood with H_0: $\rho_1 = 0$ vs. H_1: $\rho_1 \neq 0$.

2. *Assess the correlation between RNA-seq and DNase-seq TReC in the absence of genetic effects.* Conduct the LRT using the TReC likelihood with H_0: $b_R = b_C = \rho_1 = 0$ vs. H_1: $b_R = b_C = 0$,

and $\rho_1 \neq 0$.

3. *Assess the correlation between RNA-seq and DNase-seq ASReC in the presence of genetic effects.* Conduct the LRT using the ASReC likelihood with H_0: $\rho_2 = 0$ vs. H_1: $\rho_2 \neq 0$.

4. *Assess the correlation between RNA-seq and DNase-seq ASReC in the absence of genetic effects.* Conduct the LRT using the ASReC likelihood H_0: $v_R = v_C = \rho_2 = 0$ vs. H_1: $v_R = v_C = 0$, and $\rho_2 \neq 0$.

It is also desirable to test the two null hypotheses $\rho_1 = 0$ and $\rho_2 = 0$ simultaneously, as a two degree of freedom test. However, it is possible that only one of the null hypotheses is correct in certain situations. For example, if the association between gene expression and DHSs is totally due to a common *cis*-acting SNP (where $Z_C = Z_R$) and the SNP is heterozygous across all individuals, then without conditioning on SNP genotype, $\rho_1 = 0$ but $\rho_2 \neq 0$.

We conduct a genome-wide assessment of the dependency between gene expression and DHSs in the following steps. First, for each gene, we only consider the DHSs that are nearby (e.g., within 2 kb) since distant DHSs are unlikely to influence gene expression and would increase the burden of multiple testing correction. Second, for each gene and each DHS, we only consider SNPs that are close to either feature (e.g., within 2kb of either feature), which has been a common practice in previous eQTL studies (75). Our method allows different SNPs to be considered as genetic effects for the RNA-seq and DNase-seq data. However, since our focus is to account for the case where the gene expression and DHS dependence is induced by shared genetic effect, we choose to use the same SNP for RNA-seq and DNase-seq data (i.e., $Z_R = Z_C$). Another important motivation for this strategy is to reduce the multiple testing burden. For example, if there are 100 SNPs around a gene-DHS pair, we correct for the multiple tests across 100 SNPs in the case of a common SNP effect $Z_R = Z_C$. However, if we allow two distinct SNPs to be included by the RNA-seq and DNase-seq data ($Z_R \neq Z_C$), 10,000 SNP combinations will be evaluated, with much higher multiple testing correction burden and more complicated correlation structures among the 10,000 tests.

4.3 Results

4.3.1 Simulation Studies

We use simulated data to evaluate the power and type I error of the tests in section 2.3 for a triplet of (gene expression, DHS, SNP). First, TReC data were simulated from $f_{\texttt{BPLN}}$ under the combinations of the following situations.

- Sample size: $n = 50$, 100, or 300.
- SNP minor allele frequency: 0.5.
- SNP effect: $b_R = b_C = 0$, 0.05, 0.075, 0.1, 0.15, or 0.2.
- Four covariates. The first one is intercept, the other three are simulated from uniform (0,1) distribution. The coefficients are $\beta_C = (2.5, .5, .5, .5)$ and $\beta_R = (2.5, 1, 1, 1)$.
- Variance: $\Sigma = \begin{bmatrix} 0.1 & 0.1\rho_1 \\ 0.1\rho_1 & 0.1 \end{bmatrix}$, with $\rho_1 = 0$, 0.05, 0.1, 0.15, 0.2, 0.25, 0.35, or 0.5.

The simulation study results are summarized in Figure 4.1. For testing $\rho_1 = 0$ in the presence of a shared genetic effect (Figure 4.1A), there is slight inflation of Type I error for small sample sizes ($n = 50$), however such inflation disappears as sample size increases ($n = 100$ or 300). When shared genetic effects on RNA-seq and DNase-seq are ignored, testing the correlation between RNA-seq and DNase-seq TReC data has significantly inflated Type I error, and such inflation increases as the genetic effects b_R and b_C increase (Figure 4.1B). This suggests the importance of accounting for genetic effects in our model, as the correlation between TReC counts may be induced by a shared genetic effect. We also find that the power for detecting the correlation between RNA-seq and DNase-seq increases significantly with sample size (Figure 4.1C). When the sample size is 50, we achieve approximately 80% power to detect correlation $\rho_1 = 0.5$. For $n = 300$, we achieve 80% power to detect correlation $\rho_1 = 0.2$. The power calculations in Figure 4.1C corresponds to data simulated such that $b_R = b_C = 0$, while results for other values of b_R and b_C are similar.

Next, we simulated ASReC data from $f_{\texttt{BBLN}}(N_{Ri1}, N_{Ci1})$ over the following situations.

- Sample size: n=50 or 100.
- SNP minor allele frequency: 0.5.
- SNP effect: $v_R = v_C = 0$, 0.2, 0.3, or 0.4.
- $N_R, N_C \sim$ Poisson(λ), $\lambda = 5$, 20, or 100.
- Variance: $\Sigma_2 = \begin{bmatrix} 0.1 & 0.1\rho_2 \\ 0.1\rho_2 & 0.1 \end{bmatrix}$, where $\rho_2 = 0$, 0.05, 0.1, 0.15, 0.2, 0.25, .035, and 0.5.

The simulation results are shown in Figure 4.2. When we account for the shared genetic effect, testing for $\rho_2 = 0$ has little inflation of Type I error, regardless of the values of π_1 and π_2 or the total number of allele-specific reads (Figure 4.2A-B). Under model misspecification where we ignore genetic effects

(i.e., assuming $v_R = 0$ and $v_C = 0$ or equivalently $\pi_{Ri} = \pi_{Ci} = 0.5$), type I error in testing for $\rho_2 = 0$ increases as π_R and π_C deviates from 0.5 (Figure 4.2C-D). In Figure 4.2E-F, we find that the power for testing for $\rho_2 = 0$ is mostly a function of the total number of allele-specific reads, while sample size has little effect on power. For example, doubling the sample size from $n = 50$ to $n = 100$ leads only to modest gains in power, mostly at lower levels of ρ_2. Most significantly, having only 5 total allele-specific reads per site has almost zero power to detect for marginal correlation. This observation justifies our suggestion of ignoring allele-specific read data when there are few allele-specific reads.

4.3.2 Real Data Analysis

We applied our method to study the DNase-seq and RNA-seq data of 60 HapMap YRI individuals (62, 13). The data were downloaded from `http://eqtl.uchicago.edu/`.

Genotype Data Preparation

Among these 60 individuals, 42 have phased genotypes from the 1000 Genomes Project (TGP) Phase I Release Version 3 (9), consisting of 36 million SNPs. For the remaining 18 individuals we obtained their corresponding HapMap r27 genotypes consisting of approximately 3 million SNPs, and imputed the genotypes and haplotypes from the set of TGP SNPs using MACH 1.0 (47) with the TGP AFR (African population) reference panel. Prior to imputation, about 4,000 HapMap SNPs whose rsIDs have changed between human genome build hg18 and hg19 were removed using the liftRsNumber tool (`http://genome.sph.umich.edu/wiki/LiftRsNumber.py`).

After imputation, for each individual, we recorded the information of all the heterozygous SNPs (positions, rsIDs, alleles on haplotype 1 and 2 of TGP). On average, there are approximately 200,000 heterozygous SNPs (out of 36 million TGP SNPs) per individual. Then we used this list of heterozygous SNPs to extract allele-specific reads. For any read overlapping with at least one heterozygous SNP, we assigned it to haplotype 1 or 2 given its genotype at the SNP positions (75). The designation of haplotype 1 and 2 is based on TGP definition, and haplotype 1 in one individual is not necessary more similar or dissimilar to the haplotype 1 in the other individual.

Tabulating TReC and ASReC for RNA-seq and DNase-seq data

Raw data of paired-end RNA-seq reads were downloaded from `http://eqtl.uchicago.edu/RNA_Seq_data/unmapped_reads/` and were mapped to human genome build hg19 using Tophat version

2.0.6 (80) given Ensembl transcriptome annotation (GRCh37 release 66). All lanes of data pertaining to the same individual were merged subsequent to mapping.

We obtained the RNA-seq TReC for each gene by first counting the number of RNA-seq reads that overlap with exonic regions using R function `countReads` in `R/isoform` (`http://www.bios.unc.edu/~weisun/software/isoform.htm`). The allele-specific reads mapped to haplotype 1 and haplotype 2 were extracted given the list of heterozygous SNPs per individual using R function `extractAsReads` in `R/asSeq` (`http://www.bios.unc.edu/~weisun/software/asSeq.htm`). Then the Allele-specific Read Count (ASReC) per gene and per haplotype was counted using R function `countReads`. To account for possible batch effects in the RNA-seq TReC data, we computed and retained the first 6 principal components from the TReC data matrix for later association analysis. As mentioned earlier, adjusting for confounding factors is often not necessary in the allele-specific analysis since the ASReC from one haplotype is directly compared to the other haplotype within an individual, serving as its own control.

Mapped single-end DNase-seq reads were downloaded from `http://eqtl.uchicago.edu/dsQTL_data/MAPPED_READS/` and were lifted over from build hg18 to hg19 to preserve the quality controls performed in a previous study (13). The isolation of allele-specific DNase-seq reads was performed using the function `asCountsBED5` from R package `BivQTL`. Total and allele-specific DNase-seq read counts were tabulated using BedTools v2.17 (65) for each of 1.5 million 100 bp candidate regions defined in (13); and following (13), we assigned a read to a candidate region based on the 5' start position of each read. We also computed and retained the first 6 principal components from the DNase-seq TReC data matrix and used them as covariates in our RNA-seq vs. DNase-seq association studies.

We performed some additional filtering before our analysis. We removed genes and DNase-seq candidate regions without enough TReC or ASReC. Specifically, we kept features that had ≥ 10 allele-specific reads in at least 10 individuals, or with TReC ≥ 20 in at least 15 individuals. We also removed SNPs with minor allele frequency (MAF) less than 0.05. The final number of features for each data type and for each chromosome are given in Table 4.3. We only performed testing between genes and DHS candidate regions (DHS for short) that are within 2 Kb of each other, and only consider SNPs that are within 2 Kb of either feature. In total we tested 192 gene-DHS pairs (consisting of 157 genes and 190 DHSs), where on average we observed 12 SNPs per gene-DHS pair.

Suppose we test the associations of K gene-DHS pairs, and M_k SNPs are being considered as possible genetic factors of the k-th (gene, DHS candidate region) pair. These M_k SNPs may have strong LD, and thus tests across these M_k SNPs are not independent. We employed the approach of

(60) to calculate the effective number of independent tests to correct for multiple testing for a given gene-DHS pair. Specifically, the effective number of independent tests, denoted by $M_{k,\text{eff}}$, is calculated as

$$M_{k,\text{eff}} = 1 + (M_k - 1)\left(1 - \frac{\text{var}(\lambda_{\text{obs}})}{M_k}\right),$$

and $\text{var}(\lambda_{\text{obs}})$ is the variance of the observed eigenvalues from the correlation matrix of the M_k SNPs. Then we calculated adjusted p-values by $p_{1k} = \min(1, p_{0k}M_{k,\text{eff}})$, where p_{0k} is the unadjusted p-value for the k-th gene-DHS pair. Then multiple testing across all the K gene-DHS pairs are controlled via False Discovery Rate (FDR). Specifically, given a p-value cutoff p_t, we estimate FDR by $p_t K / \sum_k I(p_{1k} \leq p_t)$, where $I()$ is an indicator function. Then to control FDR at α we chose the p-value cutoff p_α such that $p_\alpha = argmax_{p_t}\{p_t : [p_t K / \sum_k I(p_{1k} \leq p_t)] \leq \alpha\}$.

We apply the BPLN and BBLN models to assess correlation between RNA-seq and DNase-seq TReC and ASReC, respectively. Given the results in Figure 1 with respect to model misspecification, we seek to test the effect for additional correlation between data types in the presence of a joint SNP effect ($\rho_1 \neq 0$, $\rho_2 \neq 0$). We summarize the number of significant results ($\alpha = 0.05$) by chromosome in Figure 4.3.

We find that the number of significant results genome wide testing for marginal correlation ρ_1 between the RNA-seq and DNase-seq TReC is relatively rare (approximately 7.56% of all tests), similar to tests for common SNP effect $\beta_{snp,R}$ and $\beta_{snp,C}$ (7.56%). It is also rare to observe significant marginal correlation ρ_1 in addition to a significant joint snp effect (Table 4.1). This is also the case for testing for an imbalance in ASE (Table 4.2). We find that we observe a relatively larger number of significant tests for ρ_2 and (v_R, v_C) in the ASE testing setting, which is expected as our simulations demonstrate that sample size has less of an effect on power compared to TReC. After performing our multiple testing correction, the total number of significant results per chromosome drops significantly, as expected (Figure 4.4). Prior to adjustment, we observe 1372 total significant results in testing for ρ_1, and after adjustment this drops to 104. Similar, prior to adjustment we observe 2546 significant results in testing for ρ_2, and after adjustment this drops down to 671 significant results.

4.4 Discussion

We introduce a new method to model relationships across multiple data types, including gene expression, epigenetic marks, and genetic variants. We demonstrate the utility and power of our method to test for bivariate correlation between RNA-seq and DNase-seq data while adjusting for a

possible shared genetic effect. Our simulation results show that there is relatively low power to detect weaker associations at smaller sample sizes, such as $n = 50$, which may explain the limited number of findings from our real data study with sample size 60. While this is a limitation for this dataset, in the near future we expect to see larger sample sizes as the cost of sequencing decreases.

For BPLN model we utilize a bivariate form of the Poisson Log-Normal distribution, similar to what was introduced by (1). The main distinction between the multivariate Poisson Log-normal model of (1) and the BPLN is that it models the mean of each marginal Poisson distribution with a set of covariates in addition to the bivariate random effect. The advantage of either approach is the flexibility in specifying the correlation structure between the bivariate counts via Σ_1. In addition, overdispersion in the RNA-seq and DNase-seq TReC is modeled via variances σ_R and σ_C, respectively, where larger variance corresponds to larger overdispersion. Most importantly, both positive and negative correlations are allowed between the bivariate counts using this approach. However, the numerical integration that is required to evaluate the BBLN likelihood and derivatives increases the complexity of the estimation procedure, and may be unstable for lower sample sizes and lower signal levels. BBLN model also share similar flexibilities as the BPLN model, and similar computational issues.

Other alternatives to the BPLN to test for correlations between data types include the bivariate negative binomial distribution introduced by (17). This model is simply the product of two marginal negative binomial distributions corresponding to each of the two random variables, plus a multiplicative term with an additional parameter λ controlling the correlation of the two random variables. This approach also allows for either positive or negative correlations between the two variables, and evaluation of the likelihood and derivatives of this distribution does not require numerical integration. However, the maximization of the corresponding likelihood with respect to λ is difficult in practice, because the plausible values of λ are bounded and such bounds are not known *apriori*. When the mean of each marginal distribution is not modeled by covariates, these bounds can be derived analytically. However in the regression setting this is difficult to determine. For ASReC, an analogous model is the Bivariate Beta Binomial Distribution (11) and it suffers from similar problems. Other approaches involving modeling bivariate or multivariate counts do not allow for negative correlations in counts or do not account for possible overdispersion in each of the random variable.

Despite the computational complexity of the BPLN and BBLN models, our implementation proved to be robust and computationally efficient. Sampling-based methods such as Monte Carlo integration could have been used to evaluate the BPLN and BBLN, however, the inherent randomness in such approaches may pose problems during maximization. Fully Bayesian approaches is not computationally

Table 4.1: Testing for Joint SNP effect and Marginal Correlation in TReC

	non-Sig. SNP	Sig. SNP
non-Sig. Corr	15661	1183
Sig. Corr	1123	189

Table 4.2: Testing for Joint SNP effect and Marginal Correlation in ASReC

	non-Sig. SNP	Sig. SNP
non-Sig. Corr	14192	1329
Sig. Corr	916	1217

efficient enough for genome-wide studies.

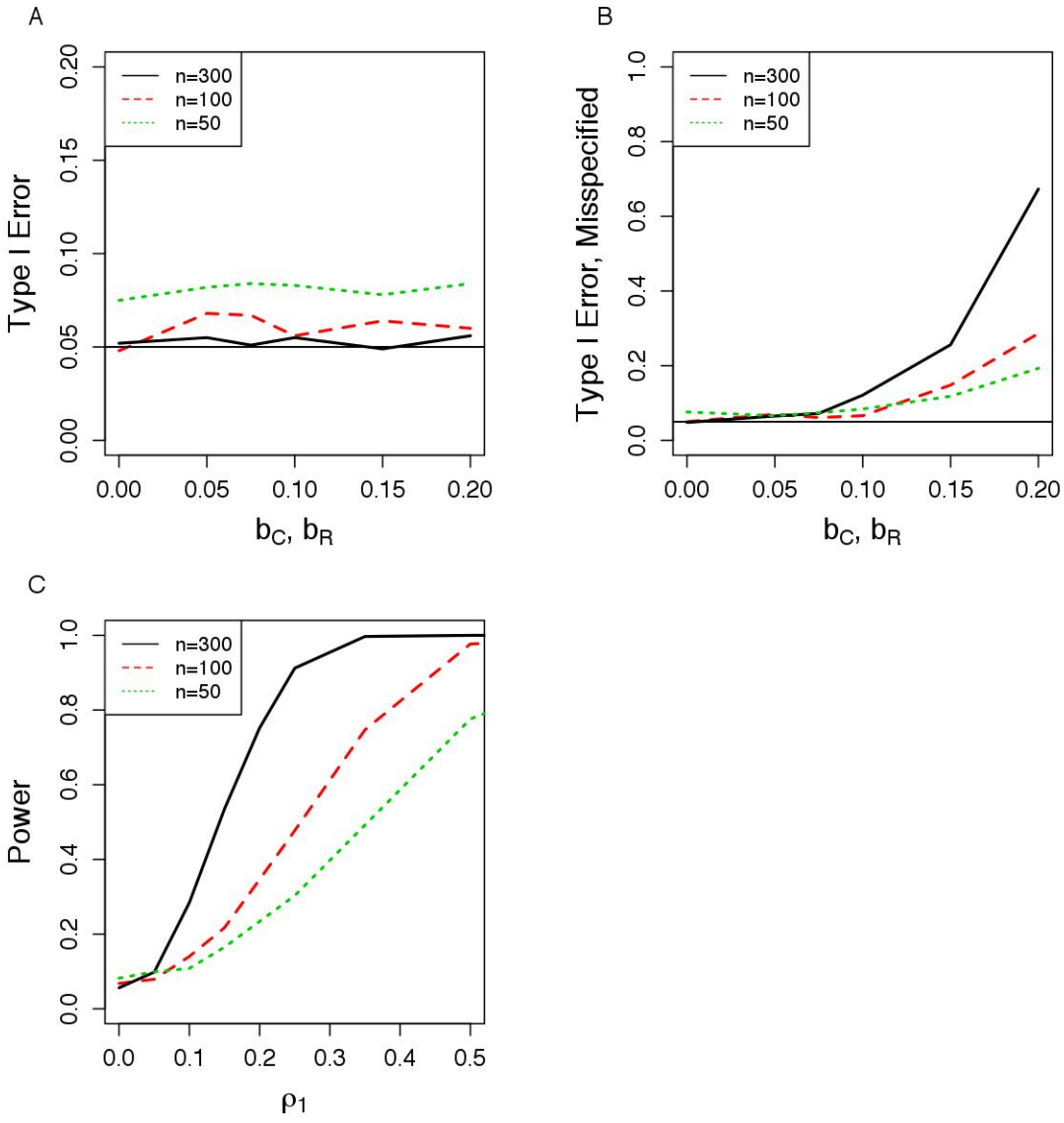

Figure 4.1: Simulation results for BPLN (Bi-variate Poisson Log Normal) model. (A) Type I error in testing for $\rho_1 = 0$ given b_C and b_R. (B) Type I error in testing for $\rho_1 = 0$ under the assumption of $b_C = 0$ and $b_R = 0$ while they may not. (C) Power in testing for $\rho_1 = 0$ with different sample sizes, given $b_C = 0$ and $b_R = 0$.

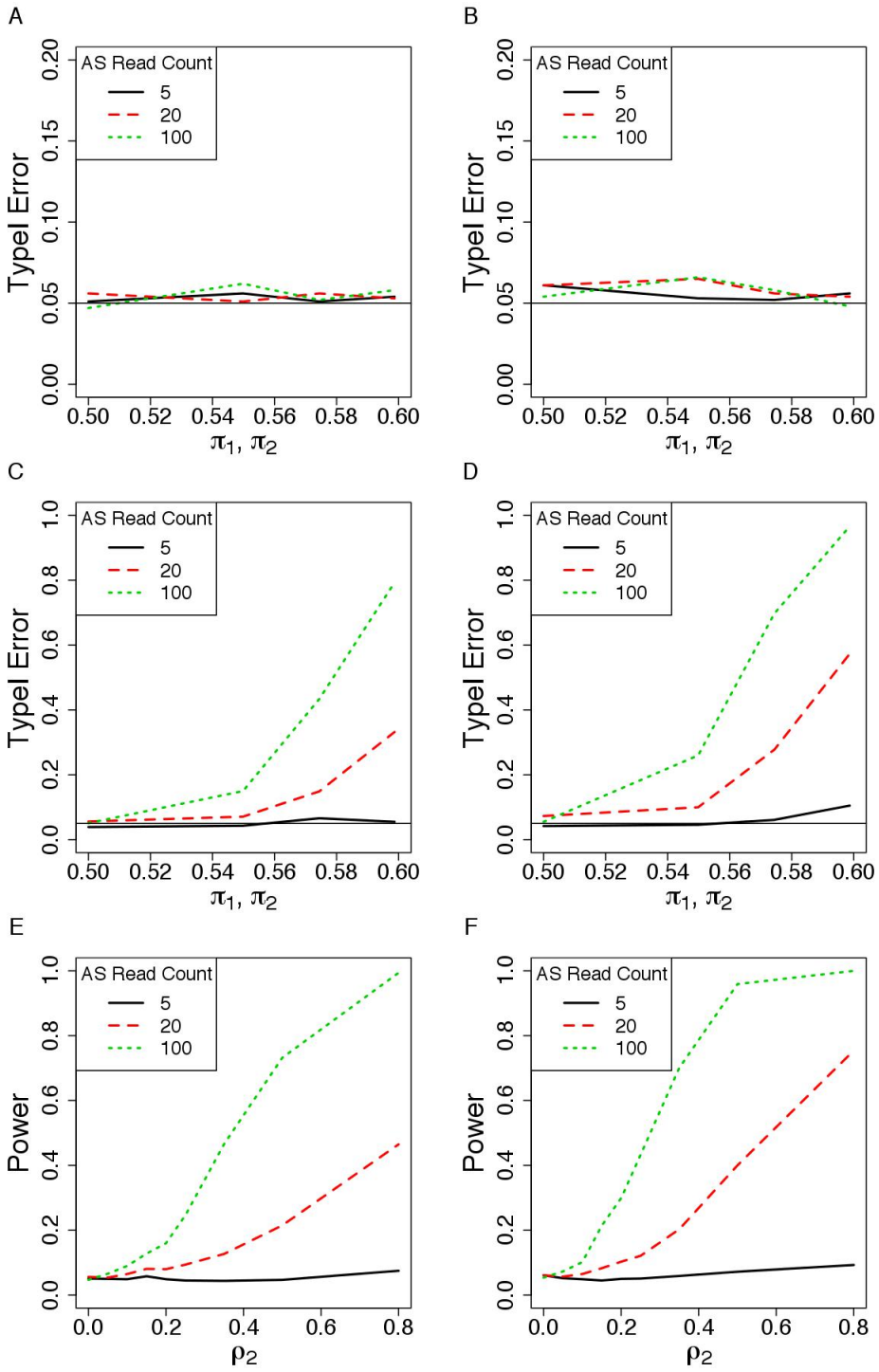

Figure 4.2: Simulation results for BBLN (Bi-variate Binomial Logistic Normal) model. (A) and (B): Type I error in testing for $\rho_2 = 0$ given π_1 and π_2 when $n = 50$ (A) or $n = 100$ (B). (C) and (D): Type I error in testing for $\rho_2 = 0$ under the assumption of $\pi_1 = 0.5$ and $\pi_2 = 0.5$ when $n = 50$ (C) or $n = 100$ (D). (E) and (F): Power in testing for $\rho_2 = 0$ when $n = 50$ (E) or $n = 100$ (F).

Figure 4.3: Significant hits by chromosome in testing for marginal TReC and ASReC correlation, adjusting for possible joint SNP effect

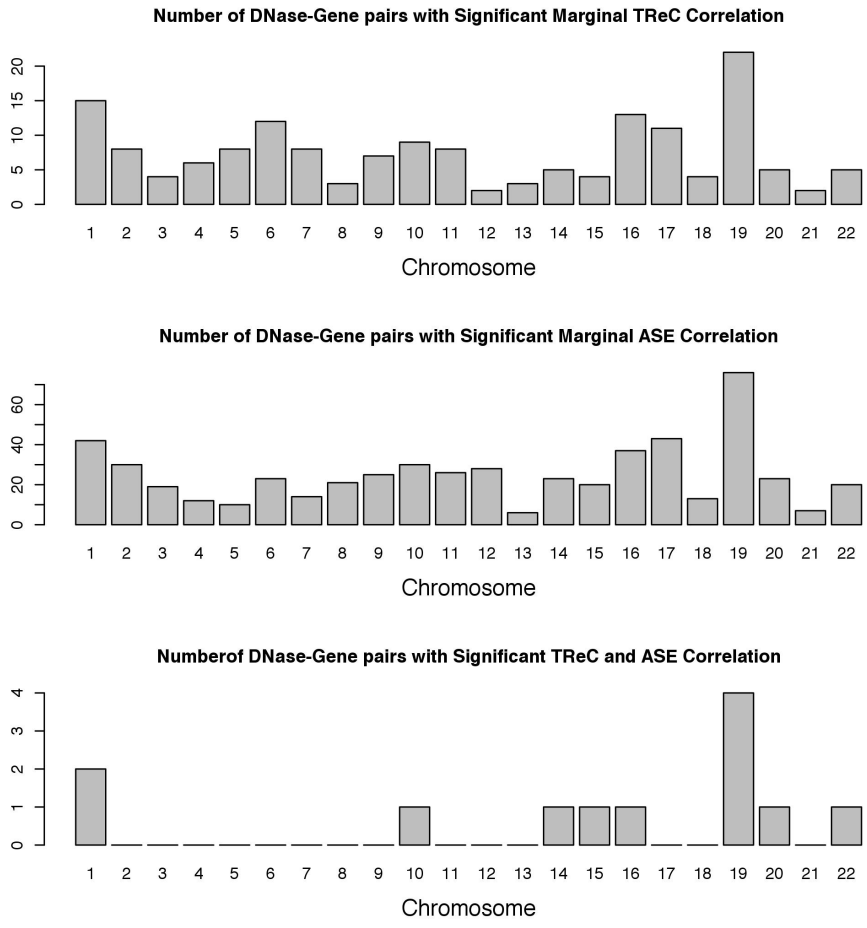

Figure 4.4: Significant hits by chromosome in testing for marginal TReC and ASReC correlation, adjusting for possible joint SNP effect and after p-value correction for multiple testing

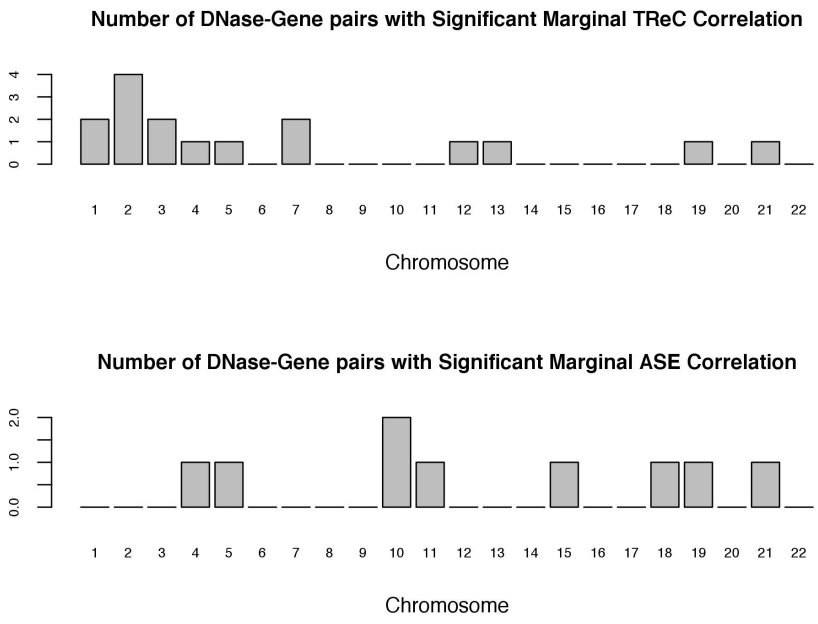

Table 4.3: Total Number of candidate regions left after filtering per feature used in the real data analysis

Chr	Transcripts	DNase sites	SNPs
1	925	881	645306
2	602	536	697842
3	500	434	596146
4	327	263	610533
5	401	354	552109
6	486	411	542274
7	442	418	488251
8	333	326	472867
9	362	378	358739
10	361	401	413961
11	549	484	413007
12	532	438	394463
13	152	142	306687
14	334	316	269737
15	339	284	245403
16	447	524	260920
17	550	626	226148
18	119	134	241130
19	642	1016	195144
20	266	303	187874
21	109	142	120730
22	269	359	115693
Total	9047	9170	8354964

Chapter 5

Conclusion

In this document we define a large class of biomedical experiments called DAE-seq experiments used widely by biomedical researchers, detailing current challenges in the analysis of such data and introduce three statistical methods to address these challenges. The first method is a three-component mixture regression model to "enriched regions", i.e., the genomic regions with more sequenced reads than expected in background regions. We demonstrate its practical utility and accuracy in detecting regions of active regulatory elements across a wide range of commonly used DAE-seq datasets. We then develop a novel Autoregressive Hidden Markov Model (AR-HMM) to account for often-ignored spatial dependence in DAE-seq data, and demonstrate that accounting for such dependence leads to increased performance in identifying biologically active genomic regions in both simulated and real datasets. We also introduce an efficient and novel variable selection procedure in the context of Hidden Markov Models when the means of the emission distributions of each state are modelled with covariates. We study the asymptotic properties of the proposed variable selection procedure and apply this approach to simulated and real DAE-seq data. Lastly, we a new method for the joint analysis of total and allele-specific read counts from DAE-seq data and RNA-seq data. In all we developed several statistical procedures for the analysis of DAE-seq data that are highly relevant to biomedical researchers with broader applicability to other areas of statistics.

APPENDIX I

Appendix for Chapter 3

AI.1 Regularity Conditions for Corollary 1

Define the standard HMM as a special case of the HMM with covariates where covariates are not utilized, $l_n(\Psi_0|\mathbf{X}, y)$ is the likelihood of the standard HMM, and Ψ_0 is the true value of Ψ. In the standard HMM, only an intercept is used to model the mean of each state distribution. Let $\hat{\Psi}$ be the MLE of Ψ and assume that the Fisher Information matrix $I(\Psi_0)$ exists and is non-singular. Then, assuming the regularity conditions A1-A6 from (5), we have the following:

1. $\hat{\Psi} \to \Psi_0$ almost surely as $n \to \infty$.
2. $n^{-\frac{1}{2}} l'_n(\Psi_0|\mathbf{X}, y) \to N(0, I(\Psi_0))$ in distribution as $n \to \infty$
3. $\frac{1}{n} l''_n(\hat{\Psi}|\mathbf{X}, y) \to -I(\Psi_0)$ in probability as $n \to \infty$
4. $n^{\frac{1}{2}}(\hat{\Psi} - \Psi_0) \to N(0, I(\Psi_0^{-1}))$ in distribution as $n \to \infty$

For the HMM with covariates case, $(Y_1, Z_1), \ldots, (Y_n, Z_n)$ is conditioned on $X_1, \ldots, X_n$ such that $Y_1|X_1, Z_1, \ldots, Y_n|X_n, Z_n$ are conditionally independent and that $(Y_1, Z_1)|X_1, \ldots, (Y_n, Z_n)|X_n$ is stationary conditional on X_i. We assume that our HMM with covariates is identifiable, such that for any set of parameters Ψ and Ψ^*, $l_n(\Psi^*|\mathbf{X}, y) = l_n(\Psi|\mathbf{X}, y)$ if and only if $\Psi* = \Psi$ and $K^* = K$ up to a permutation of the states. The above results demonstrate that the HMM with covariates likelihood have similar asymptotic properties to the typical *iid* likelihood. Given results (1)-(4) and because penalty conditions P0-P2 are similar to those from (38), Corollary 1 naturally follows from (38) Theorems 1 and 2. To avoid duplicating the proof from (38), we describe the proof of Corollary 1 as follows. The proof of Corollary 1, part (a), follows from (38) Theorem 1 with results (2) and (3) above. The proof of Corollary 1, part (b), follows from (38) Theorem 2 part a and b.1 with regularity condition A3 and result (4). Proof of part (c) of Corollary 1 follows from (38) b.2 with regularity condition results (2) and (3).

AI.2 AR-HMM Forward, Backward, and related probabilities

For any $k = 1, \ldots, K$, define

$$\begin{aligned}
\mathtt{f}_{1k} &= p(Z_1 = k, y_1 | \mathbf{X}, \Psi_A^{(s)}) = \pi_k f_k(y_1 | X_{1k}, \Psi_A^{(s)}), \\
\mathtt{f}_{ik} &= p(Z_i = k, y_1, \ldots, y_i | \mathbf{X}, \Psi_A^{(s)}) = \sum_{j=1}^{K} \alpha_{i-1,j} \gamma_{jk} f_{jk}(y_i | X_{ik}, r_{i-1,j}^{(s)}, \Psi_A^{(s)}), \text{ for } i > 1, \\
\mathtt{b}_{ik} &= p(y_{i+1}, \ldots, y_n | Z_i = k, \mathbf{X}, \Psi_A^{(s)}) = \sum_{j=1}^{K} \gamma_{kj} f_{k,j}(y_{i+1} | X_{i+1,j}, r_{i,k}^{(s)}, \Psi_A^{(s)}) \mathtt{b}_{i+1,j}, \text{ for } i < n,
\end{aligned}$$

and $\mathtt{b}_{nk} = 1$. Then we have

$$p(Z_i = k | y, \mathbf{X}, \Psi_A^{(s)}) = \frac{\mathtt{f}_{ik} \mathtt{b}_{ik}}{\sum_{l=1}^{K} \mathtt{f}_{nl}}, \tag{AI.1}$$

$$p(Z_{i-1} = j, Z_i = k \mid y, \mathbf{X}, \Psi_A^{(s)}) = \frac{\mathtt{f}_{i-1,j} \gamma_{jk}^{(s)} f_{jk}(y_i | X_{ik}, r_{i-1,j}^{(s)}, \Psi_A^{(s)}) \mathtt{b}_{ik}}{\sum_{l=1}^{K} \mathtt{f}_{nl}}. \tag{AI.2}$$

AI.3 EM + Coordinate Descent Algorithm

AI.3.1 General Overview

We first give an overview of our EM + coordinate descent algorithm for obtaining the penalized MLE under the HMM. This algorithm reduces to the algorithm for obtaining the unpenalized MLE under the HMM when the penalties are set to 0. For notational simplicity we only describe the algorithm for the HMM, as this procedure is easily generalizable for the AR-HMM.

- **Initialization:** Let y_i $(1 \leq i \leq n)$ be the number of reads in the i-th window. Denote the initial] state of the i-th window by z_i^0 such that z_i^0=1 or 2 for the background and enriched state, respectively. We create initial state assignments $z_1^0, \ldots z_n^0$ such that

$$z_i^0 = \begin{cases} 2 & \text{if } y_i > t \\ 1 & \text{if } y_i \leq t, \end{cases} \tag{AI.3}$$

given some integer threshold t, $t > 0$. Because enrichment regions may take up anywhere between 1% - 10% of the genome we typically set t to be a constant in the range of 90-99th percentile of y_i. We then estimate the following probabilities such that $p(Z_i = k) = \sum_{i=1}^{n} I[z_i^0 = k]/n$ and $p(Z_{i-1} = j, Z_i = k) = \sum_{i=1}^{n} I[z_{i-1}^0 = j \cap z_i^0 = k]/(n-1)$. These initial probabilities

are then passed to the M-step to start the EM algorithm. We may evaluate several starting values of t and chose the one that leads to the highest likelihood after several iterations of the EM algorithm. Given the large sample sizes typically used DAE-seq data analysis, we find that the final model is not very sensitive to the choice of t as long as t is not very close to 1.

- **M-step:** Denote $\Psi^{(s)} = (\beta^{(s)}, \phi^{(s)}, \pi^{(s)}, \eta^{(s)})$ be the parameter estimates in the s-th step of the ECM algorithm. In the M-step, we estimate $\Psi^{(s)}$ given $p(Z_{i-1} = j, Z_i = k|y, \mathbf{X}, \Psi^{(s-1)})$ from the E-step. When the penalty is non-zero, maximization of β is performed with respect to the penalized likelihood described in (3). In the case of the negative binomial distribution, we separate this step into two conditional maximization steps to estimate β and ϕ. In the first Conditional Maximization (CM) step, we obtain $\beta^{(s+1)}$ given $\beta^{(s)}$ and $\phi^{(s)}$. We do this through two nested loops

  - **PIRLS outer Loop:** Update β through penalized Iteratively Reweighted Least Square (PIRLS). Call inner coordinate descent loop to perform penalized estimation of β
  - **Coordinate Descent Inner Loop:** Utilizing the working residuals and weights in the current iteration of the PIRLS, maximize β with respect to the given penalty via coordinate descent.

  In the second CM step we estimate $\phi^{(s+1)}$ given $\beta^{(s+1)}$. When using this CM approach, our algorithm becomes an ECM algorithm (57).

- **E-Step:** Compute $p(Z_{i-1} = j, Z_i = k|y, \mathbf{X}, \Psi^{(s)})$ and $p(Z_{i-1} = j|y, \mathbf{X}, \Psi^{(s)}) = \sum_{k=1}^{K} p(Z_{i-1} = j, Z_i = k\ |y, \mathbf{X}, \Psi^{(s)})$. Exact computation of these values are given in the Appendix Section AI.2 of the main text.

- **Convergence:** Compute the log likelihood at the current step s, $l_n^{(s)}$. Terminate if $|l_n^{(s)} - l_n^{(s-1)}|/l_n^{(s-1)} < 10^{-7}$, where $l_n^{(s)}$ is the likelihood at step s.

AI.3.2 M-step Details

We compute γ and π in the manner described in section 3.2. In step s of the ECM algorithm, we estimate the regression coefficients for each state (k=1, or 2) using the following penalized Iteratively Re-weighted Least Squares (IRLS). The standard IRLS is a special case of the following algorithm when the penalty is 0. Employing the canonical log link function, the expected values for the negative binomial or Poisson distribution at step l of the penalized IRLS algorithm are $\mu_{ik}^{(l)} = \exp\left(X_{ik}\beta_k^{(l)}\right)$.

The variance functions are $v_{ik}^{(l)} = \mu_{ik}^{(l)} + \left[\mu_{ik}^{(l)}\right]^2 \phi_k^{(s)}$ and $v_{ik} = \mu_{ik}$ for the negative binomial distribution and Poisson distribution, respectively. We define the posterior probabilities of state membership for state k at step s of the EM algorithm as $\zeta_{ik}^{(s)} = p(Z_i = k|y, \mathbf{X}, \Psi^{(s)})$. Then we estimate $\beta_k^{(s+1)}$ with the following procedure.

Conditional Maximization 1 (CM1) for $\beta_k^{(s+1)}$:

1. **Initialization ($l = 0$):** If $s = 0$, μ_{ik} is initialized such that $\mu_{ik}^{(l)} = y_i + I[y_i = 0]/6$. If $s > 0$, then $\mu_{ik}^{(l)} = \exp\left(X_{ik}\beta_k^{(s)}\right)$. Then $v_{ik}^{(l)} = \mu_{ik}^{(l)} + \left[\mu_{ik}^{(l)}\right]^2 \phi_k^{(s)}$. For $i = 1, ..., n$, compute IRLS weights $p_{ik}^{(l)} = \zeta_{ik}^{(l)} v_{ik}^{(l)}$.

2. **Compute Working Residuals:** For $i = 1, ..., n$, compute $q_{ik}^{(l)} = X_{ik}\beta_k^{(l)} + r_i^{(l)}$, where $r_i^{(l)} = (y_i - \mu_{ik}^{(l)})/v_{ik}^{(l)}$.

3. **Estimate** β: Compute $\beta_k^{(l+1)}$ via a coordinate descent algorithm given $q_{ik}^{(l)}$ and $p_{ik}^{(l)}$ for $i = 1, ..., n$. The details of this coordinate descent algorithm are given in the next section.

4. **Convergence:** Given $\beta_k^{(l+1)}$, update $p_{ik}^{(l+1)} = \zeta_{ik}^{(l)} v_{ik}^{(l+1)}$ and $r_i^{(l+1)} = (y_i - \mu_{ik}^{(l+1)})/v_{ik}^{(l+1)}$. Compute the weighted sum of squared residuals $\texttt{wss}_k^{(l+1)} = \sum_{i=1}^n p_{ik}^{(l+1)} \left[r_i^{(l+1)}\right]^2$. This CM step converges if $|\texttt{wss}_k^{(l+1)} - \texttt{wss}_k^{(l)}|/(\texttt{wss}_k^{(l)} + 1) < 10^{-5}$, where we then set $\beta_k^{(s+1)} = \beta_k^{(l+1)}$. Otherwise set $\beta_k^{(l)} = \beta_k^{(l+1)}$ and repeat steps 2-4 until convergence.

Conditional Maximization 2 (CM2) for $\phi_k^{(s+1)}$: For the negative binomial distribution, we undertake a second maximization step to estimate ϕ_k. Estimation of $\phi_k^{(s+1)}$ is performed via Newton-Raphson given $\mu_{ik}^{(s+1)} = \exp\left(X_{ik}\beta_k^{(s+1)}\right)$ from CM1, E-step weights w_{ik}, and y_i, $i = 1, ..., n$. The starting value for ϕ_k is typically $\phi_k^{(s)}$ or the moment estimate of ϕ_k when $s = 0$.

AI.3.3 Penalized Estimation using the Coordinate Descent Algorithm

In each iteration l of the penalized IRLS, we obtain $\beta_k^{(l+1)}$ by minimizing the following objective function with respect to β_k:

$$\mathcal{Q}_k(\beta_k|\Psi^{(l)}) = \frac{1}{2}\sum_{i=1}^n p_{ik}^{(l)}(q_{ik}^{(l)} - X_{ik}\beta_k)^2 + \eta_k^{(s)}\sum_{j=1}^p \rho_{\omega_k}(\beta_{jk}), \tag{AI.4}$$

where $p_{ik}^{(l)}$ is the weight for the i-th observation belonging to the k-th state from penalized IRLS state l, and $\eta_k^{(s)}$ is the proportion of observations belonging to the k-th state (from EM iteration s). Here we only consider the estimation for a particular component k. The same algorithm will be applied for each state separately. We obtain $\beta_{jk}^{(l+1)}$, $j = 1...p$, through the following coordinate descent algorithm.

1. **Initialization ($m = 0$):** If there is no initial estimate of β_{jk}, initialize $\beta_{jk}^{(0)} = 0$ for $j = 1, ..., p$. Otherwise initialize $\beta_{jk}^{(0)}$ with the previous estimate (e.g., estimates from last EM iteration $\beta_{jk}^{(s)}$ or last penalized IRLS iteration $\beta_{jk}^{(l)}$).

   Then in the $(m+1)$-th iteration,

2. **Update intercept:** , $\beta_{0k}^{(m+1)} = \sum_{i=1}^{n} p_{ik}^{(l)}(q_{ik}^{(l)} - \sum_{j=1}^{p} X_{ijk}\beta_{jk}^{(m)})/n$.

3. **Update other regression coefficients:** Calculate $\beta_{jk}^{(m+1)} = S(\bar{\beta}_{jk}, \beta_{jk}^{(m)})$ where

$$\bar{\beta}_{jk} = \frac{\sum_{i=1}^{n} p_{ik}^{(l)} X_{ijk}(q_{ik}^{(l)} - \sum_{j \neq p}^{p} X_{ijk}\beta_{jk}^{(m)})}{\sum_{i=1}^{n} p_{ik}^{(l)} X_{ijk}^2},$$

   and $S(\cdot)$ is a penalty-specific thresholding operator. The specific forms of a few penalties are listed in the next section.

4. **Convergence**: Cycle through the estimation of intercept and p regression coefficients until the sum of squared residuals $r^{(m+1)} = \sum_{i=1}^{n} p_{ik}^{(l)}(q_{ik}^{(l)} - X_{ik}\beta_k^{(m+1)})^2$ converges, such that $|r^{(m+1)} - r^{(m)}|/r^{(m+1)} < 10^{-8}$. If convergence criteria is met, set $\beta_k^{(l+1)} = \beta_k^{(m+1)}$ and return to the penalized IRLS loop.

AI.3.4 Thresholding Operators

This section details the thresholding operators used for the LASSO (79), SCAD (18), and Log (23) penalties. The thresholding operators for LASSO and SCAD have been described elsewhere, e.g., (8). The thresholding operators for the Log penalty can be derived after applying a local linear approximation (LLA) (98) such that $\rho_{\omega_k}(\beta_{jk}) \approx \rho_{\omega_k}(\beta_{jk}^{(s)}) + \rho'_{\omega_k}(|\beta_{jk}^{(s)}|)(|\beta_{jk}| - |\beta_{jk}^{(s)}|)$.

- **Log Penalty:**

$$\beta_{jk}^{(m+1)} = \begin{cases} 0 & \text{if } |\bar{\beta}_{jk}| \leq \kappa_{jk} \\ \bar{\beta}_{jk} - \text{sgn}(\bar{\beta}_{jk})\kappa_{jk} & \text{if } |\beta_{jk}| > \kappa_{jk} \end{cases} \tag{AI.5}$$

  where $\kappa_{jk} = \lambda_k/(|\beta_{jk}^{(m)}| + \tau_k)$ and τ_k is an additional tuning parameter such that $\tau_k > 0$.

- **LASSO:**

$$\beta_{jk}^{(m+1)} = \begin{cases} 0 & \text{if } |\bar{\beta}_{jk}| \leq \lambda_k \\ \bar{\beta}_{jk} - \text{sgn}\left(\bar{\beta}_{jk}\right) \lambda_k & \text{if } |\bar{\beta}_{jk}| > \lambda_k \end{cases} \tag{AI.6}$$

- **SCAD:**

$$\beta_{jk}^{(m+1)} = \begin{cases} 0 & \text{if } |\bar{\beta}_{jk}| \leq \lambda_k \\ \bar{\beta}_{jk} - \text{sgn}\left(\bar{\beta}_{jk}\right) \lambda_k & \text{if } \lambda_k < |\bar{\beta}_{jk}| \leq 2\lambda_k \\ \left[(a-1)\bar{\beta}_{jk} - \text{sgn}\left(\bar{\beta}_{jk}\right) a\lambda_k\right] / (a-2) & \text{if } 2\lambda_k < |\bar{\beta}_{jk}| \leq a\lambda_k \\ \bar{\beta}_{jk} & \text{if } a\lambda_k < |\bar{\beta}_{jk}| \end{cases} \tag{AI.7}$$

AI.4 Rejection Controlled ECM

Typically we observe very low weights for many observations in our EM algorithm, especially for the weights corresponding to the enrichment state. We employ a rejection controlled ECM (50) where we remove observations from the computation in the M-step if they have very low weight. By doing so, we increase the stability of the M-step and reduce its computational burden. We describe the rejection controlled ECM for the HMM below. The generalization to the AR-HMM is straightforward, so it is omitted. Define the weight from the E-step corresponding to state k as $w_{ik} = p(Z_i = k|y, \mathbf{X}, \Psi^{(s)})$. Then, we compute

$$w_{ik}^{*} = \begin{cases} w_{ik} & \text{if } w_{ik} > c \\ c & \text{with probability } w_{ik}/c \text{ if } w_{ik} \leq c \\ 0 & \text{with probability } 1 - w_{ik}/c \text{ if } w_{ik} \leq c \end{cases} \tag{AI.8}$$

where w_{ik}^{*} is the new weight. We then normalize w_{ik}^{*} such that $w_{ik}^{**} = \frac{w_{ik}^{*}}{\sum_{k=1}^{K} w_{ik}^{*}}$. These weights are then passed to the M-step, where those observations with $w_{ik}^{**} = 0$ have zero contribution to the parameter estimation and hence can be removed. We choose threshold $c = 0.05$ fixed across all EM iterations.

APPENDIX II

Appendix for Chapter 4

Here we provide additional details regarding the evaluation and maximization of the Bivariate Poisson-Lognormal Regression model (BPLN) and the Bivariate Binomial Logistic-Normal Regression model (BBLN) likelihoods. The fitting of these models forms the basis of our testing framework, which utilizes the likelihood ratio test to evaluate the significance of the estimated marginal correlation and joint SNP effects on each data type.

AII.1 Data Processing for the Study of TF Binding in Promoter Regions vs. H3K36me3 Signals in Downstream Genes

We downloaded the full list of UCSC genes from the UCSC Genome Browser, and prepared the data for each gene separately. For gene i, we obtained the set of n_i tilling windows contained in this gene, and denoted the H3K36me3 window read counts by $y^{(i)} = (y_1^{(i)}, \ldots, y_{n_i}^{(i)})^T$. Next we determined whether each of the P transcription factors binds any where 2.5 kb upstream of the transcription starting site of gene i. Here, binding event of a TF within a window is defined the same way as in previous analysis of Section 6 of the main text. Specifically, a TF binding event occurs within a window if the TF binding signal is larger than 95 percentile of genome-wide TF binding signals. Then a TF binds to the promoter of gene i if a TF binding event occurs in any one of the promoter windows. We defined the covariate data for the p-th TF and for gene i to be $X_p^{(i)} = (X_{1p}^{(i)}, \ldots, X_{n_ip}^{(i)})^T$, where $X_{jp}^{(i)} = 1$ (for any $j = 1, \ldots, n_i$) if the p-th TF binds to the promoter of gene i, and $X_{jp}^{(i)} = 0$ otherwise. After collecting data for all the TFs for gene i, we had $\mathbf{X}^{(i)} = (X_1^{(i)}, \ldots, X_P^{(i)})$. Finally, concatenating data for all genes, we had $y_{n\times 1} = ({y^{(1)}}^T, \ldots, {y^{(m)}}^T)^T$, where m is the number of genes, and $n = \sum_{i=1}^m n_i$, and $\mathbf{X}_{n\times P} = ({\mathbf{X}^{(1)}}^T, \ldots, {\mathbf{X}^{(m)}}^T)^T$. For each of the n windows, we also included the corresponding set of confounding covariates and their two-way and three-way interactions as in the previous analysis. In summary, in this setup, we modeled the H3K36me3 signals in a gene as a function of the presence/absence of TF binding at promoter region, in addition to local confounding covariates. Since H3K36me3 typically covers gene bodies of actively transcribed genes, here H3K36me3-enriched and background regions would arise from those genes with high transcriptional activity vs. those with no or low transcriptional activity.

AII.2 Standard and Adaptive Bivariate Gaussian Quadrature

To increase the accuracy of the integral approximation while limiting the number of quadrature nodes, we use the adaptive quadrature approach from (51) where the quadrature nodes are scaled around the posterior mode of $f_{\texttt{BPLN}}(T_{Ri}, T_{Ci})$ with respect to $(\epsilon_{Ri}, \epsilon_{Ci})$. Let s be the number of quadrature nodes, ϵ_j is the j_{th} quadrature node from the set of s^{th} order roots of the Gauss-Hermite Polynomial, and w_j is j^{th} quadrature weight associated with ϵ_j. Then, for the ith observation, $i = 1, \ldots, n$, evaluate $f_{\texttt{BPLN}}(T_{Ri}, T_{Ci})$.

- **Step 1:** Compute $\arg\max_{(\epsilon_{Ri}, \epsilon_{Ci})} f_{\texttt{BPLN}}(T_{Ri}, T_{Ci})$ to obtain $(\hat{\epsilon}_{Ri}, \hat{\epsilon}_{Ci})$, given the data and model parameters. These are the posterior modes of random effects $(\epsilon_{Ri}, \epsilon_{Ci})$. The new quadrature nodes will be centered around these posterior modes. This maximization is also done via L-BFGS-B.

- **Step 2:** Compute $\tilde{\Sigma} = -(\nabla^2_{(\epsilon_{Ri}, \epsilon_{Ci})} f_{\texttt{BPLN}}(T_{Ri}, T_{Ci}))^{-1}$. That is, compute the negative inverse of the Hessian of $f_{\texttt{BPLN}}(T_{Ri}, T_{Ci})$. This will be used to scale the quadrature nodes around the posterior mode.

- **Step 3:** Compute adaptive quadrature nodes $(\epsilon^*_j, \epsilon^*_k)$, $j = 1, \ldots, s$, $k = 1, \ldots, s$ such that $(\epsilon^*_j, \epsilon^*_k)^T = (\hat{\epsilon}_{Ri}, \hat{\epsilon}_{Ci})^T + \sqrt{2}\tilde{\Sigma}^{\frac{1}{2}}(\epsilon_j, \epsilon_k)^T$.

- **Step 4:** Compute likelihood approximation for observation i such that

$$\begin{aligned} f_{\texttt{BPLN}}(T_{Ri}, T_{Ci}) &\approx \sum_{j=1}^{s}\sum_{k=1}^{s} 2\sqrt{|\tilde{\Sigma}|} w_j w_k \exp(\epsilon_j)\exp(\epsilon_k) f_{\texttt{BPLN}}(T_{Ri}, T_{Ci}) \\ &= \sum_{j=1}^{s}\sum_{k=1}^{s} w^*_j w^*_k f_{\texttt{BPLN}}(T_{Ri}, T_{Ci}) \\ &= \sum_{j=1}^{s}\sum_{k=1}^{s} w^*_j w^*_k f_{\texttt{P}}(T_{Ri}; \mu^*_{Ri}) f_{\texttt{P}}(T_{Ci}; \mu^*_{Ci}) \phi\left(\epsilon^*_j, \epsilon^*_k; \Sigma_1\right) \end{aligned}$$

  where $w^*_j = 2\sqrt{|\tilde{\Sigma}|} w_j \exp(\epsilon_j)$, $w^*_k = w_k \exp(\epsilon_k)$,$\log(\mu^*_R) = X_R\beta_R + Z_R b_R + \epsilon^*_j$ and $\log(\mu_C) = X_C\beta_C + Z_C b_C + \epsilon^*_k$.

Then, the total log likelihood is $L_{\texttt{BPLN}}(T_R, T_C) = \sum_{i=1}^{n} \log\left(f_{\texttt{BPLN}}(T_{Ri}, T_{Ci})\right)$. We can similarly extend this to evaluating $\log L(N_{Ci1}, N_{Ri1})$ corresponding to the $BBLN$.

AII.3 Maximization of the BPLN and BBLN log-likelihoods

Quasi-newton methods such as L-BFGS-B only require the calculation of the log-likelihood and the first derivatives of the log-likelihood for maximization. We show that the derivatives of the log-likelihood can be re-written into a form similar to the original likelihood function, allowing for straightforward evaluation of the derivatives.

The gradient of the log-likelihood takes the general form below:

$$\nabla L_{\texttt{BPLN}}\left(T_{Ri}, T_{Ci}\right) = \sum_{i=1}^{n} \frac{\nabla f_{\texttt{BPLN}}\left(T_{Ri}, T_{Ci}\right)}{L_{\texttt{BPLN}}\left(T_{Ri}, T_{Ci}\right)} \tag{AII.9}$$

Focusing on the numerator in equation (AII.9), we can derive the general forms of the derivative for the elements of β_R, b_R, Σ_1, and ρ_1. For the derivative with respect to b_R we have

$$\begin{aligned}
\frac{\partial}{\partial b_R} &= \frac{\partial}{\partial b_R} f_{\texttt{BPLN}}\left(T_{Ri}, T_{Ci}\right) \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{\partial}{\partial b_R} f_{Po}(T_{Ri};\mu_{Ri,\epsilon_{Ri}}) f_{Po}(T_{Ci};\mu_{Ci,\epsilon_{Ci}}) \phi\left(\epsilon_{Ri}, \epsilon_{Ci}; \Sigma\right) d\epsilon_{Ri} d\epsilon_{..} \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} C_i f_{Po}(T_{Ri};\mu_{Ri,\epsilon_{Ri}}) f_{Po}(T_{Ci};\mu_{Ci,\epsilon_{Ci}}) \phi\left(\epsilon_{Ri}, \epsilon_{Ci}; \Sigma\right) d\epsilon_{Ri} d\epsilon_{..} \\
&= \sum_{j=1}^{s}\sum_{k=1}^{s} C_i 2\sqrt{|\tilde{\Sigma}|} w_j w_k \exp(\epsilon_j)\exp(\epsilon_k) f_{\texttt{BPLN}}\left(T_{Ri}, T_{Ci}, \epsilon_j^*, \epsilon_k^*\right) \\
&= \sum_{j=1}^{s}\sum_{k=1}^{s} C_i w_j^* w_k^* \left(T_{Ri}, T_{Ci}, \epsilon_j^*, \epsilon_k^*\right)
\end{aligned}$$

where $C_i = Z_{Ri}\left(T_{Ri} - \exp\left(\mu_{Ri}\right)\right)$. Therefore, we can rewrite the first derivative with respect to b_R as a function of the original likelihood, allowing for evaluation with adaptive quadrature. This similarly holds for the other parameters in the model and also for the BBLN model. Using the likelihood and gradient functions, we can utilize quasi-newtown methods such as L-BFGS-b to maximize each model.

BIBLIOGRAPHY

[1] J. Aitchison and C. Ho. The multivariate poisson-log normal distribution. *Biometrika*, 76(4):643–653, 1989.

[2] T. L. Bailey, M. Boden, F. A. Buske, M. Frith, C. E. Grant, L. Clementi, J. Ren, W. W. Li, and W. S. Noble. Meme suite: tools for motif discovery and searching. *Nucleic Acids Res*, 37(Web Server issue):W202–8, 2009.

[3] A. Barski, S. Cuddapah, K. Cui, T. Roh, D. Schones, Z. Wang, G. Wei, I. Chepelev, K. Zhao, et al. High-resolution profiling of histone methylations in the human genome. *Cell*, 129(4):823–837, 2007.

[4] B. Bernstein, E. Birney, I. Dunham, E. Green, C. Gunter, M. Snyder, et al. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57, 2012.

[5] P. Bickel, Y. Ritov, and T. Ryden. Asymptotic normality of the maximum-likelihood estimator for general hidden markov models. *The Annals of Statistics*, 26(4):1614–1635, 1998.

[6] K. R. Blahnik, L. Dou, H. O'Geen, T. McPhillips, X. Xu, A. R. Cao, S. Iyengar, C. M. Nicolet, B. Ludascher, I. Korf, and P. J. Farnham. Sole-search: an integrated analysis program for peak detection and functional annotation using ChIP-seq data. *Nucleic Acids Res*, 38(3):e13, 2010.

[7] A. P. Boyle, J. Guinney, G. E. Crawford, and T. S. Furey. F-Seq: a feature density estimator for high-throughput sequence tags. *Bioinformatics*, 24(21):2537–2538, 2008.

[8] P. Breheny and J. Huang. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The annals of applied statistics*, 5(1):232, 2011.

[9] T. . G. P. Consortium. An integrated map of genetic variation from 1,092 human genomes. *Nature*, 491:1, 2012.

[10] D. Cox, G. Gudmundsson, G. Lindgren, L. Bondesson, E. Harsaae, P. Laake, K. Juselius, and S. Lauritzen. Statistical analysis of time series: Some recent developments [with discussion and reply]. *Scandinavian Journal of Statistics*, pages 93–115, 1981.

[11] P. J. Danaher and B. G. S. Hardie. Bacon with your eggs? applications of a new bivariate beta-binomial distribution. *The American Statistician*, 59(4):282–286, 2005.

[12] R. A. Davis, W. T. M. Dunsmuir, and S. B. Streett. Observation-driven models for poisson counts. *Biometrika*, 90(4):777–790, 2003.

[13] J. Degner, A. Pai, R. Pique-Regi, J. Veyrieras, D. Gaffney, J. Pickrell, S. De Leon, K. Michelini, N. Lewellen, G. Crawford, et al. Dnasei sensitivity qtls are a major determinant of human expression variation. *Nature*, 482(7385):390–394, 2012.

[14] S. Djebali, C. A. Davis, A. Merkel, A. Dobin, T. Lassmann, A. Mortazavi, A. Tanzer, J. Lagarde, W. Lin, F. Schlesinger, et al. Landscape of transcription in human cells. *Nature*, 489(7414):101–108, 2012.

[15] J. C. Dohm, C. Lottaz, T. Borodina, and H. Himmelbauer. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Res*, 36(16), 2008.

[16] B. Efron, R. Tibshirani, J. D. Storey, and V. Tusher. Empirical bayes analysis of a microarray experiment. *Journal of the American statistical association*, 96(456):1151–1160, 2001.

[17] F. Famoye. On the bivariate negative binomial regression model. *Journal of Applied Statistics*, 37(6):969–981, 2010.

[18] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

[19] J. Fan and J. Lv. A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, 20(1):101, 2010.

[20] F. Fang, E. Hodges, A. Molaro, M. Dean, G. Hannon, and A. Smith. Genomic landscape of human allele-specific dna methylation. *Proceedings of the National Academy of Sciences*, 109(19):7332–7337, 2012.

[21] A. P. Fejes, G. Robertson, M. Bilenky, R. Varhol, M. Bainbridge, and S. J. M. Jones. Findpeaks 3.1: a tool for identifying areas of enrichment from massively parallel short-read sequencing technology. *Bioinformatics*, 24(15):1729–1730, Aug 2008.

[22] D. W. Felsher, A. Zetterberg, J. Zhu, T. Tlsty, and J. M. Bishop. Overexpression of myc causes p53-dependent g2 arrest of normal fibroblasts. *Proceedings of the National Academy of Sciences*, 97(19):10544–10548, 2000.

[23] J. Friedman. Fast sparse regression and classification, 2008.

[24] R. Garcia, J. Ibrahim, and H. Zhu. Variable selection for regression models with missing data. *Statistica Sinica*, 20(1):149, 2010.

[25] P. G. Giresi, J. Kim, R. M. McDaniell, V. R. Iyer, and J. D. Lieb. Faire (formaldehyde-assisted isolation of regulatory elements) isolates active regulatory elements from human chromatin. *Genome Res*, 17(6):877–885, 2007.

[26] P. G. Giresi and J. D. Lieb. Isolation of active regulatory elements from eukaryotic chromatin using faire (formaldehyde assisted isolation of regulatory elements). *Methods*, 48(3):233–9, 2009.

[27] O. A. Hampton, P. Den Hollander, C. A. Miller, D. A. Delgado, J. Li, C. Coarfa, R. A. Harris, S. Richards, S. E. Scherer, D. M. Muzny, R. A. Gibbs, A. V. Lee, and A. Milosavljevic. A sequence-level map of chromosomal breakpoints in the mcf-7 breast cancer cell line yields insights into the evolution of a cancer genome. *Genome Res*, 19(2):167–77, 2009.

[28] J. Hartzel, A. Agresti, and B. Caffo. Multinomial logit random effects models. *Statistical Modelling*, 1(2):81–102, 2001.

[29] R. D. Hawkins, G. C. Hon, and B. Ren. Next-generation genomics: an integrative approach. *Nat Rev Genet*, 11(7):476–486, 2010.

[30] N. D. Heintzman, G. C. Hon, R. D. Hawkins, P. Kheradpour, A. Stark, L. F. Harp, Z. Ye, L. K. Lee, R. K. Stuart, C. W. Ching, K. a. Ching, J. E. Antosiewicz-Bourget, H. Liu, X. Zhang, R. D. Green, V. V. Lobanenkov, R. Stewart, J. a. Thomson, G. E. Crawford, M. Kellis, and B. Ren. Histone modifications at human enhancers reflect global cell-type-specific gene expression. *Nature*, 459(7243):108–12, May 2009.

[31] J. Hilbe. *Negative binomial regression*. Cambridge Univ Pr, 2011.

[32] L. W. Hillier, G. T. Marth, A. R. Quinlan, D. Dooling, G. Fewell, D. Barnett, P. Fox, J. I. Glasscock, M. Hickenbotham, W. Huang, V. J. Magrini, R. J. Richt, S. N. Sander, D. A. Stewart, M. Stromberg, E. F. Tsung, T. Wylie, T. Schedl, R. K. Wilson, and E. R. Mardis. Whole-genome sequencing and variant discovery in c. elegans. *Nat Methods*, 5(2):183–188, 2008.

[33] G. Hon, B. Ren, and W. Wang. Chromasig: a probabilistic approach to finding common chromatin signatures in the human genome. *PLoS Comput Biol*, 4(10):e1000201, 2008.

[34] J. G. Ibrahim. Incomplete data in generalized linear models. *Journal of the American Statistical Association*, 85(411):765–769, 1990.

[35] R. Jaenisch and A. Bird. Epigenetic regulation of gene expression: how the genome integrates intrinsic and environmental signals. *Nature genetics*, 33:245–254, 2003.

[36] H. Ji, H. Jiang, W. Ma, D. S. Johnson, R. M. Myers, and W. H. Wong. An integrated software system for analyzing chip-chip and chip-seq data. *Nat Biotechnol*, 26(11):1293–1300, 2008.

[37] R. Jothi, S. Cuddapah, A. Barski, K. Cui, and K. Zhao. Genome-wide identification of in vivo protein-DNA binding sites from chip-seq data. *Nucleic Acids Res*, 36(16):5221–5231, 2008.

[38] A. Khalili and J. Chen. Variable selection in finite mixture of regression models. *Journal of the American Statistical Association*, 102(479):1025–1038, 2007.

[39] P. V. Kharchenko, M. Y. Tolstorukov, and P. J. Park. Design and analysis of ChIP-seq experiments for DNA-binding proteins. *Nat Biotechnol*, 26(12):1351–9, 2008.

[40] T. Kim, Z. Abdullaev, A. Smith, K. Ching, D. Loukinov, R. Green, M. Zhang, V. Lobanenkov, and B. Ren. Analysis of the vertebrate insulator protein ctcf binding sites in the human genome. *Cell*, 128(6):1231, 2007.

[41] P. Kolasinska-Zwierz, T. Down, I. Latorre, T. Liu, X. S. Liu, and J. Ahringer. Differential chromatin marking of introns and expressed exons by h3k36me3. *Nature genetics*, 41(3):376–381, 2009.

[42] P. Kuan, D. Chung, G. Pan, J. Thomson, R. Stewart, and S. Keleş. A statistical framework for the analysis of chip-seq data. *Journal of the American Statistical Association*, 106(495):891–903, 2011.

[43] T. D. Laajala, S. Raghav, S. Tuomela, R. Lahesmaa, T. Aittokallio, and L. L. Elo. A practical comparison of methods for detecting transcription factor binding sites in chip-seq experiments. *BMC Genomics*, 10:618, 2009.

[44] D. Lambert. Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, 34(1):pp. 1–14, 1992.

[45] B. Li, M. Gogol, M. Carey, D. Lee, C. Seidel, and J. L. Workman. Combined action of phd and chromo domains directs the rpd3s hdac to transcribed chromatin. *Science*, 316(5827):1050–1054, 2007.

[46] F. Li, G. Mao, D. Tong, J. Huang, L. Gu, W. Yang, and G.-M. Li. The histone mark h3k36me3 regulates human dna mismatch repair through its interaction with mut$s\alpha$. *Cell*, 153(3):590–600, 2013.

[47] Y. Li, C. Willer, J. Ding, P. Scheet, and G. Abecasis. Mach: using sequence and genotype data to estimate haplotypes and unobserved genotypes. *Genetic epidemiology*, 34(8):816–834, 2010.

[48] C.-j. Liu, L. Prazak, M. Fajardo, S. Yu, N. Tyagi, and P. E. Di Cesare. Leukemia/lymphoma-related factor, a poz domain-containing transcriptional repressor, interacts with histone deacetylase-1 and inhibits cartilage oligomeric matrix protein gene expression and chondrogenesis. *Journal of Biological Chemistry*, 279(45):47081–47091, 2004.

[49] E. T. Liu, S. Pott, and M. Huss. Q&a: Chip-seq technologies and the study of gene regulation. *BMC Biol*, 8:56, 2010.

[50] J. S. Liu, R. Chen, and W. H. Wong. Rejection control and sequential importance sampling. *Journal of the American Statistical Association*, 93(443):1022–1031, 1998.

[51] Q. Liu and D. A. Pierce. A note on gauss—hermite quadrature. *Biometrika*, 81(3):624–629, 1994.

[52] C. B. Lozzio and B. B. Lozzio. Human chronic myelogenous leukemia cell-line with positive philadelphia chromosome. *Blood*, 45(3):321–334, 1975.

[53] D. S. Lun, A. Sherrid, B. Weiner, D. R. Sherman, and J. E. Galagan. A blind deconvolution approach to high-resolution mapping of transcription factor binding sites from chip-seq data. *Genome Biol*, 10(12):R142, 2009.

[54] T. Lystig and J. Hughes. Exact computation of the observed information matrix for hidden markov models. *Journal of Computational and Graphical Statistics*, 11(3):678–689, 2002.

[55] R. McDaniell, B. Lee, L. Song, Z. Liu, A. Boyle, M. Erdos, L. Scott, M. Morken, K. Kucera, A. Battenhouse, et al. Heritable individual-specific and allele-specific chromatin signatures in humans. *Science*, 328(5975):235–239, 2010.

[56] G. J. McLachlan. On the em algorithm for overdispersed count data. *Stat Methods Med Res*, 6(1):76–98, Mar 1997.

[57] X.-L. Meng and D. B. Rubin. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika*, 80(2):267–278, 1993.

[58] T. S. Mikkelsen, M. Ku, D. B. Jaffe, B. Issac, E. Lieberman, G. Giannoukos, P. Alvarez, W. Brockman, T.-K. Kim, R. P. Koche, W. Lee, E. Mendenhall, A. O'Donovan, A. Presser, C. Russ, X. Xie, A. Meissner, M. Wernig, R. Jaenisch, C. Nusbaum, E. S. Lander, and B. E. Bernstein. Genome-wide maps of chromatin state in pluripotent and lineage-committed cells. *Nature*, 448(7153):553–560, Aug 2007.

[59] D. A. Nix, S. J. Courdy, and K. M. Boucher. Empirical methods for controlling false positives and estimating confidence in chip-seq peaks. *BMC Bioinformatics*, 9:523, 2008.

[60] D. R. Nyholt. A simple correction for multiple testing for single-nucleotide polymorphisms in linkage disequilibrium with each other. *The American Journal of Human Genetics*, 74(4):765–769, 2004.

[61] S. Pepke, B. Wold, and A. Mortazavi. Computation for chip-seq and rna-seq studies. *Nat Methods*, 6(11 Suppl):S22–S32, 2009.

[62] J. Pickrell, J. Marioni, A. Pai, J. Degner, B. Engelhardt, E. Nkadori, J. Veyrieras, M. Stephens, Y. Gilad, and J. Pritchard. Understanding mechanisms underlying human gene expression variation with rna sequencing. *Nature*, 464(7289):768–772, 2010.

[63] Z. S. Qin, J. Yu, J. Shen, C. A. Maher, M. Hu, S. Kalyana-Sundaram, J. Yu, and A. M. Chinnaiyan. Hpeak: an HMM-based algorithm for defining read-enriched regions in ChIP-seq data. *BMC Bioinformatics*, 11:369, 2010.

[64] M. A. Quail, I. Kozarewa, F. Smith, A. Scally, P. J. Stephens, R. Durbin, H. Swerdlow, and D. J. Turner. A large genome center's improvements to the illumina sequencing system. *Nat Methods*, 5(12):1005–1010, 2008.

[65] A. R. Quinlan and I. M. Hall. Bedtools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842, 2010.

[66] O. J. Rando and H. Y. Chang. Genome-wide views of chromatin structure. *Annu Rev Biochem*, 78:245–71, 2009.

[67] N. Rashid, P. Giresi, J. Ibrahim, W. Sun, and J. Lieb. ZINBA integrates local covariates with DNA-seq data to identify broad and narrow regions of enrichment, even within amplified genomic regions. *Genome Biology*, 12(7):R67, 2011.

[68] B. Rhead, D. Karolchik, R. M. Kuhn, A. S. Hinrichs, A. S. Zweig, P. A. Fujita, M. Diekhans, K. E. Smith, K. R. Rosenbloom, B. J. Raney, A. Pohl, M. Pheasant, L. R. Meyer, K. Learned, F. Hsu, J. Hillman-Jackson, R. A. Harte, B. Giardine, T. R. Dreszer, H. Clawson, G. P. Barber, D. Haussler, and W. J. Kent. The ucsc genome browser database: update 2010. *Nucleic Acids Res*, 38(Database issue):D613–9, 2010.

[69] C. W. Roberts and S. H. Orkin. The swi/snf complex—chromatin and cancer. *Nature Reviews Cancer*, 4(2):133–142, 2004.

[70] J. Rozowsky, A. Abyzov, J. Wang, P. Alves, D. Raha, A. Harmanci, J. Leng, R. Bjornson, Y. Kong, N. Kitabayashi, et al. Alleleseq: analysis of allele-specific expression and binding in a network framework. *Molecular systems biology*, 7(1), 2011.

[71] J. Rozowsky, G. Euskirchen, R. K. Auerbach, Z. D. Zhang, T. Gibson, R. Bjornson, N. Carriero, M. Snyder, and M. B. Gerstein. Peakseq enables systematic scoring of ChIP-seq experiments relative to controls. *Nat Biotech*, 27(1):66–75, 2009.

[72] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

[73] L. Song, Z. Zhang, L. Grasfeder, A. Boyle, P. Giresi, B. Lee, N. Sheffield, S. Gräf, M. Huss, D. Keefe, et al. Open chromatin defined by dnasei and faire identifies

regulatory elements that shape cell-type identity. *Genome research*, 21(10):1757–1767, 2011.

[74] C. Spyrou, R. Stark, A. G. Lynch, and S. Tavaré. Bayespeak: Bayesian analysis of ChIP-seq data. *BMC Bioinformatics*, 10:299, 2009.

[75] W. Sun. A statistical framework for eqtl mapping using rna-seq data. *Biometrics*, 68(1):1–11, 2011.

[76] W. Sun, J. Ibrahim, and F. Zou. Genomewide multiple-loci mapping in experimental crosses by iterative adaptive penalized regression. *Genetics*, 185(1):349, 2010.

[77] M. L. Suvà, N. Riggi, and B. E. Bernstein. Epigenetic reprogramming in cancer. *Science*, 339(6127):1567–1570, 2013.

[78] R. E. Thurman, E. Rynes, R. Humbert, J. Vierstra, M. T. Maurano, E. Haugen, N. C. Sheffield, A. B. Stergachis, H. Wang, B. Vernot, et al. The accessible chromatin landscape of the human genome. *Nature*, 489(7414):75–82, 2012.

[79] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[80] C. Trapnell, L. Pachter, and S. Salzberg. Tophat: discovering splice junctions with rna-seq. *Bioinformatics*, 25(9):1105–1111, 2009.

[81] G. Tuteja, P. White, J. Schug, and K. H. Kaestner. Extracting transcription factor targets from ChIP-seq data. *Nucleic Acids Res*, 37(17):e113, 2009.

[82] A. Valouev, D. S. Johnson, A. Sundquist, C. Medina, E. Anton, S. Batzoglou, R. M. Myers, and A. Sidow. Genome-wide analysis of transcription factor binding sites based on ChIP-seq data. *Nat Methods*, 5(9):829–34, 2008.

[83] V. Vega, E. Cheung, N. Palanisamy, and W. Sung. Inherent signals in sequencing-based chromatin-immunoprecipitation control libraries. *PLoS One*, 4(4):e5241, 2009.

[84] G. G. Wang, L. Cai, M. P. Pasillas, and M. P. Kamps. Nup98–nsd1 links h3k36 methylation to hox-a gene activation and leukaemogenesis. *Nature cell biology*, 9(7):804–812, 2007.

[85] E. G. Wilbanks and M. T. Facciotti. Evaluation of algorithm performance in ChIP-seq peak detection. *PLoS One*, 5(7):e11471, 2010.

[86] J. Q. Wu and M. Snyder. Rna polymerase ii stalling: loading at the start prepares genes for a sprint. *Genome Biol*, 9(5):220, 2008.

[87] H. Xu, L. Handoko, X. Wei, C. Ye, J. Sheng, C. L. Wei, F. Lin, and W. K. Sung. A signal-noise model for significance analysis of ChIP-seq with negative control. *Bioinformatics*, 26(9):1199–204, 2010.

[88] H. Xu, C. L. Wei, F. Lin, and W. K. Sung. An HMM approach to genome-wide identification of differential histone modification sites from ChIP-seq data. *Bioinformatics*, 24(20):2344–9, 2008.

[89] C. Zang, D. E. Schones, C. Zeng, K. Cui, K. Zhao, and W. Peng. A clustering approach for identification of enriched domains from histone modification ChIP-seq data. *Bioinformatics*, 25(15):1952–8, 2009.

[90] S. Zeger. A regression model for time series of counts. *Biometrika*, 75(4):621, 1988.

[91] S. L. Zeger and B. Qaqish. Markov regression models for time series: a quasi-likelihood approach. *Biometrics*, 44(4):1019–31, 1988.

[92] J. Zeitlinger, A. Stark, M. Kellis, J. W. Hong, S. Nechaev, K. Adelman, M. Levine, and R. A. Young. Rna polymerase stalling at developmental control genes in the drosophila melanogaster embryo. *Nat Genet*, 39(12):1512–6, 2007.

[93] C. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

[94] Y. Zhang, T. Liu, C. A. Meyer, J. Eeckhoute, D. S. Johnson, B. E. Bernstein, C. Nusbaum, R. M. Myers, M. Brown, W. Li, et al. Model-based analysis of ChIP-Seq (MACS). *Genome Biol*, 9(9):R137, 2008.

[95] Y. Zhang, T. Liu, C. A. Meyer, J. Eeckhoute, D. S. Johnson, B. E. Bernstein, C. Nussbaum, R. M. Myers, M. Brown, W. Li, and X. S. Liu. Model-based analysis of chip-seq (macs). *Genome Biol*, 9(9), 2008.

[96] Z. D. Zhang, J. Rozowsky, M. Snyder, J. Chang, and M. Gerstein. Modeling ChIP sequencing in silico with applications. *PLoS Comput Biol*, 4(8):e1000158, 2008.

[97] H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

[98] H. Zou and R. Li. One-step sparse estimates in nonconcave penalized likelihood models. *Annals of Statistics*, 36(4):1509, 2008.