Integrative Statistical Models for Genomic Signal Detection

by

Yingying Wei

A dissertation submitted to The Johns Hopkins University
in conformity with the requirements for the degree of
Doctor of Philosophy

Baltimore, Maryland
March, 2014

Abstract

Although the cost of high-throughput technologies has decreased dramatically, it is still expensive to obtain a large number of biological replicates. On the other hand, with the wide adoption of high-throughput biology, multiple related genomic datasests are often available. The first two chapters tackle the challenging problem of borrowing information across multiple datasets, allowing context specificity, and overcoming the exponential growth of parameter space simultaneously to improve signal detection for noisy genomic data . Chapter 1 proposes a flexible Bayesian hierarchical mixture model to capture the latent correlation structures embedded in the data, named as "correlation motifs", and utilizes that piece of information to improve signal detection. The application is illustrated by differential gene expression detection when the expression datasets have only a small number of replicate samples. Chapter 2 demonstrates that a generalized version of the correlation motif approach can also help detect allele-specific protein-DNA binding from ChIP-seq data, which often suffers from low statistical power due to the limited number of sequence reads mapped to heterozygote SNPs. For both cases, the correlation motif approach substantially improves signal detection for low-signal-to-noise ratio data.

Moreover, the current high-throughput technologies such as immunoprecipitation (ChIP) with high-throughput sequencing (ChIP-seq) or tiling array hybridization (ChIP-chip) for studying protein-DNA interactions are "high-throughput" in terms of mapping a given type of transcription factor (TF) genome-widely. Nevertheless, mapping genome-wide binding sites of all TFs

in all biological contexts is a critical step toward understanding gene regulation. From this perspective, ChIP-seq and ChIP-chip are low-throughput with respect to surveying many TFs. Recent advances in genome-wide chromatin profiling, including development of technologies such as DNase-seq, FAIRE-seq and ChIP-seq for histone modifications, make it possible to predict *in vivo* TF binding sites by analyzing chromatin features at computationally determined DNA motif sites for many TFs simultaneously. Chapter 3 compares different models and discusses various issues arising from this new approach.

Readers

Hongkai Ji (Biostatistics)

Jeffrey Leek (Biostatistics)

Alan Scott (Molecular Microbiology and Immunology)

Jiou Wang (Biochemistry and Molecular Biology)

John Laterra (Alternative, Neurology)

Kasper Hansen (Alternative, Biostatistics)

Acknowledgements

I want to thank my advisor Hongkai Ji for his guidance, encouragement, and support throughout these years. I am extremely grateful for the freedom he has given me since my fourth year to explore areas that I am interested in such as survival analysis as well as his high standards for conducting scientific research. I greatly appreciate the kind help from John Laterra, Alan Scott, Michael Ochs, Rafael Irizarry, Jeffrey Leek, Kasper Hansen, and Jiou Wang, who are committee members on my oral exam or thesis defense. Their scientific vision and statistical insights help me gain deeper understandings of both science and statistics.

I always feel so fortunate to be able to spend the past five years at Hopkins Biostatistics Department, where I have been exposed to various working groups and have grown a broad interest in statistics. To begin with, my main research interests lie in statistical genomics, and I want to acknowledge faculty as well as students from the Joint Genomics Working Group. I would like to thank Rafael Irizarry, Ingo Ruczinski, Jeffrey Leek, Kasper Hansen, Elana Fertig, Hao Wu, Simina Boca, George Wu, Alyssa Frazee, Leonardo Collado Torres, and Jean-Philippe Fortin for all their help and inspiring discussions. Beyond geonomics, since my fourth year I have been truly fascinated with survival analysis by attending the weekly SLAM (Survival, Longitudinal, and Multilevel Modeling Working Group) group meetings. I want to express my deepest gratitude to Mei-cheng Wang, Chiung-yu Huang, Zheyu Wang, Gary Chan, Peng Huang, and Shanshan Li for all their enlightening discussions, precious academic opportunities such as meeting with eminent statisticians visiting SLAM, and advice

on my professional career. Meanwhile, I also want to thank my friends in the SMART group, especially Bruce Swihart, Brian Caffo, Ciprian Crainiceanu, Jeff Goldsmith, and Russell Shinohara, with whom I participated in the exciting Health Heritage Prize Competition (also together with Rafeal Irizarry) as well as Ani Eloyan and Vadim Zipunnikov. Moreover, I want to thank James Fill, Constantine Frangakis, Daniel Scharfstein, Thomas Louis, Kun-yee Liang, Elizabeth Colantuoni, Michael Rosenblum, and Martin Lindquist, whose courses prepared me for my research and piqued my interests in a wide range of topics. Consequently, I have to acknowledge Karen Bandeen-Roche for such a superb environment together with all her encouragement for me throughout my studies.

I am very thankful to my wonderful biology collaborators, John Laterra, Mingyao Ying, Michael Ochs, Joseph Califano, Junhao Mao, Kathy Niakan, Qianfei Wang, Ted Dawson, and Haisong Jiang for allowing me to work on very fascinating and thought-provoking scientific problems with them. I would also like to thank Xia Li, Jiawei Bai, and Shilu Zhang for helping me with data processing on various projects.

Special thanks to Marvin Newhouse, Fernando Pineda, Marti Gilbert, Mary Joy Argo, Ashley Gilliam, Patty Hubbard, and Jiong Yang, without whose help my thesis would be almost impossible.

I really treasure the great friendships I enjoy at Hopkins. Especially, I want to thank my classmates in various courses: Jeongyong Kim, Jenna Krall, Kirsten Lum, Paige Mass, Haochang Shou, Thomas Prior, Zhenke Wu, Sarah Khasawinah, Yang Ning, Fang Han, Yi Lu, Juemin Yang, Detian Deng, Lingyun Zhao, and Li Song . I am also deeply indebted to my friends outside the department,

Xin Zheng, Jinfang Ma, Zhen Shi, Yiyun Chen, Junjie Luo, Yi Gu, and Hua-Ling Tsai, who lent me tremendous amount of help for academics as well as life in the past five years. I save my profound gratitude to Denise Link-Farajali, a great teacher for my English, mentor for career, and friend for life.

Last but not least, I thank my grandmother, my uncle, my father and mother for their boundless love and support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Correlation Motif Model for Differential Gene Expression Detection

1.1 Introduction

Detecting differentially expressed genes is a basic task in the analysis of gene expression data. The state-of-the-art solutions to this problem, such as *limma* (Smyth, 2004), *SAM* (Tusher *and others*, 2001), edgeR (Robinson and Smyth, 2007, 2008) and DESeq (Anders and Huber, 2010), are mostly designed for analyzing data from a single experiment or study. With 1,000,000+ samples stored in public databases such as Gene Expression Omnibus (GEO), it is now very common for scientists to have data from multiple related experiments or studies. An emerging problem is how one can integrate data from multiple studies to more effectively analyze differential expression.

One example that motivated this article is a study of the vertebrate Sonic Hedgehog (SHH) signaling pathway. SHH is a signaling protein that can bind to PTCH1, a receptor protein in cell membrane (Figure 1.1(a)). PTCH1 can interact with another membrane protein SMO to repress its activity. In the absence

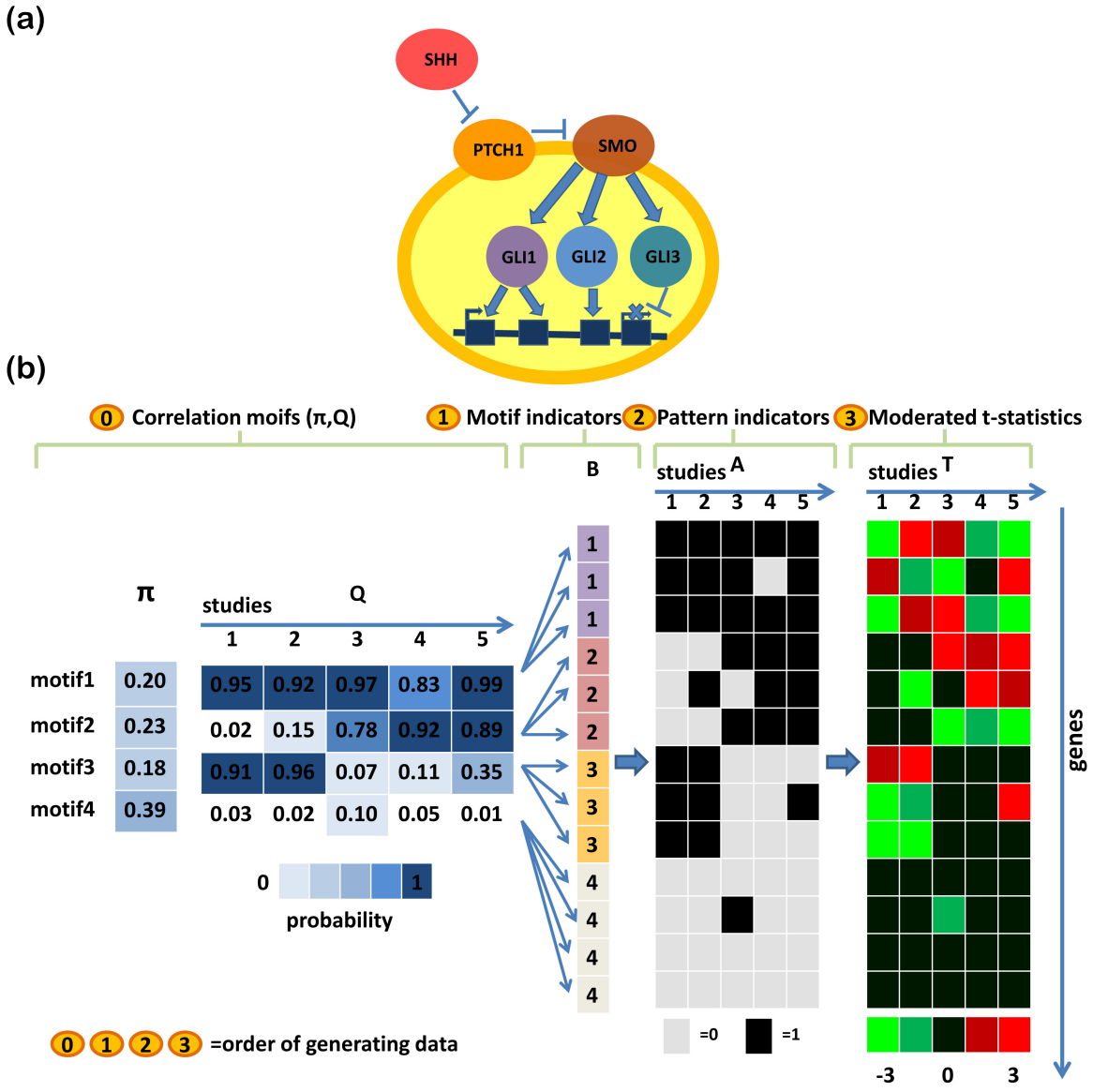


Figure 1.1: (a) A cartoon illustration of the SHH pathway. (b) A numerical example of the data generating model. There exist four motifs in the dataset, with the abundance $\boldsymbol{\pi} = (0.2, 0.23, 0.18, 0.39)$. Each row of the $\boldsymbol{Q}$ matrix represents a motif and each column corresponds to a study. The gray scale of the cells in $\boldsymbol{\pi}$ and $\boldsymbol{Q}$ illustrates the probability value. Given $\boldsymbol{\pi}$ and $\boldsymbol{Q}$, each gene is assigned a motif indicator b_g. For instance, the fifth gene belongs to motif 2. Next, the configuration of the fifth gene, $[a_{51}, a_{52}, a_{53}, a_{54}, a_{55}]$, is generated according to $\boldsymbol{q}_2 = (0.02, 0.15, 0.78, 0.92, 0.89)$. As a result, the fifth gene is differentially expressed in study 2,4 and 5. Finally, the moderated t-statistic t_{5d} within each study d is produced according to the configuration a_{5d}.

of SHH, PTCH1 keeps SMO inactive. The presence of SHH will repress PTCH1 and activate SMO. The active SMO triggers a signaling cascade by modulating activities of three transcription factors, GLI1, GLI2 and GLI3, which in turn will induce or repress the expression of hundreds of downstream target genes. The SHH pathway is one of the core signaling pathways in vertebrate development. It is associated with multiple types of tumors and birth defects (Ingham and McMahon, 2001; Villavicencio *and others*, 2000). To elucidate the underlying mechanism linking this pathway to diseases, multiple studies have been performed in different contexts to identify genes whose transcriptional activities are modulated by SHH signaling. Some studies perturb the SHH signal in different tissues by knocking out or over-expressing the pathway's key signal transduction components such as SHH, PTCH1 and SMO, while others compare disease samples with corresponding controls. Table 1.1 contains eight such datasets in mice originally generated and compiled by Tenzen *and others*, 2006 and Mao *and others*, 2006. Each dataset involves a comparison of genome-wide expression profiles between two different sample types. These data were all collected using Affymetrix Mouse Expression Set 430 arrays. The questions of biological interest include (1) which genes are controlled by the SHH signal in each dataset, (2) which genes are the core targets that respond to the SHH signal irrespective of tissue type and developmental stage, and (3) which genes are context-specific targets and are modulated by the SHH signal only in certain conditions. For simplicity, below we will call each dataset a *study*.

One simple approach to analyze these data is to analyze each study separately using existing state-of-the-art methods such as *limma* (Smyth, 2004) or *SAM* (Tusher *and others*, 2001). This approach is not ideal as it may fail to

Study ID	Condition 1 (case)	Sample No.	Condition 2 (control)	Sample No.	Reference
1	8somites_smo	3	8somites_wt	3	Tenzen *and others* (2006)
2	8somites_ptc	3	8somites_wt	3	Tenzen *and others* (2006)
3	13somites_ptc	3	13somites_wt	3	Tenzen *and others* (2006)
4	head_shh	3	head_wt	3	Tenzen *and others* (2006)
5	limb_shh	3	limb_wt	3	Tenzen *and others* (2006)
6	Medulloblastoma_tumor	3	Medulloblastoma_control	2	Mao *and others* (2006)
7	BCC_tumor	3	BCC_control	3	Mao *and others* (2006)
8	13somites_smo	3	13somites_wt	3	Tenzen *and others* (2006)

Table 1.1: SHH microarray data description. 8somites and 13somites indicate two different developmental stages of embryos; smo indicates mice with mutant Smo; ptc stands for mice with mutant Ptch1; wt means wild type; shh represents Shh mutant. Medulloblastoma and BCC (basal cell carcinoma) are two types of tumors.

detect genes with low fold changes but consistently differential in many or all studies.

Modeling all data jointly may allow one to borrow information across studies to improve the analysis. A simple model to combine data is to assume that each gene is either differential in all studies or non-differential in all studies (Conlon *and others*, 2006). This concordance model may help with identifying genes with small but consistent expression changes in all studies. However, it ignores the reality that activities of many important genes are tissue- or time-specific. This method will only produce a single gene list that reports and ranks genes in the same way for all studies. It cannot prioritize genes differently for different studies to account for context-specificity.

A more flexible approach is to consider all possible differential expression patterns. Suppose there are D studies and each gene can either be differential or non-differential in each study, there will be 2^D possible differential expression patterns. One can model the data as a mixture of 2^D different gene classes. This allows one to deal with context-specificity. However, an obvious drawback is that

as the number of studies increases, the number of possible patterns increases exponentially. Thus the model does not scale well with the increasing D.

Here, we propose a new method, *CorMotif*, for jointly analyzing multiple studies to improve differential expression detection. This method is both flexible for handling context-specificity and scalable to increasing study number. The key idea is to use a small number of latent probability vectors called "correlation motifs" to model the major correlation patterns among the studies. The motifs essentially group genes into clusters based on their differential expression patterns, and the differential gene detection is coupled with the clustering.

Previously, Kendziorski *and others* (2003) proposed a method for analyzing differential expression involving multiple biological conditions. This method, abbreviated as "eb1" hereinafter, requires users to specify all possible differential patterns, and the data are then modeled accordingly. If a user applies this method to detect differential expression between two conditions in multiple studies and wants to accommodate all possible differential patterns, the user has to enumerate all 2^D possible patterns, leading to the exponential complexity problem. Similar to Kendziorski *and others* (2003), Jensen *and others* (2009) developed a hierarchical Bayesian model and a Markov Chain Monte Carlo (MCMC) algorithm to analyze multiple conditions, again with exponential complexity due to requirement of enumerating all possible patterns. Ruan and Yuan (2011) generalized Kendziorski *and others* (2003) to a model that can integrate information from multiple studies where each study may involve comparisons of multiple conditions. Within each study, this method enumerates all possible combinatorial patterns among multiple conditions (again exponential complexity). Across studies, differential expression patterns are assumed to be

concordant; that is, each gene is assumed to have the same differential pattern in all studies. The concordance assumption does not allow study-specific differential expression.

Scharpf *and others* (2009) proposed a fully Bayesian framework, XDE, for cross-study differential expression analysis. It offers two implementations. The "Single-Indicator" implementation uses a concordance model by assuming that each gene's differential state is the same across all studies. The "Multiple-Indicator" implementation allows study-specific differential expression. However, it assumes that all genes have the same prior probability to be differential within the same study, and the differential states of each gene in different studies are a priori independent. Conceptually, these assumptions are similar to a *CorMotif* model with a single cluster, which often is insufficient to capture the heterogeneity among genes since the cross-study correlation pattern may vary from one gene to another (see details later). XDE does not have the exponential complexity problem, but it uses MCMC for posterior inference and is very slow computationally.

To capture the heterogeneity among genes, Yuan and Kendziorski (2006) developed a method for simultaneous clustering and differential expression analysis. Similar to *CorMotif*, this method also assumes that genes belong to multiple clusters, and different clusters have different propensities to show differential expression. However, Yuan and Kendziorski (2006) only considered detecting differential expression between two conditions in one study. Although one may conceptually extend this approach to handle multiple studies by combining it with the model developed by Kendziorski *and others* (2003), such a simple extension would lead to a model (called "eb10best" hereinafter) in which genes

are assumed to fall into multiple clusters and each cluster is a mixture of 2^D differential patterns. As a result, the complexity of the parameter space would become $O(K * 2^D)$, where K is the number of clusters.

In summary, none of the tools discussed above allows one to integrate information from multiple studies and also addresses study-specificity, heterogeneity among genes, and exponential complexity at the same time. These are the issues *CorMotif* attempts to solve. We organize this article as follows. Section 1.2 introduces the *CorMotif* model and algorithm. Section 1.3 uses simulations to demonstrate the approach. In Section 1.4, *CorMotif* will be applied to the SHH data. Section 1.5 will provide remarks and discussions. Here, we focus on discussing *CorMotif* for microarray data since it was motivated by the microarray analysis in the SHH study. However, the idea behind *CorMotif* is general, and it should be straight-forward to develop a similar framework for RNA-seq data. The link of the *CorMotif* R package is listed in Section 1.6, and the supplementary materials are laid out in Section 1.7.

1.2 Methods

1.2.1 Data Structure and Preprocessing

Suppose there are G genes and D microarray studies. Each study d compares two biological conditions (e.g., cancer vs. normal), and each condition l has n_{dl} replicate samples. Different studies may be related, but they can compare different biological conditions. Let x_{gdlj} denote the normalized and appropriately transformed expression value of gene g in study d, condition l and replicate j. In this article, all microarray data were normalized and log-transformed using

RMA (Irizarry *and others*, 2003). The collection of all observed data is

$$\boldsymbol{X} = \{x_{gdlj} : g = 1, \ldots, G; d = 1, \ldots, D; l = 1, 2; j = 1, \ldots, n_{dl}\}.$$

Each gene can be differentially expressed in some, all, or none of the studies. Let $a_{gd} = 1$ or 0 indicate whether gene g is differentially expressed in study d or not. $\boldsymbol{A} = (a_{gd})_{G\times D}$ is a $G \times D$ matrix that contains all a_{gd}s. Given the observed data $\boldsymbol{X}$, one is interested in inferring $\boldsymbol{A}$.

CorMotif first applies limma (Smyth, 2004) to each study separately. Define $\bar{x}_{gdl} = \sum_j x_{gdlj}/n_{dl}$, $n_d = n_{d1} + n_{d2}$ and $v_d = \frac{1}{n_{d1}} + \frac{1}{n_{d2}}$. For gene g and study d, compute the mean expression difference $y_{gd} = \bar{x}_{gd1} - \bar{x}_{gd2}$ and sample variance $s^2_{gd} = \sum_l \sum_j (x_{gdlj} - \bar{x}_{gdl})^2/(n_d - 2)$. The limma approach assumes that y_{gd}s and s^2_{gd}s within each study d follow a hierarchical model: (1) $[y_{gd}|\mu_{gd}, \sigma^2_{gd}] \sim N(\mu_{gd}, v_d\sigma^2_{gd})$, (2) $\mu_{gd} = 0$ if $a_{gd} = 0$, (3) $[\mu_{gd}|a_{gd} = 1, \sigma^2_{gd}] \sim N(0, w_d\sigma^2_{gd})$, (4) $[s^2_{gd}|\sigma^2_{gd}] \sim \frac{\sigma^2_{gd}}{n_d-2}\chi^2_{n_d-2}$, and (5) $[\frac{1}{\sigma^2_{gd}}] \sim \frac{1}{n_{0d}s^2_{0d}}\chi^2_{n_{0d}}$. Here w_d, n_{0d} and s^2_{0d} are unknown parameters. Their values can be estimated using the procedure described in Smyth (2004). This hierarchical model allows one to pool information across genes to stabilize the variance estimates. Smyth (2004) shows that it can significantly improve differential gene detection when the sample size n_d is small. For each study d, limma produces a moderated t-statistic for each gene g, computed as $t_{gd} = y_{gd}/\sqrt{v_d\tilde{s}^2_{gd}}$, where $\tilde{s}^2_{gd} = \frac{n_{0d}s^2_{0d}+(n_d-2)s^2_{gd}}{n_{0d}+n_d-2}$. This statistic summarizes gene g's differential expression information in study d. Under this model, when gene g is not differentially expressed in study d (i.e., $a_{gd} = 0$), t_{gd} follows a t-distribution $t_{n_{0d}+n_d-2}$; when $a_{gd} = 1$, t_{gd} follows a scaled t-distribution $(1 + w_d/v_d)^{1/2}t_{n_{0d}+n_d-2}$ (Smyth, 2004).

Next, we arrange all t_{gd}s into a matrix $\boldsymbol{T} = (t_{gd})_{G\times D}$. *CorMotif* will then use $\boldsymbol{T}$ instead of the raw expression values $\boldsymbol{X}$ to infer $\boldsymbol{A}$.

1.2.2 Correlation Motif Model

Organize the differential expression states of gene g into a vector $\boldsymbol{a}_g = [a_{g1}, a_{g2}, \cdots, a_{gD}]$. For D studies, $\boldsymbol{a}_g$ has 2^D possible configurations. A simple way to describe the correlation among studies is to document the empirical frequency of observing each of the 2^D configurations of $\boldsymbol{a}_g$ among all genes. This is because $f(\boldsymbol{a}_g)$, the joint distribution of $[a_{g1}, a_{g2}, \cdots, a_{gD}]$, is known once the probability of observing each configuration is given. This joint distribution will determine how a_{gd}s from different studies are correlated. While simple, this approach is not scalable since it requires $O(2^D)$ parameters and the parameter space expands exponentially with increasing D.

To avoid this limitation, *CorMotif* adopts a hierarchical mixture model (Figure 1.1(b)). The model assumes that genes fall into K different classes ($K \ll 2^D$), and the moderated t-statistics $\boldsymbol{T} = (t_{gd})_{G\times D}$ are viewed as generated as follows.

- First, each gene g is randomly and independently assigned a class label b_g according to probability $\boldsymbol{\pi} = (\pi_1, ..., \pi_K)$. Here, $\pi_k \equiv Pr(b_g = k)$ is the prior probability that a gene belongs to class k, and $\sum_k \pi_k = 1$.

- Second, given genes' class labels (i.e., b_gs), genes' differential expression states a_{gd}s are generated independently according to probabilities $q_{kd} \equiv Pr(a_{gd} = 1|b_g = k)$. For genes in the same class k, $\boldsymbol{a}_g$s are generated using the same probabilities $\boldsymbol{q_k} = (q_{k1}, ..., q_{kD})$.

- Third, given the differential expression states a_{gd}s, genes' moderated t-statistics t_{gd}s are generated independently according to $f_{d1}(t_{gd}) = f(t_{gd}|a_{gd} =$

$$1) \sim (1 + w_d/v_d)^{1/2} t_{n_{0d}+n_d-2} \text{ or } f_{d0}(t_{gd}) = f(t_{gd}|a_{gd} = 0) \sim t_{n_{0d}+n_d-2}.$$

Let $\boldsymbol{B} = (b_1, ..., b_G)$ be the class membership for all genes. Organize $\boldsymbol{q_k}$ into a matrix $\boldsymbol{Q} = (\boldsymbol{q_1^T}, \cdots, \boldsymbol{q_K^T})^T = (q_{kd})_{K\times D}$. Let $\delta(\cdot)$ be an indicator function: $\delta(\cdot) = 1$ if its argument is true, and $\delta(\cdot) = 0$ otherwise. Based on the above model, the joint probability distribution of $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{T}$ conditional on $\boldsymbol{\pi}$ and $\boldsymbol{Q}$ is:

$$Pr(\boldsymbol{T}, \boldsymbol{A}, \boldsymbol{B}|\boldsymbol{\pi}, \boldsymbol{Q}) = \prod_{g=1}^{G} \prod_{k=1}^{K} \{\pi_k \prod_{d=1}^{D} [q_{kd} f_{d1}(t_{gd})]^{a_{gd}} [(1 - q_{kd}) f_{d0}(t_{gd})]^{1-a_{gd}}\}^{\delta(b_g=k)} \tag{1.1}$$

According to this model, each gene class k is associated with a vector $\boldsymbol{q_k}$ whose elements are the prior probabilities of a gene in this class to be differential in studies $1, \ldots, D$. Each $\boldsymbol{q_k}$ represents a probabilistic differential expression pattern and therefore is called a "motif". Since q_{kd}s are probabilities, genes in the same class can have different $\boldsymbol{a}_g$ configurations. On the other hand, genes from the same class share the same $\boldsymbol{q_k}$, and hence their differential expression configuration $\boldsymbol{a}_g$s tend to be similar. Genes in different classes have different $\boldsymbol{q_k}$s, and their $\boldsymbol{a}_g$s also tend to be different. Essentially, our model groups genes into K clusters based on $\boldsymbol{a}_g$. However, unlike an usual clustering algorithm, here $\boldsymbol{a}_g$s are unknown.

Despite the assumption that a_{gd}s are a priori independent conditional on the class label b_g, a_{gd}s are no longer independent once the class label b_g is integrated out. To see this, consider the prior probability that a gene is differentially expressed in all studies. Based on our model, $Pr(\mathbf{a}_g = [1, \cdots, 1]) = \sum_k (\pi_k \prod_d q_{kd})$. A priori, the probability for a gene to be differential in study d is $Pr(a_{gd} = 1) = \sum_k \pi_k q_{kd}$. If a_{gd}s from different studies are independent, one would expect

$Pr(\mathbf{a}_g = [1, \cdots, 1]) = \prod_d Pr(a_{gd} = 1) = \prod_d (\sum_k \pi_k q_{kd})$ which is clearly different from $\sum_k (\pi_k \prod_d q_{kd})$. This explains why the hierarchical mixture model above can be used to describe the correlation among multiple studies. Since the mixture of $\boldsymbol{q_k}$s provides the key to model the cross-study correlation, each vector $\boldsymbol{q_k}$ is also called a "correlation motif".

A model with K correlation motifs requires $O(KD)$ parameters in total. Usually, a small K ($\ll 2^D$) is sufficient to capture the major correlation structure in the real data. Therefore, our method can be easily scaled up to deal with large D scenarios. When $0 < q_{kd} < 1$, each $\boldsymbol{q_k}$ will be able to generate all 2^D configurations with non-zero probabilities. Thus, our model also retains the flexibility to allow all 2^D configurations of $\boldsymbol{a}_g$ to occur at individual gene level.

1.2.3 Statistical Inference

In reality, only $\boldsymbol{T}$ is observed. $\boldsymbol{\pi}$ and $\boldsymbol{Q}$ are unknown parameters. $\boldsymbol{A}$ and $\boldsymbol{B}$ are unobserved missing data. To infer the unknowns from $\boldsymbol{T}$, we first assume that K is given and introduce a Dirichlet prior $Dir(2, ..., 2)$ for $\boldsymbol{\pi}$ and a Beta prior $B(2, 2)$ for q_{kd} such that:

$$\begin{aligned} Pr(\boldsymbol{\pi}, \boldsymbol{Q}, \boldsymbol{A}, \boldsymbol{B}|\boldsymbol{T}) \quad &\propto \quad \prod_{g=1}^{G} \prod_{k=1}^{K} \{\pi_k \prod_{d=1}^{D} [q_{kd} f_{d1}(t_{gd})]^{a_{gd}} [(1 - q_{kd}) f_{d0}(t_{gd})]^{1-a_{gd}}\}^{\delta(b_g = k)} \\ &* \quad \prod_{k=1}^{K} \pi_k \prod_{k=1}^{K} \prod_{d=1}^{D} q_{kd}(1 - q_{kd}) \end{aligned} \tag{1.2}$$

Based on the above posterior distribution, an expectation-maximization (EM) algorithm can be derived to search for the posterior mode of $\boldsymbol{\pi}$ and $\boldsymbol{Q}$ (Gelman *and others*, 2004). We chose the Dirichlet distribution $Dir(2, ..., 2)$ instead of $Dir(1, ..., 1)$ as prior since the mode of a Dirichlet distribution $Dir(\alpha_1, ..., \alpha_K)$

for the m^{th} component is $(\alpha_m - 1)/(\sum_{k=1}^{K} \alpha_k - K)$, which is zero when $\alpha_m = 1$ and not defined when all α_ks are equal to one. As a result, in the EM iterations, when a motif is associated with very few genes such that $\sum_{g=1}^{G} E(\delta(b_g = m)|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}})$ is close to zero, the estimate of π_m will become close to zero if we use $Dir(1, ..., 1)$. This will make the algorithm numerically unstable since the EM is implemented at logarithm scale (i.e., $log(\pi_m)$ instead of π_m is used in the implementation to avoid underflow when multiplying multiple probabilities). The same reason explains why $B(2, 2)$ was chosen as the prior for q_{kd}.

Using the estimated $\hat{\boldsymbol{\pi}}$ and $\hat{\boldsymbol{Q}}$, one can then compute $E(a_{gd}|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}}) = Pr(a_{gd} = 1|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}})$, the posterior probability that gene g is differentially expressed in study d. Next, we rank order genes in each study separately using $Pr(a_{gd} = 1|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}})$. The ranked lists can be used to choose follow-up targets. Users can also provide a posterior probability cutoff to dichotomize genes into *differential* or *non-differential* genes in each study. The default cutoff is 0.5.

In order to choose the motif number K, we use Bayesian Information Criterion (BIC). Details of the EM algorithm and BIC computation are provided in the Supplementary Materials Section 1.7.

CorMotif improves the differential expression detection by integrating information both across studies and across genes. $Pr(a_{gd} = 1|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}})$ can be decomposed as $\sum_{k=1}^{K} Pr(a_{gd} = 1|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}}, b_g = k) * Pr(b_g = k|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}})$. Here, $Pr(b_g = k|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}})$ is determined by jointly evaluating gene g's expression data in all studies, and $Pr(a_{gd} = 1|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}}, b_g = k)$ contains information specific to study d. According to Bayes' theorem, $Pr(a_{gd} = 1|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}}, b_g = k) \propto Pr(t_{gd}|a_{gd} = 1, \hat{\boldsymbol{Q}}, b_g = k) \times Pr(a_{gd} = 1|\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}}, b_g = k)$. t_{gd} in the first term that contains expression information for a given gene g in study

d. To compute its denominator, the limma approach also utilized information across genes to help with estimating the variance. Meanwhile, the second term $Pr(a_{gd} = 1|\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}}, b_g = k)$ involves prior probabilities given by the correlation motifs (i.e., $\hat{\boldsymbol{q}}_{\boldsymbol{k}}$s) which are estimated by examining data from all genes. Owing to this two-way information pooling (i.e., across both studies and genes), *CorMotif* uses information more effectively than methods based on only a single gene or a single study. This is especially useful for analyzing studies with relatively weak signal-to-noise ratio.

1.3 Simulations

1.3.1 Compared Methods

We compared *CorMotif* with six other methods: *separate limma*, *all concord*, *full motif*, *SAM*, *eb1*, *eb10best*. We did not compare the method in Jensen *and others* (2009) as no software was available for this method. The *separate limma* approach analyzes each study separately using limma. The moderated t-statistics in each study are assumed to be a mixture of $t_{n_{0d}+n_d-2}$ and $(1 + w_d/v_d)^{1/2} t_{n_{0d}+n_d-2}$. To better evaluate the gain from data integration, we matched this analysis to *CorMotif* as much as possible by running an EM algorithm similar to *CorMotif* to compute the posterior probability for differential expression using 0.5 as default cutoff. Conceptually, this makes *separate limma* equivalent to *CorMotif* with a single cluster ($K = 1$), and the analysis produces the same gene ranking as limma in each study. *All concord* assumes that a gene is either differentially expressed in all studies or non-differential in all studies (i.e., $\boldsymbol{a}_g = [1, 1, \ldots, 1]$ or $[0, 0, \ldots, 0]$). Conditional on $\boldsymbol{a}_g$, the

model for t_{gd} remains the same as *CorMotif* and limma. *Full motif* assumes that genes fall into 2^D classes, corresponding to the 2^D possible $\boldsymbol{a}_g$ configurations. It can be viewed as a saturated version of the *CorMotif* model. All the other methods are applied to x_{gdlj}s directly. *SAM* (Tusher *and others*, 2001) processes each study separately, whereas *eb1* and *eb10best* analyze all studies jointly. The *eb1* method corresponds to the R package EBarrays with lognormal-normal (LNN) and one cluster assumption (Kendziorski *and others*, 2003). The *eb10best* method is EBarrays with lognormal-normal and multiple cluster assumption, and the cluster number is chosen as the one with the lowest AIC among 1 to 10 (Yuan and Kendziorski, 2006). We also tried XDE (Scharpf *and others*, 2009). However, it took extremely long computing time, usually 24 hours on a machine with 2.7GHz CPU and 4Gb RAM for 1000 iterations, for an analysis involving four studies. Moreover, 1000 iterations usually were not enough for XDE to converge for an analysis consisting of four studies, which was the smallest data we analyzed here. Therefore, XDE will not be compared hereinafter. *eb10best* failed to work when it was used to jointly analyze ≥ 7 studies. *Full motif* and *eb1* failed when a dataset was composed of 20 studies.

1.3.2 Model-based Simulations

We first tested *CorMotif* using simulations. In simulation 1, we generated 10,000 genes and four studies according to the four differential patterns in Figure 1.2(a,b): 100 genes were differentially expressed in all four studies ($\boldsymbol{a}_g = [1,1,1,1]$); 400 genes were differential only in studies 1 and 2 ($[1,1,0,0]$); 400 genes were differential only in studies 2 and 3 ($[0,1,1,0]$); 9100 genes were non-differential ($[0,0,0,0]$). Each study had six samples: three cases and three

controls. The variances σ^2_{gd}s were simulated from a scaled inverse chi-square distribution $n_{0d}s^2_{0d}/\chi^2(n_{0d})$, where $n_{0d} = 4$ and $s^2_{0d} = 0.02$. Given σ^2_{gd}, the expression values were generated using $x_{gdlj} \sim N(0, \sigma^2_{gd})$. Whenever $a_{gd} = 1$, we drew μ_{gd} from $N(0, w_{0d} * \sigma^2_{gd})$ where $w_{0d} = 4$, and μ_{gd} was then added to the expression values of the three cases (i.e., x_{gd1j}s).

CorMotif was fit with the motif number K varying from 1 to 10. The K with the lowest BIC was chosen as the final motif number. In this way, four motifs were reported, and they were very similar to the true underlying differential patterns (Figure 1.2 (c)). To examine if *CorMotif* can improve gene ranking, for each study d we counted the number of true differential genes (true positives), $TP_d(r)$, among the top r ranked genes for each method, and we plotted $TP_d(r)$ versus r in Figure 1.2 (q,r,s,t). *CorMotif* consistently performed among the best in all studies. For instance, *CorMotif* identified 361 true differential genes among its top 500 gene list in study 1 (Figure 1.2(q)). This performance was almost the same as the saturated model *full motif*, which identified 362 true positives among the top 500 genes. Among the other methods, *eb10best* identified 341, *all concord* identified 292, and the others identified fewer than 292 true positives among the top 500 genes. Thus, *CorMotif* detected at least 23.6% more true positives compared to any other method except *full motif* and *eb10best*. Both *full motif* and *eb10best* have the problem of exponentially growing parameter space and will break down when the study number D is large. In addition, *eb10best* only identified 360 true positives among the top 1000 genes, whereas *CorMotif* identified 419, representing a 16.4% improvement.

In *CorMotif*, we labeled genes as differential if the posterior probability $Pr(a_{gd} = 1|\boldsymbol{T}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{Q}}) > 0.5$. Similarly, for *separate limma*, *all concord*, *full*

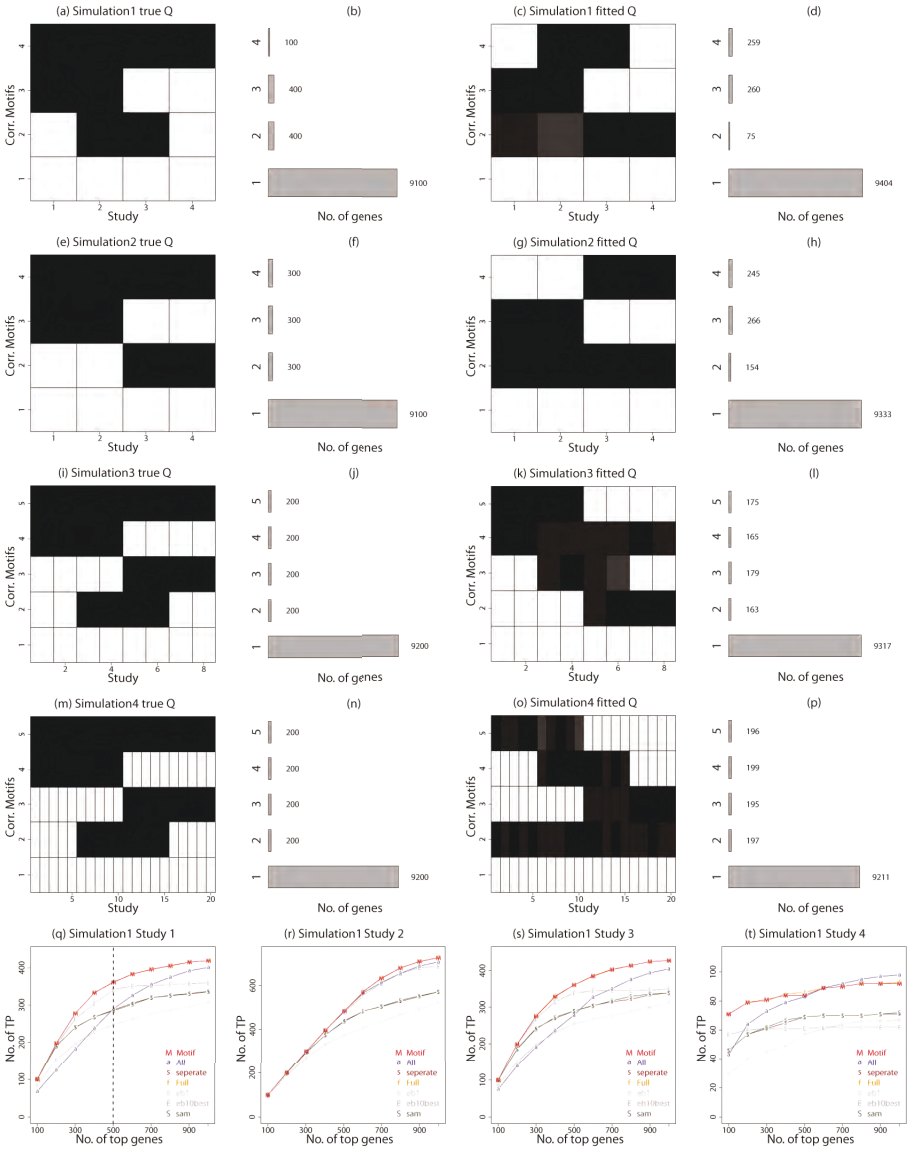


Figure 1.2: Results for the model assumption based simulations. Also see Supplementary Figure 1.4. (a),(e),(i),(m) Motif patterns for simulations 1-4. The $\boldsymbol{Q}$ of the true motifs in the simulated data. (b),(f),(j),(n) The true number of genes belonging to each motif in the simulated data (i.e., $\boldsymbol{\pi} * G$). (c),(g),(k),(o) The estimated $\hat{\boldsymbol{Q}}$ from the learned motifs. (d),(h),(l),(p) The estimated number of genes belonging to each learned motif (i.e., $\hat{\boldsymbol{\pi}} * G$). It can be seen that motif patterns learned by *CorMotif* are similar to the true underlying motif patterns. (q)-(t) Gene ranking performance of different methods in simulation 1. $TP_d(r)$, the number of genes that are truly differentially expressed in study d among the top r ranked genes by a given method, is plotted against the rank cutoff r.

Method	Motif pattern	$c(0,0,0,0)$	$c(0,1,1,0)$	$c(1,1,0,0)$	$c(1,1,1,1)$
CorMotif	$c(0,0,0,0)$	9072	161	165	16
	$c(0,1,1,0)$	3	168	3	7
	$c(1,1,0,0)$	3	2	151	6
	$c(1,1,1,1)$	0	1	0	33
	other	22	68	81	38
separate limma	$c(0,0,0,0)$	9035	144	144	16
	$c(0,1,1,0)$	0	68	0	5
	$c(1,1,0,0)$	0	0	57	6
	$c(1,1,1,1)$	0	0	0	4
	other	65	188	199	69
all concord	$c(0,0,0,0)$	9095	236	236	20
	$c(0,1,1,0)$	0	0	0	0
	$c(1,1,0,0)$	0	0	0	0
	$c(1,1,1,1)$	5	164	164	80
	other	0	0	0	0
full motif	$c(0,0,0,0)$	9072	161	164	16
	$c(0,1,1,0)$	4	172	4	7
	$c(1,1,0,0)$	3	2	155	6
	$c(1,1,1,1)$	0	1	0	35
	other	21	64	77	36
eb1	$c(0,0,0,0)$	62	0	2	0
	$c(0,1,1,0)$	2178	30	22	3
	$c(1,1,0,0)$	569	7	12	0
	$c(1,1,1,1)$	753	34	32	64
	others	5538	329	332	33
eb10best	$c(0,0,0,0)$	0	0	0	1
	$c(0,1,1,0)$	316	220	16	10
	$c(1,1,0,0)$	180	23	226	10
	$c(1,1,1,1)$	5789	77	52	63
	other	2815	80	106	16
SAM	$c(0,0,0,0)$	9099	256	279	48
	$c(0,1,1,0)$	0	20	0	3
	$c(1,1,0,0)$	0	0	9	2
	$c(1,1,1,1)$	0	0	0	1
	other	1	124	112	46

Table 1.2: Confusion matrix for simulation 1. The column labels indicate the true underlying patterns and the row labels represent the reported configurations at gene level. For *CorMotif*, *separate limma*, *all concord*, *full motif*, *eb1* and *eb10best*, differential expression in each study is determined using their default posterior probability cutoff 0.5. For *SAM*, q-value cutoff 0.1 was used to call differential expression. This yields similar number of correct classifications for pattern $[0,0,0,0]$ compared with *CorMotif*.

motif, *eb1* and *eb10best*, differential expression was determined using their default posterior probability cutoff 0.5. For *SAM*, q-value cutoff 0.1 was used to call differential expression. At this cutoff, *SAM* reported similar number of genes with $\boldsymbol{a}_g = [0, 0, 0, 0]$ (i.e., non-differential in all studies) compared with *CorMotif*. This allowed us to meaningfully compare *SAM* and *CorMotif* in terms of their ability to find differential genes. The confusion matrix in Table 1.2 shows that *CorMotif* was better at characterizing genes' true differential configurations compared to most other methods. For instance, among the 400 $[0, 1, 1, 0]$, 400 $[1, 1, 0, 0]$ and 100 $[1, 1, 1, 1]$ genes, *CorMotif* correctly reported differential label a_{gd} in all four studies for 168, 151 and 33 genes respectively. In contrast, *separate limma* only unmistakenly labeled 68, 57 and 4 genes respectively. *All concord* requires genes to have the same differential status in all studies. As such, it lacks the flexibility to handle study-specific differential expression. It correctly identified 80 out of 100 $[1, 1, 1, 1]$ genes, but none of the $[0, 1, 1, 0]$ and $[1, 1, 0, 0]$ genes were correctly labeled as study-specific. With the default cutoff, *eb1* and *eb10best* only labeled 62 and 0 out of 9100 $[0, 0, 0, 0]$ genes as completely non-differential, compared to 9072 labeled by *CorMotif* . In other words, *eb1* and *eb10best* reported more false positive differential expression events. At the same time, fewer $[0, 1, 1, 0]$ and $[1, 1, 0, 0]$ genes were correctly identified by *eb1* (30 and 12 vs. 168 and 151 by *CorMotif*). Similarly, *SAM* was also poor at identifying the differential expression patterns $[1, 1, 1, 1]$, $[1, 1, 0, 0]$ and $[0, 1, 1, 0]$. Among all the methods, only *full motif* performed slightly better than *CorMotif*. Even so, *CorMotif* was able to perform close to this saturated model. Adding up the diagonal elements in the confusion matrix for each method, *CorMotif* unmistakenly assigned $\boldsymbol{a}_g$ labels to 9424 genes,

whereas this number was 9164 for *separate limma*, 9175 for *all concord*, 9434 for *full motif*, 168 for *eb1*, 509 for *eb10best*, and 9129 for *SAM*.

Using a similar approach, we performed simulations 2-4 which involved different study numbers and differential expression patterns shown in Figure 1.2(e-p). The complete results are shown in Supplemental Material Figure 1.4 and Tables 1.4 to 1.6. The conclusions were similar to simulation 1. In particular, simulation 4 had 20 studies. *full motif*, *eb1* and *eb10best* all failed to run on this data.

1.3.3 Simulations Based on Real Data

In real data, the distributions for x_{gdlj}s may deviate from our model assumptions. Therefore, we further evaluated *CorMotif* using simulations that retained the real data noise structure. In simulation 5, 24 Human U133 Plus 2.0 Affymetrix microarray samples were downloaded from four GEO experiments. Each experiment corresponds to a different tissue and consists of six biological replicates (Supplemental Table 1.7). After RMA normalization, replicate samples in each experiment were split into three "cases" and three "controls". We then spiked in differential signals by adding random $N(0,1)$ deviates to the three cases according to patterns shown in Figure 1.5 (a-b). Data simulated in this way were able to keep the background characteristics in real data. Simulation 5 is similar to simulations 1 and 2. *CorMotif* again recovered the underlying differential patterns. It showed comparable differential gene detection performance to *full motif* and outperformed the other methods (Supplemental Figure 1.5 (e-h), Table 1.8). In a similar fashion, we performed simulations 6 and 7 based on real data (Supplemental Methods and Table 1.7). These two simulations have

the same differential signal patterns as simulations 3 and 4, respectively. Here, the motifs reported by *CorMotif* differ slightly from the underlying truth, but all the major correlation patterns were captured by the reported motifs. Once again, *CorMotif* performed the best in terms of differential gene detection (Supplemental Figure 1.5, Tables 1.9,1.10), and *eb1*, *eb10best* and *full motif* failed to run when the study number increased (when they failed, their results were not shown).

1.3.4 Motifs Are Parsimonious Representation of True Correlation Structures

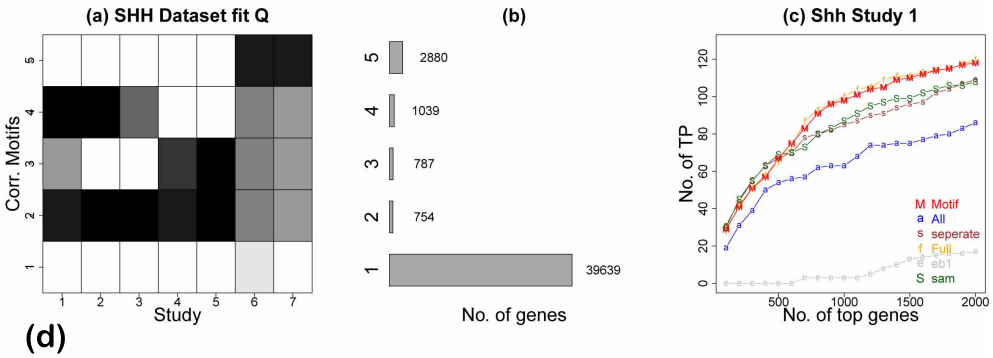
As we use probability vectors to serve as motifs, it is possible that multiple weak patterns can be merged into a single motif. For instance, two complementary patterns [1,1,0,0] and [0,0,1,1] each with n genes can be absorbed into a single motif with $\boldsymbol{q_k} = (0.5, 0.5, 0.5, 0.5)$ having $2n$ genes. To illustrate, we conducted simulations 8-10 which were composed of the same samples as in simulation 5 and various proportions of differential expression patterns (Supplemental Figure 1.6). In simulation 9 (Figure 1.6 (i-l)), the relative abundance of two complementary block motifs ([1,1,0,0] and [0,0,1,1]) was small compared to the concordance motif [1,1,1,1], and they were absorbed into a single motif. In simulations 5, 8 and 10 (Figure 1.6 (a-h),(m-p)), the complementary block motifs were more abundant, and the program successfully identified them as separate motifs. In general, we observed that weaker patterns were more likely to be merged than patterns with abundant data support. In all cases, however, *CorMotif* still provided the best gene ranking results compared to other methods (Supplemental Figure 1.7). Supplemental Figures 1.6 and 1.7 also show that

the higher the proportions of study-specific motifs (e.g., [1,1,0,0] and [0,0,1,1]), the better *CorMotif* will perform compared to the concordance analysis (i.e., *all concord*) in terms of ranking genes in each study. Together, the analyses here demonstrate that the correlation motifs only represent a parsimonious representation of the correlation structure supported by the available data. One should not expect *CorMotif* to always recover all the true underlying clusters exactly. In spite of this, our simulations show that *CorMotif* can still effectively utilize the correlation among studies to improve differential gene detection.

1.4 Application to the Sonic Hedgehog (Shh) Signaling Data Sets

We used *CorMotif* to analyze the SHH data in Table 1.1 Datasets 1 and 2 compare SMO mutant mice with wild type mice (wt) and PTCH1 mutant with wild type, respectively, in the 8 somite stage of developing embryos. Dataset 3 compares PTCH1 mutant with wild type in 13 somite stage. Datasets 4 and 5 compare SHH mutant with wild type in developing head and limb, respectively. Datasets 6 and 7 study gene expression changes in two SHH-related tumors, medulloblastoma and basal cell carcinoma (BCC), compared to normal samples (control). Dataset 8 compares SMO mutant with wild type in the 13 somite stage of developing embryos. *CorMotif* was applied to datasets 1-7. Dataset 8 was reserved for testing.

Five motifs were discovered (Figure 1.3(a,b)). Motif 1 mainly represents background. Motif 2 contains genes that have high probability to be differential in all studies. Genes in motif 3 tend to be differential in most studies except for the two involving PTCH1 mutant (i.e., studies 2 and 3). Most genes in motif 4



(d)

Gene Name	Method	Study 1	Study 2	Study 3	Study 5	Gene Name	Method	Study 1	Study 2	Study 3	Study 5
Gli1	Sep	■	■	■	■	Rab34	Sep				■
	Motif	■	■	■	■		Motif	■	■	■	■
	All	■	■	■	■		All	■	■	■	■
	Full	■	■	■	■		Full	■	■	■	■
	eb1		■	■	■		eb1				
	SAM	■	■	■	■		SAM				■
Ptch1	Sep	■	■	■	■	Hand2	Sep				■
	Motif	■	■	■	■		Motif				■
	All	■	■	■	■		All				
	Full	■	■	■	■		Full				■
	eb1		■	■	■		eb1				
	SAM	■	■	■	■		SAM				■
Ptch2	Sep	■			■	Hoxd13	Sep				■
	Motif	■	■	■	■		Motif				■
	All	■	■	■	■		All	■	■	■	■
	Full	■	■	■	■		Full				■
	eb1						eb1				■
	SAM				■		SAM				■
Hhip	Sep	■	■	■							
	Motif	■	■	■	■						
	All	■	■	■	■						
	Full	■	■	■	■						
	eb1		■	■							
	SAM		■								

Figure 1.3: Results for the SHH data. (a)-(b) Motif patterns learned from the SHH data. (c) Gene ranking performance for SHH study 1. The genes differentially expressed in dataset 8 (13somites_smo vs. 13somites_wt) were obtained using *separate limma*. They were used as the gold standard. $TP_d(r)$, the number of genes in dataset 1 that are truly differentially expressed among the top r ranked genes by each method, is plotted against the rank cutoff r. (d) Differential status claimed by each method for known SHH pathway genes. Dark blue indicates differential expression and light grey represents non-differential expression.

are not differential in the two studies involving the SHH mutant (i.e., studies 4 and 5) but tend to be differential in all other studies. Motif 5 mainly represents genes with differential expression in tumors (i.e., studies 6 and 7) but not in embryonic development (i.e., studies 1-5). In general, looking at the columns in Figure 1.3(a), the two studies involving tumors (6,7) are more similar to each other compared to other studies. The two PTCH1 mutant studies (2,3) are also relatively similar, and the same trend holds true for the two SHH mutant studies (4,5).

In this real data analysis, no comprehensive truth is available for evaluating differential expression calls. Without comprehensive knowledge about the true differential expression states of all genes in all cell types, we can only perform a partial evaluation based on existing knowledge. In this regard, we used dataset 8 as a test. Similar to dataset 1, this dataset compares SMO mutant with wild type. One expects that differential genes in these two datasets should be largely similar. Therefore, we used the top 217 differentially expressed genes detected by *separate limma* (at the posterior probability cutoff 0.5) in dataset 8 as gold standard to evaluate the gene ranking performance of different methods in dataset 1. Figure 1.3(c) shows that *CorMotif* again performed similar to *full motif* and outperformed all other methods. *eb10best* failed to run here. We note that since dataset 8 and datasets 2-7 represent more different biological contexts, one cannot use it as gold standard for evaluating these other datasets.

Finally, we examined well-studied SHH responsive target genes. Gli1, Ptch1, Ptch2, Hhip and Rab34 are known to be regulated by SHH signaling in somites and developing limb (Vokes *and others*, 2007, 2008). Therefore, we expect these genes to be differential in studies 1, 2, 3 and 5. Figure 1.3(d) shows that

Table 1.3: Ranks of known SHH target genes by each method in the SHH analysis.

Gene name	Analysis Method	Study 1	Study 2	Study 3	Study 4	Study 5	Study 6	Study 7
Gli1	*separate limma*	6	7	16	9	7	1369	515
	CorMotif	5	6	7	7	6	930	324
	all concord	9	9	9	9	9	9	9
	full motif	5	7	7	4	5	809	308
	SAM	7	6	17	9	10	1627	583
	eb1	33396	25	36	24	24	1828	720
Ptch1	*separate limma*	7	19	4	4	2	783	19
	CorMotif	6	20	8	4	3	495	12
	all concord	5	5	5	5	5	5	5
	full motif	7	16	4	3	2	409	14
	SAM	6	18	5	4	2	964	25
	eb1	13455	8	6	9	4	1464	289
Ptch2	*separate limma*	273	607	9996	1527	458	2530	117
	CorMotif	140	437	462	356	264	1848	69
	all concord	40	40	40	40	40	40	40
	full motif	145	450	482	285	256	1686	70
	SAM	303	630	9066	1431	468	2488	95
	eb1	7331	579	838	727	433	418	161
Hhip	*separate limma*	105	25	31	580	2964	13452	6
	CorMotif	61	19	27	264	652	9259	2
	all concord	22	22	22	22	22	22	22
	full motif	58	22	28	249	632	8529	2
	SAM	107	24	20	597	2903	16223	7
	eb1	6111	32	10	353	326	7462	131
Rab34	*separate limma*	927	553	299	577	396	15782	241
	CorMotif	324	401	164	176	261	10418	150
	all concord	160	160	160	160	160	160	160
	full motif	386	372	139	194	274	9546	151
	SAM	953	613	450	619	430	15923	171
	eb1	1371	1333	1042	1130	1074	12564	1019
Hand2	*separate limma*	34351	11862	6647	6061	196	20672	44939
	CorMotif	3601	3394	2794	1036	544	13371	17909
	all concord	4987	4987	4987	4987	4987	4987	4987
	full motif	3327	3021	2460	917	550	12585	14457
	SAM	34455	12375	8381	6582	207	22592	44945
	eb1	28270	2191	3040	1650	571	23269	33457
Hoxd13	*separate limma*	6805	7572	1893	10644	12	26047	9676
	CorMotif	1990	2371	1746	1223	93	15204	5734
	all concord	933	933	933	933	933	933	933
	full motif	1943	2490	1246	1064	88	14041	4722
	SAM	6724	7763	2684	10553	12	27578	8579
	eb1	6919	804	696	641	14	26742	12464

CorMotif, *all concord* and *full motif* were able to correctly identify differential expression of these genes in all these studies, whereas *separate limma*, *SAM* and *eb1* failed to do so (they missed some cases). Table 1.3 also shows that in many studies, *CorMotif*, *all concord* and *full motif* provided better rank for these genes compared to *separate limma*, *SAM* and *eb1*. Hand2 is known to be a target of SHH signaling in developing limb but not in somites (Vokes *and others*, 2008). While *separate limma*, *CorMotif*, *full motif* and *SAM* can correctly identify this, *all concord* and *eb1* failed to do so. For *all concord*, since Hand2 was not differential in studies 1-4, 6 and 7, the method thinks that this gene is not differential in any study. Similarly, Hoxd13 is a limb specific target of SHH signaling (Vokes *and others*, 2008). While the other methods correctly identified this, *all concord* failed again by claiming it to be differential in all studies. In all the genes examined, only *CorMotif* and *full motif* were able to correctly identify all known differential states. Together, our analyses show that *CorMotif* offers unique advantage over the other methods in the integrative analysis of multiple gene expression studies.

1.5 Discussion

In summary, we have proposed a flexible and scalable approach for integrative analysis of differential gene expression in multiple studies. Using a few probability vectors instead of 2^D dichotomous vectors to characterize the differential expression patterns provides the key to circumvent the challenge of exponential growth of parameter space as the study number increases. The probabilistic nature of the motifs also allows all 2^D differential patterns to occur in the data

at indiviual gene level.

The motif matrix $\boldsymbol{Q}$ can be viewed in two different ways. On one hand, each row of $\boldsymbol{Q}$ represents a cluster of genes with similar differential expression patterns across studies. Having many different motifs in $\boldsymbol{Q}$ is an indication that a concordance model, such as *all concord*, may not be sufficient to describe the correlation structure in the data. On the other hand, each column of $\boldsymbol{Q}$ represents differential expression propensities of different gene classes in a given study. If two columns are similar, the corresponding studies share similar differential expression profiles (e.g., studies 6 and 7 in the SHH data are more similar to each other compared to the other studies in the same data).

CorMotif is computationally efficient. It took $\sim$ 0.5 hour to analyze the SHH data for a given K, and 5.19 hours in total to run all Ks from 1 to 10. As a comparison, both *eb10best* and XDE failed, and *eb1* took 2.51 hours. *separate limma* (2.09 minutes) and *SAM* (1.71 minutes) were faster since each single study was processed separately each time. The relative efficiency of *CorMotif* is partly because we simplified the computation by modeling the moderated t-statistics t_{gd} instead of the raw expression values x_{gdlj}s. In addition, we used EM instead of the more time-consuming MCMC to fit the model. Despite these simplifications, our results show that the present model robustly performs comparable or better than the alternative methods. A potential future work is to couple the correlation motif idea with more sophisticated models for the raw data x_{gdlj} and explore whether the analysis can be improved further.

The *correlation motif* framework is general. Conceptually, one can modify the data generating distributions f_{d0} and f_{d1} to accommodate other data types, and use the same framework for a variety of meta-analysis problems. For

example, with appropriate modification to f_{d0}s and f_{d1}s, the *correlation motif* idea should be directly applicable to RNA-seq data. Nevertheless, a systematic treatment of RNA-seq analysis is beyond the scope of this paper.

1.6 Software

CorMotif is freely available as an R package in Bioconductor:

http://www.bioconductor.org/packages/release/bioc/html/Cormotif.html.

1.7 The Supplementary Materials for Cormotif

1.7.1 The EM Algorithm used in Cormotif

This section presents the EM algorithm used to search for posterior mode of $\hat{\boldsymbol{\pi}}$ and $\hat{\boldsymbol{Q}}$ of the distribution $Pr(\boldsymbol{\pi}, \boldsymbol{Q}|\boldsymbol{T}) = \sum_{\boldsymbol{A},\boldsymbol{B}} Pr(\boldsymbol{\pi}, \boldsymbol{Q}, \boldsymbol{A}, \boldsymbol{B}|\boldsymbol{T})$. In the EM algorithm, $\boldsymbol{A}$ and $\boldsymbol{B}$ are missing data. The algorithm iterates between the E-step and the M-step.

In the E-step, one evaluates the Q-function $Q(\boldsymbol{\pi}, \boldsymbol{Q}|\hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{Q}}^{\boldsymbol{old}})$ which is defined as $E_{old}[\ln Pr(\boldsymbol{\pi}, \boldsymbol{Q}, \boldsymbol{A}, \boldsymbol{B}|\boldsymbol{T})]$. Here the expectation is taken with respect to distribution $Pr(\boldsymbol{A}, \boldsymbol{B}|\boldsymbol{T}, \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{Q}}^{\boldsymbol{old}})$, abbreviated as $Pr_{old}(\boldsymbol{A}, \boldsymbol{B})$, where $\hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{Q}}^{\boldsymbol{old}}$ are the parameter estimates obtained from the last iteration.

We have

$$\ln Pr(\boldsymbol{\pi}, \boldsymbol{Q}, \boldsymbol{A}, \boldsymbol{B}|\boldsymbol{T}) = \sum_{g=1}^{G}\sum_{k=1}^{K} \delta(b_g = k) \ln \pi_k$$

$$+ \sum_{g=1}^{G}\sum_{k=1}^{K} \delta(b_g = k) \left\{ \sum_{d=1}^{D} a_{gd}[\ln q_{kd} + \ln f_{d1}(x_{gd})] \right\}$$

$$+ \sum_{g=1}^{G}\sum_{k=1}^{K} \delta(b_g = k) \left\{ \sum_{d=1}^{D} (1 - a_{gd})[\ln(1 - q_{kd}) + \ln f_{d0}(x_{gd})] \right\}$$

$$+\sum_{k=1}^{K}\ln\pi_k+\sum_{k=1}^{K}\sum_{d=1}^{D}[\ln q_{kd}+\ln(1-q_{kd})]+constant \tag{1.3}$$

Therefore,

$$\begin{aligned}
&Q(\boldsymbol{\pi},\boldsymbol{Q}|\hat{\boldsymbol{\pi}}^{\boldsymbol{old}},\hat{\boldsymbol{Q}}^{\boldsymbol{old}})=E_{old}[\ln Pr(\boldsymbol{\pi},\boldsymbol{Q},\boldsymbol{A},\boldsymbol{B}|\boldsymbol{T})]\\
&=\sum_{g=1}^{G}\sum_{k=1}^{K}\ln\pi_k E_{old}(\delta(b_g=k))\\
&+\sum_{g=1}^{G}\sum_{k=1}^{K}\sum_{d=1}^{D}[\ln q_{kd}+\ln f_{d1}(x_{gd})]E_{old}(\delta(b_g=k)a_{gd})\\
&+\sum_{g=1}^{G}\sum_{k=1}^{K}\sum_{d=1}^{D}[\ln(1-q_{kd})+\ln f_{d0}(x_{gd})]E_{old}(\delta(b_g=k)(1-a_{gd}))\\
&+\sum_{k=1}^{K}\ln\pi_k+\sum_{k=1}^{K}\sum_{d=1}^{D}[\ln q_{kd}+\ln(1-q_{kd})]+constant
\end{aligned} \tag{1.4}$$

In the M-step, one finds $\boldsymbol{\pi}$ and $\boldsymbol{Q}$ that maximize the Q-function $Q(\boldsymbol{\pi},\boldsymbol{Q}|\hat{\boldsymbol{\pi}}^{\boldsymbol{old}},\hat{\boldsymbol{Q}}^{\boldsymbol{old}})$. Denote them as $\hat{\boldsymbol{\pi}}^{\boldsymbol{new}}$ and $\hat{\boldsymbol{Q}}^{\boldsymbol{new}}$ and they will be used in next iteration.

By solving

$$\frac{\partial Q(\boldsymbol{\pi},\boldsymbol{Q}|\hat{\boldsymbol{\pi}}^{\boldsymbol{old}},\hat{\boldsymbol{Q}}^{\boldsymbol{old}})}{\partial\pi_k}=0 \tag{1.5}$$

$$\frac{\partial Q(\boldsymbol{\pi},\boldsymbol{Q}|\hat{\boldsymbol{\pi}}^{\boldsymbol{old}},\hat{\boldsymbol{Q}}^{\boldsymbol{old}})}{\partial q_{kd}}=0 \tag{1.6}$$

We have

$$\hat{\pi}_k^{new}=\frac{\sum_{g=1}^{G}Pr_{old}(b_g=k)+1}{G+K} \tag{1.7}$$

$$\hat{q}_{kd}^{new}=\frac{\sum_{g=1}^{G}Pr_{old}(b_g=k,a_{gd}=1)+1}{\sum_{g=1}^{G}Pr_{old}(b_g=k)+2} \tag{1.8}$$

In the formulae above, $Pr_{old}(b_g = k)$ and $Pr_{old}(b_g = k, a_{gd} = 1)$ can be computed as below

$$Pr_{old}(b_g = k) = \frac{\hat{\pi}_k^{(old)} \prod_{d=1}^{D} [\hat{q}_{kd}^{(old)} f_{d1}(t_{gd}) + (1 - \hat{q}_{kd}^{(old)}) f_{d0}(t_{gd})]}{\sum_{l=1}^{K} \hat{\pi}_l^{(old)} \prod_{d=1}^{D} [\hat{q}_{ld}^{(old)} f_{d1}(t_{gd}) + (1 - \hat{q}_{ld}^{(old)}) f_{d0}(t_{gd})]} \tag{1.9}$$

$$\begin{aligned} & Pr_{old}(b_g = k, a_{gd} = 1) = Pr_{old}(a_{gd} = 1 | b_g = k) * Pr_{old}(b_g = k) \\ & = \frac{\hat{q}_{kd}^{(old)} f_{d1}(t_{gd})}{\hat{q}_{kd}^{(old)} f_{d1}(t_{gd}) + (1 - \hat{q}_{kd}^{(old)}) f_{d0}(t_{gd})} Pr_{old}(b_g = k) \end{aligned} \tag{1.10}$$

Therefore, we can iteratively use the EM algorithm to obtain the estimates for π and Q.

1.7.2 Bayesian Information Criterion (BIC) for Choosing k

BIC is computed as

$$\begin{aligned} BIC(K) &= -2 * \ln Pr(\boldsymbol{T} | \boldsymbol{\pi}, \boldsymbol{Q}) + (K - 1 + K * D) * \ln G \qquad (1.11) \\ &= -2 * \sum_{g=1}^{G} \ln \left[\sum_{k=1}^{K} \{ \pi_k \prod_{d=1}^{D} [q_{kd} f_{d1}(t_{gd}) + (1 - q_{kd}) f_{d0}(t_{gd})] \} \right] \\ &\quad + (K - 1 + K * D) * \ln G \end{aligned}$$

BIC for different values of K are calculated and the K corresponding to the model that achieves the smallest BIC is chosen. Here K is the number of motifs in the data and $K - 1$ is the number of parameters for $\boldsymbol{\pi}$. KD is the number of parameters involved in $\boldsymbol{Q}$. G is the gene number.

1.7.3 Data for Real Data Based Simulations

Simulations 5-10 were based on real data characteristics. Each simulation contained multiple studies, and each study was composed of six samples from the same GEO experiment with the same biological condition as detailed in Table 1.7. The six samples were further split into three pseudo cases and three pseudo controls. They were used as the simulated background since one does not expect differential signals between replicate samples. We then spiked in differential signals by adding random $N(0, 1)$ numbers to the three cases according to the patterns shown in Figures 1.5 (a-b,i-j,q-r) and 1.6(a-b,e-f,i-j,m-n). Data simulated in this way were able to keep the background characteristics in real data.

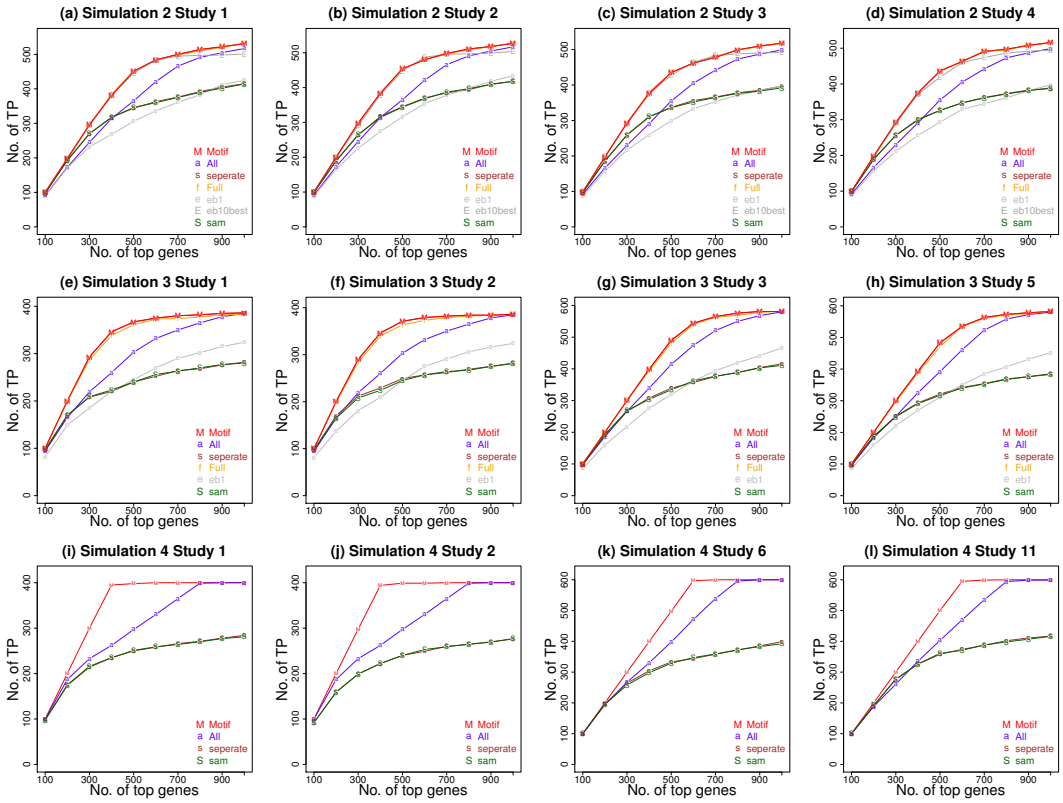


Figure 1.4: Gene ranking performance for simulations 2, 3 and 4. $TP_d(r)$, the number of genes that are truly differentially expressed in study d among the top r ranked genes by a given method, is plotted against the rank cutoff r. Simulations 3 and 4 contain more than four studies, and results for four representative studies are shown. (a)-(d) Simulation 2. (e)-(h) Simulation 3. Studies 1 and 2 are representative for patterns in studies 1, 2 and 7, 8; studies 3 and 5 are representative for patterns in studies 3 to 6. (i)-(l) Simulation 4. Studies 1 and 2 are representative for patterns in studies 1-5 and 16-20; studies 6 and 11 are representative for patterns in studies 6-15.

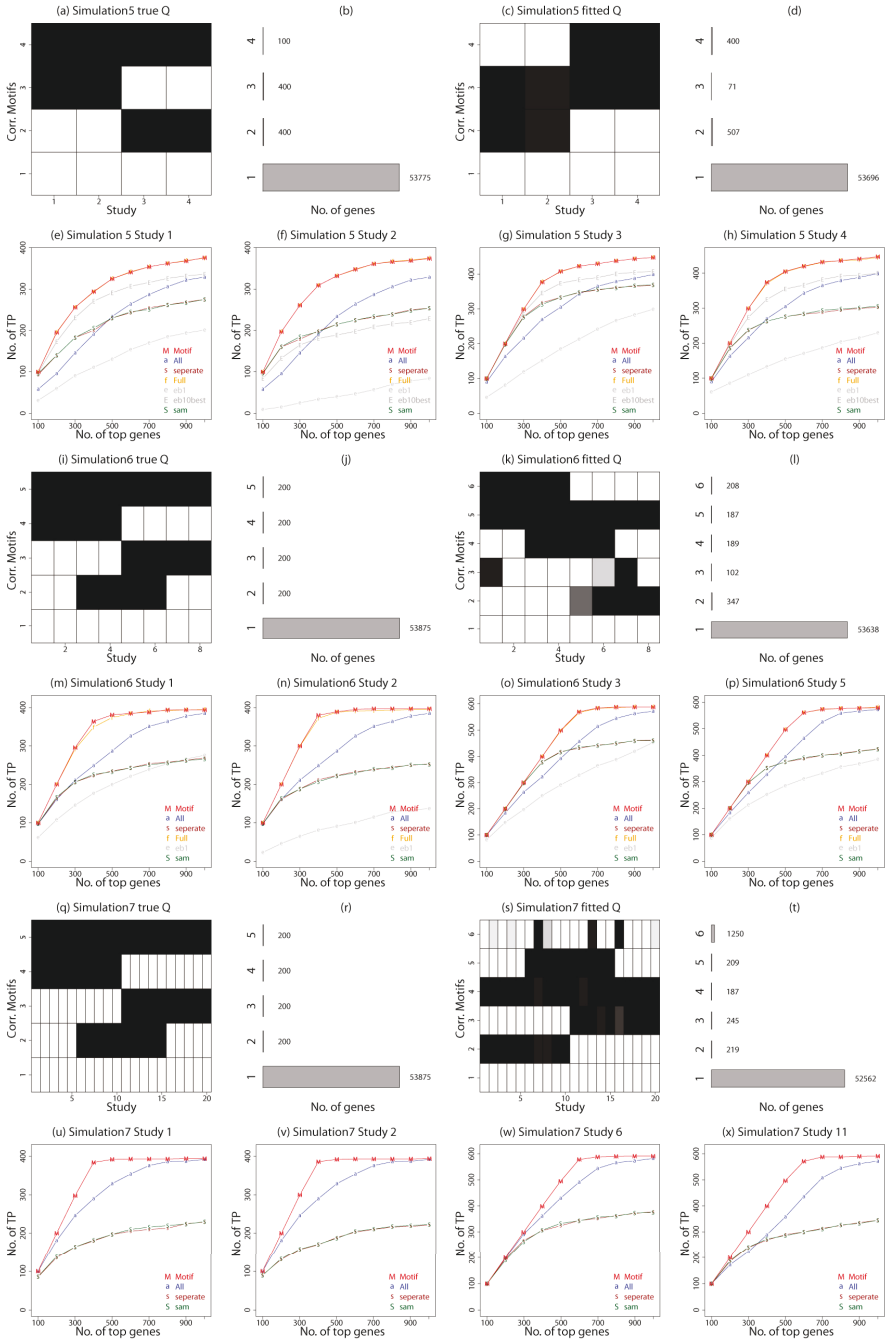


Figure 1.5: Motif patterns and gene ranking performance for simulations 5, 6 and 7. (a)-(d) True and estimated motif patterns for simulation 5. (e)-(h) Gene ranking performance for simulation 5. (i-l) Motif patterns for simulation 6. (m-p) Gene ranking performance for simulation 6. (q-t) Motif patterns for simulation 7. (u)-(x) Gene ranking performance for simulation 7.

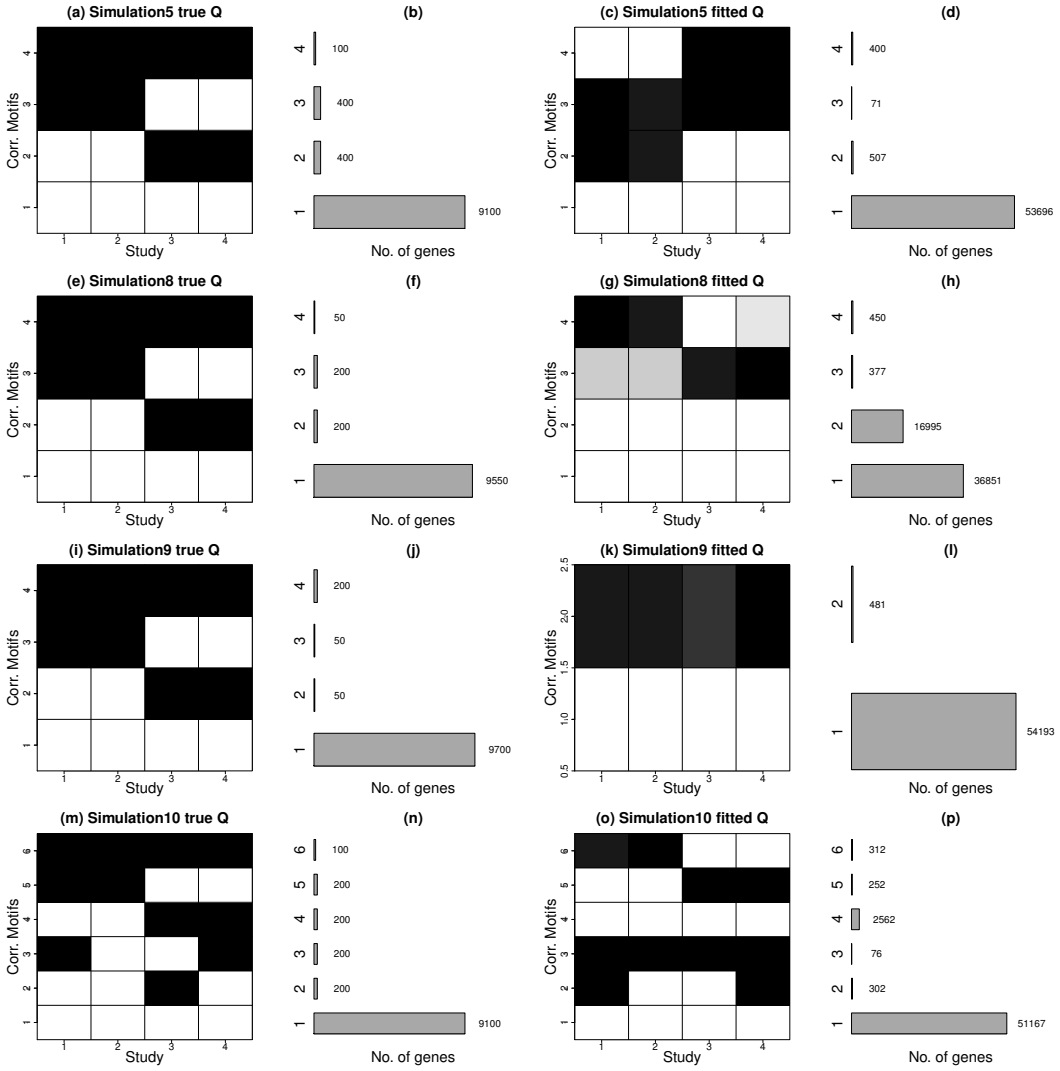


Figure 1.6: Motif patterns for simulations 5, 8, 9 and 10. (a),(e),(i),(m) The $\boldsymbol{Q}$ for the true underlying motifs in the simulated data. (b),(f),(j),(n) The true number of genes belonging to each motif in the simulated data (i.e., $\boldsymbol{\pi} * G$). (c),(g),(k),(o) The estimated $\hat{\boldsymbol{Q}}$ for the learned motifs. (d),(h),(l),(p) The estimated number of genes belonging to each learned motif (i.e., $\hat{\boldsymbol{\pi}} * G$). In the $\boldsymbol{Q}$ pattern graph (columns 1 and 3), each row indicates a motif pattern and each column represents a study. The gray scale of the cell (k, d) demonstrates the probability of differential expression in study d for pattern k. Each row of the bar chart for $(\boldsymbol{\pi} * G)$ corresponds to the motif pattern in the same row of the $\boldsymbol{Q}$ graph. The motif patterns learned by *CorMotif* are similar to the true underlying motif patterns. It can be seen that complementary block motifs, such as [1,1,0,0] and [0,0,1,1], are not likely to be absorbed into merged motifs if their relative proportions are not low.

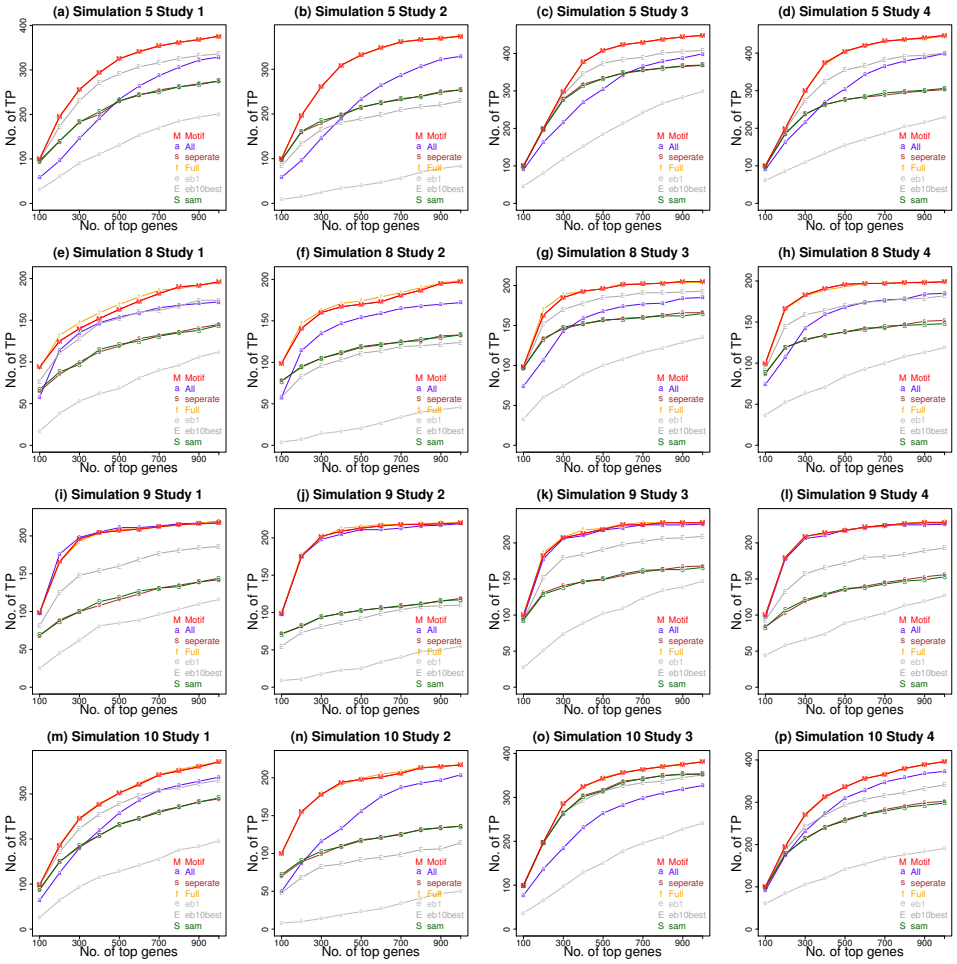


Figure 1.7: Gene ranking performance for simulations 5, 8, 9 and 10. $TP_d(r)$, the number of genes that are truly differentially expressed in study d among the top r ranked genes by a given method, is plotted against the rank cutoff r. (a)-(d) Simulation 5. (e)-(h) Simulation 8. (i)-(l) Simulation 9. (m)-(p) Simulation 10.

Table 1.4: Confusion matrix for simulation 2. The column labels indicate the true underlying patterns and the row labels represent the learned configurations.

Method	Motif pattern	$c(0,0,0,0)$	$c(0,0,1,1)$	$c(1,1,0,0)$	$c(1,1,1,1)$
Cormotif	$c(0,0,0,0)$	9069	122	99	54
	$c(0,0,1,1)$	7	127	0	30
	$c(1,1,0,0)$	3	0	153	29
	$c(1,1,1,1)$	0	1	1	89
	other	21	50	47	98
separate limma	$c(0,0,0,0)$	9024	112	89	58
	$c(0,0,1,1)$	1	44	0	13
	$c(1,1,0,0)$	0	0	57	17
	$c(1,1,1,1)$	0	0	0	8
	other	75	144	154	204
all concord	$c(0,0,0,0)$	9094	180	166	76
	$c(0,0,1,1)$	0	0	0	0
	$c(1,1,0,0)$	0	0	0	0
	$c(1,1,1,1)$	6	120	134	224
	other	0	0	0	0
full motif	$c(0,0,0,0)$	9069	122	99	54
	$c(0,0,1,1)$	7	130	0	33
	$c(1,1,0,0)$	5	0	160	29
	$c(1,1,1,1)$	0	1	1	99
	other	19	47	40	85
eb1	$c(0,0,0,0)$	4693	20	8	5
	$c(0,0,1,1)$	376	65	1	8
	$c(1,1,0,0)$	474	1	74	10
	$c(1,1,1,1)$	365	131	132	238
	other	3192	83	85	39
eb10best	$c(0,0,0,0)$	0	0	0	0
	$c(0,0,1,1)$	79	188	1	30
	$c(1,1,0,0)$	68	0	202	31
	$c(1,1,1,1)$	7793	105	87	223
	other	1160	7	10	16
SAM	$c(0,0,0,0)$	9095	209	236	193
	$c(0,0,1,1)$	0	7	0	6
	$c(1,1,0,0)$	0	0	0	0
	$c(1,1,1,1)$	0	0	0	0
	other	5	84	64	101

Table 1.5: Confusion matrix for simulation 3. The column labels indicate the true underlying patterns and the row labels represent the learned configurations.

Method	Motif pattern	$Motif1$	$Motif2$	$Motif3$	$Motif4$	$Motif5$
CorMotif	$Motif1$	9189	28	48	50	4
	$Motif2$	0	68	0	0	4
	$Motif3$	0	1	65	0	5
	$Motif4$	0	2	0	97	6
	$Motif5$	0	0	0	0	27
	other	11	101	87	53	154
separate limma	$Motif1$	9076	24	36	43	3
	$Motif2$	0	2	0	0	0
	$Motif3$	0	0	2	0	0
	$Motif4$	0	0	0	3	1
	$Motif5$	0	0	0	0	0
	other	124	174	162	154	196
all concord	$Motif1$	9200	96	117	94	5
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	104	83	106	195
	other	0	0	0	0	0
full motif	$Motif1$	9185	28	46	49	4
	$Motif2$	0	63	0	0	3
	$Motif3$	0	0	51	0	4
	$Motif4$	0	2	0	89	3
	$Motif5$	0	0	0	0	14
	other	15	107	103	62	172
eb1	$Motif1$	748	0	1	1	0
	$Motif2$	273	2	0	0	0
	$Motif3$	4	0	1	0	0
	$Motif4$	47	0	0	0	0
	$Motif5$	1239	157	149	170	183
	other	6889	41	49	29	17
SAM	$Motif1$	9200	139	170	165	134
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	0	0	0	0
	other	0	61	30	35	66

Table 1.6: Confusion matrix for simulation 4. The column labels indicate the true underlying patterns and the row labels represent the learned configurations.

Method	Motif pattern	$Motif1$	$Motif2$	$Motif3$	$Motif4$	$Motif5$
CorMotif	$Motif1$	9198	4	5	2	0
	$Motif2$	0	29	0	0	0
	$Motif3$	0	0	20	0	0
	$Motif4$	0	0	0	22	0
	$Motif5$	0	0	0	0	4
	other	2	167	175	176	196
separate limma	$Motif1$	8907	1	3	1	0
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	0	0	0	0
	other	293	199	197	199	200
all concord	$Motif1$	9200	58	69	69	0
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	142	131	131	200
	other	0	0	0	0	0
SAM	$Motif1$	9197	64	66	92	23
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	0	0	0	0
	other	3	136	134	108	177

Table 1.7: GEO data used for real data based simulations.

Simulation ID	Study ID	GEO Sample Id	GEO series number	Sample No.	Sample type
Simulations 5-10	1	GSM366065 - GSM366070	GSE14668	6	Liver tissue of liver donor
Simulations 5-10	2	GSM550623 - GSM550628	GSE22138	6	Uveal Melanoma primary tumor tissue
Simulations 5-10	3	GSM553482 - GSM553487	GSE22224	6	Peripheral blood mononuclear cells of healthy volunteer
Simulations 5-10	4	GSM494634 - GSM494639	GSE33356	6	Normal lung tissue
Simulations 6-7	5	GSM909644- GSM909649	GSE37069	6	Blood samples from controls
Simulations 6-7	6	GSM909650 - GSM909655	GSE37069	6	Blood samples from controls
Simulations 6-7	7	GSM909656- GSM909661	GSE37069	6	Blood samples from controls
Simulations 6-7	8	GSM909662 - GSM909667	GSE37069	6	Blood samples from controls
Simulations 6-7	9	GSM90968 - GSM909673	GSE37069	6	Blood samples from controls
Simulations 6-7	10	GSM909674- GSM909679	GSE37069	6	Blood samples from controls
Simulation 7	11	GSM376428 - GSM376433	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	12	GSM376434- GSM376439	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	13	GSM376440 - GSM376445	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	14	GSM376446 - GSM376451	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	15	GSM376452 - GSM376457	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	16	GSM376458 - GSM376463	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	17	GSM376464 - GSM376469	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	18	GSM376470 - GSM376475	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	19	GSM376476 - GSM376481	GSE15061	6	Non-leukemia bone marrow samples
Simulation 7	20	GSM376482 - GSM376487	GSE15061	6	Non-leukemia bone marrow samples

Table 1.8: Confusion matrix for simulation 5. The column labels indicate the true underlying patterns and the row labels represent the learned configurations.

Method	Motif pattern	$c(0,0,0,0)$	$c(0,0,1,1)$	$c(1,1,0,0)$	$c(1,1,1,1)$
CorMotif	$c(0,0,0,0)$	53670	108	164	20
	$c(0,0,1,1)$	6	286	0	18
	$c(1,1,0,0)$	29	0	200	6
	$c(1,1,1,1)$	0	0	0	31
	other	70	6	36	25
separate limma	$c(0,0,0,0)$	53615	121	171	24
	$c(0,0,1,1)$	0	79	0	8
	$c(1,1,0,0)$	0	0	46	3
	$c(1,1,1,1)$	0	0	0	1
	other	160	200	183	64
all concord	$c(0,0,0,0)$	53748	187	255	26
	$c(0,0,1,1)$	0	0	0	0
	$c(1,1,0,0)$	0	0	0	0
	$c(1,1,1,1)$	27	213	145	74
	other	0	0	0	0
full motif	$c(0,0,0,0)$	53671	108	165	20
	$c(0,0,1,1)$	5	286	0	18
	$c(1,1,0,0)$	30	0	201	6
	$c(1,1,1,1)$	0	0	1	36
	other	69	6	33	20
eb1	$c(0,0,0,0)$	49817	190	188	23
	$c(0,0,1,1)$	161	103	0	12
	$c(1,1,0,0)$	244	0	66	8
	$c(1,1,1,1)$	11	0	0	7
	other	3542	107	146	50
eb10best	$c(0,0,0,0)$	51731	109	125	36
	$c(0,0,1,1)$	5	232	0	6
	$c(1,1,0,0)$	12	0	169	4
	$c(1,1,1,1)$	0	0	0	16
	other	2027	59	106	38
SAM	$c(0,0,0,0)$	53773	283	398	83
	$c(0,0,1,1)$	0	0	0	0
	$c(1,1,0,0)$	0	0	0	0
	$c(1,1,1,1)$	0	0	0	0
	other	2	117	2	17

Table 1.9: Confusion matrix for simulation 6. The column labels indicate the true underlying patterns and the row labels represent the learned configurations.

Method	Motif pattern	$Motif1$	$Motif2$	$Motif3$	$Motif4$	$Motif5$
CorMotif	$Motif1$	53600	15	11	15	1
	$Motif2$	0	169	0	1	4
	$Motif3$	4	1	147	0	2
	$Motif4$	1	3	0	178	7
	$Motif5$	0	1	0	1	170
	other	270	11	42	5	16
separate limma	$Motif1$	53340	21	12	22	5
	$Motif2$	0	16	0	0	4
	$Motif3$	0	0	14	0	2
	$Motif4$	0	0	0	17	1
	$Motif5$	0	0	0	0	0
	other	535	163	174	161	188
all concord	$Motif1$	43	36	49	4	
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	17	157	164	151	196
	other	0	0	0	0	0
full motif	$Motif1$	53578	15	11	13	1
	$Motif2$	0	156	0	0	2
	$Motif3$	3	0	146	0	1
	$Motif4$	1	2	0	166	4
	$Motif5$	0	0	0	0	136
	other	293	27	43	21	56
eb1	$Motif1$	47986	24	14	18	0
	$Motif2$	3	47	0	0	5
	$Motif3$	23	1	42	0	1
	$Motif4$	10	0	0	69	1
	$Motif5$	3	0	0	0	38
	other	5850	128	144	113	155
SAM	$Motif1$	53851	120	138	116	89
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	0	0	0	0
	other	24	80	62	84	111

Table 1.10: Confusion matrix for simulation 7. The column labels indicate the true underlying patterns and the row labels represent the learned configurations.

Method	Motif pattern	$Motif1$	$Motif2$	$Motif3$	$Motif4$	$Motif5$
CorMotif	$Motif1$	52442	3	5	4	1
	$Motif2$	6	188	0	0	1
	$Motif3$	10	0	156	0	0
	$Motif4$	5	0	0	187	10
	$Motif5$	0	0	0	0	165
	other	1412	9	39	9	23
separate limma	$Motif1$	51999	7	24	5	4
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	0	0	0	0
	other	1876	193	176	195	196
all concord	$Motif1$	53859	27	49	18	3
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	16	173	151	182	197
	other	0	0	0	0	0
SAM	$Motif1$	53812	108	145	110	100
	$Motif2$	0	0	0	0	0
	$Motif3$	0	0	0	0	0
	$Motif4$	0	0	0	0	0
	$Motif5$	0	0	0	0	0
	other	63	92	55	90	100

Chapter 2

Integrative Analysis of Allele-specificity of Protein-DNA Interactions in Multiple ChIP-seq Datasets

[1]

2.1 Introduction

In a diploid organism, each somatic cell has two copies of the genome. At certain genomic loci, gene expression, DNA methylation, transcription factor (TF) binding or histone modification (HM) can be allele-specific. In other words, the two alleles can behave differently. These phenomena, also known as allele-specific expression (ASE), allele-specific DNA methylation (ASM) and allele-specific binding (ASB, including both allele-specific TF binding and allele-specific histone modifications), can contribute to phenotypic diversity and may

[1] A modified version of this chapter has been published: **Wei YY***, Li X*, Wang Q, Ji HK(2012) iASeq: Integrating Multiple ChIP-seq Datasets for Detecting Allele-specific Binding. *BMC Genomics*. 13:681. **Highly accessed**.($*$ joint first authors.) doi: 10.1186/1471-2164-13-681

play important roles in adaptive evolution (Bell and Beck (2009); Graze *and others* (2012); Knight (2004)). Many allele-specific (AS) events have been found to correlate with variants in genomic sequences (Chen *and others* (2012); McDaniell *and others* (2010); Kasowski *and others* (2010); Kerkel *and others* (2008); Morley *and others* (2004); Schilling *and others* (2009); Tycko (2010); Zhang *and others* (2009)). Comprehensively characterizing allele-specificity therefore can help with linking genotypes to phenotypes. Abnormal AS events have also been linked to various diseases (Cui *and others* (2003); Holt *and others* (2011); Heap *and others* (2010); Tuch *and others* (2010)). For instance, loss of imprinting in IGF2 has been associated with increased risk of colorectal cancer (Cui *and others* (2003)). This again highlights the importance of studying allele-specificity.

Early methods for analyzing AS events rely on low-throughput technologies such as real time quantitative PCR (Bell and Beck (2009)). Later, application of SNP arrays has made the AS analysis high-throughput (Ben-David *and others* (2011); Lo *and others* (2003); Palacios *and others* (2009); Serre *and others* (2008)). More recently, the rapidly evolving high-throughput sequencing technologies opened the door to produce digital read-out of AS events genome-wide without being constrained by any specific array design (Chen *and others* (2011); McDaniell *and others* (2010); Montgomery *and others* (2010); Pickrell *and others* (2010); Heap *and others* (2010); Ju *and others* (2011); Tang *and others* (2011); Tuch *and others* (2010)). This brings many new opportunities as well as analytical challenges.

ChIP-seq, a technology that couples chromatin immunoprecipitation with

high-throughput sequencing, has become the state-of-the-art approach for mapping genome-wide TF binding sites and HMs (Barski *and others* (2007); Johnson *and others* (2007); Mikkelsen *and others* (2007); Robertson *and others* (2007)). However, so far the value of this technology for studying ASB has not been fully utilized. Detecting ASB from a single ChIP-seq dataset often suffers from low statistical power. This is because only a small fraction of reads in each ChIP-seq sample are mapped to heterozygote SNPs, and only these reads are informative for inferring allele-specificity. To make the ChIP-seq based ASB analysis more useful, it is important to have either experimental or analytical innovations to increase the power for detecting allele-specificity.

ChIP-seq data in public domains grow rapidly. A recently developed database hmChIP, for instance, has compiled over 450 human and mouse ChIP-seq datasets representing approximately 2000 samples from 140+ different TFs and HMs (Chen *and others* (2011)). The large volume of data provides a new opportunity to improve detection of ASB. Conceptually, an integrative analysis of ChIP-seq data for different TFs and HMs from the same individual and cell type may allow one to discover the synergistic correlation patterns of allele-specificity among different proteins. These correlation patterns can then be utilized to integrate information from multiple datasets to improve the ASB detection. For example, if the allelic imbalance of TF A and HM B always co-occur, then analyzing their ChIP-seq data jointly will increase the effective number of reads available for allelic inference which will then increase the statistical power. Unfortunately, existing data analysis tools cannot deal with this emerging opportunity. Methods available for analyzing ASE or ASB using the next-generation sequencing data are all designed for analyzing one dataset

at a time. While a few methods are developed for solving problems such as read mapping biases (Degner *and others* (2009)), construction of individualized genome sequences (Rozowsky *and others* (2011)), and combining multiple SNPs in the same gene to infer ASE (Skelly *and others* (2011)), no methods and software tools are available for jointly analyzing multiple ChIP-seq datasets together to discover synergy patterns of allele-specificity among multiple proteins and then use the correlation patterns to increase the power of ASB detection by borrowing information across datasets.

In this chapter, we present an integrated solution to this problem by developing a new approach, iASeq, for jointly analyzing allele-specificity in multiple ChIP-seq datasets. iASeq uses a Bayesian hierarchical mixture model to describe unknown correlation patterns of allele-specificity among multiple datasets. These patterns can be discovered automatically from the data by fitting the model using an Expectation-Maximization (EM) algorithm. Using the identified correlation patterns, the model allows one to integrate information from multiple datasets to improve the ASB detection. Applying this approach, we analyzed 40 ENCODE (ENCODE Consortium (2004)) ChIP-seq datasets in GM12878 cells, representing a total of 77 samples from 34 TFs and HMs. The analysis demonstrates the ability of iASeq to automatically integrate information from multiple datasets to significantly improve the detection of allelic imbalance. iASeq is implemented as an R package which is freely available from Bioconductor.

2.2 Methods

2.2.1 Data Structure

Suppose there are D ChIP-seq datasets generated using cells from the same individual and the same cell type. Each dataset d corresponds to one TF or HM, and has J_d replicate samples (Figure 2.1a). Different datasets represent different TFs or HMs, or data generated by different labs. For the individual in question, assume one is interested in analyzing I heterozygote SNPs with known genotypes. We want to know whether the two alleles of each SNP behave differently in each dataset, and how the AS events are correlated among datasets. For each SNP, the allele consistent with the reference genome is called the *reference allele*, and the other allele is called the *non-reference allele*.

After read mapping and data preprocessing (see Supplemental Methods Section 2.6), we count reads for each allele at each heterozygote SNP. For SNP i, dataset d and replicate sample j, let x_{idj} and y_{idj} be the read counts for the reference allele and non-reference allele respectively. Let $n_{idj} = x_{idj} + y_{idj}$ be the total read count (See Figure 2.1a for a toy example). Protein-DNA binding can be skewed to the reference allele (SR), skewed to the non-reference allele (SN), or not allele-specific (NS). We use a binary variable b_{id} to indicate whether SNP i is SR ($b_{id} = 1$) or not ($b_{id} = 0$) in dataset d. If $b_{id} = 1$, then SNP i is assumed to be SR in all replicate samples in dataset d. Similarly, we introduce another binary indicator c_{id} to indicate whether SNP i is SN or not in dataset d. b_{id} and c_{id} cannot be equal to one at the same time. If $b_{id} = 0$ and $c_{id} = 0$, then SNP i is NS in dataset d. The configuration at each SNP i can be described by two vectors $\boldsymbol{B}_i = (b_{i1}, \cdots, b_{iD})$ and $\boldsymbol{C}_i = (c_{i1}, \cdots, c_{iD})$ (See Figure 2.1d for a cartoon

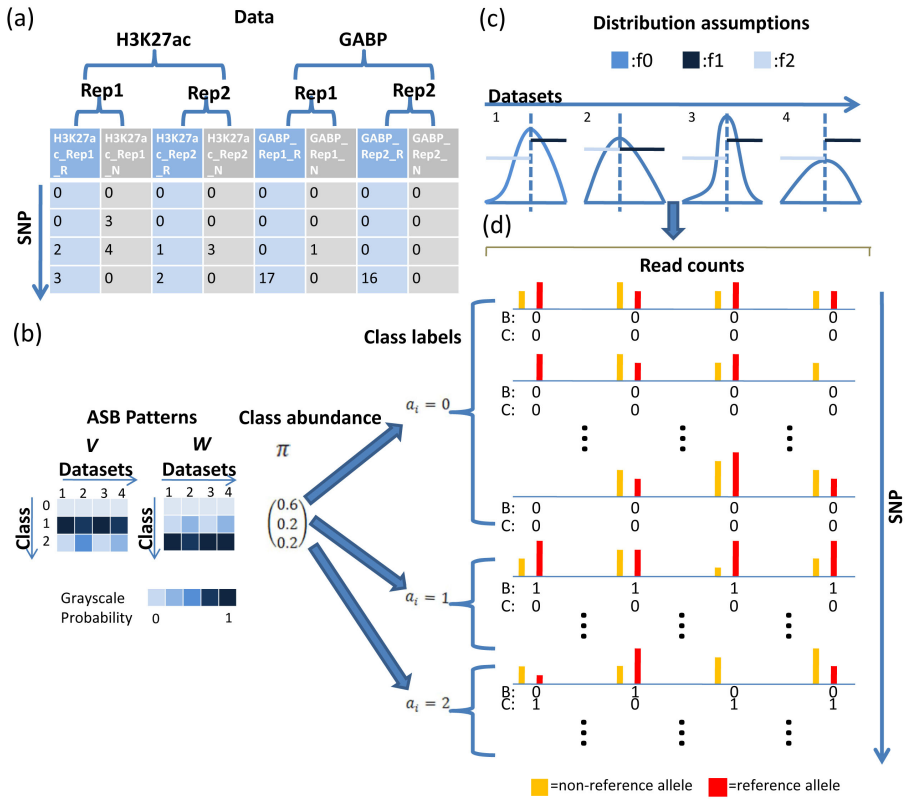


Figure 2.1: The iASeq model (a) An example of the data structure. Each row represents a SNP and each column corresponds to either the reference allele (R) or the non-reference allele (N) read counts from a ChIP-seq sample. iASeq assumes the following data generating process. (b) First, SNPs belong to $K+1$ classes with different ASB patterns. For each SNP, a class label a_i is randomly assigned according to a class abundance $\boldsymbol{\pi}$. Given the class label, a configuration $[b_{id}, c_{id}]$ is generated for each SNP in each dataset according to $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$. (c) Next, a skewing probability p_{idj} is generated for each SNP i, dataset d and replicate sample j based on $[b_{id}, c_{id}]$. The distribution of p_{idj} for NS SNPs in each sample follows a Beta distribution (blue lines). p_{idj}s for SR SNPs are uniformly distributed in the interval $[p_{dj0}, 1]$ where p_{dj0} is the mean of the background Beta distribution (dark blue lines). p_{idj}s for SN SNPs are uniformly distributed in the interval $[0, p_{dj0}]$ (light blue lines). (d) Finally, given the configuration $[b_{id}, c_{id}]$, skewing probability p_{idj} and a total read count n_{idj} for SNP i, dataset d and sample j, the read count for each allele is generated according to a binomial distribution. The length of the bar represents the read count,orange for the non-reference allele and red for the reference allele.

illustration). Based on these notations, (x_{idj}, y_{idj}), or equivalently (x_{idj}, n_{idj}), are the observed data for SNP i in sample (d, j), whereas the indicators b_{id} and c_{id} are unobserved.

2.2.2 Main Intuition and Challenge

Our primary goal is to infer for each SNP whether there is allelic imbalance in each dataset. This is equivalent to inferring b_{id} and c_{id}. A simple solution to this problem is to analyze each individual dataset separately, but this approach has low statistical power since the counts (x_{idj}, n_{idj}) usually are small.

If one knows how different datasets are correlated in terms of allelic imbalance, this knowledge may be used to improve the data analysis. For instance, if the allelic imbalance of two proteins A and B are closely correlated, then observing skewed read counts for protein A will provide information for inferring the allelic imbalance of protein B. Integrating the data from both A and B will increase the effective number of reads available for statistical inference, which will then lead to increased statistical power.

In reality, how different proteins are correlated is usually unknown. However, one may learn it by studying the data from many SNPs. Each SNP has three possible states in each dataset: SR, SN and NS. For D datasets, there are 3^D possible configurations in total. From studying many SNPs, one can know the relative frequencies (or mixing proportions) of these 3^D configurations. The mixing proportions will tell how different datasets are correlated. For instance, let $[s_1, s_2, \cdots, s_D]$ be the skewness configuration of a SNP in the D datasets. If the mixing proportions for three configurations $[NS, NS, \cdots, NS]$, $[SR, SR, \cdots, SR]$ and $[SN, SN, \cdots, SN]$ are 0.9, 0.05 and 0.05, then no other

configurations exist in the data and all datasets are perfectly correlated in terms of the allelic imbalance. In other words, at a particular SNP, if one dataset is SR, then all the other datasets are also SR. If one is SN, then all the others are also SN. On the other hand, if other configurations have non-zero mixing proportions, then not all datasets are perfectly correlated, and at a particular SNP, one allows the possibility that only a subset of datasets are correlated. For instance, if the mixing proportion for a configuration $[SR, SR, NS, \cdots, NS]$ is 0.03, then there will be 3% of SNPs that are skewed to the reference allele in the first two datasets but not skewed in the other datasets. Therefore, knowing the mixing proportions of all 3^D configurations will tell one the correlation structure in the data. This knowledge can then be used to improve statistical inference at each individual SNP by facilitating information sharing across datasets. For example, if the configuration $[SR, SR, SN]$ has a much higher mixing proportion than $[SR, SR, NS]$, then observing strong skewness towards the reference allele of a SNP in the first two datasets will imply that, a priori, the SNP is highly likely to be skewed to the non-reference allele in the third dataset and has much lower probability to be non-skewed for both alleles. The principle here is the same as the principle represented by the Bayesian hierarchical models in the statistical literature.

A limitation of this approach is that one has to enumerate all 3^D AS configurations in order to describe the correlation. As the number of datasets increases, the number of possible configurations increases exponentially. Thus this approach does not scale well with the increasing D. Later, in our analysis of GM12878 data, $D = 40$ and $3^D > 10^{19}$. This simple approach is clearly intractable.

To circumvent the difficulty of documenting the frequencies of all 3^D configurations, iASeq employs a technique that can describe the major correlation patterns in the data using a few probability vectors whose values vary from 0 to 1 rather than being dichotomous (i.e., 0 or 1). This approach significantly reduces the model complexity but keeps the flexibility to account for all 3^D configurations. It is easily scalable to increasing dataset number. The correlation structure in the model can then be used to improve the statistical inference of allelic imbalance at each SNP in each individual dataset.

2.2.3 Probability Model

iASeq is based on the Bayesian hierarchical mixture model below that uses several probability vectors to describe the major correlation patterns among multiple datasets (Figure 2.1). The model assumes that SNPs can be grouped into $K+1$ classes with different allele-specificity patterns ($K \ll 3^D$), and the observed data are viewed as generated as follows:

- First, a class label a_i is randomly assigned to each SNP i according to a probability vector $\boldsymbol{\pi} = (\pi_0, \pi_1, \cdots, \pi_K)$. Here, $\pi_k = Pr(a_i = k)$ is the prior probability to assign a SNP to class k. $\sum_k \pi_k = 1$.

- If the class label $a_i = 0$, then $\boldsymbol{B}_i = (0, \cdots, 0)$ and $\boldsymbol{C}_i = (0, \cdots, 0)$. In other words, all SNPs in class 0 are background SNPs, and they are NS in all datasets. If $a_i = k$ and $k \neq 0$, then SNP i can be skewed, and its $[b_{id}; c_{id}]$s in different datasets are generated independently according to the following probabilities: $Pr(b_{id} = 1, c_{id} = 0 | a_i = k) = v_{kd}$ and $Pr(b_{id} = 0, c_{id} = 1 | a_i = k) = w_{kd}$. We assume $v_{kd} + w_{kd} < 1$, i.e., $Pr(b_{id} =$

$0, c_{id} = 0 | a_i = k) = 1 - v_{kd} - w_{kd} > 0$. The model implies that each class is associated with two vectors of probabilities $\boldsymbol{V}_k = (v_{k1}, \cdots, v_{kD})$ and $\boldsymbol{W}_k = (w_{k1}, \cdots, w_{kD})$. For SNPs in class k, $\boldsymbol{B}_i$ and $\boldsymbol{C}_i$ are generated according to the probabilities in $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$.

- Next, the observed read counts are generated based on the AS configurations specified by $\boldsymbol{B}_i$s and $\boldsymbol{C}_i$s. Consider SNP i and dataset d. If $b_{id} = 1$, then (x_{idj}, n_{idj}) in each replicate sample (d, j) is generated according to a probability distribution $Pr(x_{idj}, n_{idj} | b_{id} = 1, c_{id} = 0) = Pr(n_{idj} | b_{id} = 1, c_{id} = 0) Pr(x_{idj} | n_{idj}, b_{id} = 1, c_{id} = 0) \equiv Pr(n_{idj}) f_{idj1}(x_{idj})$. Here we assume that the marginal distribution of n_{idj} does not depend on b_{id} and c_{id}, and we use $f_{idj1}(x_{idj})$ to denote the conditional distribution $Pr(x_{idj} | n_{idj}, b_{id} = 1, c_{id} = 0)$. Data in different replicate samples are assumed to be generated independently. Similarly, if $c_{id} = 1$, then (x_{idj}, n_{idj})s are generated according to $Pr(x_{idj}, n_{idj} | b_{id} = 0, c_{id} = 1) = Pr(n_{idj}) f_{idj2}(x_{idj})$. If $b_{id} = 0$ and $c_{id} = 0$, then (x_{idj}, n_{idj})s are generated according to $Pr(x_{idj}, n_{idj} | b_{id} = 0, c_{id} = 0) = Pr(n_{idj}) f_{idj0}(x_{idj})$.

For SNP i and dataset d, we organize data from all replicates $j = 1, \cdots, J_d$ into $\boldsymbol{X}_{id} = (x_{id1}, \cdots, x_{idJ_d})$ and $\boldsymbol{N}_{id} = (n_{id1}, \cdots, n_{idJ_d})$. For SNP i, $\boldsymbol{X}_i = (\boldsymbol{X}_{i1}, \cdots, \boldsymbol{X}_{iD})$ and $\boldsymbol{N}_i = (\boldsymbol{N}_{i1}, \cdots, \boldsymbol{N}_{iD})$ contain data from all datasets. The final observed data are $\boldsymbol{X} = (\boldsymbol{X}_1, \cdots, \boldsymbol{X}_I)$ and $\boldsymbol{N} = (\boldsymbol{N}_1, \cdots, \boldsymbol{N}_I)$ which are the ensemble of data from all SNPs.

Let $\boldsymbol{A} = (a_1, \cdots, a_I)$ be the collection of class membership indictors of all SNPs, and let $\boldsymbol{B} = (\boldsymbol{B}_1, \cdots, \boldsymbol{B}_I)$ and $\boldsymbol{C} = (\boldsymbol{C}_1, \cdots, \boldsymbol{C}_I)$ be the SR and SN

indictors for all SNPs. $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are the unobserved missing data one wants to infer.

Organize the probability vectors $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$ from different classes into two matrices $\boldsymbol{V}_{K\times D} = (\boldsymbol{V}_1^T, \cdots, \boldsymbol{V}_K^T)^T$ and $\boldsymbol{W}_{K\times D} = (\boldsymbol{W}_1^T, \cdots, \boldsymbol{W}_K^T)^T$. $\boldsymbol{V}$, $\boldsymbol{W}$, and the probability vector $\boldsymbol{\pi}$ that describes the class abundance are the unknown model parameters. K is assumed to be fixed. The choice of K and specification of data generating distributions $Pr(n_{idj})$, $f_{idj0}(x_{idj})$, $f_{idj1}(x_{idj})$ and $f_{idj2}(x_{idj})$ will be discussed later.

Based on this model, each SNP class k ($k \neq 0$) is associated with two vectors of probabilities $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$ which characterize the allelic imbalance preferences in different datasets for SNPs belonging to class k. For example, if a class has $[\boldsymbol{V}_k; \boldsymbol{W}_k] = [(0.8, 0.7, 0.1, 0.1); (0.1, 0.1, 0.8, 0.1)]$, then SNPs in this class have high probability to be SR in datasets 1 and 2, and high probability to be SN in dataset 3, but they have low probability to be allele-specific in dataset 4. Since $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$ are probabilities rather than 0-1 vectors, each class k can generate all 3^D AS configurations. Therefore, SNPs in the same class are not required to have the same AS configuration (e.g., a class can have one SNP with configuration $[SR, SR, NS, NS]$ while at the same time another SNP with configuration $[SR, NS, SR, NS]$), although they usually have similar AS configurations because SNPs in the same class are all generated using the same probability vectors. Meanwhile, there are K different classes, and each class has a different $[\boldsymbol{V}_k; \boldsymbol{W}_k]$ which specifies a different preference to generate the skewing configurations. Thus, whereas SNPs in the same class tend to have similar $[\boldsymbol{B}_i; \boldsymbol{C}_i]$ configurations, SNPs from different classes tend to have very different configurations. Conceptually, this is similar to a model-based

clustering analysis in which SNPs are grouped into $K+1$ clusters based on their $[\boldsymbol{B}_i; \boldsymbol{C}_i]$ configurations. However, an important difference here is that $[\boldsymbol{B}_i; \boldsymbol{C}_i]$s are unknown.

Our model assumes that $[b_{id}; c_{id}]$s of the same SNP in different datasets are a priori independent conditional on the class membership a_i. However, $[b_{id}; c_{id}]$s from different datasets are not independent marginally if one integrates out the class label a_i. For example, the marginal probability $Pr([b_{id}; c_{id}] = [1; 0]) = \sum_k Pr([b_{id}; c_{id}] = [1; 0] | a_i = k) Pr(a_i = k) = \sum_{k=1}^{K} \pi_k v_{kd}$. On the other hand, the joint probability $Pr([\boldsymbol{B}_i; \boldsymbol{C}_i] = [(1, 1, \cdots, 1); (0, 0, \cdots, 0)]) = \sum_{k=1}^{K} \pi_k (\prod_d v_{kd})$, which is clearly different from the product of the marginals $\prod_d Pr([b_{id}; c_{id}] = [1; 0]) = \prod_d (\sum_{k=1}^{K} \pi_k v_{kd})$. This explains why our model can be used to describe the correlation among multiple datasets despite the conditional independence assumption. Intuitively, if one views the model as a clustering analysis of SNPs based on $[\boldsymbol{B}_i; \boldsymbol{C}_i]$, then each cluster will represent a co-occurrence pattern of allele-specificity across multiple proteins. The marginal correlation among multiple datasets is described by multiple clusters, whereas within each cluster the data in different datasets are generated independently. In real data, a small K (i.e., a small number of SNP classes) usually is sufficient to describe the major correlation structure among datasets. Using $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ to describe the correlation among datasets only requires $O(KD)$ parameters, which is significantly less complex than $O(3^D)$ parameters. At the same time, the iASeq model still provides the flexibility to accommodate all 3^D possible $[\boldsymbol{B}_i; \boldsymbol{C}_i]$ configurations as all of them have non-zero probability to occur.

2.2.4 Data Generating Distributions

To fully specify the model, one also needs to specify the data generating distributions $Pr(x_{idj}, n_{idj}|b_{id}, c_{id}) = Pr(n_{idj})Pr(x_{idj}|n_{idj}, b_{id}, c_{id})$. The primary goal of iASeq is to infer whether two alleles are different. We assume that information on allele-specificity is only contained in $Pr(x_{idj}|n_{idj}, b_{id}, c_{id})$, and therefore the exact form of $Pr(n_{idj})$, i.e., the marginal probability distribution of the total read count, is irrelevant for our purpose. As such, we mainly focus on modeling the conditional distribution of x_{idj} given n_{idj}, b_{id} and c_{id}, i.e., the three distributions $f_{idj0}(x)$, $f_{idj1}(x)$ and $f_{idj2}(x)$.

iASeq models these distributions hierarchically in two steps. First, x_{idj} is assumed to follow a binomial distribution $x_{idj}|n_{idj}, p_{idj} \sim Bin(n_{idj}, p_{idj})$, where p_{idj} is the probability that a read generated at SNP i in sample (d, j) represents the reference allele. Next, we model p_{idj} depending on the values of b_{id} and c_{id}.

If $b_{id} = 0$ and $c_{id} = 0$, SNP i is NS in dataset d. In this case, we assume that p_{idj} follows a Beta distribution $Beta(\alpha_{dj}, \beta_{dj})$ with mean $p_{dj0} = \alpha_{dj}/(\alpha_{dj} + \beta_{dj})$. Note that a simpler model for p_{idj} would be to set it to a constant p_{dj0} which reflects the background ratio of read counts between two alleles. However, previous studies have shown that many background SNPs can have p_{idj} slightly different from the average background p_{dj0} even though they do not have biologically meaningful allele-specificity Skelly *and others* (2011). As a result, a constant p_{dj0} is not sufficient to describe the background variation. For this reason, we adopt the Beta distribution to describe p_{idj} instead of setting it to a constant (See the blue lines illustrated for $f(p_{idj}|b_{id} = 0, c_{id} = 0)$ in Figure 2.1c). In the ideal world, the mean of the Beta distribution, p_{dj0}, would be

equal to 0.5. However, in reality p_{dj0} may be slightly different from 0.5 due to various sources of read mapping biases. For example, allowing the same number of mismatches, reads from the reference allele are easier to be mapped back to the reference genome than reads from the non-reference allele. Therefore, in iASeq p_{dj0} may take values different from 0.5. Indeed, it is determined by the parameters α_{dj} and β_{dj} in the Beta distribution which are estimated from the data using a moment matching approach (see Supplemental Method in Section 2.6). Once estimated, α_{dj}, β_{dj} and p_{dj0} are treated as fixed and known parameters. Based on the model for p_{idj}, we integrate out all possible values of p_{idj} to obtain the distribution of x_{idj} conditional on $b_{id} = 0$ and $c_{id} = 0$, which is a beta-binomial distribution:

$$\begin{aligned} f_{idj0}(x_{idj}) &= Pr(x_{idj}|n_{idj}, b_{id} = 0, c_{id} = 0) \\ &= \int_0^1 Pr(x_{idj}|n_{idj}, p_{idj}, b_{id} = 0, c_{id} = 0) f(p_{idj}|b_{id} = 0, c_{id} = 0) dp_{idj} \\ &= \frac{C_{n_{idj}}^{x_{idj}}}{B(\alpha_{dj}, \beta_{dj})} \int_0^1 p^{x_{idj}+\alpha_{dj}-1}(1-p)^{n_{idj}-x_{idj}+\beta_{dj}-1} dp \\ &= \frac{C_{n_{idj}}^{x_{idj}} B(x_{idj} + \alpha_{dj}, n_{idj} - x_{idj} + \beta_{dj})}{B(\alpha_{dj}, \beta_{dj})} \end{aligned} \tag{2.1}$$

Here C_n^k is the binomial coefficients "n choose k", and $B(.,.)$ is the beta function.

If $b_{id} = 1$ and $c_{id} = 0$, SNP i is SR in dataset d. In this case, we assume that p_{idj} follows a uniform distribution $U[p_{dj0}, 1]$(See the dark blue lines illustrated for $f(p_{idj}|b_{id} = 1, c_{id} = 0)$ in Figure 2.1c). Here $p_{dj0} = \alpha_{dj}/(\alpha_{dj} + \beta_{dj})$ is defined as above. After integrating out p_{idj}, the distribution of x_{idj} conditional on $b_{id} = 1$ and $c_{id} = 0$ is

$$f_{idj1}(x_{idj}) = Pr(x_{idj}|n_{idj}, b_{id} = 1, c_{id} = 0)$$

$$= \int_0^1 Pr(x_{idj}|n_{idj}, p_{idj}, b_{id}=1, c_{id}=0) f(p_{idj}|b_{id}=1, c_{id}=0) dp_{idj}$$

$$= \frac{C_{n_{idj}}^{x_{idj}}}{1-p_{dj0}} \int_{p_{dj0}}^1 p^{x_{idj}} (1-p)^{n_{idj}-x_{idj}} dp \quad (2.2)$$

If $b_{id} = 0$ and $c_{id} = 1$, SNP i is SN in dataset d, and we assume that p_{idj} follows a uniform distribution $U[0, p_{dj0}]$ (See the light blue lines illustrated for $f(p_{idj}|b_{id}=0, c_{id}=1)$ in Figure 2.1c). After integrating out p_{idj}, the distribution of x_{idj} conditional on $b_{id} = 0$ and $c_{id} = 1$ is

$$f_{idj2}(x_{idj}) = Pr(x_{idj}|n_{idj}, b_{id}=0, c_{id}=1)$$

$$= \int_0^1 f(x_{idj}|n_{idj}, p_{idj}, b_{id}=0, c_{id}=1) f(p_{idj}|b_{id}=0, c_{id}=1) dp_{idj}$$

$$= \frac{C_{n_{idj}}^{x_{idj}}}{p_{dj0}} \int_0^{p_{dj0}} p^{x_{idj}} (1-p)^{n_{idj}-x_{idj}} dp \quad (2.3)$$

2.2.5 Joint Probabilities and Model Fitting

Based on the model above, the complete data likelihood can be derived as:

$$Pr(\boldsymbol{X}, \boldsymbol{N}, \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}|\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) = Pr(\boldsymbol{N}) Pr(\boldsymbol{X}, \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}|\boldsymbol{N}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) \quad (2.4)$$

$$= Pr(\boldsymbol{N}) \prod_{i=1}^{I} Pr(\boldsymbol{X}_i, a_i, \boldsymbol{B}_i, \boldsymbol{C}_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$$

Define $L_{id0} = \prod_{j=1}^{J_d} f_{idj0}(x_{idj})$, $L_{id1} = \prod_{j=1}^{J_d} f_{idj1}(x_{idj})$ and $L_{id2} = \prod_{j=1}^{J_d} f_{idj2}(x_{idj})$. Define $\delta(.)$ to be an indicator function. $\delta(.) = 1$ if its argument is true, and $\delta(.) = 0$ otherwise. We have

$$Pr(\boldsymbol{X}_i, a_i, \boldsymbol{B}_i, \boldsymbol{C}_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$$

$$= Pr(a_i|\boldsymbol{\pi}) \prod_{d=1}^{D} Pr(b_{id}, c_{id}|a_i, \boldsymbol{V}, \boldsymbol{W}) Pr(\boldsymbol{X}_{id}|\boldsymbol{N}_{id}, a_i, b_{id}, c_{id})$$

$$= \{\pi_0 \prod_{d=1}^{D} L_{id0}\}^{\delta(a_i=0)} \prod_{k=1}^{K} \{\pi_k \prod_{d=1}^{D} [v_{kd} L_{id1}]^{b_{id}} [w_{kd} L_{id2}]^{c_{id}}$$

$$[(1 - v_{kd} - w_{kd}) L_{id0}]^{1-b_{id}-c_{id}}\}^{\delta(a_i=k)} \quad (2.5)$$

To infer $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$, we employ a Bayesian approach by imposing a Dirichlet prior $D(\eta, \cdots, \eta)$ on $\boldsymbol{\pi}$ and imposing independent Dirichlet priors $D(\eta, \eta, \eta)$ on all triplets $(v_{kd}, w_{kd}, 1 - v_{kd} - w_{kd})$. The joint posterior distribution of unknown parameters and indicators given the observed data is:

$$\begin{aligned} Pr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \boldsymbol{X}, \boldsymbol{N}) &\propto Pr(\boldsymbol{X}, \boldsymbol{N}, \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C} | \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) f(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) \\ &\propto \prod_{i=1}^{I} Pr(\boldsymbol{X}_i, a_i, \boldsymbol{B}_i, \boldsymbol{C}_i | \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) \quad (2.6) \\ &\quad \{\prod_{k=0}^{K} \pi_k^{\eta-1}\} \{\prod_{k=1}^{K} \prod_{d=1}^{D} v_{kd}^{\eta-1} w_{kd}^{\eta-1} (1 - v_{kd} - w_{kd})^{\eta-1}\} \end{aligned}$$

Conditional on the observed data, $Pr(\boldsymbol{N})$ is a constant that does not contain parameters of interest, therefore it is absorbed into a proportionality constant not shown in the formula above. Using this joint posterior, an EM algorithm can be derived to search for posterior mode $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{V}}, \hat{\boldsymbol{W}})$ of $Pr(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \boldsymbol{X}, \boldsymbol{N}) = \sum_{\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}} Pr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \boldsymbol{X}, \boldsymbol{N})$ in which the missing indictors $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are all integrated out (see Supplemental Method 2.6.4).

For the Dirichlet prior, we use $\eta = 2$ (See Supplemental Method 2.6.3 for a discussion on the choice of parameter for the Dirichlet prior). In the EM algorithm, we assume that the class number K is given. In order to choose the optimal K, we run the algorithm multiple times using different values of K. We choose the best K using the Bayesian Information Criterion (BIC) (see Supplemental Method 2.6.5).

2.2.6 Statistical Inference of Allele-specificity

The estimated $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ can describe the correlation patterns of allele-specificity among datasets. Given $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$, one can infer whether SNP i belongs to class k based on the posterior probability $Pr(a_i = k|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$. One can then infer whether each SNP i is skewed in each individual dataset d based on the posterior probability

$Pr(b_{id}, c_{id}|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) = \sum_{a_i} Pr(a_i, b_{id}, c_{id}|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$

after summing over all possible values of a_i. Note that

$$Pr(b_{id}, c_{id}|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) \tag{2.7}$$

$$= \sum_k Pr(a_i = k|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) Pr(b_{id}, c_{id}|a_i = k, \boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$$

Define

$$\tilde{P}_{id} = max\left\{Pr(b_{id} = 1, c_{id} = 0|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}), Pr(b_{id} = 0, c_{id} = 1|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})\right\} \tag{2.8}$$

Using $\tilde{P}_{id}$, SNPs can be rank ordered for biologists to choose candidates to design follow-up studies. For each top ranked SNP, one can determine its skewing direction by comparing $Pr(b_{id} = 1, c_{id} = 0|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$ and $Pr(b_{id} = 0, c_{id} = 1|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$. The one with the larger value determines the direction. Finally, the posterior probabilities of top N SNPs can be converted to an estimate of false discovery rate (FDR) using $FDR(N) = \sum_{i \in top\ N\ SNPs}(1 - \tilde{P}_{id})/N$.

Formula 2.8 shows that two types of information contribute to $Pr(b_{id}, c_{id}|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$: (1) $Pr(a_i = k|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$, which is determined using information from all D datasets, and (2) $Pr(b_{id}, c_{id}|a_i =$

$k, \boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$, which only uses information specific to dataset d conditional on $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$. Thus for each particular dataset d, the dataset-specific information is weighted by information obtained from other datasets to determine the SNP ranking. Intuitively, if allelic imbalance in two datasets are correlated, then observing an AS event in one dataset will suggest that a relatively weak skewing event observed at the same SNP in the other dataset is very likely to be a true AS event. In contrast, if no AS event is observed in one dataset, then a relatively weak skewing event observed at the same SNP in the other dataset is likely to be a false positive. This is the underlying nature of using $Pr(a_i = k|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$ to re-weigh information in $Pr(b_{id}, c_{id}|a_i = k, \boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$, and it provides the foundation for improving SNP ranking by borrowing information across datasets. In real applications, $\boldsymbol{\pi}$, $\boldsymbol{V}$,$\boldsymbol{W}$ are unknown, and they are replaced by the posterior mode obtained from the EM algorithm.

2.3 Results

2.3.1 GM12878 Data and Preprocessing

We collected 40 ENCODE Consortium (2012) ChIP-seq datasets with a total of 77 samples together with a genomic DNA sample in GM12878 lymphoblastoid cells. GM12878 is a female and is one of the most extensively studied cell lines in ENCODE. Within each dataset, the number of replicate samples varied from 1 to 3. We downloaded the raw sequence reads of all 78 samples and mapped them to human genome (hg18) (see details in Section 2.6). We removed repeated sequences from the ChIP-seq datasets to avoid PCR duplicates, which may skew

the determination of allelic biases. In other words, if multiple reads have exactly the same sequence, only one copy is retained. We obtained the genotype data for GM12878 from

ftp://ftp-trace.ncbi.nih.gov/1000genomes/ftp/pilot_data/release/2010_07/trio/snps.

As previously described in Degner *and others* (2009), there are two different types of read mapping biases that may affect the analysis of AS events: the reference bias and the inherent bias. The reference bias often occurs when one maps sequence reads to a reference genome. If one allows the same number of mismatches in the alignment, a read from the non-reference allele is less likely to be mapped back to the reference genome compared to a read from the reference allele, since the non-reference read has one more mismatch to the reference genome. This phenomenon is known as the reference bias. This type of bias, if it exists, is automatically taken care of by the iASeq model through the parameter p_{dj0} which models the background skewing probability and is estimated using all reads mapped to heterozygote SNPs in each sample. If there is reference mapping bias, p_{dj0} will take a value different from 0.5 to adjust for the bias. One may remove reference bias before the analysis by masking SNPs in the reference genome during the alignment or by aligning reads to a diploid personal genome. This situation will also be automatically recognized by iASeq through the estimation of p_{dj0} from the data (if there is no bias, $p_{dj0} = 0.5$). Therefore, regardless of whether the reference bias has been removed from the data in the preprocessing or not, the iASeq model is able to automatically handle it and adjust the inference accordingly.

The intrinsic bias is a different type of bias. As shown by Degner *and others* (2009), even if the reference bias is removed (e.g., by masking SNPs

in the reference genome), the inherent bias still exists. For example, suppose sequence 1 (e.g., xxxAxxx) and sequence 2(e.g., xxxTxxx) are two reads that differ only in one position (i.e., A/T). It is possible that sequence 1 is easier to be mapped back to its correct location in the genome than sequence 2 if the second sequence has many repeats in the genome. This bias reflects the inherent characteristics of the genome and cannot be removed by masking variants in the reference genome or by mapping reads to a diploid personal genome. In the above example, masking A and T in the original reads is also not a solution, since a priori one does not know which position in a read corresponds to a SNP position and therefore should be masked without first aligning the read to the genome. When a heterozygote SNP has inherent bias, one allele will have higher read counts than the other even if the two alleles have the same binding level. To avoid this bias, we used the approach described in Pickrell *and others* (2010); Degner *and others* (2009) to remove SNPs with the inherent bias.

We began with 1,704,166 heterozygote SNPs and filtered out 149,996 (8.8%) SNPs with inherent bias. Next, we eliminated SNPs that were not bound by any TF or associated with any HM in any dataset (see Supplemental Methods Section 2.6.1 for details). After applying these filters, 94,519 heterozygote SNPs remained. These 94,519 SNPs were then analyzed by iASeq.

2.3.2 A Simulation Study

Before we apply iASeq to the real data, we first tested its performance in simulations that took into account real data characteristics. Our simulations kept the same design as the real GM12878 ChIP-seq data, with the same number of datasets and the same number of replicates within each dataset, except

that the genomic DNA sample was not used here since we knew the truth in the simulations and did not need genomic DNA as a control for potential bias. To create the simulation data, we first applied iASeq to the real GM12878 data to identify 86,353 SNPs that were not skewed in any dataset using $Pr(a_i = 0|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) > 0.5$ as cutoff. To mimic the real background noise, these SNPs were resampled by a bootstrap procedure to create the background SNPs in the simulations, and we kept the read counts (x_{idj}, n_{idj}) of each background SNP as is in the simulated data. Next, we simulated ASB SNPs and added them to the background. Simulations were carried out under two different scenarios (Figure 2.2).

- Scenario 1: Two types of ASB SNPs (classes 1 and 2) were created in addition to the background SNPs (class 0). The SNP number for class 0, 1, and 2 was 85,069, 4,725 and 4,725 respectively. Thus the true π_k for the three classes was 0.90, 0.05 and 0.05 respectively. SNPs in class 1 were SR in datasets 1 to 30 (i.e., their $b_{id} = 1$ for $d = 1, \cdots, 30$). SNPs in class 2 were SN in datasets 1 to 30 (i.e., $c_{id} = 1$ for $d = 1, \cdots, 30$). In datasets 31 to 40, no SNPs had allelic imbalance. Class 2 can be viewed as the mirror image of class 1. This symmetric design reflects the symmetry of allele-specificity, that is, the skewing to the reference allele and to the non-reference allele is approximately symmetric. The class abundance (0.90,0.05,0.05) roughly matched the abundance observed in the analysis of real GM12878 data.

- Scenario 2: Four correlation patterns (classes 1-4) were created in addition to the background class (class 0). Class 1 and class 2 were the same

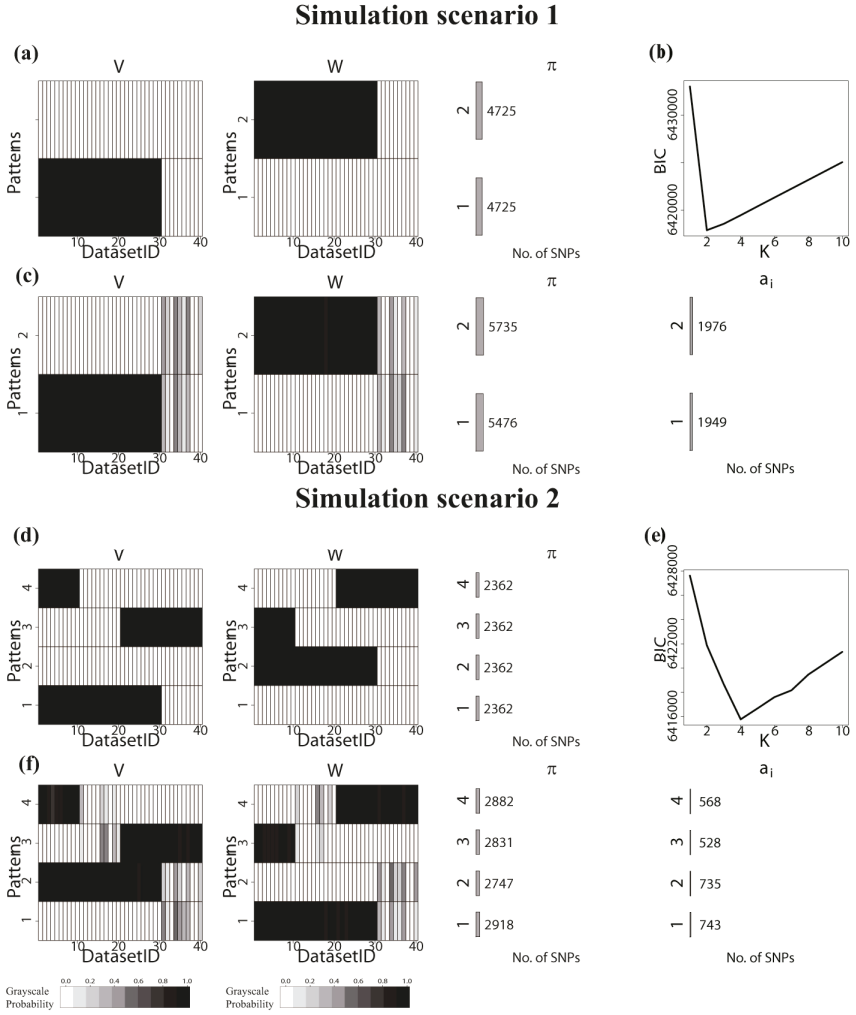


Figure 2.2: Simulation design and patterns discovered by iASeq (a) The true ASB patterns in simulation 1. Two non-background patterns were simulated in addition to the background pattern and shown here. Each row in the plot represents a SNP class, and each column represents a dataset. The color in the cell (k, d) demonstrates the SR or SN probability in class k and dataset d. Black means skewed, and white means not skewed. (b) The BIC values for different class number K in simulation 1. (c) Patterns discovered by iASeq in simulation 1. The plot shows the estimated $\boldsymbol{V}$ and $\boldsymbol{W}$ when $K = 2$. The numbers shown under π are the estimated number of SNPs in each class (i.e., $\hat{\pi}_k*$ the total number of SNPs). The numbers shown under a_i are the number of SNPs identified for the corresponding class using the posterior probability $Pr(a_i = k|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) > 0.9$ as cutoff. (d) The true ASB patterns in simulation 2. Four non-background patterns were simulated in addition to the background pattern and shown here. (e) The BIC values for different class number K in simulation 2. (f) The patterns discovered by iASeq in simulation 2.

as in simulation 1. Classes 3 and 4 were two new patterns. SNPs in class 3 were SR in datasets 21-40, and SN in datasets 1-10. Class 4 was the mirror image of class 3. The abundance of the classes 0 to 4 was (0.90,0.025,0.025,0.025,0.025).

Given the simulated $[\boldsymbol{B}_i; \boldsymbol{C}_i]$ configurations, we then simulated the read count data for ASB SNPs as described in detail in Supplemental Methods S.6.6. Simulations done in this way was able to keep the major characteristics of real data while allowing us to benchmark the performance of different methods since we knew the truth.

We applied iASeq to both simulations. In both cases, iASeq was able to identify the correct number of SNP classes using BIC (Figures 2.2a,b,d,e). Figures 2.2c and f show that the ASB patterns reported by iASeq matched the true patterns well. In order to test whether iASeq can improve the statistical power of detecting SNPs with allelic imbalance, we compared the SNP ranking provided by iASeq with rankings provided by five other methods that analyze each dataset separately (Figure 2.3). In iASeq, SNPs were ranked in each dataset d according to the posterior probability $\tilde{P}_{id}$ defined by Formula 2.8. Since we know the truth, we can count how many of the top N SNPs were true positives. Here the true positives were defined as SNPs that were truly allele-specific and also had the skewing direction correctly inferred. The five single-dataset based methods for ranking SNPs include a *deviation statistic* d, *naive z statistic*, *naive Bayes statistic*, *empirical Bayes statistic* and *single dataset EM*. These methods were applied to each individual dataset. For each dataset d, we merged data from all replicates to obtain $x_{id} = \sum_{j=1}^{J_d} x_{idj}$ and $n_{id} = \sum_{j=1}^{J_d} n_{idj}$. We then

computed the statistics used for SNP ranking as described below.

1. *Deviation statistic (d)*: SNPs were ranked based on $|x_{id}/n_{id} - p_{d0}|$. Here we estimated $p_{d0} = \frac{1}{I'}\sum_{i:n_{id}\neq 0} p_{id} = \frac{1}{I'}\sum_{i:n_{id}\neq 0} \frac{x_{id}}{n_{id}}$, where I' is the number of SNPs for which $n_{id} \neq 0$.

2. *Naive z statistic (z)*: SNPs were ranked based on $\frac{|x_{id}/n_{id} - p_{d0}|}{\sqrt{(p_{d0} * (1-p_{d0})/n_{id})}}$. Here p_{d0} was estimated as in the *deviation statistic d*.

3. *Naive Bayes statistic (b)*: SNPs were ranked using $|(x_{id}+2*\tilde{p}_{d0})/(n_{id}+2) - \tilde{p}_{d0}|$. Here $\tilde{p}_{d0} = \frac{1}{I}\sum_i \frac{x_{id}+2*p_{d0}}{n_{id}+2}$ where p_{d0} was estimated as in the *deviation statistic d*. The implicit assumption here is that $x_{id}|p_{id} \sim Bin(n_{id}, p_{id})$ and $p_{id} \sim Beta(\alpha_d, \beta_d)$ with $\alpha_d = 2\tilde{p}_{d0}$ and $\beta_d = 2(1-\tilde{p}_{d0})$. The posterior mean of p_{id} is used to construct the ranking statistic.

4. *Empirical Bayes statistic (B)*: SNPs were ranked using $|(x_{id} + \hat{\alpha}_d)/(n_{id} + \hat{\alpha}_d + \hat{\beta}_d) - \check{p}_{d0}|$. We estimated $\check{p}_{d0} = \frac{\hat{\alpha}_d}{\hat{\alpha}_d + \hat{\beta}_d}$. The implicit assumption is the same as the *naive Bayes statistic*, but now we estimate α_d and β_d based on the observed data using the method of moments as in iASeq (see Supplemental Method Section 2.6.2).

5. *Single dataset EM (singleEM)*: We fitted a mixture model of SR, SN and NS with distributions $f_{idjp}(\cdot), p = 0, 1, 2$ and mixing probabilities v_d, w_d and $1 - v_d - w_d$ for each dataset d without considering other datasets. SNPs were ranked using a posterior probability similar to $\tilde{P}_{id}$, but now determined based on information in dataset d only (see Supplemental Method Section 2.6.7 for details).

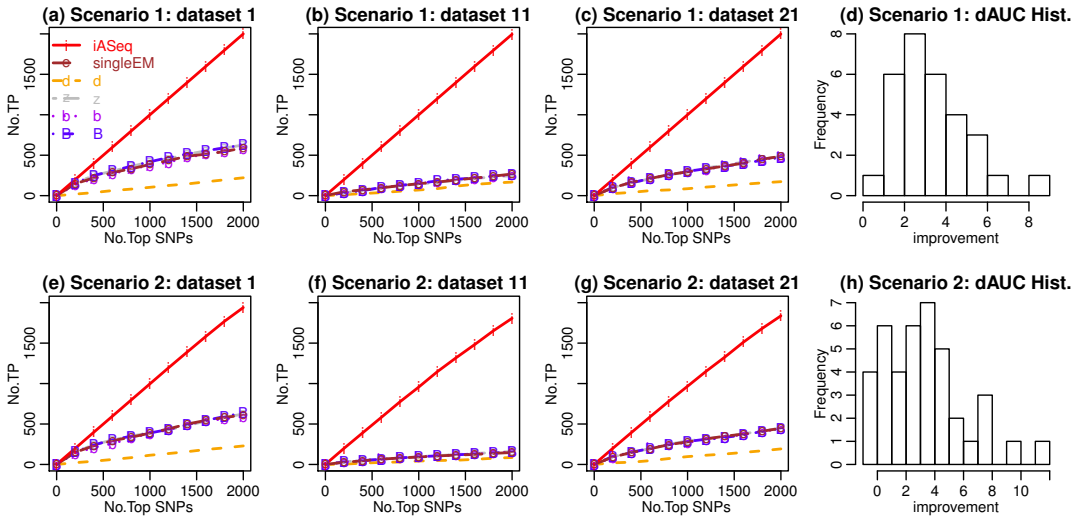


Figure 2.3: The Receiver Operating Characteristic (ROC) curves for simulations (a)-(c) We plot the number of true allele-specific SNPs (i.e., true positives, TP) among the top q ranked SNPs in each dataset against the rank cutoff q. Results for different methods in three representative datasets in simulation 1 are shown. Results in all other datasets were similar. (d) For each ranking method and each dataset, we computed the area under the ROC curve (AUC) using the 2000 top ranked SNPs. dAUC, the proportion of improvement of AUC brought by iASeq over the best AUC obtained from the single-dataset based methods, was computed for each dataset. $dAUC > 0$ means iASeq brings improvement. The distribution of dAUC in all 40 datasets is shown for simulation 1. (e)-(g) Results in three representative datasets from simulation 2. Results in all other datasets were similar. (h) The distribution of dAUC in all 40 datasets is shown for simulation 2.

Figure 2.3 compares the number of true positives, $TP_d(q)$, in the top q SNPs reported by each method in each dataset d. In Figures 2.3a-c and e-g, $TP_d(q)$ is plotted as a function of q in a few representative datasets. These plots show that iASeq outperformed all single-dataset based methods, and it was able to substantially improve the power for detecting allele-specificity.

In general, the observed differences between iASeq and the *d, z, b* and *B* statistics could be caused by many factors such as use of different statistical models, ranking statistics, or methods for parameter estimation. However, the comparison between iASeq and the single dataset EM represents a well-controlled comparison since these two methods used exactly the same distributional assumptions and parameter estimation methods. The only difference between them was that iASeq used information from multiple datasets whereas *singleEM* was based on one dataset only. This well-controlled comparison shows that jointly modeling multiple datasets is able to improve the allelic inference.

To examine whether iASeq was able to bring improvement in all datasets, we computed the Area under the Receiver Operating Characteristic (ROC) curves (AUC) for each method in each dataset using the top 2000 ranked SNPs. In each dataset, we computed the proportion of improvement in terms of AUC brought by iASeq over the best single-dataset based ranking method (i.e., $dAUC = \frac{AUC_{iAseq} - AUC_{bestsingle}}{AUC_{bestsingle}}$). $dAUC > 0$ means iASeq is able to bring improvement. Figures 2.3d and h show the distribution of $dAUC$ across all 40 datasets as a histogram. The results show that iASeq was able to improve the SNP ranking in almost all datasets.

In Figure 2.4, we converted the iASeq posterior probabilities of top N SNPs to FDR estimates and plotted the estimated FDR against the true FDR. The

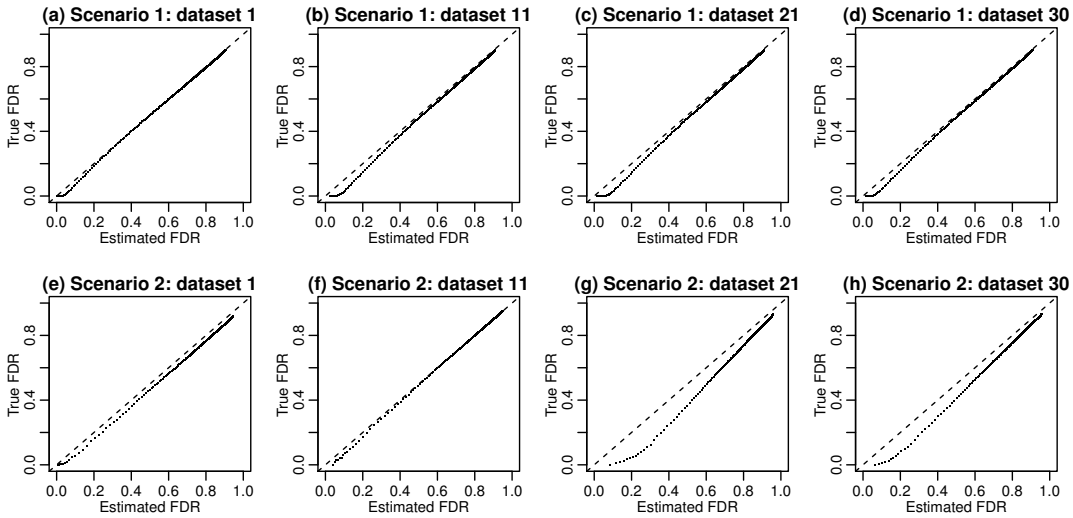


Figure 2.4: Estimated FDR against true FDR in simulations (a)-(d) Results for four representative datasets in simulation 1. (e)-(h) Results for four representative datasets in simulation 2. Results for all other datasets were similar.

figure shows that iASeq was able to provide reasonable FDR estimates as well. Shown in the figure are a few representative datasets. Results in all other datasets were similar.

2.3.3 Analysis of real data

Our simulation study demonstrates the ability of iASeq to discover correlation patterns of allele-specificity and improve the detection of skewed SNPs. Next, we applied iASeq to analyze the 41 real datasets (78 samples) in GM12878 cells. In real data, we do not have comprehensive knowledge about the truth. Therefore, unlike simulations, we were not able to assess the FDR estimates. For this reason, we mainly focused on analyzing the correlation patterns of allele-specificity and testing whether iASeq can improve the SNP ranking.

Correlation patterns of allele-specificity

Figure 2.5a shows the BIC in the real data. Based on BIC, the optimal K was 2. In other words, in addition to the background class ($k = 0$), iASeq discovered two other SNP classes, representing different allele-specificity patterns. For these two non-background classes, π_k was estimated to be 0.0696 and 0.0691 respectively, suggesting that they cover 6.96% and 6.91% of the analyzed SNPs. Due to the background noises, not all SNPs in these two classes can be confidently detected. At the 0.90 posterior probability cutoff, iASeq reported 1868 and 2138 SNPs for classes 1 and 2 respectively (Figure 2.5b). Note that our simulations had similar settings as the real data analysis, and they showed that iASeq was able to discover more than two patterns if they are supported by the data. Therefore our discovery of two correlation patterns here is likely driven by the data, that is, the information in the data is only sufficient for supporting robust discovery of two patterns. Figures 2.5b and c show the posterior mode of $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$ for the two non-background classes. It turned out that these two classes corresponded to two global directions of allele-specificity, SR and SN, respectively. Since the assignment of reference or non-reference allele depends on the reference genome, the assignment *per se* is not of biological interest. However, recall that GM12878 is a single person, therefore at each single SNP, the nucleotide representing the reference or non-reference allele is the same across all datasets analyzed here. Given this fact, what these results essentially tell is that at each single SNP, most TFs and HMs in our analysis were highly correlated in terms of allele-specificity, and if they are skewed, they tend to be skewed toward the same direction (i.e., the same allele).For instance, for SNPs

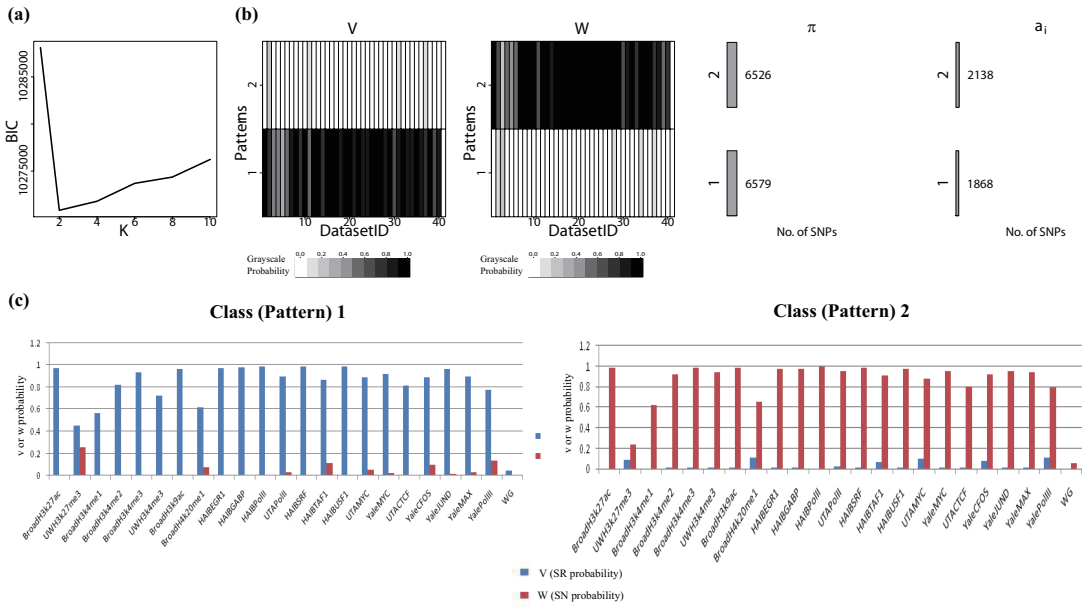


Figure 2.5: Correlation patterns of allele-specificity among different TFs and HMs in GM12878 cells discovered by iASeq (a) The BIC values for different class number K. The BIC achieves the minimum at $K = 2$. (b) The estimated $\boldsymbol{V}$ and $\boldsymbol{W}$ when $K = 2$. Each row corresponds to a class. Each column represents a dataset. The color in the cell (k, d) represents the SR or SN probability in class k and dataset d. From white to dark, the probability increases from 0 to 1. The bar plot and the numbers shown under π are the estimated number of SNPs in each class (i.e., $\hat{\pi}_k *$ the total number of SNPs). The bar plot and the numbers shown under a_i are the number of SNPs identified for the corresponding class using the posterior probability $Pr(a_i = k|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) > 0.9$ as cutoff. (c) A closer look at $\boldsymbol{V}$ and $\boldsymbol{W}$ in a number of representative datasets. The barplots show the estimated SR and SN probabilities v_{kd} and w_{kd} in a number of selected datasets. Left: the skewing probabilities in class 1. Right: the skewing probabilities in class 2. The height of each bar represents the SR or SN probability.

in class 1, both H3K4me3 (from the Broad Institute) and H3K27ac (Broad) had high probability to be SR, with (v_{kd}, w_{kd}) equal to (0.9337,0.0070) and (0.9730,0.0041) respectively (Figure 2.5c). The probability that one is SR and the other one is SN was small. Similarly, for SNPs in class 2, both H3K4me3 and H3K27ac were highly likely to be SN simultaneously ((v_{kd}, w_{kd})=(0.0061,0.9835) for H3K4me3 (Broad) and (0.0040,0.9897) for H3K27ac (Broad)). While the allelic imbalance of most TFs and HMs were highly correlated, H3K27me3, a HM involved in gene repression, was an exception. In both non-background classes, H3K27me3 had much lower skewing probabilities compared to the other proteins (Figure 2.5c). Within each class, the difference in the skewing probability between the two alleles was also much weaker for H3K27me3 as compared to the other proteins. For instance, in class 1, while most other proteins showed strong preference to be skewed toward the reference allele, H3K27me3 can be skewed to the reference allele at some SNPs and skewed to the non-reference allele at many other SNPs. Therefore, the allelic imbalance in H3K27me3 is not strongly correlated with the allelic imbalance of the other proteins analyzed here. For the genomic DNA which was used as control here, the skewing probabilities (v_{kd}, w_{kd}) in both classes were fairly low as shown in Figure 2.5b-c. In both classes, the probability for not being skewed in the genomic DNA (i.e., $1 - v_{kd} - w_{kd}$) was bigger than 0.95. This indicates that the high probability of skewing observed in the other datasets was not an artifact.

The coordinated allelic imbalance of different proteins toward the same allele has also been observed in a recent study (Reddy *and others* (2012)). In that study, the authors analyzed AS of 24 TFs and found that when multiple TFs bind to the same SNP, they frequently bind to the same allele. Moreover, those

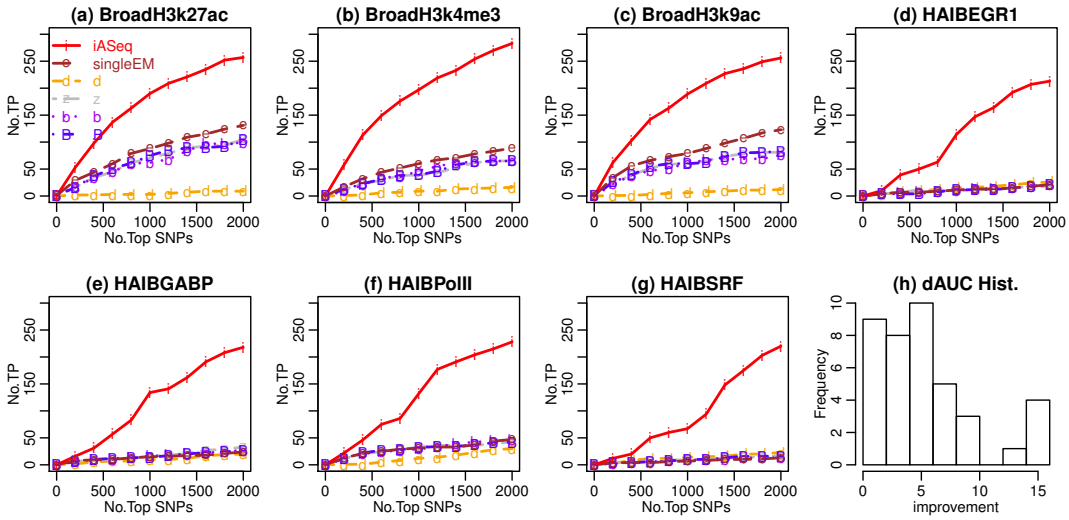


Figure 2.6: The ROC curves with chrX-npa SNPs as gold standard in the GM12878 analysis. We plot the number of non-pseudoautosomal region X chromosome SNPs, denoted by $TP_d(q)$, among the top q ranked SNPs in dataset d as a function of the rank cutoff q for each method. (a)-(g) Results in 7 representative datasets. (h) In each dataset, we computed the area under the ROC curve (AUC) using the 2000 top ranked SNPs for each method. dAUC, the proportion of improvement of AUC brought by iASeq over the best AUC from the single-dataset based methods, was computed for each dataset. The distribution of dAUC in all 40 datasets is shown.

authors did not observe any pair of TFs that regularly bind the same position on alternate alleles. Our observation here therefore is consistent with their finding.

Increased power for detecting allele-specificity compared with single dataset analysis

We ranked SNPs based on the posterior probabilities $\tilde{P}_{id}$ in each dataset. The iASeq ranking was compared with the rankings provided by the five single-dataset based methods described above. Since we do not know the truth, we used two types of independent information as gold standard to benchmark the ranking results.

First, we evaluated different methods by counting how many of their top ranked SNPs were located in the non-pseudoautosomal regions of chromosome X (chrX-npa) (Figure 2.6). GM12878 is a female lymphoblastoid cell line. In GM12878, SNPs in chrX-npa are expected to be allele-specific due to cells rapidly become clonal in culture leading to a skewed X-inactivation (McDaniell *and others* (2010); Kucera *and others* (2011); Reddy *and others* (2012)). Therefore, given a fixed number of top SNPs, the more chrX-npa SNPs one can find, the more powerful a method is. Figure 2.6 shows that iASeq clearly increased the power for detecting allele-specificity in each dataset compared to the single-dataset based analysis. For example, Figure 2.6 a shows that in the H3K27ac dataset generated by the Broad Institute, iASeq was able to identify 122 chrX-npa SNPs among the top 500 SNPs. This represents 126% improvement compared to *singleEM*, the best single-dataset based ranking method in that dataset, which only identified 54 chrX-npa SNPs. Figures 2.6a-g show results in a few representative datasets. Figure 2.6h shows the distribution of $dAUC$ (i.e., the proportion of improvement of AUC by iASeq over the best single-dataset based ranking method in each dataset) in all 40 datasets. These plots clearly show that iASeq outperformed all single-dataset based methods in all datasets and the average improvement in AUC was 354%.

Second, we evaluated different methods by using independent RNA-seq data. From RNA-seq, one can identify exonic ASE SNPs and use them as gold standard. We collected one RNA-seq datasets in GM12878 from the California Institute of Technology (Caltech). We identified the top 400 exonic ASE SNPs using the naive Bayes statistics. Using the other methods to identify the gold standard ASE SNPs produced similar results which, for simplicity, will not be

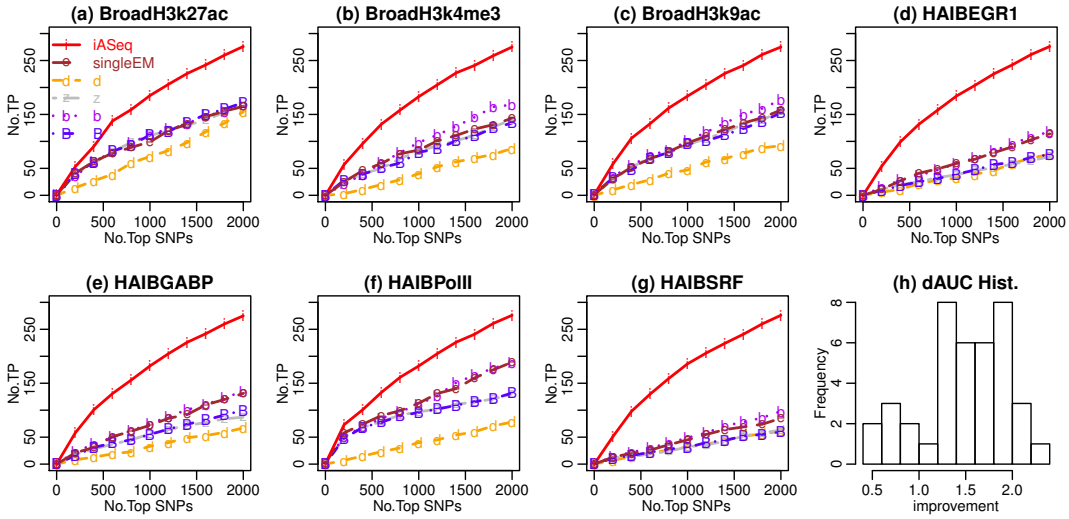


Figure 2.7: The ROC curves in GM12878 data using Caltech RNA-seq ASE SNPs as gold standard. We plot $TP_d(q)$, the number of true allele-specific SNPs among the top q ranked SNPs in dataset d, against the rank cutoff q for each method. The true allele-specific SNPs are defined as SNPs that have ≥ 1 RNA-seq exonic ASE SNPs in their 10kb neighborhood. (a)-(g) Results in 7 representative datasets. (h) In each dataset, we computed the area under the ROC curve (AUC) using the 2000 top ranked SNPs for each method. dAUC, the proportion of improvement of AUC brought by iASeq over the best AUC from the single-dataset based methods, was computed for each dataset. The distribution of dAUC in all 40 datasets is shown.

shown here. Based on these exonic ASE SNPs, we defined a SNP in our ChIP-seq analysis as truly allele-specific if there was an exonic ASE SNP in its Xkb neighborhood. Here we tried both $X = 10$kb and $X = 1$kb (data now shown) and obtained similar results. Below we illustrate the results using $X = 10$kb as an example. Among the 94,519 SNPs analyzed in the ChIP-seq data, 20,526 had one or more exonic SNPs within its 10kb neighborhood and therefore could potentially be linked to an exonic ASE SNP. Figure 2.7 compare rankings of these SNPs provided by different methods in terms of how many of the top ranked SNPs are true positives (i.e., associated with ASE). iASeq again outperformed all the other single-dataset based ranking methods. For instance, based on the Caltech gold standard, iASeq on average identified 144% more true positive SNPs among the top 500 SNPs (Figure 2.7a-g). The average improvement in terms of AUC (i.e., $dAUC$) across all 40 datasets was 148% (Figure 2.7h).

To ensure that the increased statistical power was not completely attributed to X chromosome SNPs, we repeated the benchmark analysis based on RNA-seq using only SNPs in autosomal chromosomes, and we obtained similar results (Figure 2.8). This shows that the increased power is not only contributed by chrX SNPs.

Comparisons with other methods

Most existing studies on allele-specificity were conducted using in-house data analysis pipelines. A tool developed by Skelly et al. (Skelly *and others* (2011)) and AlleleSeq (Rozowsky *and others* (2011)) are two software tools accessible to third-party users for AS analysis. The method proposed by Skelly et al. (Skelly *and others* (2011)) is designed for analyzing ASE in RNA-seq data. It first fits a

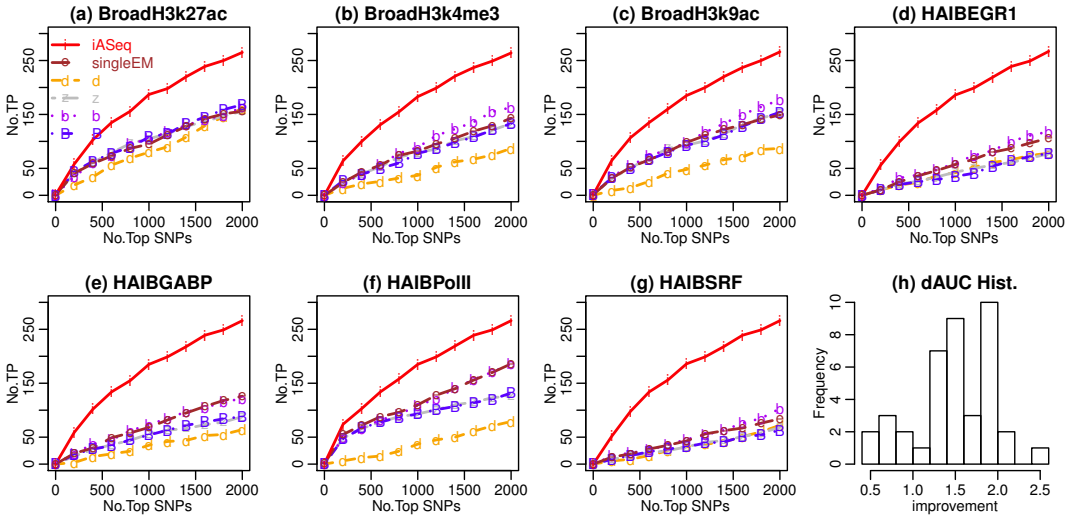


Figure 2.8: The ROC curves in GM12878 data using Caltech RNA-seq autosomal ASE SNPs as gold standard. We plot $TP_d(q)$, the number of true allele-specific SNPs among the top q ranked autosomal SNPs in dataset d, against the rank cutoff q for each method. The true allele-specific SNPs are defined as autosomal SNPs that have ≥ 1 RNA-seq exonic ASE SNPs in their 10kb neighborhood. (a)-(g) Results in 7 representative datasets. (h) In each dataset, we computed the area under the ROC curve (AUC) using the 2000 top ranked SNPs for each method. dAUC, the proportion of improvement of AUC brought by iASeq over the best AUC from the single-dataset based methods, was computed for each dataset. The distribution of dAUC in all 40 datasets is shown.

background model using genomic DNA and then feeds the estimated parameters into a Bayesian model that combines information from multiple SNPs within a gene to infer ASE. When we applied this method to analyzing the GM12878 ChIP-seq data, two problems occurred. First, the method uses Markov Chain Monte Carlo (MCMC) to fit the background model from the genomic DNA which, as alerted by Skelly *and others* (2011), is well-known for its slow speed and difficulties for users to monitor the convergence. Our genomic DNA data had 94,519 SNPs which covered 12,417 genes. Running this algorithm on this data using the parameter settings recommended by Skelly *and others* (2011) on a machine with 2.7 GHz CPU and 4 Gb RAM took more than 60 days. Second, after feeding the background model parameters obtained from the first step to the inference model in the second step, the algorithm stopped execution after a few iterations. This is because the original model was developed for deeply sequenced RNA-seq rather than ChIP-seq, where the average read count covering a heterozygote SNP in a ChIP-seq dataset is only 0.64. As a result, the model developed in Skelly *and others* (2011) did not fit the real data in ChIP-seq experiments. This lack-of-fit caused the program to stop early, likely due to the abnormally fitted parameters causing various computation problems (e.g., overflow). For this reason, although the method proposed by Skelly *and others* (2011) represents an advanced solution for analyzing RNA-seq ASE, it cannot be directly used to analyze ASB in ChIP-seq data without significantly redesigning the model and algorithm. For this reason, it is not further compared here.

AlleleSeq (Rozowsky *and others* (2011)) is another tool for AS analysis. It has been used to analyze ASB of several TFs in GM12878 (Rozowsky *and others*

Gold standard	ChrX			All Caltech ASE exonic SNPs			Autosomal Caltech ASE exonic SNPs		
TF	T_d	AlleleSeq	iASeq	T_d	AlleleSeq	iASeq	T_d	AlleleSeq	iASeq
YaleCFOS	41	3	4	9	5	3	9	5	3
YaleMYC	122	9	22	39	5	10	38	5	10
YaleJUND	289	13	31	24	4	8	23	4	7
YaleMAX	105	3	18	18	3	1	18	3	2
YalePolIII	25	2	2	0	0	0	0	0	0

Table 2.1: Comparison of iASeq and AlleleSeq. Column 1: TF name. Column 2: T_d is the number of AlleleSeq reported ASB SNPs. Columns 3-4: the number of non-pseudoautosomal region X chromosome SNPs among the top T_d allele-specific SNPs reported by AlleleSeq and iASeq. Column 5: T_d is the number of AlleleSeq reported ASB SNPs that had an exonic SNP within their 10kb neighborhood. Columns 6-7 show among the top T_d allele-specific SNPs reported by AlleleSeq and iASeq, how many SNPs had ≥ 1 exonic ASE SNP in their 10kb neighborhood according to the Caltech RNA-seq experiment. Column 8: T_d is the number of AlleleSeq reported autosomal ASB SNPs that had an exonic SNP within their 10kb neighborhood. Columns 9-10 show among the top T_d autosomal allele-specific SNPs reported by AlleleSeq and iASeq, how many SNPs had ≥ 1 exonic ASE SNP in their 10kb neighborhood according to the Caltech RNA-seq experiment.

(2011)). AlleleSeq is more focused on the preprocessing step. Its pipeline first constructs a diploid personal genome sequence according to family trio data and then maps ChIP-seq reads to this personal genome. After removing various biases, the method then analyzes allele-specificity in each individual ChIP-seq dataset separately. Rozowsky *and others* (2011) applied AlleleSeq to analyze 7 different TF datasets in GM12878, among them 5 were also included in our iASeq analysis. We compared iASeq and AlleleSeq using these same 5 datasets. We first obtained the ASB SNPs reported by AlleleSeq from Rozowsky *and others* (2011). Let T_d denote the number of reported ASB SNPs for each TF dataset d. We next obtained the top T_d SNPs ranked by iASeq. We then compared these two methods based on how many of their top T_d SNPs were in chrX-npa, and how many of them were associated with exonic ASE SNPs determined by RNA-seq. We also performed the comparison after excluding the chromosome X SNPs. Table 2.1 shows that iASeq either outperformed or performed comparable to AlleleSeq in all datasets. Sometimes, the improvement was substantial (e.g., YaleMYC).

2.4 Discussion

In summary, we have proposed a Bayesian hierarchical mixture model iASeq to integrate multiple ChIP-seq datasets for analyzing allele-specificity. The primary goal of iASeq is to increase the statistical power of AS detection, and it does so by taking the advantage of correlations among datasets. Since the correlation structure may not be known before the data analysis, iASeq learns it from the data automatically. Application of iASeq to the ENCODE GM12878

data shows that allelic imbalance of most analyzed TFs and HMs have strong preference to be skewed toward the same direction. Analysis of both the simulated and real data show the effectiveness of iASeq to improve detection of allele-specificity compared to single-dataset based methods.

2.4.1 Interpretation of the correlation patterns

When analyzing the real data in GM12878, iASeq found two non-background AS patterns, representing two opposite directions of allelic imbalance. Since the assignment of reference and non-reference allele depends on the reference genome, whether a SNP is skewed toward reference or non-reference allele *per se* does not have direct biological meaning. What these two patterns essentially suggest is that the allelic imbalances of multiple proteins at a single SNP are correlated and have high preference to be skewed toward the same allele. In other words, the two patterns should be viewed as a pair and interpreted together.

In general, although one may view different allelic imbalance patterns in iASeq as different clusters of SNPs, these clusters only describe the similarities among SNPs in terms of their skewness directions, rather than the similarities in terms of their functions. The direction is defined using the reference/non-reference allele. The reference or non-reference allele for different SNPs can have different meanings (e.g., for one SNP, the maternal allele may be the reference allele, whereas for another SNP the paternal allele may be the reference allele). Therefore within each cluster, even though SNPs have similar skewness pattern, they are not necessarily functionally related to each other. One should not confuse the SNP clusters here with the clusters obtained from the traditional

gene expression microarray data analysis, where co-expressed genes in a cluster often have similar functions. In iASeq, the clusters only serve as a tool to describe the correlation structure among different datasets (i.e., proteins), rather than the functional correlation among different SNPs. The correlation patterns among datasets are used by iASeq to inform one how to integrate information across datasets (i.e., which datasets are highly correlated and therefore can borrow information from each other) to improve detection of AS events for each individual SNP and dataset. In order to understand functions of the detected AS events, one needs to further correlate the iASeq results with other information (e.g., one may determine the parent-of-origin of each SNP first and then study various phenomena such as imprinting).

Our observation that different proteins prefer to be skewed in the same direction is consistent with a recent observation reported in Reddy *and others* (2012) that AS of 24 different TFs are skewed toward the same allele. A number of factors could contribute to the observed correlation. First, biologically it is plausible that functionally related HMs and TFs have correlated allele-specificity. For instance, both H3K4me2 and H3K4me3 are markers for active transcription. Therefore, for a specific SNP, if the reference allele is associated with a gene with active transcription but the non-reference allele is not, then it is very likely that both H3K4me2 and H3K4me3 will be skewed toward the reference allele. For another SNP, if the non-reference allele is transcribed but the reference allele is not, then both H3K4me2 and H3K4me3 will have high probability to be skewed toward the non-reference allele. In the genome, H3K4me2 and H3K4me3 are skewed toward reference allele for some SNPs, and skewed toward non-reference allele for some other SNPs. Therefore the skewed SNPs

could naturally fall into two clusters, representing two opposite AS directions. Second, as pointed out by Reddy *and others* (2012), the coordinated AS could also occur as a result of the difference in the chromatin landscape between the two alleles. For instance, if the chromatin on one allele is more open and accessible, it could increase the overall binding probability of multiple different proteins, leading to correlated allelic skewing.

While our results show that most analyzed TFs/HMs tend to be skewed toward the same direction, these results do not imply that these proteins are perfectly correlated in terms of allele-specificity at each and every SNP. In iASeq, the correlation patterns $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$ are probabilistic patterns rather than 0-1 vectors. Each correlation class k can generate all 3^D AS configurations. For instance, for a class with $[\boldsymbol{V}_k; \boldsymbol{W}_k] = [(0.9, 0.9, 0.9, 0.1); (0.1, 0.1, 0.1, 0.1)]$, it is possible to have one SNP with configuration $[SR, SR, NS, NS]$ and at the same time another SNP with configuration $[SR, NS, SR, NS]$. Therefore, SNPs in the same class are not required to have the same AS configuration, even though they tend to have similar AS configurations. The probabilistic patterns are used here to provide a parsimonious description of the complex correlation structure in the data, so that one can circumvent the difficulty of handling 3^D AS configurations whose complexity increases exponentially. As a consequence of using this parsimonious model, multiple weak correlation patterns without strong enough data support could be merged into a bigger class. For instance, consider two AS patterns $[\boldsymbol{V}_k; \boldsymbol{W}_k] = [(1, 1, 0, 0); (0, 0, 0, 0)]$ (i.e., [SR,SR,NS,NS]) and $[\boldsymbol{V}_k; \boldsymbol{W}_k] = [(0, 0, 1, 1); (0, 0, 0, 0)]$ (i.e., [NS,NS,SR,SR]). Suppose both patterns are equally likely to occur in the data. If each pattern is only associated with a small number of SNPs, then a parsimonious model will prefer merging them

together into one single class with $[\boldsymbol{V}_k; \boldsymbol{W}_k] = [(0.5, 0.5, 0.5, 0.5); (0, 0, 0, 0)]$. For this reason, iASeq only discovers correlation patterns that have sufficient data support so that they can be distinguished from other patterns. It will not report weak patterns, which could be real but do not have enough data support to allow them to be robustly recovered. For users, this means that at the cluster level, they may not be able to see weak but real AS correlation patterns if these patterns are not associated with enough number of SNPs. On the other hand, for the purpose of inferring whether or not each SNP is allele-specific in each dataset, these parsimonious correlation patterns are sufficient for describing the correlation structure in the data and serving as a prior to guide the information sharing across datasets. The information sharing will lead the increased ASB detection power, and the eventual AS configuration at each individual SNP will be determined by the posterior probabilities of (b_{id}, c_{id}) (i.e., $\tilde{P}_{id}$) rather than the cluster-level prior probabilities $[\boldsymbol{V}_k; \boldsymbol{W}_k]$. Therefore, in the final AS calls, the model still allows each SNP to have its own AS configuration which may not necessarily be the same as the AS configurations of other SNPs from the same cluster.

Consistent with Reddy *and others* (2012), in the two non-background AS patterns discovered here, proteins skewed toward the same direction did not always correspond to known protein-protein interactions. As pointed out by Reddy *and others* (2012), this could happen as a result of allelic imbalances of different proteins being caused by a common underlying factor such as allelic difference in chromatin landscape. It could also reflect unknown protein-protein interactions. For iASeq specifically, there is a third reason, that is,

multiple small patterns can be merged into a bigger probabilistic class as described before. For example, because of the use of probabilistic patterns, two patterns [SR,SR,NS,NS] and [NS,NS,SR,SR] may be merged into a single SNP class (e.g., $[\boldsymbol{V}_k; \boldsymbol{W}_k] = [(0.5, 0.5, 0.5, 0.5); (0, 0, 0, 0)]$). As a result, only looking at the pattern represented by $[\boldsymbol{V}_k; \boldsymbol{W}_k]$, one cannot tell the details of protein-protein interactions, such as these interactions only exist between datasets 1 and 2, or between 3 and 4, but not between the other pairs of datasets. What one can tell from this merged pattern is that, when the allelic imbalance occurs in these four datasets, they will be skewed toward the same direction, i.e., the reference allele in this example.

In summary, while the correlation patterns in iASeq provide some insights on the correlation of allelic imbalance among different datasets, one should not over-interpret them. The primary goal of these patterns is to describe the correlation structure in the data so that information from different datasets can be shared in a principled way to increase the power of statistical inference. This also points to an important difference between this study and previous studies that reported coordinated allele-specificity among multiple proteins. The previous studies only reported the correlation as a biological finding, but did not provide a statistical method to further utilize the correlation structure to improve the statistical inference. In contrast, iASeq provides a general and rigorous statistical method that utilizes the automatically discovered correlation patterns to increase the statistical power of AS detection. As such, it represents a novel development for the analysis of allele-specificity.

2.4.2 Model, algorithm, and possible extensions

Unlike tools such as AlleleSeq which mainly focus on the preprocessing steps for the AS analysis (e.g., construction of diploid personal genome), iASeq is developed as a general model working downstream of the preprocessing pipelines. The input data for iASeq are the read counts in the format shown in Figure 2.1a. With this design, iASeq can be easily coupled with different data preprocessing protocols. For instance, some investigators may map their reads to a reference genome, while others may map their reads to a diploid personal genome. Both types of investigators can use iASeq to integrate information from multiple datasets once they obtained the allelic read counts.

In iASeq, we used an EM algorithm to find the posterior mode of parameters and carried out statistical inference accordingly. In principle, one may also use a full Bayesian approach and Markov Chain Monte Carlo (MCMC) to perform the posterior inference. However, since MCMC usually takes much longer to run for a big dataset and it is not easy for users to monitor convergence, we decided to use the posterior mode and EM-based approach in our implementation. For analyzing the GM12878 data with 94,519 SNPs, iASeq took 5 hours to run the EM algorithm to fit a single model with $K = 1$ on a machine with 2.7 GHz CPU and 4Gb RAM. To fit a single model with $K = 10$ on the same machine, the EM took 16 hours. Running the EM for all 10 Ks between 1 and 10 on a single core took 4.6 days. However, when we run these 10 jobs in parallel on 10 cluster nodes, we were able to select the best model within 1 day. Therefore, running the algorithm on a single machine is a little time-consuming, but the computation time can be reduced by parallelization. Also,

our analysis of GM12878 data indicates that the optimal K in that real data was 2. For a K not extremely large, even if running the full BIC selection on a single machine takes some time, it usually requires less than a week, which is acceptable compared to the time devoted to preparing samples and generating data.

In principle, the statistical model developed in iASeq may also be applied to analyze other types of AS events, such as ASE and ASM. In the future, we plan to improve the model by incorporating information from the spatial correlation among closely located SNPs. For example, for the ASE analysis, one may jointly model SNPs from the same gene, similar to Skelly *and others* (2011).

2.4.3 Implications on future studies

The analysis of AS events using the high-throughput sequencing data frequently faces the problem of low statistical power due to the limited amount of information available at heterozygote SNPs. One way to increase the power is to increase the sequencing depth for one data type (e.g., MYC ChIP-seq). An alternative approach is to spend the same amount of money to generate data for multiple different but related data types (e.g., ChIP-seq for MYC, H3K4me1, H3K4me3, etc.), each with a lower coverage. One can then integrate the multiple datasets to increase the statistical power of allele-specificity analysis. The merit of the second approach is that one can collect multiple different types of information which might be useful for other purposes (e.g., in addition to studying MYC binding using MYC ChIP-seq, one may couple H3K4me1 ChIP-seq data with DNA motif information to locate active enhancers and predict binding sites of other TFs in the genome). If the second approach is used in the

study design, then iASeq will offer a flexible, powerful and scalable framework for better analyzing the AS events in the data. As ChIP-seq data continue to grow rapidly, this integrative approach will allow us to use the data more efficiently to characterize the allele-specificity.

2.5 Software

iASeq is freely available as an R package in Bioconductor:

http://www.bioconductor.org/packages/release/bioc/html/iASeq.html

2.6 Supplementary Materials

2.6.1 Data preprocessing

Data collection

Both ChIP-seq and RNA-seq data for GM12878 cells were downloaded from the ENCODE Project Consortium (2012). The datasets used in the analysis are summarized in Additional File 2 at http://www.biomedcentral.com/1471-2164/13/681/additional. For each ChIP-seq sample, we downloaded the FASTQ file and mapped the raw sequence reads to human genome (hg18) using MAQ (Version 0.7.1) with default parameters Li *and others* (2008). Uniquely mapped reads with the mapping quality score above 0 were extracted.

Heterozygote SNP collection, bias filtering and protein binding filtering

The genotype data for GM12878 was retrieved from the

ftp://ftp-trace.ncbi.nih.gov/1000genomes/ftp/pilot_data/release/2010_07/trio/snps.

A total of 1,704,166 heterozygote SNPs were obtained. As shown by Degner *and*

others (2009), in addition to the reference mapping bias (i.e., reference alleles are easier to be mapped back), certain SNPs have intrinsic biases toward one of the alleles. For these SNPs, genomic DNA extracted computationally from one allele (not necessarily the reference allele) is intrinsically easier to be mapped back compared to DNA extracted from the other allele. Using the method described in Degner *and others* (2009); Pickrell *and others* (2010), we identified and removed these 149,996 intrinsically biased SNPs. Next, we called ChIPseq peaks for each dataset using CisGenome (Ji *and others* (2008)) at 1% FDR level and retained only those heterozygous SNPs without intrinsic bias located in the protein binding regions in at least one dataset. After these two steps of filtering, 94,519 SNPs were included in our subsequent analysis.

Collection of exonic SNPs and ASE SNPs

The exonic ASE SNPs were used as one of our gold standards to evaluate iASeq and other methods. To determine which SNPs are exonic, we downloaded the hg18 Ensemble gene annotation file Homo_sapiens.NCBI36.54.gtf from

ftp://ftp.ensembl.org/pub/release-54/gtf/homo_sapiens/.

Exonic SNPs were annotated using the exonic regions from the gene annotation file. The exonic ASE SNPs were determined using RNA-seq. For a given RNA-seq dataset, the naive Bayes statistic was calculated for each exonic SNP. The top 400 exonic SNPs (6.61% of the total 6051 exonic SNPs) ranked based on the naive Bayes statistic were called as exonic ASE SNPs. Subsequently, in the analysis of ChIP-seq data, a SNP was claimed to be a true positive if there was an exonic ASE SNP within its Xkb neighborhood. Results for X=10kb and results for X=1kb gave similar results, and only results for X=10kb are shown.

2.6.2 Method of moment for estimating parameters in the Beta distribution

For a Beta distribution $Beta(\alpha, \beta)$, the mean is $\alpha/(\alpha+\beta)$, and the variance is $\alpha\beta/(\alpha+\beta)^2(\alpha+\beta+1)$. For sample (d, j), each SNP has a p_{idj} which can be roughly estimated as $\hat{p}_{idj} = (x_{idj} + 2 * p^{(0)}_{dj0})/(n_{idj} + 2)$. Here $p^{(0)}_{dj0} = \frac{1}{I^{(0)}_d} \sum_{i:n_{idj}\neq 0} x_{idj}/n_{idj}$, and $I^{(0)}_d$ is the number of SNPs in dataset d for which $n_{idj} \neq 0$. Let $p_{dj0} = \sum_i \hat{p}_{idj}/I$, and $v_{dj} = \frac{1}{I-1}\sum_i (p_{idj} - \hat{p}_{dj0})^2$. By matching p_{dj0} and v_{dj} to the theoretical mean and variance of a Beta distribution, we obtain

$$\hat{\alpha}_{dj} = p_{dj0} * [\frac{p_{dj0}(1-p_{dj0})}{v_{dj}} - 1] \tag{2.9}$$

$$\hat{\beta}_{dj} = (1-p_{dj0}) * [\frac{p_{dj0}(1-p_{dj0})}{v_{dj}} - 1] \tag{2.10}$$

In principle, one may develop a more sophisticated algorithm to estimate α and β by fitting beta-binomial distributions to $x_{idj}|n_{idj}$, but the computation will be more involved. Therefore we did not pursue this solution and instead used the simple method described above to approximately estimate α and β.

2.6.3 Parameter choice for the Dirichlet prior

Although $\eta = 1$ can specify an uniform prior and seems to be a natural choice, it will make the EM algorithm numerically unstable. This is because the EM searches for posterior mode and is implemented on log scale. The mode of a Dirichlet distribution $D(\eta_1, \cdots, \eta_M)$ for the m-th component is $(\eta_m - 1)/\sum_{l=1}^{M}(\eta_l - 1)$. It is not defined if all η_ms are equal to one. As a result, if $\eta = 1$ is used as the prior, and when the expectation of the counts $\sum_i \delta(a_i = k)$,

$\sum_i \delta(a_i = k) b_{id}$ or $\sum_i \delta(a_i = k) c_{id}$ in the E-step of the algorithm is close to zero, then the algorithm can easily lose its numerical stability due to issues such as $log(0)$ and ill-defined posterior mode. These issues can be avoided by using $\eta = 2$ which still imposes a relatively non-informative prior.

2.6.4 The EM algorithm used in iASeq

This section presents the EM algorithm used to search for posterior mode $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{V}}, \hat{\boldsymbol{W}})$ of the distribution:

$Pr(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}|\boldsymbol{X}, \boldsymbol{N}) = \sum_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}} Pr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}|\boldsymbol{X}, \boldsymbol{N})$. In the EM algorithm, $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are the missing data. The algorithm iterates between an E-step and an M-step.

In the E-step, one evaluates the Q-function $Q(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}|\hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}})$ which is defined as $E_{old}[\ln Pr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}|\boldsymbol{X}, \boldsymbol{N})]$. Here the expectation is taken with respect to probability distribution $Pr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}|\boldsymbol{X}, \boldsymbol{N}, \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}})$, abbreviated as $Pr_{old}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$, where $\hat{\boldsymbol{\pi}}^{\boldsymbol{old}}$, $\hat{\boldsymbol{V}}^{\boldsymbol{old}}$ and $\hat{\boldsymbol{W}}^{\boldsymbol{old}}$ are the parameter estimates obtained from the last iteration.

When we use $\eta = 2$ in the Dirichlet priors for $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$, we have

$$
\begin{aligned}
& \ln Pr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}|\boldsymbol{X}, \boldsymbol{N}) \\
= & \sum_{i=1}^{I} \{\delta(a_i = 0)(\ln \pi_0 + \sum_{d=1}^{D} \ln L_{id0}) \\
+ & \sum_{k=1}^{K} \delta(a_i = k)[\ln \pi_k + \sum_{d=1}^{D} b_{id}(\ln v_{kd} + \ln L_{id1}) + \sum_{d=1}^{D} c_{id}(\ln w_{kd} + \ln L_{id2}) \\
+ & \sum_{d=1}^{D} (1 - b_{id} - c_{id})(\ln(1 - v_{kd} - w_{kd}) + \ln L_{id0})]\}
\end{aligned}
$$

$$+ \quad \sum_{k=0}^{K} \ln \pi_k + \sum_{k=1}^{K} \sum_{d=1}^{D} [\ln v_{kd} + \ln w_{kd} + \ln(1 - v_{kd} - w_{kd})]$$

$$+ \quad constant \tag{2.11}$$

Therefore,

$$Q(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}}) \tag{2.12}$$

$$= \quad E_{old}[\ln Pr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \boldsymbol{X}, \boldsymbol{N})]$$

$$= \quad \sum_{k=0}^{K} \{\sum_{i=1}^{I} E_{old}[\delta(a_i = k)] + 1\} \ln \pi_k + \sum_{k=1}^{K} \sum_{d=1}^{D} \{(\sum_{i=1}^{I} E_{old}[\delta(a_i = k) b_{id}] + 1) \ln v_{kd}$$

$$+ \quad (\sum_{i=1}^{I} E_{old}[\delta(a_i = k) c_{id}] + 1) \ln w_{kd}$$

$$+ \quad + (\sum_{i=1}^{I} E_{old}[\delta(a_i = k)(1 - b_{id} - c_{id})] + 1) \ln(1 - v_{kd} - w_{kd})\} + constant$$

In the M-step, one finds $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ that maximize the Q-function $Q(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}})$. Denote them by $\hat{\boldsymbol{\pi}}^{\boldsymbol{new}}$, $\hat{\boldsymbol{V}}^{\boldsymbol{new}}$ and $\hat{\boldsymbol{W}}^{\boldsymbol{new}}$. These will give the new parameter estimates.

By solving

$$\frac{\partial Q(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}})}{\partial \pi_k} = 0 \tag{2.13}$$

$$\frac{\partial Q(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}})}{\partial v_{kd}} = 0 \tag{2.14}$$

$$\frac{\partial Q(\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W} | \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}})}{\partial w_{kd}} = 0 \tag{2.15}$$

We have

$$\hat{\pi}_k^{new} = \frac{\sum_{i=1}^{I} Pr_{old}(a_i = k) + 1}{I + K + 1} \tag{2.16}$$

$$\hat{v}_{kd}^{new} = \frac{\sum_{i=1}^{I} Pr_{old}(a_i = k, b_{id} = 1) + 1}{\sum_{i=1}^{I} Pr_{old}(a_i = k) + 3} \tag{2.17}$$

$$\hat{w}_{kd}^{new} = \frac{\sum_{i=1}^{I} Pr_{old}(a_i = k, c_{id} = 1) + 1}{\sum_{i=1}^{I} Pr_{old}(a_i = k) + 3} \tag{2.18}$$

In the formulas above, $Pr_{old}(a_i = k)$, $Pr_{old}(a_i = k, b_{id} = 1)$ and $Pr_{old}(a_i = k, c_{id} = 1)$ are computed as follows. To compute $Pr_{old}(a_i = k) = Pr(a_i = k|\boldsymbol{X}_i, \boldsymbol{N}_i, \hat{\boldsymbol{\pi}}^{\boldsymbol{old}}, \hat{\boldsymbol{V}}^{\boldsymbol{old}}, \hat{\boldsymbol{W}}^{\boldsymbol{old}})$, recall

$$Pr(a_i|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) = Pr(\boldsymbol{X}_i, a_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})/Pr(\boldsymbol{X}_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) \tag{2.19}$$

Since

$$Pr(\boldsymbol{X}_i, a_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) = \sum_{\boldsymbol{B}_i, \boldsymbol{C}_i} Pr(\boldsymbol{X}_i, a_i, \boldsymbol{B}_i, \boldsymbol{C}_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) \tag{2.20}$$

$$= \{\pi_0 \prod_{d=1}^{D} L_{id0}\}^{\delta(a_i=0)} \prod_{k=1}^{K} \{\pi_k \prod_{d=1}^{D} [v_{kd} L_{id1} + w_{kd} L_{id2} + (1 - v_{kd} - w_{kd}) L_{id0}]\}^{\delta(a_i=k)}$$

and

$$Pr(\boldsymbol{X}_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) = \sum_{a_i} Pr(\boldsymbol{X}_i, a_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) \tag{2.21}$$

$$= \pi_0 \prod_{d=1}^{D} L_{id0} + \sum_{k=1}^{K} \{\pi_k \prod_{d=1}^{D} [v_{kd} L_{id1} + w_{kd} L_{id2} + (1 - v_{kd} - w_{kd}) L_{id0}]\}$$

Therefore, $Pr_{old}(a_i = k)$ can be computed by replacing $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ with $\hat{\boldsymbol{\pi}}^{\boldsymbol{old}}$, $\hat{\boldsymbol{V}}^{\boldsymbol{old}}$ and $\hat{\boldsymbol{W}}^{\boldsymbol{old}}$.

$Pr_{old}(a_i = k, b_{id} = 1) = Pr_{old}(a_i = k) Pr_{old}(b_{id} = 1|a_i = k)$. $Pr_{old}(a_i = k)$ is computed as above. However,$Pr(b_{id}, c_{id}|a_i = k, \boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$ can be computed as

$$\frac{Pr(b_{id}, c_{id}, \boldsymbol{X}_i|a_i = k, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})}{Pr(\boldsymbol{X}_i|a_i = k, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})}$$

$$= \frac{Pr(b_{id}, c_{id}, \boldsymbol{X}_{id}|a_i = k, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})}{\sum_{b_{id},c_{id}} Pr(b_{id}, c_{id}, \boldsymbol{X}_{id}|a_i = k, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})}$$

$$= \frac{[v_{kd}L_{id1}]^{b_{id}}[w_{kd}L_{id2}]^{c_{id}}[(1 - v_{kd} - w_{kd})L_{id0}]^{1-b_{id}-c_{id}}}{v_{kd}L_{id1} + w_{kd}L_{id2} + (1 - v_{kd} - w_{kd})L_{id0}} \tag{2.22}$$

$Pr_{old}(b_{id} = 1|a_i = k)$ and $Pr_{old}(c_{id} = 1|a_i = k)$ can be obtained by plugging in $\hat{\boldsymbol{\pi}}^{\boldsymbol{old}}$, $\hat{\boldsymbol{V}}^{\boldsymbol{old}}$ and $\hat{\boldsymbol{W}}^{\boldsymbol{old}}$ into the formula above to replace $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$.

2.6.5 Bayesian Information Criterion (BIC) for choosing K

We compute BIC as

$$BIC(K) = -2 * \ln\{\prod_{i=1}^{I} Pr(\boldsymbol{X}_i|\boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})\} + (K + 2 * K * D) * \ln I$$

$$= -2 * \sum_{i=1}^{I} \ln \left[\pi_0 \prod_{d=1}^{D} L_{id0} + \sum_{k=1}^{K}\{\pi_k \prod_{d=1}^{D}[v_{kd}L_{id1} + w_{kd}L_{id2} + (1 - v_{kd} - w_{kd})L_{id0}]\}\right]$$

$$+K(2D + 1) \ln I \tag{2.23}$$

We calculate BIC for different values of K, and choose the K with the smallest BIC. Here $K+1$ is the number of classes. K is also the number of parameters in $\boldsymbol{\pi}$. $2KD$ is the number of parameters involved in $\boldsymbol{V}$ and $\boldsymbol{W}$. I is the SNP number. Strictly speaking, the data likelihood also involve terms $Pr(\boldsymbol{N}_i|\boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W})$. However, based on our assumption, these terms do not depend on K, $\boldsymbol{\pi}$, $\boldsymbol{V}$ and $\boldsymbol{W}$, and can be reduced to $Pr(\boldsymbol{N}_i)$. They can be viewed as constants for the purpose of choosing the optimal K. We do not include them in the BIC computation.

2.6.6 Data generation in simulation studies

To simulate a ASB SNP i, we first sampled a SNP from the 8166 non-background SNPs in the real GM12878 data. Here the non-background SNPs in the real data were determined by iASeq using $Pr(a_i = 0|\boldsymbol{X}_i, \boldsymbol{N}_i, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{W}) < 0.5$ as cutoff. Additionally, we also sampled a SNP from the 86,353 background SNPs in the real GM12878 data. Next, with these two real SNPs in hand, we went through each dataset d to generate the read counts for the simulated SNP. If $[b_{id}, c_{id}] = [0, 0]$, then we used the background SNP's read count data (x_{idj}, n_{idj}) in sample (d, j) to serve as the data of the simulated SNP in dataset d sample j. If $[b_{id}, c_{id}] = [1, 0]$, then we used the non-background SNP's read count data to simulate read counts as follows. For each replicate sample j in dataset d, we obtained the observed total read count n_{idj} for the non-background SNP. We then randomly drew a number p_{idj} from $U[p_{dj0}, 1]$, where p_{dj0} is the mean of $\frac{x_{idj}}{n_{idj}}$ over all background SNPs in the same sample (d, j). Subsequently, we simulated x_{idj} from a binomial distribution $Bin(n_{idj}, p_{idj})$ to serve as the simulated SNP's data in dataset d and sample j. If $[b_{id}, c_{id}] = [0, 1]$, we applied a similar procedure but the p_{idj} was drawn from $U[0, p_{dj0}]$.

2.6.7 The single dataset based EM analysis

This approach analyzes each dataset separately. Let $\boldsymbol{X}^d = (\boldsymbol{X}_{1d}, \cdots, \boldsymbol{X}_{Id})$ and $\boldsymbol{N}^d = (\boldsymbol{N}_{1d}, \cdots, \boldsymbol{N}_{Id})$ be the data from dataset d. We assumed that in each dataset d, a SNP i can be SR ($b_{id} = 1$), SN ($c_{id} = 1$) or NS ($b_{id} = 0$ and $c_{id} = 0$) with probability $(v_d, w_d, 1 - v_d - w_d)$. Let $\boldsymbol{B}^d = (b_{1d}, \cdots, b_{Id})$ and $\boldsymbol{C}^d = (c_{1d}, \cdots, c_{Id})$ be the ensemble of all ASB indicators in dataset d.

Adopting the same distributional assumption as in Equations 1-3 in the main mainuscript, the complete data likelihood can be derived as:

$$Pr(\boldsymbol{X}^d, \boldsymbol{N}^d, \boldsymbol{B}^d, \boldsymbol{C}^d | v_d, w_d) Pr(\boldsymbol{N}^d) Pr(\boldsymbol{X}^d, \boldsymbol{B}^d, \boldsymbol{C}^d | \boldsymbol{N}^d, v_d, w_d)$$

$$= Pr(\boldsymbol{N}^d) \prod_{i=1}^{I} \{[v_d L_{id1}]^{b_{id}} [w_d L_{id2}]^{c_{id}} [(1 - v_d - w_d) L_{id0}]^{(1-b_{id}-c_{id})}\} \quad (2.24)$$

By imposing a Dirichlet prior $D(2, 2, 2)$ on $(v_d, w_d, 1 - v_d - w_d)$, we obtain the posterior distribution of the unknown parameters and missing indicators:

$$Pr(\boldsymbol{B}^d, \boldsymbol{C}^d, v_d, w_d | \boldsymbol{X}^d, \boldsymbol{N}^d) \quad (2.25)$$

$$\propto \prod_{i=1}^{I} \{[v_d L_{id1}]^{b_{id}} [w_d L_{id2}]^{c_{id}} [(1 - v_d - w_d) L_{id0}]^{(1-b_{id}-c_{id})}\} v_d w_d (1 - v_d - w_d)$$

An EM algorithm can be similarly derived as in iASeq to estimate the parameters v_d and w_d by searching for the posterior mode of $Pr(v_d, w_d | \boldsymbol{X}^d, \boldsymbol{N}^d)$.

In the E-step, we compute the Q-function $Q(v_d, w_d | \hat{v}_d^{old}, \hat{w}_d^{old})$. Since

$$\ln Pr(\boldsymbol{B}^d, \boldsymbol{C}^d, v_d, w_d | \boldsymbol{X}^d, \boldsymbol{N}^d) = \sum_{i=1}^{I} \{b_{id} [\ln v_d + \ln L_{id1}] + c_{id} [\ln w_d + \ln L_{id2}]$$

$$+ \quad (1 - b_{id} - c_{id}) [\ln(1 - v_d - w_d) + \ln L_{id0}]\} + \ln v_d$$

$$+ \quad \ln w_d + \ln(1 - v_d - w_d) + constant \quad (2.26)$$

We have

$$Q(v_d, w_d | \hat{v}_d^{old}, \hat{w}_d^{old}) = E_{old}[\ln Pr(\boldsymbol{B}^d, \boldsymbol{C}^d, v_d, w_d | \boldsymbol{X}^d, \boldsymbol{N}^d)] \quad (2.27)$$

$$= \{\sum_{i=1}^{I} E_{old}(b_{id}) + 1\} \ln v_d + \{\sum_{i=1}^{I} E_{old}(c_{id}) + 1\} \ln w_d$$

$$+ \quad \{\sum_{i=1}^{I} E_{old}(1 - b_{id} - c_{id}) + 1\} \ln(1 - v_d - w_d) + constant$$

In the M-step, we find v_d and w_d that maximize $Q(v_d, w_d|\hat{v}_d^{old}, \hat{w}_d^{old})$. By solving

$$\frac{\partial Q(v_d, w_d|\hat{v}_d^{old}, \hat{w}_d^{old})}{\partial v_d} = 0 \tag{2.28}$$

$$\frac{\partial Q(v_d, w_d|\hat{v}_d^{old}, \hat{w}_d^{old})}{\partial w_d} = 0 \tag{2.29}$$

We obtain

$$\hat{v}_d^{new} = \frac{\sum_{i=1}^{I} Pr_{old}(b_{id} = 1) + 1}{I + 3} \tag{2.30}$$

$$\hat{w}_d^{new} = \frac{\sum_{i=1}^{I} Pr_{old}(c_{id} = 1) + 1}{I + 3} \tag{2.31}$$

Here

$$\begin{aligned} Pr_{old}(b_{id} = 1) &= Pr(b_{id} = 1|\boldsymbol{X}_{id}, \boldsymbol{N}_{id}, \hat{v}_d^{old}, \hat{w}_d^{old}) \\ &= \frac{Pr(b_{id} = 1, \boldsymbol{X}_{id}|\boldsymbol{N}_{id}, \hat{v}_d^{old}, \hat{w}_d^{old})}{Pr(\boldsymbol{X}_{id}|\boldsymbol{N}_{id}, \hat{v}_d^{old}, \hat{w}_d^{old})} \\ &= \frac{\hat{v}_d^{old} L_{id1}}{\hat{v}_d^{old} L_{id1} + \hat{w}_d^{old} L_{id2} + (1 - \hat{v}_d^{old} - \hat{w}_d^{old}) L_{id0}} \end{aligned} \tag{2.32}$$

$$\begin{aligned} Pr_{old}(c_{id} = 1) &= Pr(c_{id} = 1|\boldsymbol{X}_{id}, \boldsymbol{N}_{id}, \hat{v}_d^{old}, \hat{w}_d^{old}) \\ &= \frac{Pr(c_{id} = 1, \boldsymbol{X}_{id}|\boldsymbol{N}_{id}, \hat{v}_d^{old}, \hat{w}_d^{old})}{Pr(\boldsymbol{X}_{id}|\boldsymbol{N}_{id}, \hat{v}_d^{old}, \hat{w}_d^{old})} \\ &= \frac{\hat{w}_d^{old} L_{id2}}{\hat{v}_d^{old} L_{id1} + \hat{w}_d^{old} L_{id2} + (1 - \hat{v}_d^{old} - \hat{w}_d^{old}) L_{id0}} \end{aligned} \tag{2.33}$$

Using the posterior mode, one can similarly compute $Pr(b_{id}, c_{id}|\boldsymbol{X}^d, \boldsymbol{N}^d, v_d, w_d)$ and $\tilde{P}_{id}$ to detect and rank AS SNPs.

Chapter 3

Global Mapping of Transcription Factor Binding Sites by Sequencing Chromatin Surrogates

[1]

3.1 Introduction

One major goal of functional genomics is to comprehensively characterize the regulatory circuitry behind coordinated spatial and temporal gene activities. In order to achieve this goal, a critical step is to monitor downstream regulatory programs of all transcription factors (TFs). With the capability of mapping genome-wide transcription factor binding sites (TFBSs), chromatin immunoprecipitation coupled with high-throughput sequencing (ChIP-seq) (Barski *and others* (2007), Johnson *and others* (2007),Mikkelsen *and others* (2007),Robertson

[1] A modified version of this chapter has been published: **Wei YY**, Wu G, Ji HK (2013) Global Mapping of Transcription Factor Binding Sites by Sequencing Chromatin States: A Perspective on Experimental Design, Data Analysis, and Open Problems. *Statistics in Biosciences*.5: 156-178. doi: 10.1007/s12561-012-9066-5

and others (2007)) or tiling array hybridization (ChIP-chip) (Cawley *and others* (2004),Ren *and others* (2000)) have become standard approaches for studying gene regulation. Both technologies are now being widely used by investigators world-wide as well as consortium projects such as the ENCODE (Consortium (2007)) , modENCODE (Celniker *and others* (2009)) and Epigenome Roadmap (Bernstein *and others* (2010)) to map functional cis-regulatory elements. Although ChIPx (i.e., ChIP-seq and ChIP-chip) offers the power to survey binding sites genome-wide, a number of limitations make this technology low-throughput with respect to surveying a large number of TFs. First, successful application of ChIPx requires high-quality antibodies specifically recognizing the TF of interest. Unfortunately for many TFs, ChIP-quality antibodies are not available. Second, each individual ChIPx experiment can only analyze one TF in one cell type. To analyze many TFs, one has to test to ensure sensitive antibodies, optimize the protocol, and perform experiments repeatedly, which is both costly and labor intensive. For these reasons, currently it is unrealistic to use ChIPx to directly monitor genome-wide TFBSs for all TFs. Therefore, the development of innovative methods and technologies that allow high-throughput mapping of *in vivo* TFBSs of all TFs is both important and urgently needed.

Computational predictions based on mapping DNA sequence motifs to genome sequences offer an alternative approach to analyze TFBSs (Jensen *and others* (2004),Ji and Wong (2006),Stormo (2000),Tompa *and others* (2005)). Predictions based purely on DNA sequences, however, are known to have low specificity. In addition, *in vivo* TF binding is highly context-dependent. Without further information, computationally determined motif sites cannot describe the highly dynamic TF binding activities in different cell types and conditions.

Recent technological advances have made it possible to analyze genome-wide chromatin profiles (Barski *and others* (2007),Boyle *and others* (2008),Ernst and Kellis (2010),Ernst *and others* (2011),He *and others* (2010),Heintzman *and others* (2007),Hon *and others* (2009),Mikkelsen *and others* (2007),Song *and others* (2011)). For example, a variety of histone modifications (HMs) (e.g., H3K27ac, H3K4me1, H3K4me2, H3K4me3) can now be measured by ChIP-seq (Barski *and others* (2007),Ernst *and others* (2011),He *and others* (2010),Heintzman *and others* (2007)). Additionally, DNase-seq and FAIRE-seq have been developed for mapping DNase I hypersensitivity (DHS) and open chromatin (Boyle *and others* (2008),Gaulton *and others* (2010),Song *and others* (2011)). Analyses of data generated by these technologies show that many chromatin features correlate with TF binding (Figure 3.1). As a result, HM ChIP-seq, DNase-seq and FAIRE-seq can serve as a surrogate in place of TF ChIPx for mapping TFBSs (Boyle *and others* (2011),Cheng *and others* (2011),Pique-Regi *and others* (2011),Whitington *and others* (2009),Won *and others* (2010)). Coupling analyses of these surrogate data with computationally determined motif sites allows one to predict *in vivo* TF binding. This predictive approach has several unique advantages. First, the requirement for antibodies is easier to satisfy, because ChIP-quality antibodies are available for many HMs, and DNase-seq and FAIRE-seq do not require TF-specific antibodies. Second, measurements offered by HM ChIP-seq, DNase-seq and FAIRE-seq are context-dependent, hence TFBS predictions based on these data are specific to the biological contexts in question (Figure 3.1a). Third, this approach makes analysis

of TFs high-throughput. Among the approximately 1400 human TFs, sequence-specific DNA binding motifs have been determined for about 500 TFs by high-throughput means such as protein microarrays (Hu *and others* (2009),Robasky and Bulyk (2011),Sandelin *and others* (2004),Wingender *and others* (1996),Xie *and others* (2010)). Thus, the predictive approach allows one to infer TFBSs for hundreds of different TFs simultaneously in one assay. For these reasons, predicting TFBSs based on sequencing chromatin surrogates offers a promising new solution to the global analysis of gene regulation.

As a new approach, many open issues remain to be addressed. Examples include what principles to follow when designing experiments, which guidelines to use to choose informative surrogate data types, and what methods will analyze the data optimally. For statisticians and computational scientists, it is of interest to know what are the crucial analytical challenges and opportunities for developing new methodology. The purpose of this chapter is two-fold. First, through an analysis of the ENCODE data, we will demonstrate some basic characteristics of this approach which will shed light on several important experimental design and data analysis issues. Second, we will use the data to introduce several analytical challenges to investigators who are interested in exploring this new field.

3.2 Key Questions

Our analyses were designed to shed light on the following questions.

(1) Overall prediction performance: what is the overall accuracy and sensitivity for predicting TFBSs by using chromatin surrogates?

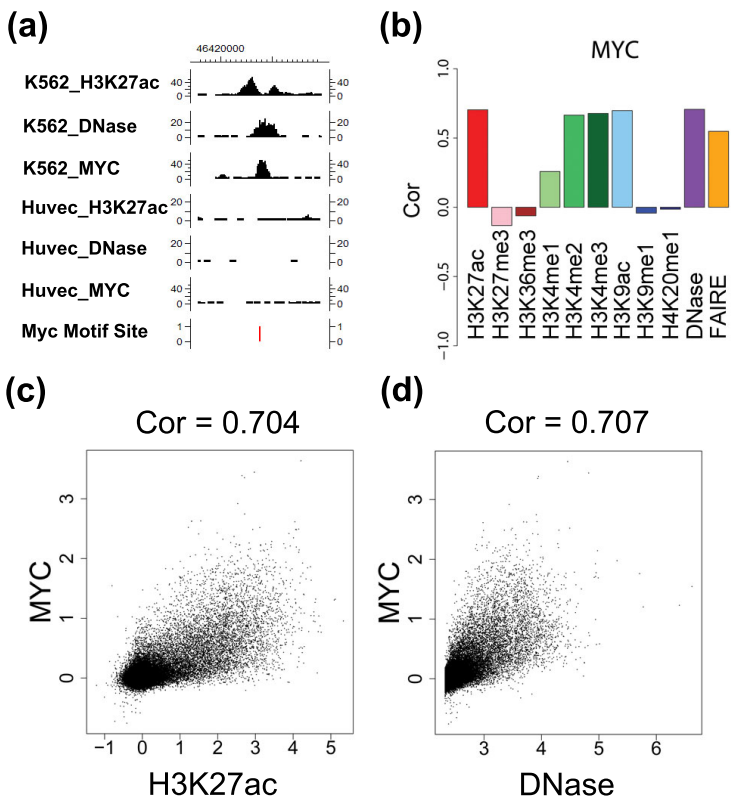


Figure 3.1: Correlation between TF binding and chromatin features. (a) Histone modification H3K27ac ChIP-seq and DNase-seq profiles at a MYC motif site are shown along with ChIP-seq data for TF MYC in two cell lines K562 and Huvec. The profiles shown are read counts in 100bp sliding windows at 25bp resolution. MYC binding can be inferred from the H3K27ac and DNase data. In this example, the motif site is bound by MYC in the K562 cell line but not in the Huvec cell line. The cell-type specific binding is correlated with the cell-type specific H3K27ac and DNase I hypersensitivity. In the K562_H3K27ac track, MYC binding leads to nucleosome displacement. As a result, the binding site is surrounded by two nucleosomes carrying the H3K27ac signals (He *and others* (2010)), causing the dip shape in the signal curve. In the K562_DNase track, the peak reflects the chromatin accessibility due to TF binding. (b) Pearson correlation coefficients between different types of chromatin data and the actual MYC ChIP-seq binding intensities in K562 across all MYC motif sites. Certain chromatin features (e.g., H3K27ac, H3K4me2, H3K4me3, H3K9ac, DNase and FAIRE) clearly correlate with MYC binding. (c) A scatter plot demonstrating the correlation between H3K27ac and MYC ChIP-seq binding intensities in K562 across all MYC motif sites. Each dot is a motif site. The binding intensities are normalized and log2 transformed read counts. 'Cor': Pearson correlation coefficient. (d) Correlation between DNase-seq and MYC ChIP-seq binding intensities in K562.

(2) Best surrogate data type: which surrogate data type(s), individually or in combination, can produce the best prediction performance?

(3) Supervised versus unsupervised learning: predictions can be made by two different approaches. In the unsupervised approach, only surrogate chromatin data are collected. The TFBSs are then predicted based on analyzing the surrogate data at the DNA motif sites. In the supervised approach, one collects ChIP-seq data for at least one TF in addition to generating the surrogate chromatin data. One then uses these data to train a model to predict TFBSs based on the surrogate data. The trained model will be applied to predict binding sites of all other TFs. The supervised approach seems to use more information and intuitively should outperform the unsupervised approach. Is this true? Should one use the supervised approach or the unsupervised approach? For the supervised approach, is it possible to eliminate the need for generating the training TF ChIP-seq data by coupling ones' own surrogate data with TF ChIP-seq data from other labs (e.g., existing data in public databases) to train a model, and then apply the model to make predictions?

Answers to these questions have important implications to future studies. They may help one to design future experiments to better allocate available resources. Answers to (1) and (3) may help statisticians and computational biologists to decide where to invest their efforts for developing the most needed analytical tools.

Table 3.1: Summary of surrogate chromatin data

Lab	Data type	K562	Gm12878	Description
Broad	H3K27ac	√	√	acetylation of H3 Lysine 27
Broad	H3K27me3	√	√	trimethylation of H3 Lysine 27
Broad	H3K36me3	√	√	trimethylation of H3 Lysine 36
Broad	H3K4me1	√	√	monomethylation of H3 Lysine 4
Broad	H3K4me2	√	√	dimethylation of H3 Lysine 4
Broad	H3K4me3	√	√	trimethylation of H3 Lysine 4
Broad	H3K9ac	√	√	acetylation of H3 Lysine 9
Broad	H3K9me1	√		monomethylation of H3 Lysine 9
Broad	H4K20me1	√	√	monomethylation of H4 Lysine 20
Duke	DNase (DHS)	√	√	DNase I hypersensitivity
UNC	FAIRE	√	√	nucleosome-depleted regions

Available HM ChIP-seq, DNase-seq and FAIRE-seq data in the ENCODE consortium for cell lines K562 and Gm12878 were analyzed. Each row is a dataset containing 1-3 replicate samples.

Table 3.2: Summary of TF ChIP-seq data

Lab	TF	TF type	K562	Gm12878
HudsonAlpha (HA)	EGR1	activator	√	√
HudsonAlpha (HA)	GABP	activator	√	√
HudsonAlpha (HA)	SRF	activator	√	√
HudsonAlpha (HA)	USF	activator	√	√
HudsonAlpha (HA)	NRSF	repressor	√	√
Yale	E2F4	activator	√	
Yale	E2F6	activator	√	
UTA	MYC	activator	√	√
UTA	CTCF	insulator	√	√

We analyzed 9 different TFs from 3 different labs in the ENCODE consortium for cell lines K562 and Gm12878. Each row is a dataset containing 1-3 replicate samples.

3.3 Data

To answer these questions, we have analyzed 11 different surrogate data types (Table 3.1), and constructed various models to predict binding sites of 9 different TFs (Table 3.2). These data were generated by 6 different labs in the ENCODE consortium and involved two different cell lines for which rich data are available: K562 and Gm12878. The data analyzed represent those available to us from ENCODE at the time the study was initiated, and only TFs with known DNA binding motifs were considered.

Nine of the eleven surrogates are histone modifications. Among them, H3K27ac, H3K4me1, H3K4me2, H3K4me3 and H3K9ac correlate with active promoters or enhancers, whereas H3K27me3 is a mark for gene repression (Barski *and others* (2007),Heintzman *and others* (2007),Wang *and others* (2008)). H3K36me3 is enriched in the gene body of actively transcribed genes (Barski *and others* (2007)). H4K20me1 and H3K9me1 have been previously linked to repressive chromatin (Sims *and others* (2006)), but recent studies also found correlation between these two HMs with active transcription (Barski *and others* (2007)). As the current understanding of HM functions is incomplete, it is possible that some HMs individually or in combination have unknown new functions. Besides these nine HMs, our surrogates also included DNase I hypersensitivity measured by DNase-seq, which is a signature for DNA binding by trans-acting factors in place of canonical nucleosomes, and open chromatin measured by FAIRE-seq, which is a mark for nucleosome-depleted regions. Among the nine TFs considered, NRSF is a repressor that inactivates neuronal gene transcription in non-neuronal cells. CTCF is a protein that binds to insulators

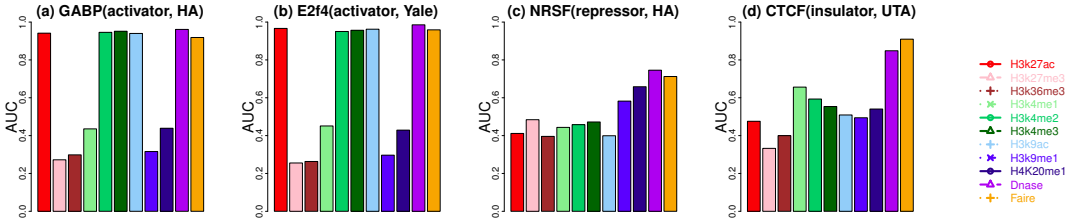


Figure 3.2: Area under the receiver operating characteristic curves for predicting TFBSs in K562 based on single surrogate. (a) GABP; (b) E2F4; (c) NRSF; (d) CTCF. Results for other TFs are in Supplemental Figure 3.9.

and may also serve as a transcriptional repressor. The other TFs all have roles in activating gene expression.

Since analyses of the two cell lines have reached essentially the same conclusions, this chapter will use K562 as an example to demonstrate the main results.

3.4 Which surrogates are informative predictors individually?

We first investigated which surrogates (i.e., DHS, FAIRE, and various HMs) are most informative for predicting TFBSs. We downloaded aligned ChIP-seq, DNase-seq and FAIRE-seq reads (human genome build 36/hg18) from the ENCODE website (http://genome.ucsc.edu/ENCODE/). Consider J surrogate datasets. To predict binding sites of a TF, the DNA binding motif of the TF was mapped to human genome by CisGenome (Ji *and others* (2008)) using the default parameters. For each motif site s and surrogate dataset j, the normalized read count x_{sj} in a 500bp flanking window centered at the motif site was obtained to represent the surrogate signal intensity (see Supplemental Method

Section 3.10.1). For each motif site, the actual TF binding intensity y_s was also computed using the ENCODE ChIP-seq data for the TF (see Supplemental Method Section 3.10.1). We used the surrogate signal intensities x_{sj} to rank order motif sites. Top ranked sites were predicted to be bound by the TF. We varied the cutoff and evaluated the predictions using the actual TF binding intensities y_s. For evaluation, motif sites with $y_s > 1$ were treated as true binding sites. Intuitively, $y_s > 1$ means the log2 ratio between the normalized ChIP and Input control read counts is bigger than one (or 2 fold enrichment). Using these as gold standard, we obtained a curve for each surrogate data type that describes the positive predictive values (PPV, i.e., the percentage of true positives among top predictions) at varying cutoffs. We also computed the area under the receiver operating characteristic curve (AUC) for each surrogate and compared different surrogates in terms of AUC.

When each surrogate was used individually as the predictor, DHS performed the best in most situations based on the global PPV curves and AUC (Figure 3.2, Figures 3.9, 3.10, Supplemental Table 3.4). Only for CTCF, FAIRE outperformed DHS. Several HMs, including H3K27ac, H3K4me2, H3K4me3 and H3K9ac also performed well in most but not all datasets. In general, the predictive power of HMs depends on the TF. H3K27ac, H3K4me2, H3K4me3 and H3K9ac predicted TFBSs well for EGR1, GABP, SRF, USF, E2F4, E2F6 and MYC (Figure 3.2, Figures 3.9, 3.10). However, for NRSF, H4K20me1 and H3K9me1 performed better than the other HMs. For CTCF, H3K4me1 performed the best among the tested HMs. These results are consistent with the patterns we saw in supplemental Figure 3.11 where Pearson correlation between the predictors x_{sj} and the actual binding y_s are compared.

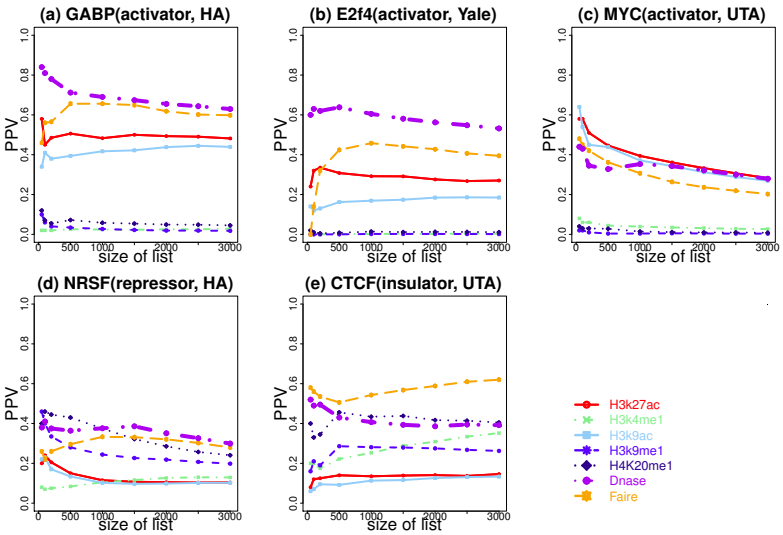


Figure 3.3: Positive predictive value curves for predicting TFBSs in K562 based on single surrogate. The x axis is the number of the top ranked motif sites. The y axis is the positive predictive value. (a) GABP; (b) E2F4; (c) MYC; (d) NRSF; (e) CTCF. Only representative surrogates and TFs are shown. See Supplemental Figure 3.12 for comprehensive results.

The analysis above compares the prediction performance globally based on all motif sites. We also examined the PPVs for the top ranked predictions which are most likely to be picked up for follow-up experimental studies (Figure 3.3, Supplemental Figure 3.12). While DHS still performed the best in most datasets, we found a few cases where other surrogates predicted TFBSs better than DHS among the top predictions. For example, for MYC (i.e., c-Myc), H3K27ac and H3K9ac performed better. For NRSF, H4K20me1 outperformed DHS for top 800 motif sites. For CTCF, FAIRE and H4K20me1 performed the best.

In summary, we found DNase I hypersensitivity to be the most consistently accurate predictor for TFBSs, whereas the predictive power of HMs depend on the TF-of-interest.

3.5 How do surrogates perform jointly?

Next, we asked whether using multiple surrogates together can improve prediction. Let $\mathbf{x}_s = (x_{s1}, \ldots, x_{sJ})^T$ be the vector that contains all surrogate intensities at motif site s, we constructed models that use $\mathbf{x}_s$ to predict y_s.

Before constructing any model, we first investigated whether binding sites of each TF fall into different classes exhibiting different chromatin patterns. For each TF, we clustered its bound motif sites (i.e., sites for which $y_s > 1$) based on $\mathbf{x}_s$. It turns out that for the same TF, most motif sites bound by the TF share a similar pattern in $\mathbf{x}_s$ (Supplemental Figure 3.13a). Next, for each TF, we asked whether its motif sites have regionalized patterns of $\mathbf{x}_s$. In this regard, we clustered all motif sites of the TF based on $\mathbf{x}_s$ (Supplemental Figure 3.13b).

We then examined whether the distribution of the motif sites in each cluster are concentrated on certain genomic regions. However, we did not observe such a phenomenon (Supplemental Figure 3.13c,d). Furthermore, we checked the correlation between y_s and each surrogate in each chromosome. We found that the correlation patterns in different chromosomes were similar (Supplemental Figure 3.14). Based on these explorations and due to considerations of computational efficiency, we decided not to construct regionalized prediction models with varying forms or parameters for different genomic regions. Instead, for each TF, we constructed models whose form and parameters remain the same across the genome.

Eight prediction methods were tested, including one unsupervised approach and seven supervised learning methods (Table 3.3; Supplemental Method Section 3.10.2). The methods employed include both linear and non-linear models. In the unsupervised approach, the first principal component (PC1) of $\mathbf{x}_s$ was computed using all motif sites. The motif sites were then rank ordered based on PC1. Since the direction of unique PCs can only be determined up to a positive or negative sign, motif sites were sorted based on PC1 and $-$PC1 separately. Both rankings were tested, and the one with better prediction performance was reported. In the supervised approach, the prediction model was trained using ChIP-seq data for one TF and then applied to other TFs to make predictions. The training methods include linear regression (L) using all surrogates (AS) as predictors, principal component regression (PCR) using the first two principal components of $\mathbf{x}_s$s, classification and regression tree (CART), random forest (RF), and support vector regression with linear (SVR_L) and Gaussian

(SVR_G) kernels. For the linear regression, we also enumerated all combinations of multiple surrogates (MS), identified the best subset of surrogates using the Cp statistic, and then obtained the linear model based on the best surrogate set (MS_L). For the non-linear models, we did not analyze different surrogate combinations since it would require a tremendous amount of computation time.

Interestingly, we found that even though the best methods based on multiple or all surrogates improved predictions for some TFs compared to analyses based on DHS alone, none of these methods consistently outperformed DHS for all test TFs (Figure 3.4, Supplemental Figure 3.15). For instance, RF and SVR_G trained using EGR1 ChIP-seq data and all surrogates outperformed DHS for E2F4 and E2F6, but performed worse than DHS for NRSF and CTCF. A recent study based on an unsupervised approach has reported that adding HM ChIP-seq did not improve the power for predicting TFBSs using DHS Pique-Regi *and others* (2011). Our results are consistent with that observation. Different from Pique-Regi *and others* (2011), however, our analyses here also examined a number of supervised learning approaches. The analyses show that integrating multiple surrogates by these supervised approaches did not improve predictions consistently.

3.6 Supervised versus unsupervised learning

Ranking motif sites based on DHS alone is essentially an unsupervised approach. Figure 3.4 and Supplemental Figure 3.15 show that when DHS is included in the predictors, the gain of using all surrogates and supervised learning over this simple unsupervised method is not universally guaranteed. We speculate

Table 3.3: Methods used for prediction

Abbreviation	Category	Description
SS	unsupervised	Single surrogate
AS_PC1	unsupervised	All surrogates, the first principal component
MS_L	supervised	The best subset of surrogates, linear regression
AS_L	supervised	All surrogates, linear regression
AS_PCR	supervised	All surrogates, principal component regression
AS_CART	supervised	All surrogates, classification and regression tree
AS_RF	supervised	All surrogates, random forest
AS_SVR_L	supervised	All surrogates, linear kernel support vector regression
AS_SVR_G	supervised	All surrogates, Gaussian kernel support vector regression

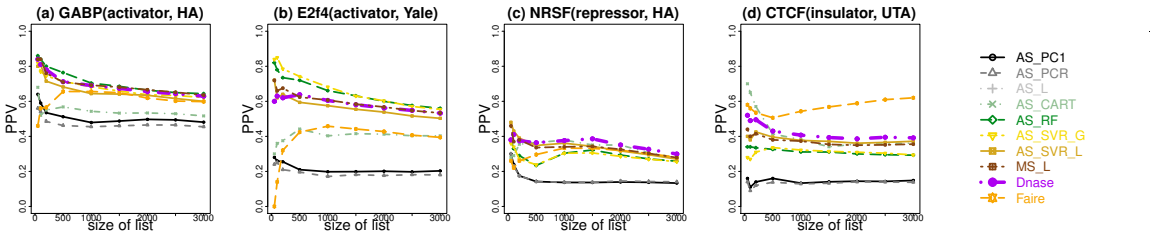


Figure 3.4: Positive predictive value curves for predicting TFBSs in K562 based on models trained using EGR1. (a) Prediction for GABP; (b) prediction for E2F4;(c) prediction for NRSF; (d) prediction for CTCF. Predition results for other TFs are in Supplemental Figure 3.15. Using other TFs to train the model produced similar results (data not shown).

that part of the reason is that the supervised approach trains models using one TF and applies it to another TF. Due to intrinsic differences between TFs, the model may be optimized for the training TF, but may not be optimal for the test TF. To examine whether this is the case, we compared two prediction scenarios. In scenario 1, a prediction model was trained using surrogate and ChIP-seq data for TF A in a subset of chromosomes (chromosomes 1-16). The model was then applied to predict binding sites of TF A in other chromosomes (chromosomes 17-22 and X). The prediction performance was evaluated using ChIP-seq data for TF A in the test chromosomes (Figures 3.16,3.17). In scenario 2, a prediction model was trained using ChIP-seq data for TF A, and then applied to predict binding sites of TF B. The prediction performance was evaluated using ChIP-seq for TF B (Figure 3.4, Supplemental Figure 3.15). In scenario 1, the prediction model trained using AS_L, MS_L, AS_CART, AS_RF,AS_SVR_L and AS_SVR_G all performed better than using DHS alone, and supervised learning on average performed better than unsupervised approaches. In contrast, in scenario 2, supervised prediction based on all surrogates did not consistently outperform DHS (e.g., compare NRSF and CTCF in Figure 3.4 and Supplemental Figure 3.15). This demonstrates that supervised learning was able to improve the prediction for the training TF, but cannot guarantee an improvement when the trained model is applied to another TF.

An investigator may decide to collect HM ChIP-seq data without DNase-seq for other considerations (e.g., if one is primarily interested in studying HMs and the budget does not allow additional DNase-seq). With only HMs as predictors, we observed similar phenomena, that is, the supervised approach did

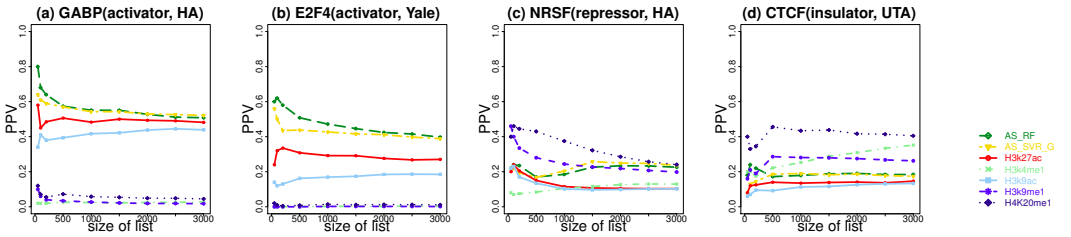


Figure 3.5: Positive predictive value curves for predicting TFBSs in K562 based on models trained on EGR1 using only HM ChIP-seq data. (a) Prediction for GABP; (b) prediction for E2F4; (c) prediction for NRSF; (d) prediction for CTCF. Only representative methods and TFs are shown. See Supplemental Figure 3.18 for comprehensive results.

not consistently outperform the unsupervised approach (Figure 3.5, Supplemental Figure 3.18). However, the difference in prediction accuracy between the best supervised method and the best unsupervised method became much bigger. For instance, RF and SVR_G trained using EGR1 ChIP-seq data now performed substantially better than the best unsupervised ranking based on H3K27ac for predicting GABP, SRF, USF, E2F4 and E2F6. For predicting NRSF and CTCF, RF and SVR_G trained by EGR1 performed substantially worse than unsupervised rankings based on H4K20me1.

To further shed light on when the supervised methods can outperform the unsupervised methods, we clustered the nine TFs based on the eleven surrogates. For each TF, the TF's ChIP-seq data was used to group motif sites into two classes: bound ($y_s > 1$) and not bound ($y_s \leq 1$). The enrichment of the surrogate signals in the bound class compared to the non-bound class were used to cluster TFs (Supplemental Method Section 3.10.3). The TFs fall into two distinct classes (Figure 3.6). The repressor and insulator proteins NRSF and CTCF were clearly separated from the other TFs which can serve as transcriptional activators. H3K9ac, H3K4me2, H3K4me3 and H3K27ac were clearly

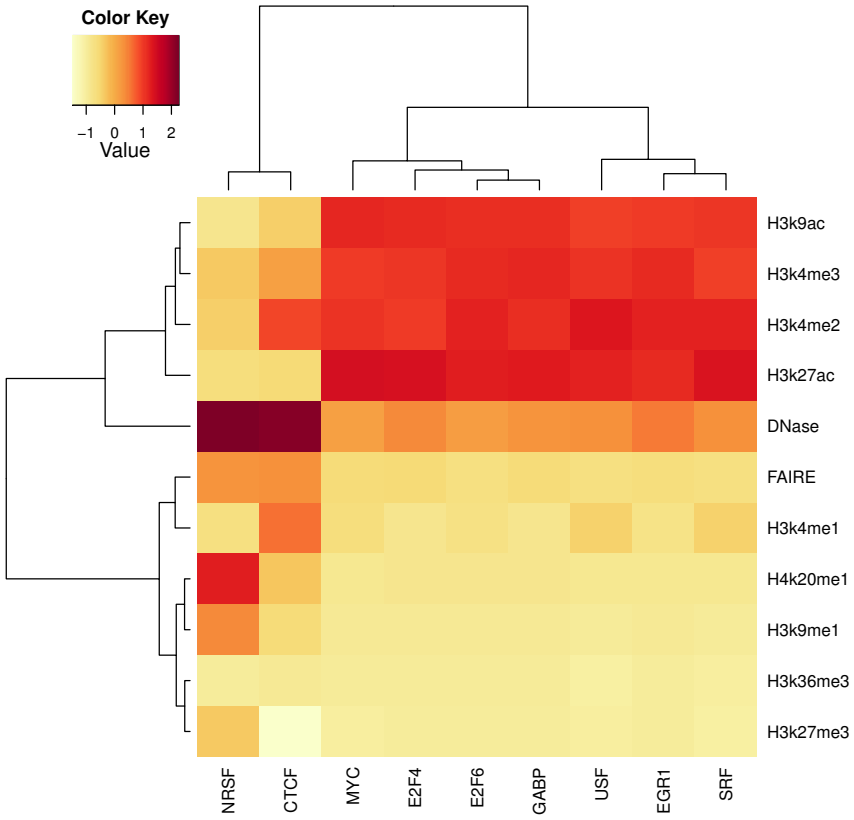


Figure 3.6: Hierarchical clustering of TFs and surrogates based on the enrichment of the surrogate signals in the bound motif sites compared to the signals in the non-bound motif sites.

enriched in the bound motif sites for those activators but not for NRSF and CTCF.

A careful examination of Figures 3.4, 3.5 and 3.6 reveals that whether or not the supervised approach improves the unsupervised approach depends on whether or not the training and test TFs are of similar types. For instance, supervised models trained using EGR1 predicted GABP and E2F4 well as they are in the same class (also see SRF, USF, E2F6 in Supplemental Figures 3.15, 3.18), but it did not perform so well for NRSF and CTCF. Interestingly, when we attempted to predict CTCF using models trained by NRSF and only using HMs as predictors, or predict NRSF using models trained by CTCF, supervised learning improved the prediction performance a lot in both cases, compared to predictions based on DHS and FAIRE (Supplemental Figure 3.19).

Figure 3.6 shows that DHS is enriched in bound motif sites for all TFs, consistent with the observation that it is the most consistently accurate predictor for all analyzed TFs. This also explains why we observed bigger differences between the best supervised prediction and the best single surrogate based ranking in Figure 3.5 after excluding DHS from the predictors, compared with Figure 3.15 in which DHS was included as a predictor.

Together, our results suggest that the intrinsic differences among TFs are an important reason why supervised learning based on all surrogates does not guarantee a gain over the unsupervised ranking based on DHS alone. Therefore, when developing future supervised learning methods for predicting TFBSs using surrogate data, it is important to consider the heterogeneity of the TFs. One may need to group TFs into different categories (e.g., activators, repressors, etc.) so that TFs within each category have similar characteristics. One could

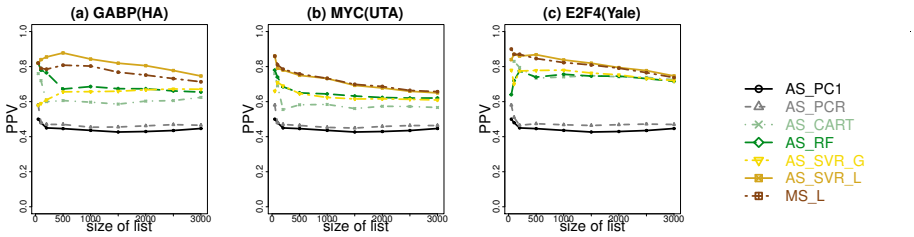


Figure 3.7: Positive predictive value curves for prediction on EGR1 by models trained using ChIP-seq data from different labs. (a) Models trained using GABP (HudsonAlpha); (b) models trained using MYC (UTA); (c) models trained using E2F4 (Yale). Only representative methods and training TFs are shown. See Supplemental Figure 3.20 for comprehensive results.

then train a model for each category in order to take the full advantage of the supervised learning, which may eventually lead to improved prediction accuracy.

3.7 Cross-lab prediction

Both unsupervised and supervised approaches require one to generate surrogate data for the cell type of interest. For the supervised approach, one also needs to collect training TF ChIP-seq data for different TF classes. If TF ChIP-seq data for the same cell type are available in public databases, a natural question is whether one can couple these public TF ChIP-seq data (typically generated by a different lab) with his/her own surrogate data to train the prediction model, thereby eliminating the needs for generating ones' own TF ChIP-seq. In our analyses, EGR1, GABP and NRSF came from one lab. E2F4 and E2F6 came from another lab. Figure 3.5 and Supplemental Figures 3.15 and 3.18 show that using the random forest and support vector regression trained by EGR1, one achieved comparable or better prediction performance for predicting E2F4 and E2F6 as compared to predicting GABP and NRSF. Futhermore, when

we attempted to predict binding sites for EGR1 by models trained using data from different labs, including USF (HudsonAlpha), SRF (HudsonAlpha), GABP (HudsonAlpha), MYC (UTA), E2F4 (Yale) and E2F6 (Yale), models trained by data from different labs performed similarly (Figure 3.7, Supplemental Figure 3.7). Collectively, these suggest that cross-lab training is feasible, and as ChIP-seq data in public domains continue to grow rapidly, the need to generate one's own training TF ChIP-seq data may be partially eliminated in future.

3.8 Sensitivity

Since DHS has robustly performed among the best, our subsequent analyses were focused on DHS. To evaluate sensitivity, we analyzed ChIP-seq data for each TF using CisGenomev2 algorithm Ji *and others* (2008) and called peaks using 1% FDR as the cutoff. Peaks that contained the motif of the corresponding TF were used as gold standard. In parallel, we ranked motif sites by DHS, used the top ranked sites to predict TFBSs, and estimated the FDR among the predicted sites by comparing their DHS signal distribution to the DHS signal distribution at randomly chosen genomic loci (Supplemental Method Section 3.10.4). The receiver operating characteristics (ROC) in Figure 3.8 show that at the 25% FDR level, the predictions were able to recover 50-90% of the ChIP-seq peaks containing the motifs. SRF is an exception. For SRF, the data were noisy and the lowest prediction FDR we can obtain was 58%. In practice, this means that none of the predicted SRF binding sites can be claimed as statistically significant. It should be noted that the ROC will change if peaks and motif sites are called using different cutoffs, or if different motifs are used to

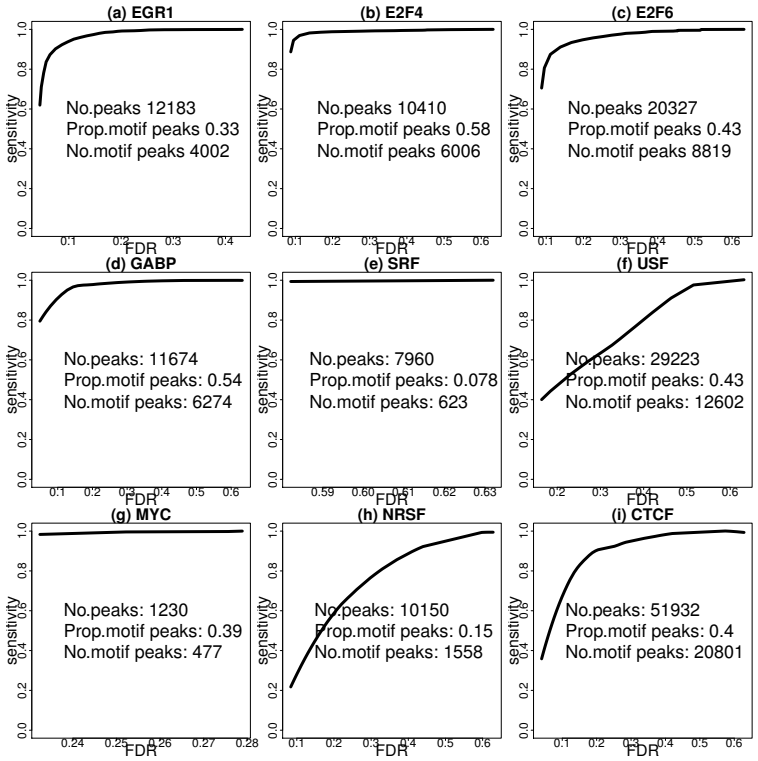


Figure 3.8: Sensitivity against FDR plot. The x axis is the FDR of DHS at candidate sites. The y axis is the percentage of gold standard motif peaks discovered. "No. peaks" is the total number of gold standard peaks called by CisGenome at FDR 1%. "Prop. motif peaks" is the proportion of gold standard peaks containing motif sites, called as motif peaks. "No. motif peaks" is the total number of motif peaks. (a) EGR1; (b) E2F4; (c) E2F6; (d) GABP; (e) SRF; (f) USF; (g) MYC; (h) NRSF; (i) CTCF.

make predictions. Therefore Figure 3.8 should be interpreted as a rough picture of the sensitivity of the prediction approach.

3.9 Conclusions and Discussion

Through the analyses of ENCODE data, we have verified that TFBSs can be predicted using chromatin surrogates with reasonable accuracy and sensitivity. This approach offers an attractive alternative to ChIP-seq and ChIP-chip as it allows one to survey many TFs together in one assay. Our analyses show that DNase I hypersensitivity profiled by DNase-seq consistently performed among the best as a predictor, whereas the performance of using a specific HM as the predictor may depend on TFs. Thus if the available resources only allow one to sequence one surrogate data type, one may consider DNase-seq.

When TF ChIP-seq data are available in addition to multiple types of surrogate data, one may choose to use these data to train a prediction model and then apply the model to predict binding sites for other TFs. Our analyses show that the improvement the supervised learning can provide over the unsupervised method is not significant when DHS is included as a predictor. When only HMs are used as predictors, the gain of supervised learning over the unsupervised approach depends on whether the training and test TFs belong to similar classes. If these two TFs have distinct properties (e.g., one is activator, whereas the other one is repressor), then the supervised learning approach may not improve over the unsupervised methods. Therefore, the advantage of supervised learning for HMs only is also not universally guaranteed. Investigators developing such methods may need to develop different prediction models for different TF

classes.

A recent study of 203 yeast TFs have shown that yeast TFs fall into two categories: histone-sensitive TFs and histone-insensitive TFs (Cheng *and others* (2011)). The target genes of histone-sensitive TFs have relatively higher HM signals and are easier to be predicted using HMs. The histone-sensitive TFs are also more likely to interact with chromatin modifiers and are enriched in the upper layers of regulatory hierarchy. Whether these phenomena hold true in human is an interesting problem. Our results suggest that human TFs are very likely to fall into different categories as well. On the other hand, since we only have data from a limited number of human TFs, including only one repressor NRSF and one insulator binding protein CTCF, and since the knowledge of the TF network in human is still incomplete as most human TFs do not have ChIP data, we were not able to meaningfully examine the statistical association between different TF categories and their ability to interact with histone modifiers, or their positions in the regulatory hierarchy. These issues are worthwhile to be re-examined in the future as sufficient data become available.

Our analyses did not use the curve shape information in the surrogate chromatin data. Several studies show that DNase-seq and some HM ChIP-seq profiles have characteristic footprints surrounding TFBSs. For instance, many of these surrogates have a characteristic dip structure around the *bona fide* binding sites (Figure 3.1a). Incorporating the shape information into the prediction model may further increase the prediction power (Boyle *and others* (2011),Pique-Regi *and others* (2011)).

Supervised learning requires training TF ChIP-seq data. As more ChIPx data become available in public domains, it may be possible to couple these

public data with one's own surrogate data to train the prediction models. This may allow one to reduce the experimental cost.

The two applications of public ChIPx data highlights the value of compiling such data. Importantly, methods for assessing data quality are needed to ensure that bad quality datasets will be excluded to avoid misleading supervised learning or candidate site identification. Statistical methods that can integrate the quality measures into the prediction pipeline may also be needed.

Predictions based on DNase-seq and other surrogate data are complementary to ChIPx. ChIPx are still useful to accurately determine direct binding of a TF of interest. When designing future experiments, one may couple DNase-seq for surveying many TFs with relatively low accuracy and sensitivity with ChIP-seq for analyzing selected TFs with high accuracy and sensitivity. With DNase-seq available, one question that remains to be addressed but not discussed in this paper is whether one can reduce the sequencing depth of the ChIP-seq library but still keep similar sensitivity by integrating DNase-seq data into ChIP-seq analysis. If so, this will allow one to reduce the experimental cost, which is particularly useful if one wishes to analyze many TFs using ChIPx in detail, or analyze the same TF in many different developmental time points or biological conditions. For statisticians, this will create a need for new data integration methods.

The observation that DHS alone predicted TFBSs reasonably well seems to suggest that there is no much room for statisticians to develop new methods. However, this is not true if one realizes that predicting TFBSs is not our final goal. It remains unclear how one should resolve the one-motif-multiple-TF ambiguity. Moreover, only a small fraction of binding sites are functional. How

to identify the small subset of functional binding targets remains a significant challenge. These examples show that research related to predicting TFBSs by sequencing chromatin states is filled with unsolved open problems. Data scientists will find this research to be both challenging and exciting.

3.10 Supplementary Materials

3.10.1 Supplemental Method 1: Data preprocessing

In order to predict TF binding based on chromatin surrogates at motif sites, DNA binding motif of each TF was mapped to human genome using CisGenome (Ji *and others* (2008)) with default parameter settings. For each TF, the corresponding DNA binding motif was obtained from either TRANSFAC (Wingender *and others* (1996)) or publication (Kim *and others* (2007)). Consider a single TF. For each motif site s and surrogate dataset j, a normalized read count x_{sj} was computed to represent the signal intensity of HM, DNase I hypersensitivity or open chromatin in a 500bp flanking window centered at the motif site. For each motif site, the actual TF binding intensity y_s was also computed using the ChIP-seq dataset for the TF. Conceptually, our goal is to predict which motif sites are bound by the TF based on x_{sj}s. The predictions can be evaluated based on y_s.

To compute x_{sj} and y_s, sequence reads in each ChIP-seq sample were extended 150bp to 3' end to approximately reconstruct the original DNA fragments. Based on ENCODE annotations, 150bp reflects the most typical DNA fragment length in these samples. After dividing the genome into 10bp non-overlapping bins, the number of DNA fragments covering each bin was counted.

Let C_{ijk} be the raw count for bin i, dataset j and replicate k. Let N_{jk} be the total fragment count in sample (j,k). Normalize C_{ijk} by $c_{ijk} = C_{ijk} * min_{(j',k')}N_{j'k'}/N_{jk}$, and transform the normalized value to $b_{ijk} = log_2(\Delta + c_{ijk})$. Δ is an offset added to avoid log(0) and unstable estimate of fold changes when c_{ijk} is small. Different values of Δ (1, 5, and 10) were tried, and they produced the same qualitative conclusions. In the paper, we only show results based on $\Delta = 5$ for simplicity. After obtaining b_{ijk}s, replicates were averaged to obtain $a_{ij} = \sum_k b_{ijk}/n_j$. n_j is the number of replicate samples in dataset j. If control samples were available (e.g., Input controls in ChIP-seq experiments), control read counts were subtracted from the ChIP read counts to obtain:

$$a_{ij} = \sum_k b_{ijk}/n_j - \sum_k b_{ij'k}/n_{j'}. \tag{3.1}$$

Here j' indicates the control dataset corresponding to the ChIP dataset j. a_{ij} provides a one number summary for each bin i and dataset j. Next, each motif site was extended 250bp to both ends. The a_{ij}s within the 500bp window were averaged to obtain x_{sj}. y_s was computed similarly.

3.10.2 Supplemental Method 2: Various prediction methods

We compared nine different methods for predicting TFBSs.

Unsupervised methods:

1. Single surrogate (SS): TFBSs are predicted based on each individual surrogate. It was assumed that y_s is a monotone function of x_{sj}, and ranking of motif sites based on x_{sj} determines the ranking of motif sites based on y_s. Top ranked motif sites were predicted to be TFBSs.

2. Principal component of all surrogates (AS_PC1): Let $\mathbf{x}_s = (x_{s1}, \ldots, x_{sJ})^T$ be the vector that contains the intensity values of all surrogates at motif site s. The first principal component (PC1) (Jolliffe (2002)) of all $\mathbf{x}_s$s was computed and motif sites were ranked accordingly. Since the direction of unique PCs can only be determined up to a positive or negative sign, motif sites can be ranked based on either PC1 scores or minus one times PC1. Both rankings were tested, and the ranking that produced better results was reported.

Supervised methods:

1. Best subset of multiple surrogates, linear regression (MS_L): Using a training TF ChIP-seq dataset, the following linear model $y_s = \beta_0 + \beta_{j_1} x_{sj_1} + \cdots + \beta_{j_k} x_{sj_k} + error$ is fit using a subset of surrogates $(j_1, \cdots, j_k)$. All possible combinations of surrogates were enumerated and tested. The exhaustive search finds the best subset of surrogates using the Mallows' Cp as the selection criterion. The linear model based on the best surrogate combination will be used as the final prediction model to predict TFBSs for other TFs (R package leaps).

2. All surrogates, linear regression (AS_L): Using a training TF ChIP-seq dataset, a prediction model $y_s = \beta_0 + \sum_j \beta_j x_{sj} + error$ was fit using all surrogate data types. The trained model was used to predict TFBSs for other TFs.

3. All surrogates, principal component regression (AS_PCR): The first two PCs (Jolliffe (2002)) of $\mathbf{x}_s$s were used as covariates to fit a regression using

a training TF ChIP-seq dataset: $y_s = \beta_0 + \beta_1 PC1 + \beta_2 PC2 + error$. The trained model was used to predict TFBSs for other TFs.

4. All surrogates, CART (AS_CART): Using a training TF ChIP-seq dataset and all surrogates, a prediction model was trained using the classification and regression tree algorithm (Hastie *and others* (2002)) (R package rpart), which was then applied to make predictions for new TFs.

5. All surrogates, Random Forest (AS_RF): The prediction model was trained using all surrogates and random forest (Breman (2001))(R package randomForest).

6. All surrogates, linear kernel SVR (AS_SVR_L): The prediction model was trained using all surrogates and support vector regression with a linear kernel (Hastie *and others* (2002)),(R package e1071): $K(\mathbf{x}, \mathbf{x}') =< \mathbf{x}, \mathbf{x}' >$.

7. All surrogates, Gaussian kernel SVR (AS_SVR_G): The prediction model was trained using all surrogates and support vector regression with a Gaussian kernel (Hastie *and others* (2002)),(R package e1071).

To compare different methods, TF ChIP-seq was used as gold standard. For instance, we train a model using TF A, say EGR1, and use the model to predict binding sites of TF B, say GABP. We can then use TF B ChIP-seq data to benchmark prediction performance. TF B motif sites with $y_s > 1$ were treated as true binding sites. For each prediction method, motif sites were rank ordered based on the predicted TF binding intensities. The positive predictive value (PPV) (i.e., the percentage of true positives among the top predictions) was reported for the top N predictions, where $N = 1, 2, \cdots, etc.$ This created

a curve showing the PPV as a function of N. Curves of different methods were compared. The area under the receiver operating characteristic curves (AUC) was also computed and compared across methods. In parallel, we also computed and compared the Pearson correlation between y_s (from the actual TF B ChIP-seq data) and the predicted binding intensities for each prediction model.

3.10.3 Supplemental Method 3: Clustering analysis

To generate Figure 3.6, we first took the average of x_{sj} as computed in Supplemental Method 1 over the TF bound motif sites (where the TF ChIP-seq $y_s > 1$) and the TF non-bound sites (where $y_s \leq 1$) respectively for TF t and surrogate data type j. This created the average surrogate signals for the bound and non-bound sites, denoted by u_{tj} and v_{tj} respectively. Next, we subtracted v_{tj} from u_{tj} to obtain $r_{tj} = u_{tj} - v_{tj}$. Since x_{sj}, u_{tj} and v_{tj} were all on log2 scale, r_{tj} describes the enrichment of the surrogate signals in the bound class compared to the non-bound class for surrogate data type j and TF t. Using r_{tj}s, we then conducted a hierarchical clustering using Euclidean distance and complete linkage, and the result is shown as a heat map.

3.10.4 Supplemental Method 4: Sensitivity analysis

For Figure 3.8, to calculate the false discovery rate (FDR) for each candidate site, we need to learn the null distribution of DNase intensities. For that purpose, we randomly sampled 1,000,000 loci without replacement from the whole genome and used these loci as our null motif sites. Then we counted DNase read numbers for the null motif sites in the same way as before, and obtained

the background null distribution $p_0(x)$. For a given TF A and a candidate binding sites list, we ranked sites according to the decreasing order of DNase read count. At each cutoff k of the rank list, we computed the p-value based on the null distribution $p_0(x)$ and estimated the false discovery rate (FDR) using the Benjamini-Hechberg procedure (Benjamini and Hochberg (1995)).

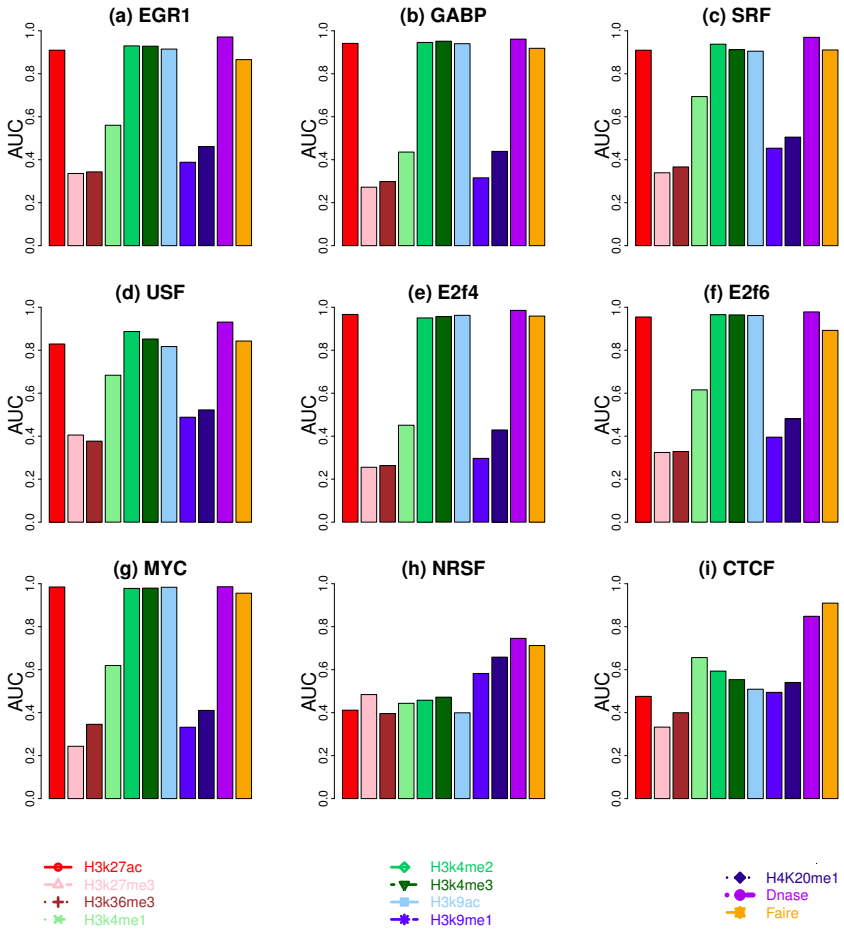


Figure 3.9: Area under the receiver operating characteristic curves for predicting TFBSs in K562 based on single surrogate. (a) EGR1; (b) GABP; (c) SRF; (d) USF; (e) E2F4; (f) E2F6; (g) MYC; (h) NRSF; (i) CTCF.

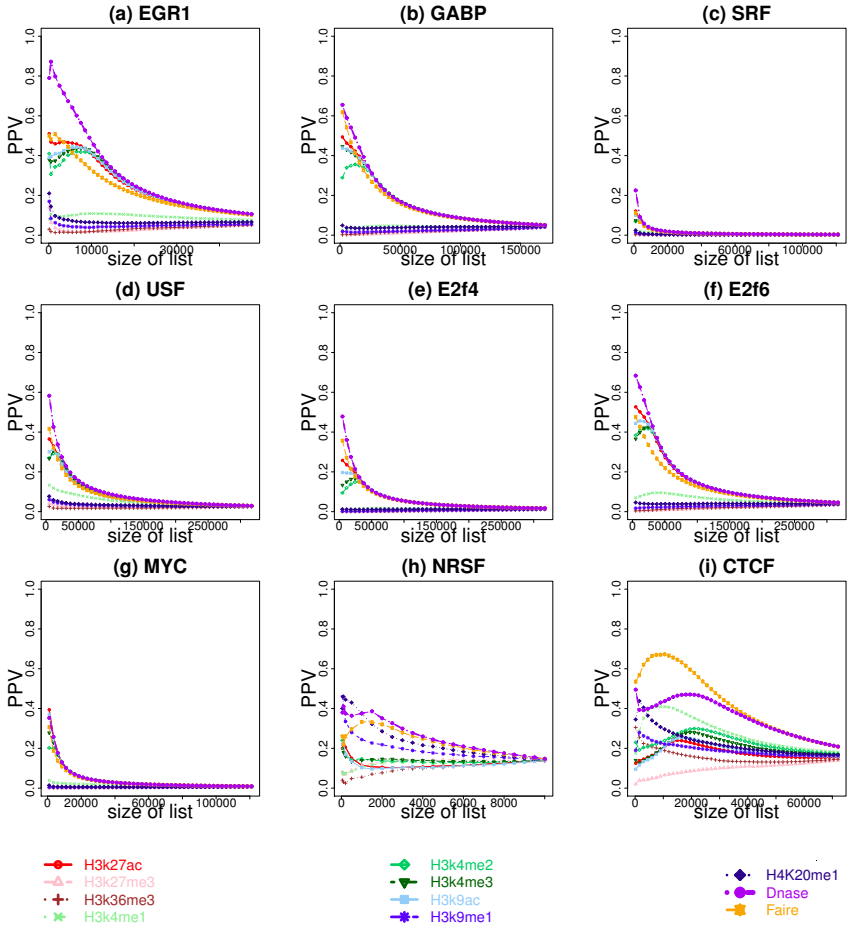


Figure 3.10: Positive predictive value curves for predicting TFBSs in K562 based on single surrogate over all motif sites' ranges. The x axis is the number of the top ranked motif sites. The y axis is the positive predictive value, i.e., the percentage of true positives among the top predictions. (a) EGR1; (b) GABP; (c) SRF; (d) USF; (e) E2F4; (f) E2F6; (g) MYC; (h) NRSF; (i) CTCF.

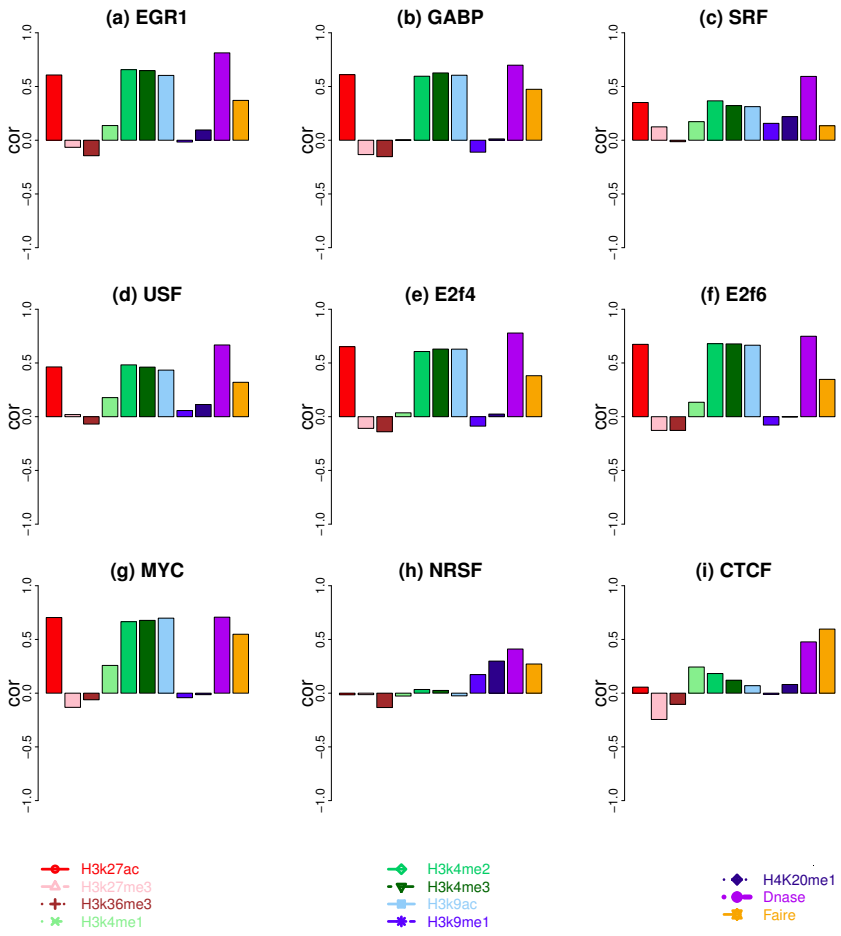


Figure 3.11: Pearson correlation coefficients between the predictors and the actual ChIP-seq binding intensity in K562. (a) EGR1; (b) GABP; (c) SRF; (d) USF; (e) E2F4; (f) E2F6; (g) MYC; (h) NRSF; (i) CTCF.

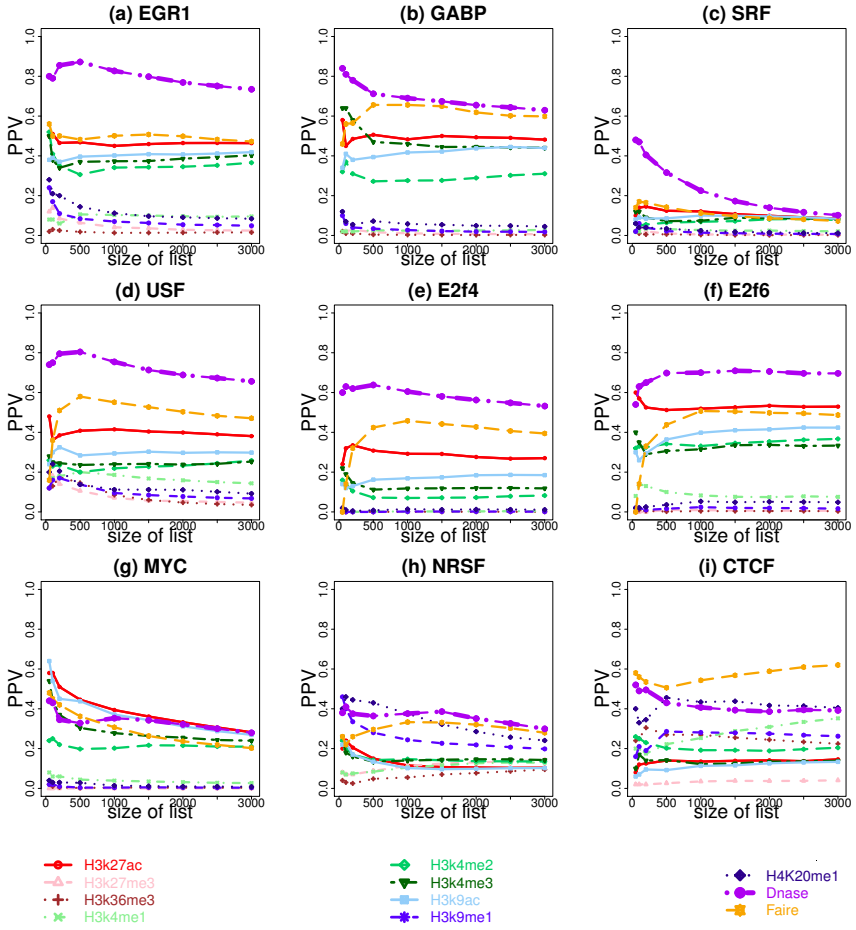


Figure 3.12: Positive predictive value curves for predicting TFBSs in K562 based on single surrogate. The x axis is the number of the top ranked motif sites. The y axis is the positive predictive value. (a) EGR1; (b) GABP; (c) SRF; (d) USF; (e) E2F4; (f) E2F6; (g) MYC; (h) NRSF; (i) CTCF.

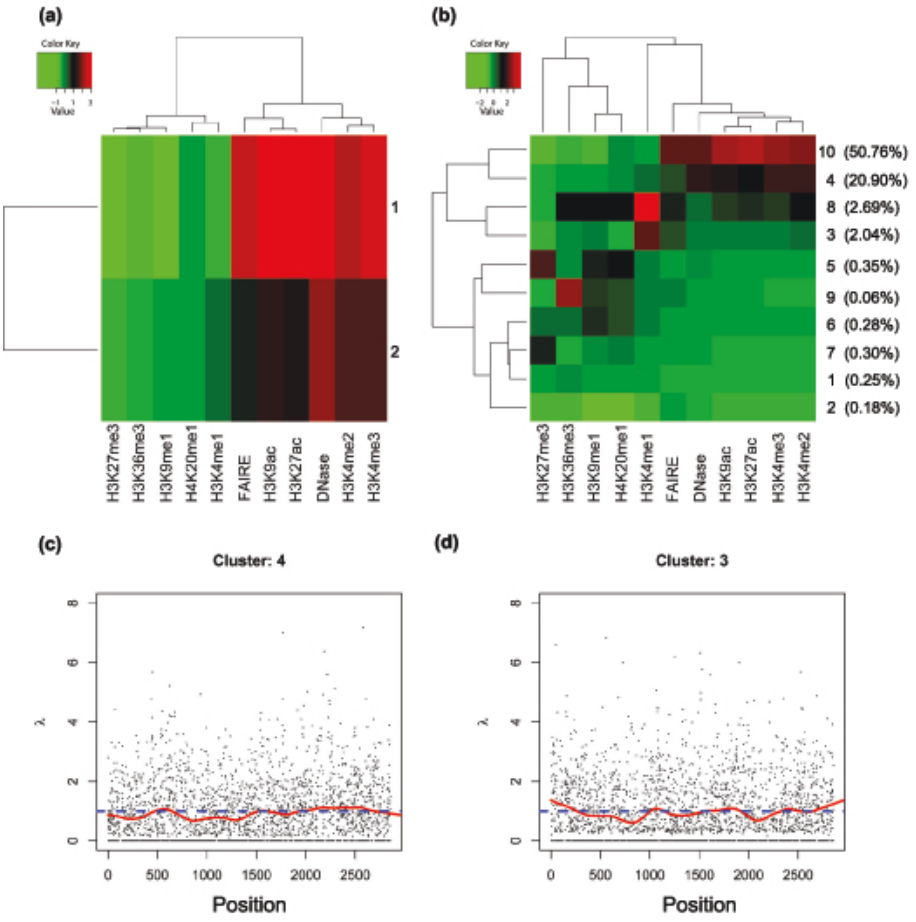


Figure 3.13: K-means clustering of GABP motif sites based on chromatin surrogate signals $\mathbf{x}_s$. (a) The GABP bound motif sites were clustered based on the Euclidean distance. The plot shows results for $k = 2$ clusters. For each cluster and each surrogate, the average signal across all motif sites is shown. (b) All GABP motif sites (bound and non-bound) were clustered into $k = 10$ clusters based on $\mathbf{x}_s$. The percentage of actually bound motif sites for each cluster is shown in the brackets. Clusters 10 and 4 are enriched in true GABP binding sites and both show patterns similar to (a), indicating that most bound motif sites share a similar chromatin pattern. (c),(d) We cut the whole genome into 1Mbp non-overlaping bins. The relative enrichment of cluster k motif site in bin j compared to the genome-wide proportion λ_{jk} is computed and plotted across the genome for two representative clusters: (c) cluster 4, and (d) cluster 3. Different chromosomes are concatenated together in the plots. The smoothing splines for more stable estimates of λ_{jk} (red curve) are shown, which fluctuate around 1 (blue line) across the genome with relatively mild fluctuation, indicating no strong regionalized distribution of motif sites. Other TFs and clusters gave similar results (data not shown).

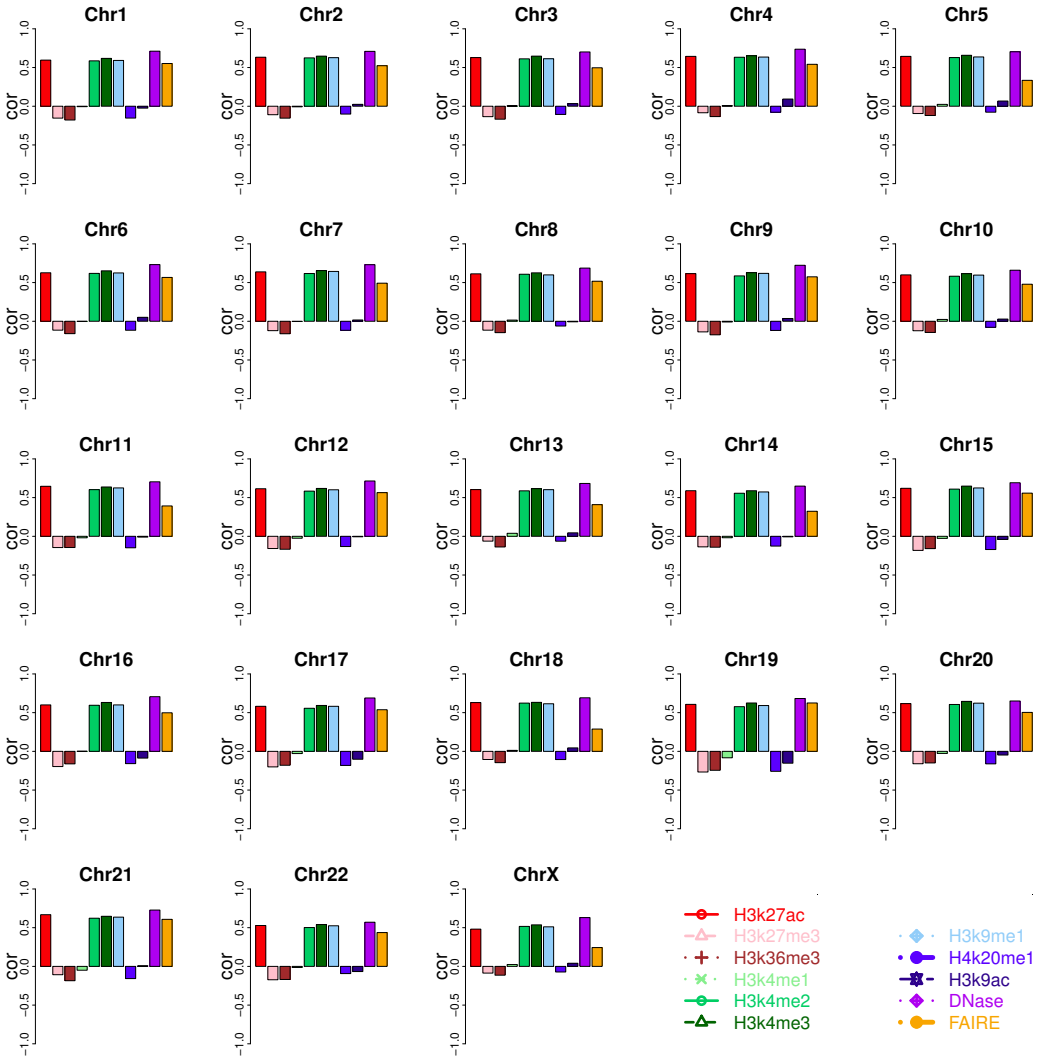


Figure 3.14: Pearson correlation coefficients between GABP ChIP-seq binding intensity and various chromatin surrogates across all GABP motif sites in each chromosome. Different chromosomes show similar correlation patterns. GABP is a representative example. Similar analyses were performed for all other TFs and yielded similar results (data not shown).

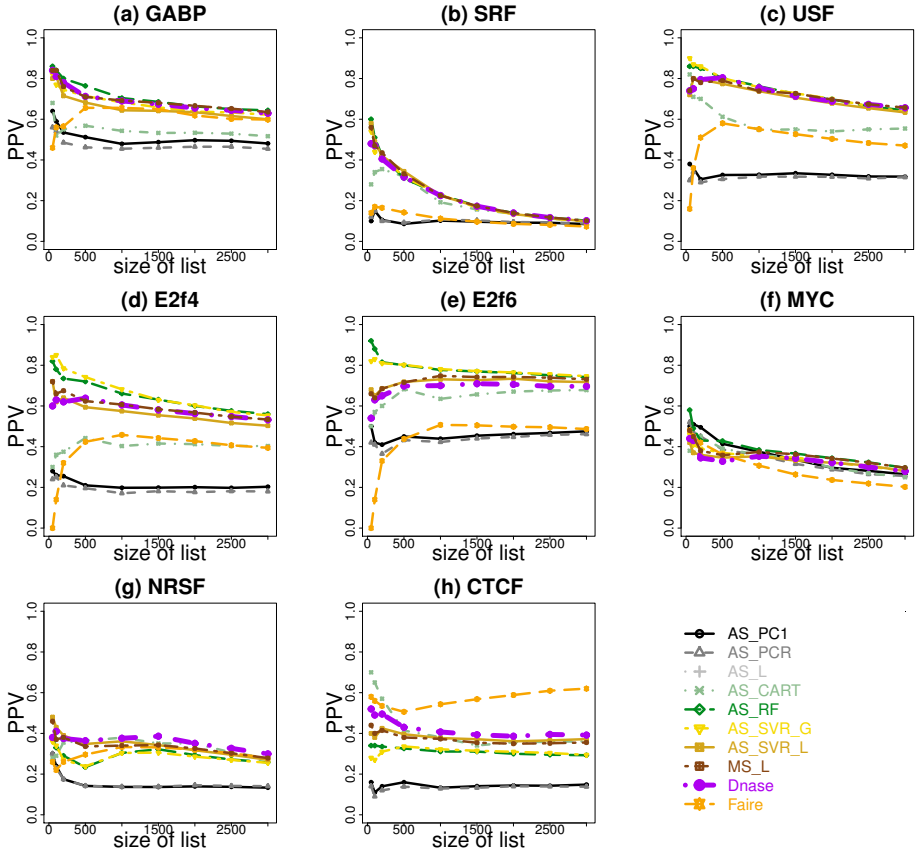


Figure 3.15: Positive predictive value curves for predicting TFBSs in K562 based on models trained using EGR1. (a) Prediction for GABP; (b) prediction for SRF; (c) prediction for USF (d) prediction for E2F4; (e) prediction for E2F6; (f) prediction for MYC; (g) prediction for NRSF; (h) prediction for CTCF. Using other training and test TF pairs produced similar results (data not shown).

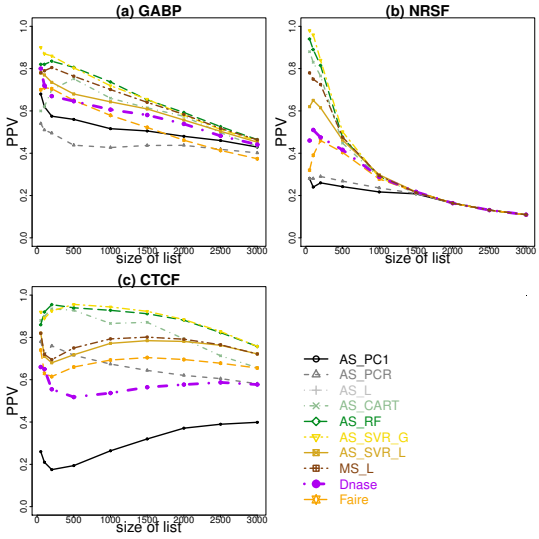


Figure 3.16: Positive predictive values of prediction for chromosomes 17-22 and chromosome X by models trained using chromosomes 1-16 for (a) GABP; (b) NRSF; (c) CTCF. The training and test TFs are the same. Single surrogate predictions by DNase and FAIRE are also added for comparison. The three TFs shown are representative examples of all analyzed TFs.

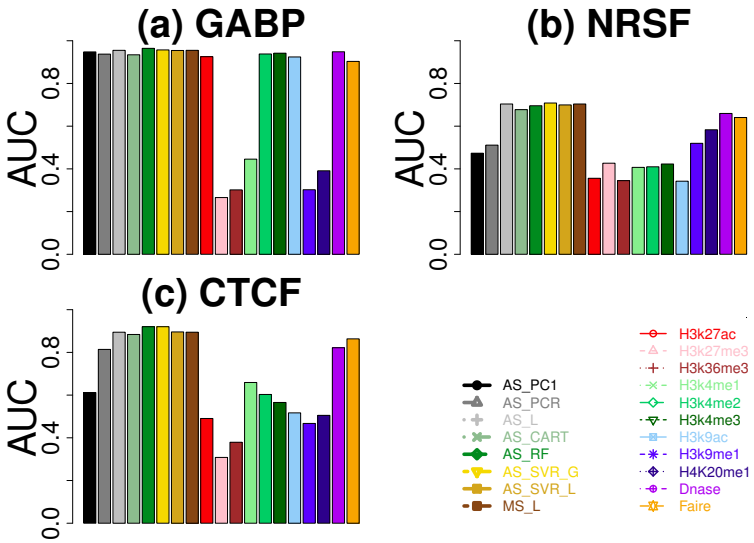


Figure 3.17: AUC of prediction for chromosomes 17-22 and chromosome X by models trained using chromosomes 1-16 for (a) GABP; (b) NRSF; (c) CTCF. The training and test TFs are the same. Single surrogate predictions by DNase and FAIRE are also added for comparison. The three TFs shown are representative examples of all analyzed TFs.

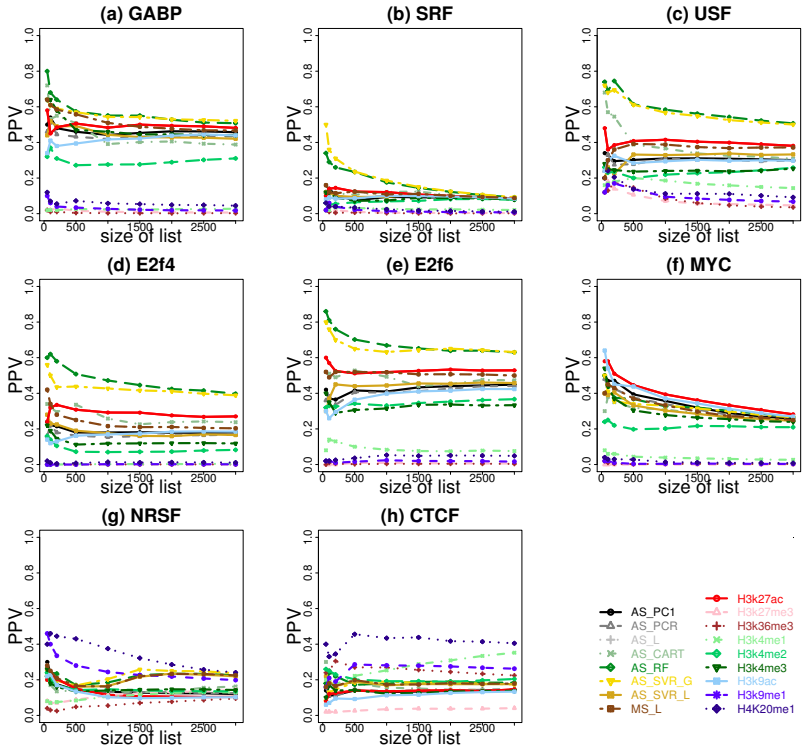


Figure 3.18: Positive predictive value curves for predicting TFBSs in K562 based on models trained on EGR1 using only HM ChIP-seq data. (a) prediction for GABP; (b) prediction for SRF; (c) prediction for USF (d) prediction for E2F4; (e) prediction for E2F6; (f) prediction for MYC; (g) prediction for NRSF; (h) prediction for CTCF.

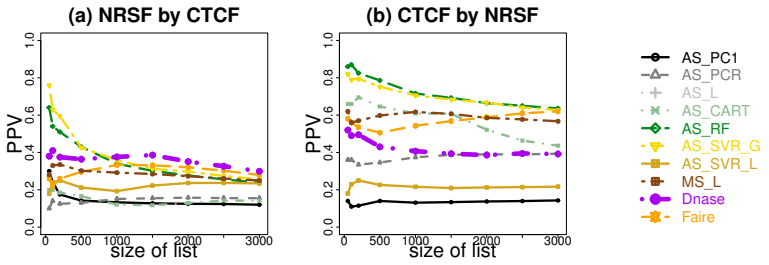


Figure 3.19: Positive predictive value curves for (a) prediction for NRSF by models trained on CTCF using only HM ChIP-seqs and (b) prediction for CTCF by models trained on NRSF using only HM ChIP-seqs. Single surrogate predictions by DNase and FAIRE are also added for comparison.

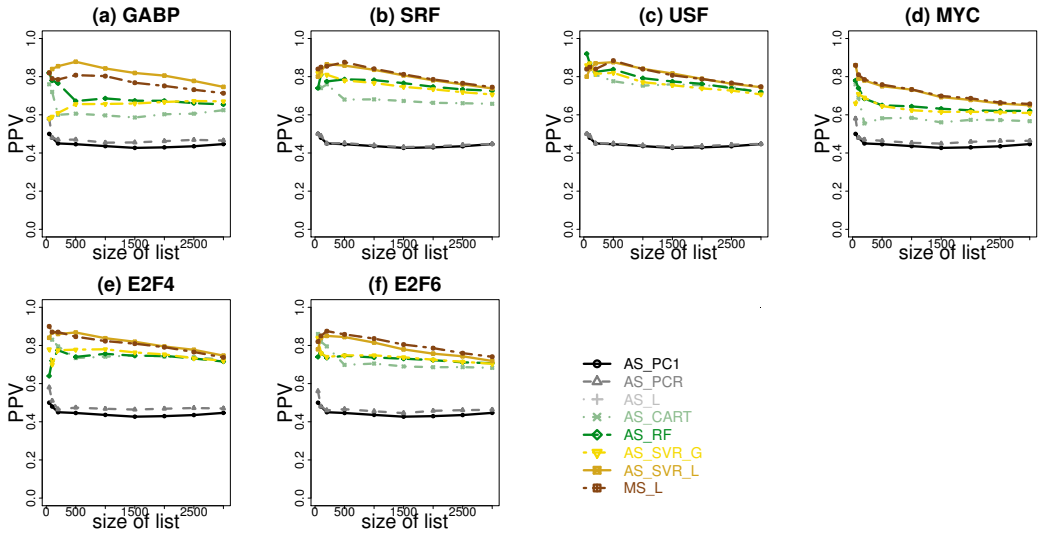


Figure 3.20: Positive predictive value curves for prediction on EGR1 by models trained using ChIP-seq data from different labs. (a) Models trained using GABP (HudsonAlpha); (b) models trained using SRF (HudsonAlpha); (c) models trained using USF (HudsonAlpha); (d) models trained using MYC (UTA); (e) models trained using E2F4 (Yale); (f) models trained using E2F6 (Yale).

Table 3.4: Top ranked surrogate data types based on AUC

TF	rank1	rank2	rank3	rank4	rank5
EGR1	DNase	H3K4me2	H3K4me3	H3K9ac	H3K27ac
GABP	DNase	H3K4me3	H3K4me2	H3K27ac	H3K9ac
SRF	DNase	H3K4me2	H3K4me3	FAIRE	H3K27ac
USF	DNase	H3K4me2	H3K4me3	FAIRE	H3K27ac
E2F4	DNase	H3K27ac	H3K9ac	FAIRE	H3K4me3
E2F6	DNase	H3K4me2	H3K4me3	H3K9ac	H3K27ac
MYC	DNase	H3K27ac	H3K9ac	H3K4me3	H3K4me2
NRSF	DNase	FAIRE	H4K20me1	H3K9me1	H3K27me3
CTCF	FAIRE	DNase	H3K4me1	H3K4me2	H3K4me3

We rank the surrogate data types in terms of AUC for predicting TFBSs of each TF and list the top five surrogate data types.

Bibliography

Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome Biology* **11**, R106.

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

Bell, C.G. and Beck, S. (2009). Advances in the identification and analysis of allele specific expression. *Genome Medicine* **1**, 56.

Ben-David, E., Granot-Hershkovitz, E., Monderer-Rothkoff, G., Lerer, E., Levi, S., Yaari, M., Ebstein, R.P., Yirmiya, N. and Shifman, S. (2011). Identification of a functional rare variant in autism using genome-wide screen for monoallelic expression. *Human Molecular Genetics* **20**, 3632–3641.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B(Methodological)* **57(1)**, 289–300.

Bernstein, B.E., Stamatoyannopoulos, J.A., Costello, J.F., Ren, B., Milosavljevic, A., Meissner, A., Kellis, M., Marra, M.A.,

Beaudet, A.L., Ecker, J.R., Farnham, P.J., Hirst, M., Lander, E.S., Mikkelsen, T.S. *and others*. (2010). The nih roadmap epigenomics mapping consortium. *Nature Biotechnology* **28**, 1045–8.

Boyle, A.P., Davis, S., Shulha, H.P., Meltzer, P., Margulies, E.H., Weng, Z., Furey, T.S. and Crawford, G.E. (2008). High-resolution mapping and characterization of open chromatin across the genome. *Cell* **132**, 311–22.

Boyle, A.P., Song, L., Lee, B.K., London, D., Keefe, D., Birney, E., Iyer, V.R., Crawford, G.E. and Furey, T.S. (2011). High-resolution genome-wide in vivo footprinting of diverse transcription factors in human cells. *Genome Research* **21**, 456–64.

Breman, L. (2001). Random forests. *Machine Learning* **45**, 5–32.

Cawley, S., Bekiranov, S., Ng, H.H., Kapranov, P., Sekinger, E.A., Kampa, D., Piccolboni, A., Sementchenko, V., Cheng, J., Williams, A.J., Wheeler, R., Wong, B., Drenkow, J., Yamanaka, M., Patel, S., Brubaker, S., Tammana, H., Helt, G., Struhl, K. *and others*. (2004). Unbiased mapping of transcription factor binding sites along human chromosomes 21 and 22 points to widespread regulation of non-coding rnas. *Cell* **116**, 499–509.

Celniker, S.E., Dillon, L.A., Gerstein, M.B., Gunsalus, K.C., Henikoff, S., Karpen, G.H., Kellis, M., Lai, E.C., Lieb, J.D., MacAlpine, D.M., Micklem, G., Piano, F., Snyder, M., Stein, L.,

White, K.P., Waterston, R.H. *and others.* (2009). Unlocking the secrets of the genome. *Nature* **459**, 927–930.

Chen, L., Wu, G. and Ji, H. (2011*a*). hmchip: a database and web server for exploring publicly available human and mouse chip-seq and chip-chip data. *Bioinformatics* **27**, 1447–1448.

Chen, P.Y., Feng, S., Joo, J.W., Jacobsen, S.E. and Pellegrini, M. (2011*b*). A comparative analysis of dna methylation across human embryonic stem cell lines. *Genome Biology* **12**, R62.

Chen, R., Mias, G.I., Li-Pook-Than, J., Jiang, L., Lam, H.Y., Miriami, E., Karczewski, K.J., Hariharan, M., Dewey, F.E., Cheng, Y., Clark, M.J., Im, H., Habegger, L., Balasubramanian, S., O'Huallachain, M., Dudley, J.T., Hillenmeyer, S., Haraksingh, R., Sharon, D., Euskirchen, G., Lacroute, P., Bettinger, K., Boyle, A.P., Kasowski, M., Grubert, F., Seki, S., Garcia, M., Whirl-Carrillo, M., Gallardo, M., Blasco, M.A. *and others.* (2012). Personal omics profiling reveals dynamic molecular and medical phenotypes. *Cell* **148**, 1293–1307.

Cheng, C., Shou, C., Yip, K.Y. and Gerstein, M. (2011). Genome-wide analysis of chromatin features identifies histone modification sensitive and insensitive yeast transcription factors. *Genome Biology* **12**, R111.

Conlon, E.M., Song, J. J. and Liu, J.S. (2006). Bayesian models for pooling microarray studies with multiple sources of replications. *BMC Bioinformatics* **7**, 1979 – 1985.

Consortium, ENCODE Project. (2004). The encode (encyclopedia of dna elements) project. *Science* **306**, 636–640.

Consortium, ENCODE Project. (2007). Identification and analysis of functional elements in 1% of the human genome by the encode pilot project. *Nature* **447**, 799–816.

Consortium, ENCODE Project. (2012). An integrated encyclopedia of dna elements in the human genome. *Nature* **489**, 57–74.

Cui, H., Cruz-Correa, M., Giardiello, F.M., Hutcheon, D.F., Kafonek, D.R., Brandenburg, S., Wu, Y., He, X., Powe, N.R. and Feinberg, A.P. (2003). Loss of igf2 imprinting: a potential marker of colorectal cancer risk. *Science* **299**, 1753–1755.

Degner, J.F., Marioni, J.C., Pai, A.A., Pickrell, J.K., Nkadori, E., Gilad, Y. and Pritchard, J.K. (2009). Effect of read-mapping biases on detecting allele-specific expression from rna-sequencing data. *Bioinformatics* **25**, 3207–3212.

Ernst, J. and Kellis, M. (2010). Discovery and characterization of chromatin states for systematic annotation of the human genome. *Nature Biotechnology* **28**, 817–25.

Ernst, J., Kheradpour, P., Mikkelsen, T.S., Shoresh, N., Ward, L.D., Epstein, C.B., Zhang, X., Wang, L., Issner, R., Coyne, M., Ku, M., Durham, T., Kellis, M. *and others.* (2011). Mapping and analysis of chromatin state dynamics in nine human cell types. *Nature* **473**, 43–9.

Gaulton, K.J., Nammo, T., Pasquali, L., Simon, J.M., Fogarty, P.G. Giresiand M.P., Panhuis, T.M., Mieczkowski, P., Secchi, A., Bosco, D., Berney, T., Montanya, E., Mohlke, K.L., Lieb, J.D. *and others*. (2010). A map of open chromatin in human pancreatic islets. *Nature Genetics* **42**, 255–9.

Gelman, A., Carlin, J.B., Stern, H.S. and Rubin, D.B. (2004). *Bayesian Data Analysis, Second Edition*. New York, NY: Chapman Hall/CRC.

Graze, R.M., Novelo, L.L., Amin, V., Fear, J.M., Casella, G., Nuzhdin, S.V. and McIntyre, L.M. (2012). Allelic imbalance in drosophila hybrid heads: exons, isoforms, and evolution. *Molecular Biology and Evolution* **29**, 1521–1532.

Hastie, T., Tibshirani, R. and Friedman, J. (2002). *The elements of statistical learning: data mining, inference, and prediction, second edition*. New York: Springer.

He, H.H., Meyer, C.A., Shin, H., Bailey, S.T., Wei, G., Wang, Q., Zhang, Y., Xu, K., Ni, M., Lupien, M., Mieczkowski, P., Lieb, J.D., Zhao, K., Brown, M. *and others*. (2010). Nucleosome dynamics define transcriptional enhancers. *Nature Genetics* **42**, 343–7.

Heap, G.A., Yang, J.H.M., Downes, K., Healy, B.C., Hunt, K.A., Bockett, N., Franke, L., Dubois, P.C., Mein, C.A., Dobson, R.J., Albert, T.J., Rodesch, M.J., Clayton, D.G., Todd, J.A., Van-Heel, D.A. *and others*. (2010). Genome-wide analysis of allelic expression

imbalance in human primary cells by high-throughput transcriptome resequencing. *Human Molecular Genetics* **19**, 122–134.

Heintzman, N.D., Stuart, R.K., Hon, G., Fu, Y., Ching, C.W., Hawkins, R.D., Barrera, L.O., Van-Calcar, S., Qu, C., Ching, K.A., Wang, W., Weng, Z., Green, R.D., Crawford, G.E. *and others.* (2007). Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome. *Nature Genetics* **39**, 311–8.

Holt, R.J., Zhang, Y., Binia, A., Dixon, A.L., Vandiedonck, C., Cookson, W.O., Knight, J.C. and Moffatt, M.F. (2011). Allele-specific transcription of the asthma-associated phd finger protein 11 gene (phf11) modulated by octamer-binding transcription factor 1 (oct-1). *Journal of Allergy and Clinical Immunology* **127**, 1054–1062 e1051–1052.

Hon, G., Wang, W. and Ren, B. (2009). Discovery and annotation of functional chromatin signatures in the human genome. *PLoS Computational Biology* **5**, e1000566.

Hu, S., Xie, Z., Onishi, A., Yu, X, Jiang, L., Lin, J., Woodard, H.S. Rhoand C., Wang, H., Jeong, J.S., Long, S., He, X., Wade, H., Blackshaw, S., Qian, J. *and others.* (2009). Profiling the human protein-dna interactome reveals erk2 as a transcriptional repressor of interferon signaling. *Cell* **139**, 610–22.

Ingham, P.W. and McMahon, A.P. (2001). Hedgehog signaling in animal development: paradigms and principles. *Genes and Development* **15**, 3059–3087.

Irizarry, R.A., Hobbs, B., Collin, F., Beazer-Barclay, Y.D., Antonellis, K.J., Scherf, U. and Speed, T.P. (2003). Exploration, normalization, and summaries of high density oligonucleotid array probe level data. *Biostatistics* **4(2)**, 249–264.

Jensen, S.T., Erkan, I., Arnardottir, E.S. and Small, D.S. (2009). Bayesian testing of many hypothesis*many genes: a study of sleep apnea. *Annals of Applied Statistics* **3(3)**, 1080–1101.

Jensen, S.T., Liu, X.S., Zhou, Q. and Liu, J.S. (2004). Computational discovery of gene regulatory binding motifs: A bayesian perspective. *Statistical Science* **19**, 188–204.

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

Ji, H. and Wong, W.H. (2006). Computational biology: toward deciphering gene regulatory information in mammalian genomes. *Biometrics* **62**, 645–63.

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

Jolliffe, I. (2002). *Principal component analysis*. New York: Springer.

Ju, Y.S., Kim, J.I., Kim, S., Hong, D., Park, H., Shin, J.Y., Lee, S., Lee, W.C., Kim, S., Yu, S.B., Park, S.S., Seo, S.H., Yun, J.Y., Kim, H.J., Lee, D.S., Yavartanoo, M., Kang, H.P., Gokcumen, O.,

Govindaraju, D.R., Jung, J.H., Chong, H., Yang, K.S., Kim, H., Lee, C. *and others*. (2011). Extensive genomic and transcriptional diversity identified through massively parallel dna and rna sequencing of eighteen korean individuals. *Nature Genetics* **43**, 745–U747.

Kasowski, M., Grubert, F., Heffelfinger, C., Hariharan, M., Asabere, A., Waszak, S.M., Habegger, L., Rozowsky, J., Shi, M., Urban, A.E., Hong, M.Y., Karczewski, K.J., Huber, W., Weissman, S.M., Gerstein, M.B., Korbel, J.O. *and others*. (2010). Variation in transcription factor binding among humans. *Science* **328**, 232–235.

Kendziorski, C.M., M.A. Newton, M. A.and H. Lan and Gould, M.N. (2003). On parametric empirical bayes methods for comparing multiple groups using replicated gene expression profiles. *Statistics in Medicine* **22**, 3899–3914.

Kerkel, K., Spadola, A., Yuan, E., Kosek, J., Jiang, L., Hod, E., Li, K., Murty, V.V., Schupf, N., Vilain, E., Morris, M., Haghighi, F. *and others*. (2008). Genomic surveys by methylation-sensitive snp analysis identify sequence-dependent allele-specific dna methylation. *Nature Genetics* **40**, 904–908.

Kim, T.H., Abdullaev, Z.K., Smith, A.D., Ching, K.A., Loukinov, D.I., Green, R.D., Zhang, M.Q., Lobanenkov, V.V. and Ren, B. (2007). Analysis of the vertebrate insulator protein ctcf-binding sites in the human genome. *Cell* **128**, 1231–45.

Knight, J.C. (2004). Allele-specific gene expression uncovered. *Trends in Genetics* **20**, 113–116.

Kucera, K.S., Reddy, T.E., Pauli, F., Gertz, J., Logan, J.E., Myers, R.M. and Willard, H.F. (2011). Allele-specific distribution of rna polymerase ii on female x chromosomes. *Human Molecular Genetics* **20**, 3964–3973.

Li, H., Ruan, J. and Durbin, R. (2008). Mapping short dna sequencing reads and calling variants using mapping quality scores. *Genome Research* **18**, 1851–1858.

Lo, H.S., Wang, Z., Hu, Y., Yang, H.H., Gere, S., Buetow, K.H. and Lee, M.P. (2003). Allelic variation in gene expression is common in the human genome. *Genome Research* **13**, 1855–1862.

Mao, J., Ligon, K.L., Rakhlin, E.Y., Thayer, S.P., Bronson, R.T., Rowitch, D. and McMahon, A.P. (2006). A novel somatic mouse model to survey tumorigenic potential applied to the hedgehog pathway. *Cancer Research* **66(20)**, 10171–10178.

McDaniell, R., Lee, B.K., Song, L., Liu, Z., Boyle, A.P., Erdos, M.R., Scott, L.J., Morken, M.A., Kucera, K.S., Battenhouse, A., Keefe, D., Collins, F.S., Willard, H.F., Lieb, J.D., Furey, T.S., Crawford, G.E., Iyer, V.R. *and others.* (2010). Heritable individual-specific and allele-specific chromatin signatures in humans. *Science* **328**, 235–239.

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

Montgomery, S.B., Sammeth, M., Gutierrez-Arcelus, M., Lach, R.P., Ingle, C., Nisbett, J., Guigo, R. and Dermitzakis, E.T. (2010). Transcriptome genetics using second generation sequencing in a caucasian population. *Nature* **464**, 773–U151.

Morley, M., Molony, C.M., Weber, T.M., Devlin, J.L., Ewens, K.G., Spielman, R.S. and Cheung, V.G. (2004). Genetic analysis of genome-wide variation in human gene expression. *Nature* **430**, 743–747.

Palacios, R., Gazave, E., Goni, J., Piedrafita, G., Fernando, O., Navarro, A. and Villoslada, P. (2009). Allele-specific gene expression is widespread across the genome and biological processes. *PLoS One* **4(1)**, e4150.

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

Pique-Regi, R., Degner, J.F., Pai, A.A., Gaffney, D.J., Gilad, Y. and Pritchard, J.K. (2011). Accurate inference of transcription factor

binding from dna sequence and chromatin accessibility data. *Genome Research* **21**, 447–55.

Reddy, T.E., Gertz, J., Pauli, F., Kucera, K.S., Varley, K.E., Newberry, K.M., Marinov, G.K., Mortazavi, A., Williams, B.A., Song, L., Crawford, G.E., Wod, B., Willard, H. *and others.* (2012). Effects of sequence variation on differential allelic transcription factor occupancy and gene expression. *Genome Research* **22**, 860–869.

Ren, B., Robert, F., Wyrick, J.J., Aparicio, O., Jennings, E.G., Simon, I., Zeitlinger, J., Schreiber, J., Hannett, N., Kanin, E., Volkert, T.L., Wilson, C.J., Bell, S.P. *and others.* (2000). Genome-wide location and function of dna binding proteins. *Science* **290**, 2306–9.

Robasky, K. and Bulyk, M.L. (2011). Uniprobe and update 2011: expanded content and search tools in the online database of protein-binding microarray data on protein-dna interactions. *Nucleic Acids Research* **39**, D124–8.

Robertson, G., Hirst, M., Bainbridge, M., Bilenky, M., Zhao, Y., Zeng, T., Euskirchen, G., Bernier, B., Varhol, R., Delaney, A., Thiessen, N., Griffith, O.L., He, A., Marra, M., Snyder, M. *and others.* (2007). Genome-wide profiles of stat1 dna association using chromatin immunoprecipitation and massively parallel sequencing. *Nature Methods* **4**, 651–657.

Robinson, M.D. and Smyth, G.K. (2007). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics* **23**, 2881–2887.

Robinson, M.D. and Smyth, G.K. (2008). Small-sample estimation of negative binomial dispersion, with applications to sage data. *Biostatistics* **9**, 321–332.

Rozowsky, J., Abyzov, A., Wang, J., Alves, P., Raha, D., Harmanci, A., Leng, J., Bjornson, R., Kong, Y., Kitabayashi, N., Bhardwaj, N., Rubin, M., Snyder, M. *and others*. (2011). Alleleseq: analysis of allele-specific expression and binding in a network framework. *Molecular Systems Biology* **7**, 522.

Ruan, L. and Yuan, M. (2011). An empirical bayes approach to joint analysis of multiple microarray gene expression studies. *Biometrics* **67**, 1617C–1626.

Sandelin, A., Alkema, W., Engstrom, P., Wasserman, W.W. and Lenhard, B. (2004). Jaspar: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research* **32**, D91–4.

Scharpf, R.B., Tjelmeland, H., Parmigiani, G. and Nobel, A.B. (2009). A bayesian model for cross-study differential gene expression. *Journal of the American Statistical Association* **104(488)**, 1295–1310.

Schilling, E., Chartouni, C.E. and Rehli, M. (2009). Allele-specific dna methylation in mouse strains is mainly determined by cis-acting sequences. *Genome Research* **19**, 2028–2035.

Serre, D., Gurd, S., Ge, B., Sladek, R., Sinnett, D., Harmsen, E., Bibikova, M., Chudin, E., Barker, D.L., Dickinson, T., Fan, J.B. *and others*. (2008). Differential allelic expression in the human genome:

a robust approach to identify genetic and epigenetic cis-acting mechanisms regulating gene expression. *PLoS Genetics* **4**, e1000006.

Sims, J.K., Houston, S.I., Magazinnik, T. and Rice, J.C. (2006). A trans-tail histone code defined by monomethylated h4 lys-20 and h3 lys-9 demarcates distinct regions of silent chromatin. *The Journal of Biological Chemistry* **281**, 12760–6.

Skelly, D.A., Johansson, M., Madeoy, J., Wakefield, J. and Akey, J.M. (2011). A powerful and flexible statistical framework for testing hypotheses of allele-specific gene expression from rna-seq data. *Genome Research* **21**, 1728–1737.

Smyth, G.K. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology* **3**, 3.

Song, L., Zhang, Z., Grasfeder, L.L., Boyle, A.P., Giresi, P.G., Lee, B.K., Sheffield, N.C., Graf, S., Huss, M., Keefe, D., Liu, Z., London, D., McDaniell, R.M., Shibata, Y., Showers, K.A., Simon, J.M., Vales, T., Wang, T., Winter, D., Zhang, Z., Clarke, N.D., Birney, E., Iyer, V.R., Crawford, G.E., Lieb, J.D. *and others.* (2011). Open chromatin defined by dnasei and faire identifies regulatory elements that shape cell-type identity. *Genome Research* **21**, 1757–6.

Stormo, G.D. (2000). Dna binding sites: Representation and discovery. *Bioinformatics* **16**, 16–23.

Tang, F.C., Barbacioru, C., Nordman, E., Bao, S.Q., Lee, C., Wang, X.H., Tuch, B.B., Heard, E., Lao, K.Q. and Surani, M.A. (2011). Deterministic and stochastic allele specific gene expression in single mouse blastomeres. *PLoS One* **6(6)**, e21208.

Tenzen, T., Allen, B.L., Cole, F., Kang, J.S., Krauss, R.S. and McMahon, A.P. (2006). The cell surface membrane proteins cdo and boc are components and targets of the hedgehog signaling pathway and feedback network in mice. *Developmental Cell* **10(5)**, 647–656.

Tompa, M., Li, N., Bailey, T.L., Church, G.M., De-Moor, B., Eskin, E., Favorov, A.V., Frith, M.C., Fu, Y., Kent, W.J., Makeev, V.J., Mironov, A.A., Noble, W.S., Pavesi, G., Pesole, G., Rgnier, M., Simonis, N., Sinha, S., Thijs, G., van Helden, J., Vandenbogaert, M., Weng, Z., Workman, C., Ye, C. *and others*. (2005). Assessing computational tools for the discovery of transcription factor binding sites. *Nature Biotechnology* **23**, 137–144.

Tuch, B.B., Laborde, R.R., Xu, X., Gu, J., Chung, C.B., Monighetti, C.K., Stanley, S.J., Olsen, K.D., Kasperbauer, J.L., Moore, E.J., Broomer, A.J., Tan, R.Y., Brzoska, P.M., Muller, M.W., Siddiqui, A.S., Asmann, Y.W., Sun, Y.M., Kuersten, S., Barker, M.A., De-La-Vega, F.M. *and others*. (2010). Tumor transcriptome sequencing reveals allelic expression imbalances associated with copy number alterations. *PLoS One* **5(2)**, e9317.

Tusher, V.G., Tibshirani, R. and Chu, G. (2001). Significance analysis

of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences* **98(9)**, 5116–5121.

Tycko, B. (2010). Allele-specific dna methylation: beyond imprinting. *Human Molecular Genetics* **19**, R210–220.

Villavicencio, E.H., Walterhouse, D.O. and Iannaccone, P.M. (2000). The sonic hedgehogcpatchedcgli pathway in human development and disease. *The American Journal of Human Genetics* **67(5)**, 1047–1054.

Vokes, S.A., Ji, H., McCuine, S., Tenzen, T., Giles, S., Zhong, S., Longabaugh, W.J.R., Davidson, E.H. and McMahon, A.P. (2007). Genomic characterization of gli-activator targets in sonic hedgehog-mediated neural patterning. *Development* **134**, 1977–1989.

Vokes, S.A., Ji, H., Wong, W.H. and McMahon, A.P. (2008). Whole genome identification and characterization of gli cis-regulatory circuitry in hedgehog-mediated mammalian limb development. *Genes Development* **22**, 2651–2663.

Wang, Z., Zang, C., Rosenfeld, J.A., Schones, D.E., Barski, A., Cuddapah, S., Cui, K., Roh, T.Y., Peng, W., Zhang, M.Q. *and others.* (2008). Combinatorial patterns of histone acetylations and methylations in the human genome. *Nature Genetics* **40**, 897–903.

Whitington, T., Perkins, A.C. and Bailey, T.L. (2009). High-throughput chromatin information enables accurate tissue-specific prediction of transcription factor binding sites. *Nucleic Acids Research* **37**, 14–25.

Wingender, E., Dietze, P., Karas, H. and Knuppel, R. (1996). Transfac: a database on tran- scription factors and their dna binding sites. *Nucleic Acids Research* **24**, 238–41.

Won, K.J., Ren, B. and Wang, W. (2010). Genome-wide prediction of transcription factor binding sites using an integrated model. *Genome Biology* **11**, R7.

Xie, Z., Hu, S., Blackshaw, S., Zhu, H. and Qian, J. (2010). hpdi: a database of experimental human protein-dna interactions. *Bioinformatics* **26**, 287–9.

Yuan, M. and Kendziorski, C.M. (2006). A unified approach for simultaneous gene clustering and differential expression identification. *Biometrics* **62**, 1089–1098.

Zhang, K., Li, J.B., Gao, Y., Egli, D., Xie, B., Deng, J., Li, Z., Lee, J.H., Aach, J., Leproust, E.M., Eggan, K. *and others.* (2009). Digital rna allelotyping reveals tissue-specific and allele-specific gene expression in human. *Nature Methods* **6**, 613–618.

Yingying Wei

Johns Hopkins University, Department of Biostatistics

Email: ywei:ywei@jhsph.edu Phone: 410-502-3365

Education

Ph.D., Biostatistics, Johns Hopkins University, 2014

Advisor: Dr. Hongkai Ji

B.S., Mathematics, Tsinghua University, 2009

Honors and Awards

2013 The Margaret Merrell Award, for outstanding research by a Biostatistics doctoral student, *the Johns Hopkins University*

2013 ENAR Distinguished Student Paper Award, *the International Biometric Society*

2012 The June B. Culley Award, for outstanding achievement on the school wide examination paper, *the Johns Hopkins University*

2012 First Place, Biostatistics Section of the Delta Omega Poster Competition,*the Johns Hopkins University*

2009 Department of Biostatistics Entrance Scholarship, *the Johns Hopkins University*

2009 Excellent Graduate of Tsinghua University

2008 Meritorious Award for the Mathematical Contest in Modeling (MCM), *the Society for Industrial and Applied Mathematics (SIAM), the*

National Security Agency (NSA), and the Institute for Operations Research and the Management Sciences (INFORMS), USA

2006-2008 Outstanding Academic Performance Scholarship, *Tsinghua University*

2006 National Scholarship, *Ministry of Education of the People's Republic of China*

Research Experience

2011-2014 Research Assistant to Dr. John J. Laterra, The Johns Hopkins University School of Medicine and the Kennedy Krieger Research Institute, MD USA

2010-2011 Research Assistant to Dr. Michael F. Ochs and Dr. Joseph A. Califano, Division of Oncology Biostatistics and Bioinformatics, the Sidney Kimmel Comprehensive Cancer Center of the Johns Hopkins University, and Department of Otolaryngology-Head and Neck Surgery, the Johns Hopkins Medical Institutions, MD USA

2007-2009 Research Assistant to Dr. Shao Li, Bioinformatics Division, National Laboratory for Information Science and Technology, Tsinghua University, Beijing, China

Papers

Li S, Zhang B, Jiang D, **Wei YY**, Zhang N (2010) Herb network construction and co-module analysis for uncovering the combination rule of traditional Chinese herbal formulae. *BMC Bioinformatics*. 11(S11):S6.

Wei YY*, Li X*, Wang Q, Ji HK(2012) iASeq: Integrating Multiple ChIP-seq Datasets for Detecting Allele-specific Binding. *BMC Genomics*. 13:681. **Highly accessed**.($*$ joint first authors.)

Wei YY, Wu G, Ji HK (2013) Global Mapping of Transcription Factor Binding Sites by Sequencing Chromatin States: A Perspective on Experimental Design, Data Analysis, and Open Problems. *Statistics in Biosciences*.5: 156-178.

Wang JY, Park JS, **Wei YY**, Rajurkar M, Cotton JL, Fan Q, Lewis BC, Ji HK, Mao JH (2013) TRIB2 acts downstream of Wnt/TCF in liver cancer cells to regulate YAP and C/EBPα function. *Molecular Cell*. 51: 211-225.

Ochs MF, Farrar JE, Considine M, **Wei YY**, Meschinchi S, Arcei RJ (2013) Outlier gene set analysis combined with top scoring pair provides robust biomarkers of pathway activity. *Pattern Recognition in Bioinformatics: Lecture Notes in Computer Science*. 7986: 47-58.

Ochs MF, Farrar JE, Considine M, **Wei YY**, Meschinchi S, Arcei RJ (2013) Outlier analysis and top scoring pair for integrated data analysis and biomarker discovery. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. Accepted

Software

coGPS an R/Bioconductor package to conduct gene set enrichment analysis on P-value based statistics for outlier gene detection in dataset

merged from multiple genomic data types

http://www.bioconductor.org/packages/release/bioc/html/coGPS.html

Cormotif an R/Bioconductor package for jointly analyzing multiple gene expression datasets to simultaneously detect differentially expression genes and patterns

http://www.bioconductor.org/packages/release/bioc/html/Cormotif.html

iASeq an R/Bioconductor package for detecting allele-specific binding by jointly analyzing multiple ChIP-seq datasets

http://www.bioconductor.org/packages/release/bioc/html/iASeq.html

Presentations

Wei YY, Ji HK. Measuring Co-binding among transcription factors. August 8, 2013. Montreal, CA.

Wei YY, Li X, Wang Q, Ji HK iASeq: Integrating Multiple ChIP-seq Datasets for Detecting Allele-specific Binding. Eastern North America Region of the International Biometric Society. March 13, 2013. Orlando, FL.

Wei YY, Li X, Wang Q, Ji HK iASeq: Integrating Multiple ChIP-seq Datasets for Detecting Allele-specific Binding. Joint Statistical Meetings. July 31, 2012. San Diego, CA. (poster)

Wei YY, Li X, Wang Q, Ji HK iASeq: Integrating Multiple ChIP-seq Datasets for Detecting Allele-specific Binding. 20th Annual International

Conference on Intelligent Systems for Molecular Biology. July 16, 2012. Long Beach, CA. (poster)

Wei YY, Li X, Wang Q, Ji HK iASeq: Integrating Multiple ChIP-seq Datasets for Detecting Allele-specific Binding. 8th ISCB Student Council Symposium. July 13, 2012. Long Beach, CA.

Professional Activities

Referee	*Biostatistics*
Assisted review for	*Annals of Applied Statistics*
Assisted review for	*Bioinformatics*
Assisted review for	*Genome Biology*
Assisted review for	*Genomics*
Assisted review for	*Plos Computational Biology*
Volunteer	*The Eastern North American Region Meetings, 2012*
Society membership	*The American Statistical Association*
Society membership	*The International Biometric Society*

Working Group

Member Joint Genomics Working Group,The Johns Hopkins University, Department of Biostatistics

Member Survival, Longitudinal, and Multilevel Modeling Working Group, The Johns Hopkins University, Department of Biostatistics

Statistical Consulting

2013-2014 The Department of Environmental Health Sciences, JHU.

2011-2012 The Department of Health Policy and Management, JHU

Teaching Assistant

2013 Biostatistics 654: Methods in Biostatistics IV, JHU.

2013 Biostatistics 653: Methods in Biostatistics III, JHU.

2012 Biostatistics 652: Methods in Biostatistics II, JHU.

2012 Biostatistics 651: Methods in Biostatistics I, JHU.

2011 Biostatistics 624: Statistical Methods in Public Health IV, JHU.

2011 Biostatistics 623: Statistical Methods in Public Health III, JHU.

2010 Biostatistics for Public Health (undergraduate course), JHU.

2009 Application of statistics and probability (undergraduate course), Tsinghua University.

Computing Proficiency

Programming Language:

Comprehensive: C, C++, R, WinBUGS, Perl, MySQL

Functional: MATLAB, SAS

Operating System: Linux, Windows