Ph. D. Thesis

Computational Methods for Analyzing the Architecture and Evolution of the Regulatory Genome

Pradipta Ray

CMU-LTI-12-018

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:
Eric P. Xing, Carnegie Mellon (co-chair)
Veronica F. Hinman, Carnegie Mellon (co-chair)
Jaime Carbonell, Carnegie Mellon
Ziv-Bar Joseph , Carnegie Mellon
Martin Kreitman, University of Chicago

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Keywords: Regulatory Genomics, Comparative Genomics, Regulatory Evolution, motif finding, Hierarchical HMM, Generalized HMM, Graphical Model, Conditional Random Field, Latent Dirichlet Allocation

Dedicated to the memory of two friends
Partha Basu (1979 - 1996)
who introduced me to my schoolboy love of computer programming
Subhankar Nag (1979 - 2006)
who introduced me to the wonderful world of college trivia

Contents

Abstract

Diversity in forms of animal life, diversity in function of an organisms cells, and diversity in function of a single cell over its lifetime or in response to different stimuli drives a whole plethora of processes in biology. Gene control circuitry actually dictates whether or not, and when and by how much a particular gene should be expressed in a cell in order to create such diversity. Such control mechanisms are often present in the genome in the form of cis-regulatory modules: regions in the neighborhood of each gene which contain sequence motifs (particular genetic subsequences which are noisy copies of each other) where proteins (called Transcription Factors) that regulate the gene expression bind. The large amounts of genomic (and often other corresponding experimental) data involved, and the complexity of the resulting analysis lends itself well to a machine learning setting.

One goal of this thesis is to explore supervised motif detection in regulatory sequences by maximally utilizing the inherent grammar or structure of the cis-regulatory modules. We achieve this goal by using hierarchical and generalized Hidden Markov Models in a Bayesian setting. Another goal is to explore supervised motif detection by using multiple sequence alignments, specifically modeling functional turnover : a confounding phenomenon in phylogenetic analysis where orthologous sequences across even closely related species have varying functionality due to rapid coordinated evolutionary change. We developed a generative graphical model which models the multiple sequence alignment as the output of a mixture of phylogenies and perform inference on it to identify regulatory regions and turnover events in related species. A third goal is to analyze diverse sources of evidence and conclude which genetic and epigenetic features correlate well with binding site locations, and to use such information to create a discriminative model for supervised prediction of binding sites. We use the discriminative framework of a conditional random field for the purpose, which assigns weights to genomic, evolutionary, translational, compositional, as well as epigenetic features.

A final goal is to model the evolutionary dynamics of regulatory regions. We modeled co-evolving regions inside cis-regulatory modules by spectral clustering evolutionary parameters in different regulatory subsequences. We also analyzed evolutionary forces in the regulatory genome by identifying which k-mers are preferentially present in regulatory regions across species by modeling regulatory regions as being constructed from evolving mixtures of stochastic dictionaries. This thesis provides novel statistical frameworks for identifying regulatory regions, and analyzing them in terms of their architecture, function, evolutionary properties and correlation with other genomic and epigenomic features in a computationally optimal and statistically sound way.

Acknowledgements

I would like to thank my advisors Eric Xing and Veronica Hinman, who were encouraging when I was successful, patient when I struggled, and insightful when I needed help. Thank you : your example, advice, counsel and support has impacted me in both my research and the way I live life . I would also like to thank my other thesis committee members Jaime Carbonell, Ziv-Bar Joseph and Martin Kreitman, for their great counsel during my thesis proposal, job hunt and afterwards. Combined, all of you enabled me to access great expertise in Statistics, Computer Science and Biology that has helped shape the thesis. I would further like to thank Chuck Ettehnsohn, Robert Frederking, Tom Mitchell and Robert Murphy for their wise counsel. I would also like to thank all my colleagues at SAILING Lab and Hinman Lab, especially Le Song, Mladen Kolar, Suyash Shringarpure, Amr Ahmed, Andre Martins, Seyoung Kim, Hetunandan K, Kriti Puniyani, Kyung Ah Sohn, Geir Kjetil Sandve, Selen Uguroglu, Wenjie Fu, Henry Lin, Charlotte Jennings, Stephanie Hughes, Kristen Yankura, Alys Cheatle and Brenna McCauley, all of whose collaborative input was invaluable.

Many conversations with friends and colleagues at Carnegie Mellon and the University of Pittsburgh shaped my Ph D in many subtle ways : I am especially thankful to Andreas Zollmann, Vijaylaxmi Manoharan, Andrew Schlaijker, Oznur Tastan, Pinar Donmez, Jason Ernst, Narges Razavian, Lingyun Gu, Chun Jin, Hideki Shima, Sourish Choudhuri, Anindita Dutta, Debdutta Roy, and Mohit Kumar.

More generally, I would like to thank all my past and present mentors and collaborators : especially my undergraduate research advisor at Jadavpur University (Chandan Mazumdar) , and my mentors at the Indian Institute of Technology Kharagpur (Sudeshna Sarkar and Anupam Basu). Monojit Choudhury introduced me to Computational Linguistics and Computational Biology : fascinating fields which have held me in their thrall ever since : without your influence, I wonder whether I would be working on these fields . I would like to extend special thanks to my present mentor Michael Zhang for his great guidance : your support has meant a lot to me. I also extend sincere thanks to Michelle Martin, Monica Hopes, Diane Stidle, Radha Rao, Linda Hager, Stacey Young, Thom Gulish, Al Scheuring and the entire staff of Language Technologies Institute, Machine Learning Department, Lane Center and Biological Sciences who keep the departments' wheels rolling : thank you for helping me iron out red tape on innumerable occasions.

I would like to thank my friends : who have with a lot of love and good humour and care, put up with my singularities and contrarian nature. It is hard for me to enumerate the names of all my well wishers, but I would like to thank friends made during my time in Pittsburgh, especially Melaine Furman, Rahul Parikh, Kusum Parikh and folks from the Carnegie Mellon Quiz Club. My thanks also go to all my friends and folks who have opened their doors (and kitchens) to me during deadlines, conference travel, relocation and furlough, and made me feel welcome and loved : there are so many of you but to name some would be : Arindam Mallik, Ambarish Dutta, Hirok Banerjee, Matt Gulish, Sandeep Bhadra, Shamik Dattagupta, Sunandan Chakraborty, Suranjan Chakraborty, Rajdeep Sensharma, Prateek Shah, Sayan Dey, Ruchira and Swarnangsu, Anindita and Saugata, Atrayee and Jaydeep, Sayani and Anindya, Shibalee and Rudra, Tanni and Santanu, Shohini and Shubhagata, Kamalini and Souvik, Kajal and Soumya, Ranjna and Nilanjan, Saswati and Kaustav, Paroma and Ritwik, Payoshni and Ashok, Mrittika and Somjeet, Paramita and Anirban, Sucharita

and Sanjay, Parika and Susmit, Pritha and Vikram, and of course, Shubho.

Finally, I would like to thank people I have come to think of as family. I would like to thank my parents Sikha Ray and Prasad Ray : who gave up many opportunities in their own lives so that I could have them in mine : I am deeply grateful for their continuing support . My mother's pursuit of her doctorate under great duress and hardship and my father's unbelievable work ethic has provided a lot of inspiration to me during difficult times. I would also like to thank my extended family in the United States : Ranajoy, Stephen, Sumana, Saswato, and my two little nieces : Aruna and Ushoshi. My thanks go out to my first adoptive family in the United States : my housemates Ankur Mukherjee, Shaswati Mukherjee, Samsiddhi Bhattacharjee, and Suman Bhattacharyya : you helped me learn how to manage my time, how to drive, and how to cook : without your help I am not half the person I am. Last but not the least, I have to thank Dipanjan Das, Rohini Chaki, Sourav Bhattacharya, Sudarshana Bhattacharya, and recent additions Tathagata Dasgupta and Soumitree Dasgupta : you have been my home away from home, people I share my little joys with : I would not be here without you today.

Chapter 1

Introduction

1.1 The regulatory genome and *cis*-regulatory mechanisms

Diversity in forms of animal life, diversity in function of an organism's cells, and diversity in function of a single cell over its lifetime or in response to different stimuli drives a whole plethora of processes in biology. Questions about genomic processes underlying such diversity and their resulting biological role have been well studied in the past few decades, and their contribution lies in the gene regulatory mechanisms encoded in the DNA. The gene control circuitry actually dictates whether or not, and when and by how much a particular gene should be expressed in a cell. Such control mechanisms are most often present in the genome in the form of *cis*-regulatory modules (CRMs). These are genomic regions in the neighborhood of each gene which contain sequence "motif"s (particular genetic subsequences which are noisy copies of each other) where proteins that regulate its expression bind, causing up-regulation or down-regulation of expression levels [35].

The regulatory genome therefore consists of such regulatory regions, as well as the coding regions of the proteins (known as transcription factors (TFs)) performing the regulation, making up the *cis* and *trans* components of the regulatory mechanism respectively. The entire regulatory process is often conveniently viewed in terms of a regulatory network, with directed edges from transcription factors to regulated genes [35].

Detection of regulatory regions and motifs require significant computational analysis. Unlike gene coding sequences, there are no high-throughput, accurate biological experiments for determining the location of these motifs. The explosion of genomic and population genetic data in the 2000s have led to new avenues of exploring and understanding regulatory mechanisms in organisms at the genomic level. Detection of the exact location of the transcription factor binding sites (TFBSs) in the organism's genome, analyzing their evolutionary dynamics, and reconstructing the underlying regulatory networks are among the biggest problems which are required to understand their function in development, cell differentiation and other critical biological processes. The large amounts of genomic (and often corresponding experimental) data involved, and the complexity of the resulting analysis lends itself well to a machine learning setting. There is presently a large and growing body of work in this area [15, 194].

The rest of this document is organized as follows: in the remaining part of this chapter, we look at contemporary approaches to computational analyses of the *cis*-regulatory mechanism and the goals of this thesis. The next three chapters analyze in detail our work so far on analyzing *cis*-regulatory mechanisms.

The second chapter looks at generalized hierarchical hidden Markov models for capturing the intrinsic organization and grammar of *cis*-regulatory regions and predicting binding site locations in a supervised setting [107].

The third chapter explores a graphical model for modelling an evolutionary phenomenon called functional turnover, and predicting binding sites in a supervised fashion from multiple aligned genomes [149].

The fourth chapter outlines a conditional random field based discriminative model for supervised binding site prediction and analyzing what genomic and epigenomic feature correlate well with binding site locations [57].

The fifth chapter outlines the evolutionary analysis of regulatory regions, exploring a way to cluster together co-evolving parts of regulatory regions and then outlining a bag-of-words model for analyzing how a mixture of stochastic dictionaries evolve across species (preliminary work in [150]).

The sixth chapter is the concluding chapter which summarizes the chronological changes in the field of motif analysis and *cis*-regulatory analysis, and of the changing nature of motif analysis in the light of Chromatin Immunoprecipitation technologies, and the work presented in this thesis in the context of the literature.

1.2 Contemporary approaches to computational *cis*-regulatory analysis

Prediction of the location of TFBS and CRMs in the vicinity of coding regions is typically modelled as a classification problem with each nucleotide in the sequences of interest being assigned a class label denoting its functionality (like binding site, background nucleotide, etc) either by unsupervised methods without training data [7], or by supervised methods using training data where some instances of TFBS and CRMs are already known from the outcome of biological experiments.

Computational models of the TFBS of a single TF have existed for many years, with the most effective and common model being the position weight matrix (PWM), which was introduced more than 20 years ago [182].The popularity of the PWM is associated with the simplicity of the model, associating a TFBS with an ordered set of multinomials of A, T, G and C. More intricate models for TFBS have been suggested like modelling position with Markov models rather than independent multinomials, and using Dirichlet priors rather than directly estimating the multinomials, in a Bayesian setting [8, 204]. Over the past decade, the focus has been on predicting CRMs comprising several binding sites, typically of a few different TFs, as opposed to predicting binding sites for one single TF.

The major challenge in supervised TFBS prediction is in the fact that TFBSs are noisy copies of each other, but can easily confound an inference algorithm into predicting more false positives than

true positives. This is due to the fact that nucleotide k-mer distributions are often skewed inside regulatory regions based on the fact that selection is at work, and other evolutionary events like duplication, while suppressed are not entirely absent [72]. In terms of the underlying classification problem, typical precision and recall values for TFBS prediction in eukaryotic genomes is in the range of 0.1 to 0.4, with the typical precision - recall tradeoff seen in P-R curves [107]. This is surprisingly low given that concerted efforts have been made to model such problems from as early on as 1984 [182].

One early line of approach for CRM and TFBS discovery is a window-based approach, by simply counting the number of matches above a certain predefined match score for a particular motif pattern inside a window of fixed length in the DNA [41, 148, 151, 168]. Such methods are akin to hypothesis testing over the length of the window to determine positions of TFBSs. However, the length of the window, and the threshold for the match score are parameters which are typically decided on an ad-hoc basis and are not model-based, and are thus difficult to set in a new dataset.

A second line of methods takes an entirely different approach by modeling the occurrences of motifs and CRMs as the output of a first-order hidden Markov process on the genomic sequence [54, 55, 74, 175, 193, 213]. This approach does not suffer from the problem of having to set arbitrary window sizes and thresholds, and enjoys the rigorous guarantees of estimation and inference using Hidden Markov Models (HMMs) and its variants. It takes into account not only the strengths of motif matches, but also the spatial distances between matches (arguably more informative than co-occurrence within a window). The HMM translates to a set of soft specifications of the expected CRM length and the inter-CRM distance (i.e., in terms of geometric distributions). However, such generative models typically train parameters by maximizing the joint likelihood over the observed sequence and the hidden labels, and are thus often misled by noise in the data [57]. Further, the law of diminishing returns is in effect on using Markov Models for supervised learning of location of CRMs and TFBSs. Sophisticated models like BayCis (usng a generalized hierarchical HMM)[107] obtain only mild performance gains over the state of the art.

Additional sources of evidence need to be combined besides the regulatory sequences themselves to help improve performance, but generative models do not lend themselves well to integrating various kinds of evidence. Typically, it is not always intuitive how to generate values for new kinds of evidence, especially continuous valued ones, and even when additional evidence can be combined into the model, like evolutionary data, they typically result in an exponential increase in the state space of the model [172] (number of class labels), causing the performance of estimation and inference algorithms to go down drastically.

However, the most commonly integrated source of additional evidence into generative models for predicting regulatory function is to use multiple aligned genomes of related species. Comparative genomic methods for CRM prediction started with the Loots *et al* paper [111]. However, these approaches are restricted to very closely related organisms, because for evolutionarily distant organisms, not only are the non-coding regions hard to align, but the assumption that the aligned sequences are orthologous is also often not substantiated for small and typically degenerate functional elements such as motifs and CRMs. A number of recent investigations have shown that TFBS loss and gain are fairly common events during genome evolution [113, 129]. For example,

Patel et al [112] showed that aligned "motif sites" in orthologous CRMs in the *Drosophila* clade may have varying functionality in different taxa. Such cases usually occur in regions with reduced evolutionary constraints, such as regions where motifs are abundant, or near a duplication event. The sequence dissimilarities of CRMs across taxa include indel events in the spacers, as well as gains and losses of binding sites for TFs. Nevertheless, the fact that sequence similarity is absent does not necessarily mean that the overall functional effect of the CRM as a whole is vastly different. In fact, for the *Drosophila* clade, despite the substantial sequence dissimilarity in gap-gene CRMs such as *eve2*, the expression of these gap genes shows similar spatio-temporal stripe patterns across the taxa [112, 113]. Orthology-based motif detection methods developed so far are mainly based on nucleotide-level conservation. Some of the methods do not resort to a formal evolutionary model [15], but are guided by either empirical conservation measures [16, 42, 160], such as parsimonious substitution events or window-based nucleotide identity, or by empirical likelihood functions not explicitly modeling sequence evolution [10, 89, 199]. The advantage of these non-phylogeny based methods lies in the simplicity of their design, and their non-reliance on strong evolutionary assumptions. However, since they do not correspond to explicit evolutionary models, their utility is restricted to purely pattern search, and not for analytical tasks such as ancestral inference or evolutionary parameter estimation. Some of these methods employ specialized heuristic search algorithms that are difficult to scale up to multiple species, or generalize to aligned sequences with high divergence. Phylogenetic methods such as EMnEM [127], MONKEY [128], and CSMET [149] employ rigorous model based techniques for inferring the position of CRMs and TFBSs.

Discriminative models have also been used for predicting regulatory function. Craven *et al* [17] first applied such a scheme to identify regulatory signals in prokaryotic sequences; but their model employs a simple feature set to resolve the motif sequence overlap problem, and also requires a prescreening of motif scores via basic PWM-based models. Discriminative models explicitly tailor towards maximizing the likelihood of predicting motifs, rather than maximizing the joint likelihood - which often confounds the analysis in the case of generative models. Secondly, discriminative frameworks like Conditional Random Fields [57] employ a comprehensive set of features carefully selected from the literature designed to capture a variety of characteristics of the motif and CRM patterns. An additional goal of such models is to empirically test the correlation of various types of genetic and epigenetic features with binding site locations.

In the past few years, advances on the side of biological experimentation have provided two methods of detecting protein-DNA binding events *in vivo* on a high throughput basis. The first experiments to expressly investigate transcription factor - DNA binding was ChIP-Chip, which combined chromatin immunoprecipitation with DNA microarray technology to identify which parts of the genome are bound by transcription factors. [83, 105, 153]. However, ChIP-Chip suffers from noise inherent in the microarray chip read-out, and has in the past few years been primarily replaced with ChIP-Seq technology, which combines chromatin immunoprecipitation with next-generation sequencing technology to identify the areas of the genome where the transcription factor binds *in vivo*. This is less noisy than ChIP-Chip methods, as sequencing techniques are inherently less noisy than DNA microarray technology [85, 156]. However, the output of ChIP-Seq experiments cannot directly be translated into positions of TFBSs. The DNA subsequences to which the TF

binds (known as "tag"s), and which are consequently sequenced, are of magnitude 20 bp - 200 bp, whereas typically the exact length of a TFBS is typically 6 - 20 bp long.

Finally, a lot of work has been done in the recent past analyzing the evolutionary trends in regulatory regions. Selection is a driving force of evolution, and is well known to constrain functionally important regions of the genome, causing negative selection [93]. Positive selection, on the contrary, is defined to be faster than normal rates of nucleotide evolution which can affect the functionality of coding or regulatory regions. Recently, positive selection has been reported in regulatory regions, using population genetics and phylogenetic tests based on indel events and substitutions in binding sites versus nonbinding sites [75, 157, 158]. A recent spurt of work in detecting selection and other evolutionary parameters in eukaryotes has coincided with the appearance of a large volume of comparative genomic data, including the 12-way *Drosophila* and other insect multiple alignment, and the 30-way mammalian multiple alignment. These works include the analysis of evolutionary signatures for different functional annotations on the *Drosophila* clade [189], analysis of selection in conserved non coding sequences in humans,chimpanzee, mouse, rat and dog by Kim and Pritchard [92], a study of adaptive substitutions in *Drosophila* [102], a study of selection in TFBS inside repeats in humans [144], and a genome-wide analysis of selection on human *cis*-elements [166].

1.3 An overview of the thesis

One goal of this thesis is to explore supervised motif detection in regulatory sequences by maximally utilizing the inherent "grammar" or structure of the *cis*-regulatory modules. We achieve this goal by using hierarchical and generalized Hidden Markov Models (HMMs) in a Bayesian setting. Hierarchical HMMs help capture correlation among binding sites should they exist, as well as being able to model flanking regions specific to different kinds of binding sites, Generalized HMMs help model spacer distances between motifs and a bayesian framework ensures that whatever prior knowledge we have about the architecture (possible correlations of types of binding sites, etc) can be incorporated into the model by using priors on the parameters. The work is presented in detail in Chapter 2 [107].

Another goal of this thesis is to explore supervised motif detection by using comparative genomic data (multiple sequence alignment), with a specific focus of taking into account the phenomenon of functional turnover. Functional turnover is a phenomenon where orthologous sequences across even closely related species may have varying functionality due to gain or loss in functionality in the specific subsequence in question. Functional turnover is one of the biggest confounding factors plaguing comparative genomic analyses, and we developed a generative graphical model which models the multiple sequence alignment as the output of a mixture of phylogenies. The mixture variables themselves are not drawn from a simple Bernoulli distribution [127], but are themselves the product of a higher level phylogenetic tree modelling the evolution of binary function indicators. The work is presented in detail in Chapter 3 [149].

A third goal of this thesis is to analyze diverse sources of evidence and conclude which genetic and epigenetic features correlate well with binding site locations, and to use such information to create a discriminative model for supervised prediction of binding sites. We use the discrimina-

tive framework of a conditional random field (CRF) for the purpose, which assigns weights to each genetic or epigenetic feature or "score". Evolutionary features, annotation of transcribed and translated regions, features like GC content related to chromatin stability, as well as epigenetic features like nucleosome binding affinity were explored. The work is presented in detail in Chapter 4 [57]. DISCOVER aims to be the standard tool for integrative analysis based on Conditional Random Fields, with the ability to integrate differing datasets like epigenetic marks, transcription factor binding, genomic information, and evolutionary context.

Obtaining a deeper understanding of the evolution of the regulatory genome is crucial to be able to model generative processes which account for evolution of regulatory regions like CSMET [149], and EMnEM [127], as well as for analyzing what kinds of evolutionary features may prove discriminative with respect to motif-finding in discriminative models like DISCOVER [57]. A final goal of this thesis is to model the evolutionary dynamics of regulatory regions. We modelled co-evolving regions inside cis-regulatory modules by analyzing and spectral clustering evolutionary parameters in different parts of regulatory regions. Another goal was analyzing selectional forces in the regulatory genome by identifying which k-mers are preferentially present in regulatory regions across species by modelling regulatory regions as evolving mixtures of stochastic dictionaries. We explored the predictive ability of the mixture components in our stochastic dictionaries, as well as understanding how we can track the evolution of such stochastic dictionaries across species. This work is presented in detail in Chapter 5, with preliminary work having been presented in [150].

This thesis provides novel statistical frameworks for identifying regulatory regions, and analyzing them in terms of their architecture, function, evolutionary properties and correlation with other genomic and epigenomic features in a computationally optimal and statistically sound way.

Chapter 2

Variations on Markov Models for cis-regulatory analysis

2.1 Related work

Motif models of binding sites for a single transcription factor have existed for many years, currently the most common model being the position weight matrix (PWM) introduced more than twenty years ago [182]. In recent years, focus has shifted from predicting binding sites for a single TF towards predicting CRMs comprising several binding sites, often for several distinct TFs. Several models have been proposed, making use of certain architectural features of the CRMs. Some of these models apply comparative genomic methods for CRM prediction [111, 127, 171, 174]. These approaches are, however, restricted to very closely related organisms, because non-coding sequences are hard to align and more subject to events like duplication and shuffling which make orthology prediction difficult. A large number of CRM and motif prediction algorithms, including the one we propose thus rely on single species data. One line of methods for the discovery of CRMs count the number of matches (of some minimal strength) to given motif patterns within a certain window of DNA sequence [41, 148, 151, 168] From a modeling point of view, this family of algorithms assumes that motifs are uniformly and independently distributed within each window; an *ad hoc* window size needs to be specified, and careful statistical analysis of matching strength is required to determine a good cutoff or scoring scheme [81, 168]. Rajewsky *et al.* addressed the issue of compensating the matching scores for co-occurring weak motif sites using an updatable "word-frequency" measure, which leads to higher scores for motifs co-occurring more frequently within a window of a given size. This algorithm also contains an important extension for unsupervised CRM prediction, in which representations of novel motifs are estimated directly from the input DNA sequences. However, under a modular formulation of the CRM prediction problem (cf. the LOGOS model [207]), the prediction of motif instances from given representations, and the estimation of motif representations from predicted instances, can be treated as two orthogonal sub-problems. These sub-problems may be solved separately and coupled as two components of a higher-level joint model, with estimates exchanged in an iterative fashion. We only focus on the aspect of CRM prediction given motif representations. [148].

Another approach to the problem involves modelling the occurrences of motifs and CRMs as the output of a first-order hidden Markov process. This approach alleviates the necessity of both the window size and the score cutoff, and takes into account not only the strengths of motif matches, but also the spatial distances between matches (arguably more informative than co-occurrence within a window). The hidden Markov model (HMM) translates to a set of soft specifications of the expected CRM length and the inter-CRM distance (i.e., in terms of geometric distributions). However, since training data for fitting the HMM parameters hardly exist, these parameters typically have to be specified based on empirical guesses. HMMs and similar models that captures binding site distributions, as well as intra-CRM and inter-CRM backgrounds, have been used in several CRM discovery methods, e.g. in Cister [54], Cluster-Buster [55], CisModule [213] and EMCModule [74]. As these methods employ a general inter-motif background, they do not infer any ordering between motifs. This model is extended to include distinct motif-to-motif transition probabilities in the methods Stubb [175] and Module Sampler [193].

2.2 The formal model

To model the complex architecture of metazoan transcriptional regulatory sequences (TRS), we propose to use a *hierarchical hidden Markov model* (hHMM) that can encode a set of stochastic syntactic rules presumably underlying the CRM organizations and motif dependencies. A first-order Markov process over a hierarchy of states allows us to describe the structure of regulatory regions at different levels of granularity, offering more modeling power than existing methods.

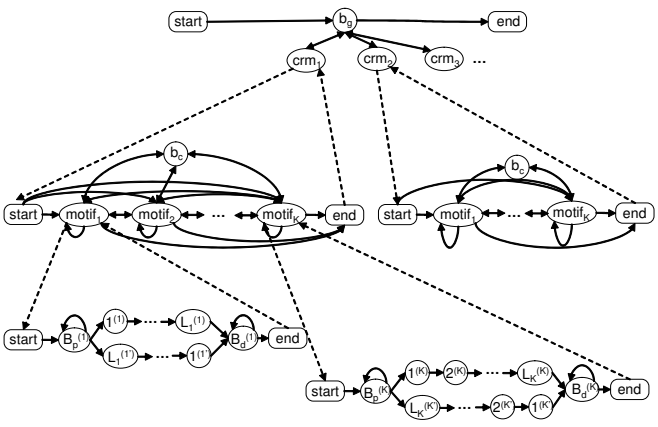


Figure 2.1: The BayCis hHMM state transition diagram with 3-level hierarchy. Circular nodes represent functional states in DNA sequences, and round boxes represent start and end states in each sub-model. CRM and motif states are sub-models invoked by higher level models. Arrows between nodes represent permissible state transtions, including horizontal transitions denoted as black arrows, and verticle transitions denoted as dashed arrows.

2.2.1 A Hierarchical HMM of TRS

As first proposed in [51], the hHMM is an extension of the classical HMM for modeling domains with hierarchical structures. In an hHMM, all hidden states are not equal, but follow a hierarchical organization that constrains stochastic transitions among states—transitions are only permissible for (certain pairs of) states at the same level or adjacent levels in the hierarchy; different states can emit either single observations or strings of observations, depending on their position in the state hierarchy; and the strings emitted from the non-leaf states in the hierarchy are themselves governed by a sub-hHMM (or more generally, by an arbitrary generative model, which would further extend the overall model beyond an hHMM).

An hHMM can explicitly capture nested generative structures (e.g., TRS $\rightarrow$ CRM $\rightarrow$ Motif $\rightarrow$ Single Nucleotide Site) underlying complex sequential data, and dependencies among elements at different levels of granularity (e.g., motif versus motif, site versus site, etc.), which makes it a powerful and natural approach to model genomic regions harboring transcriptional regulatory sequences. Figure 2.1 shows an example of an hHMM encoding typical hierarchical structures of the metazoan TRSs we are concerned with in this study. At the top (i.e., coarsest) level, this hHMM represents a TRS as a concatenation of long stretches of sequences corresponding to global backgrounds and CRMs. We can think of this top level as an HMM whose states emit whole CRMs and inter-CRM (global) background sequences. Formally, we let $\mathbb{Q}^1 \equiv \{b_g, c_1, c_2, \ldots, c_I\}$ denote the set of these possible states. At the next level, each CRM is represented as a sequence of motifs and intra-CRM (local) background states. Accordingly we have $\mathbb{Q}^2 \equiv \{b_c, m_1, m_2, \ldots, m_K\}$. At a finer level below, each motif is represented as a sequence of buffer states and nucleotide sites. (We will explain shortly why we include non-motif buffer states at this level.) Accordingly, we define $\mathbb{Q}^3 \equiv \mathbb{B} \cup (\cup_i \mathbb{M}_i)$, where $\mathbb{B}$ corresponds to the non-motif buffer states padding right before and after the motif sequences and $\mathbb{M}_i$ corresponds to all possible sites within motif i. More specifically, we define: $\mathbb{M}_i \equiv \mathbb{M}_i^f \cup \mathbb{M}_i^r$, where $\mathbb{M}_i^f = \{1^{(i)} \ldots L_i^{(i)}\}$ is the set of all possible sites within motif i on the forward DNA strand, and $\mathbb{M}_i^r$ is the set of all possible sites within motif i if it is on the reverse complementary DNA strand.; $\mathbb{B} \equiv \mathbb{B}^p \cup \mathbb{B}^d$, where $\mathbb{B}^p = \{b_p^{(1)}, \ldots, b_p^{(K)}\}$ denotes the set of *proximal-buffer* states associated with each type of motif [1], and $\mathbb{B}^d = \{b_d^{(1)}, \ldots, b_d^{(K)}\}$ denotes the set of *distal-buffer* states associated with each type of motif.

The possible transitions between these states are made explicit by the arrows in the hierarchical state diagram in Figure 2.1. (Note that to make the hHMM model well-defined, we also introduce *dummy* states `START` and `END` at appropriate levels to enable instantiation of state-traversal, and proper termination of subsequences at each level.) The biological motivation for such a state hierarchy is that we expect to see occasional motif clusters in a large ocean of global background sequences (represented by state b_g); each motif instance in a cluster is like an island in a sea of intra-cluster background sequences (b_c); and adjacent motifs may be statistically coupled (we will elaborate on this point in the next section). Our model assumes that the distance between clusters is geometrically distributed with mean $1/(1-\beta_{g,g})$, and the span of the intra-cluster background is also geometrically distributed with mean $1/(1-\beta_{c,c})$. These modeling choices are intended to

[1] Here, proximal-buffer refers to the background sites immediately next to the proximal-end of the motif. For consistency, orientations are defined with respect to the initial position of the input sequence. That is, the 1st position of the input sequence corresponds to the proximal end, and the last position corresponds to the distal end.

not only reflect our uncertainty about the CRM structure, but also to offer substantial flexibility to accommodate potential 1st-order syntactic characteristics within the CRMs. In this hHMM, only the bottom-level motif-site and motif-buffer states, as well as the global and local background states, are capable of emitting individual nucleotides constituting the TRS, according to a stochastic emission model (which we will elaborate in §2.1.3). A stochastic traversal of the hHMM states according to the hHMM state-transition diagram would generate a TRS of arbitrary length but with a structure consistent with our empirical knowledge of the functional organization of the metazoan TRS. Note that this hHMM model does not impose rigid constrains on the number of motif instances or modules; the actual number of instances is determined by the posterior distribution of the hHMM states given the observed sequence. Also note that we have not included functional states related to gene annotation and basic promoters, but such extensions are straightforward if co-identification of CRMs and genes is desired.

Given the observed sequences, and proper (i.e., biologically meaningful) construction of the state space and its hierarchical organization, one can infer the latent state-traversal path, which correspond to a plausible annotation or segmentation of the input sequence, using a number of exact posterior inference algorithms. The original algorithms given by [51] is a variant of the inside-outside algorithm for stochastic context free grammar, and takes $O(T^3Q^D)$, where T is the length of the sequence, Q is the total number of states, and D is the depth of the hierarchy. A linear time algorithm was developed by [130] based on a transformation of hHMM into an equivalent dynamic Bayesian network. It is also possible to flatten the hHMM to an HMM with a block-structured sparse transition, and use a modified forward-backward algorithm for linear-time inference. We use a Bayesian extension of hHMM.

Motif bigram via hHMM

An hHMM not only encodes hierarchical segmental structures in a sequence, but it can also be used to capture dependencies between sequence elements at different levels of granularity at a cost much smaller than that would be needed by a "flat" Markovian model which must resort to heavily parameterized high-order conditional probabilities. For example, we can capture the dependencies between neighboring CRMs in a TRS by modeling transitions between the CRM states. Of particular importance here, we use hHMM to capture the dependencies between occurrences of motifs within a CRM. As discussed earlier, the spatial arrangement of motifs within a CRM may encode intricate combinatorial transcriptional regulatory signal. Thus modeling at least 1st-order dependencies between motifs may be beneficial to the unraveling of motifs in long TRS bearing complex regulatory function, as well-known in the case of *Drosophila* enhancers. Note that a direct transition between trivially defined motif states (e.g., last site of motif i and first site of motif j) would suggest that coupled motifs always occur right next to each other, which is biologically not always true. To capture possible dependencies between motifs in the vicinity of each other, we define the emission of a motif state (in $\mathbb{Q}^2$) to contain not only the motif sequence itself, but also non-motif sequences denoted as proximal and distal buffers. Such an emission can be understood as an extended instance of a motif, which we referred to as a *motif envelope*. Thus cross-background (i.e., high-order) dependencies between motifs can be captured by immediate (i.e., 1st-order) dependencies between the motif envelopes.

We write $A_2 \equiv \{a_{i,j}\}$ as the stochastic matrix for transitions among states in $\mathbb{Q}^2$, which defines a *bigram* of motifs (and their local backgrounds) within CRMs. The length of the proximal and distal buffers of a motif is geometrically distributed with mean $1/(1-\alpha_{i,i})$ and $1/(1-\beta_{i,i})$, and can be generated via self-transitions of the corresponding states at the third level (i.e., in $\mathbb{Q}^3$) with probability $\alpha_{i,i}$ and $\beta_{i,i}$, respectively. Then with equal probability $\alpha_{i,m}/2$, a proximal buffer state $b_p^{(i)}$ reaches the start states $1^{(i)}$ (resp. $L_i^{(i')}$) of motif i on the forward (resp. reverse) strand, deterministically passes through all internal sites of motif i, and transitions to the distal-buffer state $b_d^{(i)}$, thereby stochastically generating a non-empty motif envelope [2]. Each b_d^i has probability $\beta_{i,j}$ of transitioning to the proximal-buffer state of another motif j (or of the same motif when $j = i$) to concatenate another motif envelope, or it may choose to pad with some inter-cluster background before adding more envelopes, with probability $\beta_{i,c}$. All distal-buffer states also have probability $\beta_{i,g}$ of returning to the global background, terminating a CRM.

Spacer length distribution via GhHMM

A *spacer* is the interval seperating adjacent motif instances, modeled as b_c, b_p, and b_d states in BayCis. It has been suggested that the range of spacer length is under selection forces according to comparative genomics data of several *Drosophila* species [114]. Empirically, we found that the distribution of spacer lengths can be approximated by a negative binomial distribution, whereas under an hHMM, the state durations of cluster backgrounds is distributed as a goemetric distribution, which is not a good approximation of the space length distribution. Our generalized hierarchical hidden Markov model (GhHMM) implements an approximate negative binomial distribution of spacer lengths by joining several geometrically distributed cluster background states.

The emission models: PWM and higher-order Markov background

Once the hHMM enters the motif-site states, we resort to a *motif model* to generate the nucleotides at the corresponding sites. To maintain our focus on the hHMM and relevant algorithmic issues, we only consider the scenario of searching for known motifs here (although extending our model for *de novo* motif detection is straightforward based on, for example, the LOGOS framework [207]). For motif model we choose the classical product-multinomial (PM) model, which can be represented by a PWM [182].

Several previous studies have stressed the importance of using a richer background model for the non-motif sequences [109, 192]. In accordance with these results, BayCis uses a standard global k-th order Markov model for the emission probability of the global background state. For the intra-CRM states, we used locally estimated Markov models. Since the models are defined to be *local*, the conditional probability of a nucleotide at position t is now estimated only from a window of length $2d$ centered at t. These probabilities can still be computed off-line and stored for subsequent uses, by using a careful bookkeeping scheme (i.e., using a "sliding-window" to

[2]Note that the distinction between the proximal and distal buffers avoids generating empty envelops (because otherwise, a single buffer state would not be able to remember whether a motif has been generated beyond k positions prior to the current position under a k-th order Markov model.)

compute the local Markov model of each successive position, each with a constant "update cost" based on the previous one).

2.2.2 Bayesian hHMM

One caveat of the standard HMM approach for CRM modeling is the difficulty of fitting the model parameters, such as the state-transition probabilities, due to rarity of fully annotated CRM-bearing genomic sequences. In principle, using the Baum-Welsh algorithm one can learn the maximal-likelihood (ML) estimates of the model parameters directly from the unannotated sequences while analyzing them. In practice, however, such a completely likelihood-driven approach tends to result in spurious results, such as over-estimation of the motif and CRM frequencies and poor stringency of the learned models for potential motif patterns. Previous methods tried to overcome this by reducing the number of parameters needed as much as possible, and by setting them according to some good guesses of the motif/CRM frequencies or CRM sizes [54]. But as a result, such remedies compromise the expression power of the already simple HMM, and risk mis-representing the actual CRM structures. In the following, we propose a Bayesian approach that introduces the desired "soft constraints" and smoothing effect for an HMM of rich parameterization, using only a small number of *hyper-parameters*. Essentially, this approach defines a posterior probability distribution of all possible value-assignments of the HMM parameters, given the observed un-annotated sequences and empirical prior distributions of the parameters that reflect general knowledge of CRM structures. The resulting model allows probabilistic queries (i.e., estimating the probability of a functional state) to be answered based on the aforementioned posterior distribution rather than on fixed given values of the HMM parameters.

We assume that the self-transition probability of the global background state $\beta_{g,g}$, and the total probability mass of transitioning into a motif-buffer state $\sum_{k\in\mathbb{B}^p}\beta_{g,k}$ (note that $\beta_{g,g} = 1 - \sum_{k\in\mathbb{B}^p}\beta_{g,k}$), admit a beta distribution, $Beta(\xi_{g,1},\xi_{g,2})$. We choose a small value for $\frac{\xi_{g,2}}{\xi_{g,1}+\xi_{g,2}}$, corresponding to a prior expectation of a low CRM frequency. Similarly, we define a beta prior $Beta(\xi_{c,1},\xi_{c,2})$ for the self- and total motif-buffer-going transition probabilities $[\beta_{c,c},\ \sum_{k\in\mathbb{B}^p}\beta_{c,k}]$ associated with the intra-cluster background state; and another beta prior $Beta(\xi_{p,1},\xi_{p,2})$ for the self- and motif-going transition probabilities $[\alpha_{i,i},\ \alpha_{i,m}]$ associated with the proximal-buffer state of a motif. Finally, we assume that for the distal-buffer state, the self-transition probability, the total mass of transition probabilities into a proximal-buffer state, the probability of transitioning into the intra-cluster background, and the probability of transitioning into the global background, $[\beta_{i,i},\ \sum_{k\in\mathbb{B}^p}\beta_{i,k},\ \beta_{i,c},\ \beta_{i,g}]$, admit a 4-dimensional gamma distribution, $Gamma(\xi_{d,1},\xi_{d,2},\xi_{d,3},\xi_{d,4})$.

To define priors for the GhHMM parameters, the GhHMM with a single cluster background state (b_c) is considered as an HMM with several cluster background states ($\{b_c^1,\cdots,b_c^{g_{cr}}\}$) sharing the same self-transition probability $\beta_{c,c}$. Similar to other background states, we define a beta prior $Beta(\xi_{c,1},\xi_{c,2})$ on the total probability mass of transitions into motif-buffer states $\sum_{k\in\mathbb{B}^p}\beta_{c,k}$ (note that $\beta_{c,c} = \sum_{k\in\mathbb{B}^p}\beta_{c,k}$).

Note that due to conjugacy between the prior distributions described above and the corresponding transition probabilities they model, the hyper-parameters of the above prior distributions can be understood as *pseudo-counts* of the corresponding transitioning events, which can be

roughly specified according to empirical guesses of the motif and CRM frequencies. But unlike the standard HMM approach, of which the transition probabilities are fixed once specified, the hyper-parameters only lead to a soft enforcement of the empirical syntactic rules of CRM organization in terms of prior distributions, allowing controlled posterior update of the HMM transition probabilities while analyzing the un-annotated sequences. For the BayCis hHMM, we specify the hyper-parameters (i.e., the pseudo-counts) using estimated frequencies of the corresponding state-transition events, multiplied by a "prior strength" N, which corresponds to an imaginary "total number of events" from which the estimated frequencies are "derived". That is, for the beta priors, we let $[\xi_{[\cdot,1]}, \xi_{[\cdot,2]}] = [1 - \omega_{[\cdot]},\ \omega_{[\cdot]}] \times N$, where the "$\cdot$" in the subscript denotes either the g, c, or p state, and $\omega_{[\cdot]}$ is the corresponding frequency. For the gamma prior, we let $[\xi_{d,1}, \xi_{d,2}, \xi_{d,3}, \xi_{d,4}] = [\omega_{d,1},\ 1 - \sum_j \omega_{d,j},\ \omega_{d,2},\ \omega_{d,3}] \times N$. Overall, we need to specify 7 hyper-parameters (of course one can use different "strengths" for different priors, with a few additional parameters), a modest increase compare to, say, 3 needed in Cister [54].

2.3 Experiments

We evaluated BayCis on both synthetic transcriptional regulatory sequences and a rich set of carefully compiled real genomic TRSs of *Drosophila melanogaster* (available at our website). The prediction performance of BayCis was compared with 5 popular published methods for supervised discovery of motifs/CRMs based on a wide spectrum of models: Cister [54], Cluster-Buster [55], MSCAN [1], Ahab [148] and Stubb [175] (all of which were applied to the real data, and two seemingly superior ones to the semi-synthetic data), which cover a wide spectrum of different models/algorithms (e.g., HMMs, windows) for motif search. We ran other methods with default parameters, specifying 500 bp CRM window where needed. Overall, the prediction performance of BayCis is competitive or superior to all chosen benchmark methods on this quite comprehensive selection of data sets, according to a wide assortment of performance measures. By employing sound and flexible probabilistic modeling of regulatory regions, BayCis is also able to strike a good balance between precision and recall with its default MAP solution.

Synthetic TRSs are useful in that the ground truth for motif/CRM locations is known exactly. To generate semi-realistic synthetic TRSs, we planted selected TFBS from the Transfac [203] database in simulated background sequences according to the model assumptions underlying BayCis on the background distribution and the length distributions between CRMs and between motif instances within CRMs. Specifically, 30 sequences of length 20,000 bp were generated, each containing zero to three CRMs. The length of the CRM is uniformly distributed between 200 bp and 1600 bp, while the average motif spacer length is 50 bp. Each CRM contains 3 to 6 motif types and about 14 motif instances. To simulate motif co-occurrence, about 25% of the motif instances in each CRMs appear in predefined pairs. The background sequences inside or outside the CRM are simulated by a 3rd-order Markov model learned from an intergenic region.

As shown in Figure 2.2, the performance of BayCis using either hHMM or GhHMM is significantly better than CISTER and ClusterBuster in terms of the overall precision/recall (P/R) trade-off at the MAP prediction. The P/R curve of BayCis is also well above the default predictions from other methods. It also shows that GhHMM performs consistently better than hHMM in both preci-

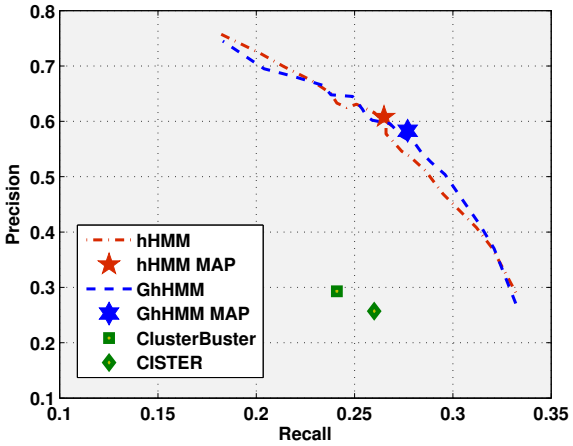


Figure 2.2: The precision-recall curves of two models of BayCis (hHMM and GhHMM) versus the P/R of default predictions by CISTER and ClusterBuster. From [107].

sion and recall, although the difference is not very large. CISTER and ClusterBuster were chosen for the simulation study based on their good performance on real data.

The synthetic TRSs are generated partially based on the same model assumptions underlying BayCis, and thus the results cannot be interpreted as conclusive. In this section we present an empirical evaluation based on a rich and carefully compiled *Drosophila* TRS dataset, although it is noteworthy that even though we have tried our best to gather the most complete annotations for each test sequence based on footprinting results from the literature, this "gold standard" is still possibly only a subset of the ground truth.

The dataset

We created a manually curated dataset containing 97 CRMs pertaining to 35 early developmental genes. This collection was compiled based on a filtering of all known CRMs from a number of public databases (e.g., the REDfly CRM database [61] and the *Drosophila cis*-regulatory Database at the National University of Singapore [132]), through which we only chose CRMs that are at least 200 bp long, and contain at least 5 experimentally confirmed motif instances (2 CRMs with a borderline count of 4 motif instances were also included). Each test sequence consists of the CRMs pertinent to a particular gene, all intra-CRM background inbetween, with flanking regions on either side of the extremally located CRMs such that the entire sequence is at least 10 kbp long, and the boundaries of the sequence are at least 2 kbp from the extremal CRMs. We included the exonic regions of the genes only when they fell in the aforementioned selected region, and not otherwise. This database is available at `http://www.sailing.cs.cmu.edu/BayCis`, where the BayCis tool is publicly available.

Experimental setup

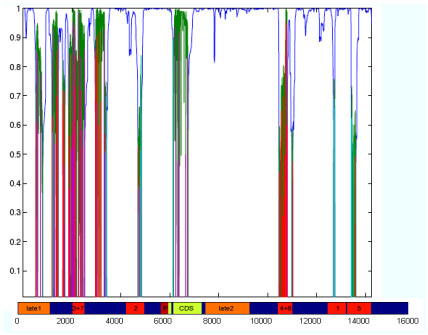
BayCis is a Bayesian framework based on hHMMs and GhHMMs to model the organization and distribution of TFBS. Prior beliefs pertaining to the parameters of the model thus could be specified by the user before running on experimental data in the form of hyperparameters (i.e., pseudocounts) of the hHMM or GhHMM parameters. The PWMs of the motifs to be searched for also need to be provided because here we are interested in identifying TFBS of existing TF motifs, rather than *de novo* motif detection. As mentioned in previous sections, extending BayCis for this function is straightforward by introducing an EM step for the PWM estimation.

Hyperparameters: The choice of hyperparameters should in principle be dealt with via an "empirical Bayes scheme", which employs maximal likelihood estimates of these hyperparameters based on some fully labeled training sequences. Upon prediction on an unannotated sequence, the hHMM or GhHMM parameters themselves can be adjusted in an unsupervised fashion via the variational EM algorithm. We specify the hyperparameters as follows: for the global background, $\omega_g = 0.002$; for the inter-module background, $\omega_c = 0.05$; for the proximal motif buffer, $\omega_p = 0.25$; for the distal buffer hyperparameters, $\omega_{d,1} = 0.125$ (distal to global background), $\omega_{d,2} = 0.125$ (distal to clustal background), and $\omega_{d,3} = 0.25$ (distal to proximal buffer). Finally, the "strength" of the hyperparameters are set to $1/10$ of the expected counts of the transitions on a 15 kbp dataset, with the exception of ω_g which is set to $10,000$. The background probability of the nucleotide at each position was computed locally using a 2nd-order Markov model from a sliding window of 1100 bp centered at the corresponding position. For the GhHMM, based on visual inspection of spacer length distributions between motifs, we choose the parameter as $r = 2$.

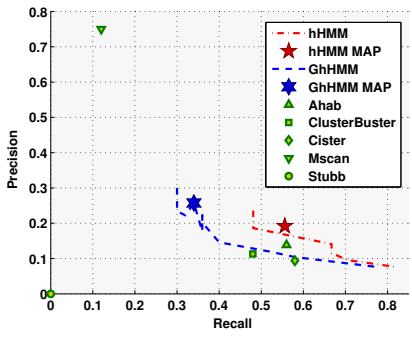
Prediction scheme: BayCis provides three kinds of prediction schemes for motifs. The *maximum a posteriori* (MAP) prediction is based on the posterior probabilities of the labeling state at each site, which allows overlapping motifs. A Viterbi prediction, which gives a consistent prediction in the Bayesian setting analogous to an ML prediction under a classical setting can also be used. A third scheme is based on a simple but effective thresholding scheme where we directly predict motifs based on whether the motif states have a higher probability than the specified threshold in the posterior probabilities. For simplicity, we only present the MAP results and the P/R curve of the threshold method. Note that unlike many other scoring schemes for motif/CRM detection, such as logodds (i.e., the PSSM score) or a likelihood score regularized by word frequencies, our MAP prediction does not require a cutoff value for the scores, nor a window to measure the local concentration of motif instances, both of which are difficult to set optimally.

Evaluation measures: There is no unanimous way of evaluating the prediction performance of a motif/CRM discovery method against annotations. To avoid reliance on a single evaluation procedure and measure, we have chosen to present the performance of BayCis in comparison with other methods using several different evaluation procedures. This also ensures a thorough and objective presentation of results. For an overall evaluation we compare the prediction performance of BayCis with other methods using both the F1-score of precision and recall, and the coefficient

of correlation (CC) score at nucleotide-level [194] as single point measures. We do this by first summing true/false positives/negatives across datasets at the nucleotide level, and then computing F1/CC from these combined counts. To present the behavior of BayCis with respect to site-level precision/recall, we plot the binding-site level P/R curve from different thresholds in extracting predictions, along with the P/R at MAP predictions.



(b)



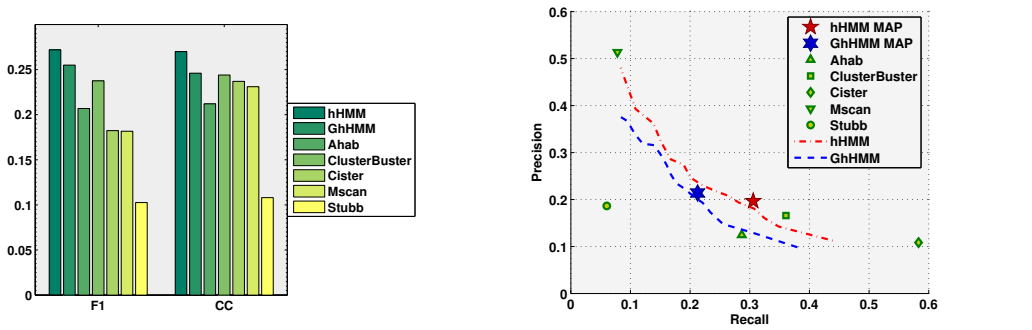
(a)

Figure 2.3: Performance of BayCis (hHMM) on a representative *eve* TRS. (a) The posterior probability plot of the global background (blue), cluster background (green) and motif specific (red and other colors) states. (b) The precision versus recall performance of the MAP and thresholded predictions of the hHMM and GhHMM algorithms, as compared to those made by other methods. From [107].

Motif prediction performance

As an illustration, Figure 2.5a shows a plot of the MAP prediction along the *even-skipped* gene TRS, under a particular hyperparameter setting. As revealed in the ground-truth annotation bar bellow the plot, this region contains 5 CRMs (from left to right): *stripe3+7*, *stripe2*, *stripe4+6*, *stripe1*, and *stripe5*. BayCis picks out all of them, although the CRM boundary appears to be more stringent in most cases. We believe this can be improved by adopting a more specialized cluster background model (i.e., local higher-Markov model, better GhHMM model, etc.), which we have not fully explored yet. BayCis also identifies motif-rich regions proximal and distal to the *stripe3+7* CRM, which is not reported before, and it also finds another putative motif-rich region spanning the core promoter and the CDS of *eve*, which can be a false positive or a putative CRM. The overall MAP prediction score of BayCis, and the P/R curves resulted from applying different threshold values under BayCis, are shown in Figure 2.5b, along with the scores of 5 other competing algorithms in their default configurations. The BayCis MAP predictions seem significantly better than other methods, and strike a good balance between recall and precision. It is important to realize that although the threshold method can reach high precision or recall at both extremes, in practice it is very hard to pick the optimal threshold without knowing the prediction results, and typically a threshold optimal for one sequence is not necessarily good for another sequence; significance-test based determination of threshold is also difficult for a complex model or large sequence. Thus, a default prediction such as MAP, which automatically finds an appropriate trade-off between precision and recall, is highly desirable. The overall CC and F1-scores of running BayCis

and five competing methods on the full set of *Drosophila melanogaster* sequences are shown in Figure 2.4a. According to either measure, both the hHMM and the GhHMM version of BayCis outperforms all existing methods. The hHMM version of BayCis performs slightly better overall compared to GhHMM according to both measures. For both versions of BayCis, the MAP solution was chosen. To look at the behavior of BayCis in the precision recall landscape on our entire dataset, we



(a) (b)

Figure 2.4: (a) F1 and CC scores, and (b) Precision - Recall performances of the MAP and thresholded predictions of the hHMM and GhHMM, in comparison with other algorithms on the full *Drosophila* TRS dataset. From [107].

plot the precision-recall curve resulting from different thresholds for BayCis predictions. For other methods we provide the single points in precision-recall landscape corresponding to their default output. As is apparent from Figure 2.4b, the 5 competing methods strike different balances between precision and recall in their default output. MSCAN focuses on very high precision predictions, while Cister is geared towards high values of recall. The precision-recall curves of both versions of BayCis span a balanced range in the precision-recall landscape, with MAP estimates lying in the middle of the curves. Again, in practice the precision and recall values are not available for use by methods, so the balance between precision and recall has to be found based solely on the input data. Thus the ability to appropriately balance the precision and recall automatically is essential. To further investigate the prediction performance, we look at the variation of individual dataset prediction performance across all datasets. The boxplot in figure 2.5 shows the median CC-score for each method, as well as upper and lower quartiles and minimum/maximum values. We see that prediction scores varies much between datasets for all methods, and that the overall performance differences between methods is not very large compared to the variation of individual methods across datasets. This confirms what has long been acknowledged in the motif discovery field, that even the best performing methods will in many cases give misleading predictions (although some of the low scores may be due to lack of annotations). Among the high scoring methods (hHMM, GhHMM, Cluster-Buster and Cister), GhHMM and Cister come out as the most stable with low variance across datasets, a criterion which is useful when handling a

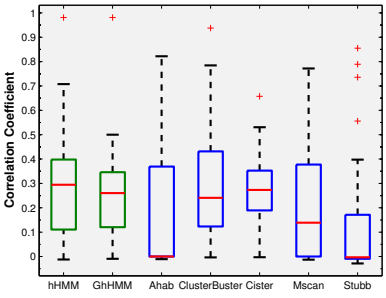


Figure 2.5: A boxplot showing variation in CC across datasets. From [107].

varied set of data. The posterior expectations of the hHMM/GhHMM parameters also carry rich architectural information of each TRS we processed, and merits systematic analyses.

Chapter 3

Modelling functional turnover in regulatory regions

3.1 Related work

Uncovering motifs in eukaryotic *cis*-regulatory modules from multiple evolutionarily related species, such as the members from the *Drosophila* clade is a natural extension of modelling regulatory regions of a single species using HMMs. Due to high degeneracy of motif instances, and complex motif organization within the CRMs, pattern-matching-based motif search in higher eukaryotes remains a difficult problem, even when representations such as the position weight matrices (PWMs) of the motifs are given. Extant methods that operate on a single genome or simpler organisms such as yeast often yield a large number of false positives, especially when the sequence to be examined spans a long region (e.g., tens of thousands of bps) beyond the basal promoters, where possible CRMs could be located. As in gene finding, having orthologous sequences from multiple evolutionarily related taxa can potentially benefit motif detection because a reasonable alignment of these sequences could enhance the contrast of sequence conservation in motifs with respect to that of the non-motif regions, However, the alignment quality of non-coding regions is usually significantly worse than that of the coding regions, so that the aligned motif sequences are not reliably orthologous. This is often unavoidable even for the best possible local alignment software because of the short lengths and weak conservation of TFBSs. When applying a standard shadowing model on such alignments, motif instances aligned with non-orthologous sequences or gaps can be hard to identify due to low overall shadowing score of the aligned sequences. In addition to the *incomplete orthology* due to imperfect alignment, a more serious concern comes from a legitimate uncertainty over the actual functional orthology of regions that are *alignment-wise* orthologous. A number of recent investigations have shown that TFBS loss and gain are fairly common events during genome evolution [113, 129]. For example, Patel et al [112] showed that aligned "motif sites" in orthologous CRMs in the *Drosophila* clade may have varying functionality in different taxa. Such cases usually occur in regions with reduced evolutionary constraints, such as regions where motifs are abundant, or near a duplication event. The sequence dissimilarities of CRMs across taxa include indel events in the spacers, as well as gains and losses of binding sites for TFs. A recent statistical

analysis of the *Zeste* binding sites in several *Drosophila* taxa also revealed existence of large-scale functional turnover [129]. Nevertheless, the fact that sequence similarity is absent does not necessarily mean that the overall functional effect of the CRM as a whole is vastly different. In fact, for the *Drosophila* clade, despite the substantial sequence dissimilarity in gap-gene CRMs such as *eve2*, the expression of these gap genes shows similar spatio-temporal stripe patterns across the taxa [112, 113].

Although a clear understanding of the evolutionary dynamics underlying such inter- and intra-taxa diversity is still lacking, it is hypothesized that regulatory sequences such as TFBSs and CRMs may undergo adaptive evolution via stabilizing selections acting synergistically on different loci within the sequence elements [113, 129], which causes site evolution to be non-*iid* and non-*isotropic* across all taxa. In such a scenario, it is crucial to be able to model the evolution of biological entities not only at the resolution of individual nucleotides, but also at more macroscopic levels, such as the functionality of whole sequence elements such as TFBSs over lineages. To our knowledge, so far there have been few attempts along this line, especially in the context of motif detection. The CSMET model intends to address this issue.

Orthology-based motif detection methods developed so far are mainly based on nucleotide-level conservation. Some of the methods do not resort to a formal evolutionary model [15], but are guided by either empirical conservation measures [16, 42, 160], such as parsimonious substitution events or window-based nucleotide identity, or by empirical likelihood functions not explicitly modeling sequence evolution [10, 89, 199]. The advantage of these non-phylogeny based methods lies in the simplicity of their design, and their non-reliance on strong evolutionary assumptions. However, since they do not correspond to explicit evolutionary models, their utility is restricted to purely pattern search, and not for analytical tasks such as ancestral inference or evolutionary parameter estimation. Some of these methods employ specialized heuristic search algorithms that are difficult to scale up to multiple species, or generalize to aligned sequences with high divergence. Phylogenetic methods such as EMnEM [127], MONKEY [128], and our in-house implementation of PhyloHMM (originally implemented in [18] for gene finding, but in our own version tailored for motif search) explicitly adopt a *complete* and *independent* shadowing model at the nucleotide level. These methods are all based on the assumption of homogeneity of functionality across orthologous nucleotides, which is not always true even among relatively closely related species (e.g., of divergence less than 50 mya in *Drosophila*). Empirical estimation and simulation of turnover events is an emerging subject in the literature [82, 129], but to our knowledge, no explicit evolutionary model for functional turnover has been proposed and brought to bear in comparative genomic search of non-conserved motifs. Thus our CSMET model represents an initial foray in this direction. Closely related to our work, two recent algorithms, rMonkey [129] - an extension over the MONKEY program, and PhyloGibbs [171] - a Gibbs sampling based motif detection algorithm, can also explicitly account for differential functionality among orthologs, both using the technique of shuffling or reducing the input alignment to create well conserved local subalignments. But in both methods, no explicit functional turnover model has been used to infer the turnover events. Another recent program PhyME [174] partially addresses the incomplete orthology issue via a heuristic that allows motifs only present in a pre-chosen *reference* taxon to be also detectable, but it is not clear how to generalize this ability to motifs present in arbitrary combination of other taxa,

and so far no well-founded evolutionary hypothesis and model is provided to explain the heuristic. Non-homogeneous conservation due to selection across aligned sites has also been studied in DLESS [172] and PhastCons [121], but unlike in CSMET, no explicit substitution model for lineage-specific functional evolution was used in these algorithms, and the HMM-based model employed there makes it computationally much more expensive than CSMET to systematically explore all possible evolutionary hypotheses. A notable work in the context of protein classification proposed a phylogenomic model over protein functions, which employs a regression-like functional to model the evolution of protein functions represented as feature vectors along lineages in a *complete* phylogeny [46], but such ideas have not been explored so far for comparative genomic motif search. Various nucleotide substitution models, including the Jukes-Cantor 69 (JC69) model [86], and the Felsenstein 81 (F81) model [47], have been employed in current phylogenetic shadowing or footprinting algorithms. PhyloGibbs and PhyME use an analogue of F81 proposed in [176], which is one of the simplest models to handle arbitrary stationary distributions, necessary to model various specific PWMs of motifs. Both PhyME and PhyloGibbs also offer an alternative to use a simplified star-phylogeny to replace the phylogenetic tree when dealing with a large number of taxa, which corresponds to an even simpler substitution process.

3.2 The generative model

3.2.1 The CSMET approach

Our CSMET model differs from these existing methods in several important ways. First, it uses a different evolutionary model based on a coupled-set of both functional and nucleotide substitution processes, rather than a single nucleotide substitution model to score every alignment block. Second, it uses a more sophisticated and popular nucleotide substitution process based on the Felsenstein84 (F84) model [49], which captures the transition/transversion bias. Third, it employs a hidden Markov model that explicitly models autocorrelation of evolutionary rates on successive sites in the genome. Fourth, it uses an efficient deterministic inference algorithm that is linear to the length of the input sequence and either exponential (under a full functional phylogeny) or linear (under a star-shaped functional phylogeny) to the number of the aligned taxa, rather than the Monte Carlo or heuristic search algorithms that require long convergence times. Essentially, CSMET is a context-dependent probabilistic graphical model that allows a single column in a multiple alignment to be modeled by multiple evolutionary trees conditioned on the functional specifications of each row (i.e., the functional identity of a substring in the corresponding taxon) (Figure 3.1). When conjoined with a hidden Markov model that auto-correlates the choices of different evolutionary rates on the phylogenetic trees at different sites, we have a stochastic generative model of phylogenetically related CRM sequences that allows both binding site turnover in arbitrary subsets of taxa, and coupling of evolutionary forces at different sites based on the motif organizations within CRMs. Overall, CSMET offers an elegant and efficient way to take into consideration complex evolutionary mechanisms of regulatory sequences during motif detection. When such a model is properly trained on annotated sequences, it can be used for comparative genomic motif search in all aligned taxa based on a posterior probabilistic inference algorithm. This model can be also

used for *de novo* motif finding as programs such as PhyloGibbs and PhyME, with a straightforward extension of the inference procedure that couples the training and prediction routines in an expectation-maximization (EM) iteration on unannotated sequence alignments. We focus on supervised motif search in higher eukaryotic genomes. We compare CSMET with representative competing algorithms, including EMnEm, PhyloHMM, PhyloGibbs, and a mono-genomic baseline Stubb (which uses an HMM on single species) on both simulated data, and a pre-aligned Drosophila dataset containing 14 developmental CRMs for 11 aligned *Drosophila* species. Annotations for motif occurrences in *D. melanogaster* of 5 gap-gene TFs - *Bicoid*, *Caudal*, *Hunchback*, *Kruppel* and *Knirps* - were obtained from the literature. We show that CSMET outperforms the other methods on both synthetic and real data, and identifies a number of previously unknown occurrences of motifs within and near the study CRMs. The CSMET program, the data used in this analysis, and the predicted TFBS in *Drosophila* sequences, are available for download at `http://www.sailing.cs.cmu.edu/csmet/`.

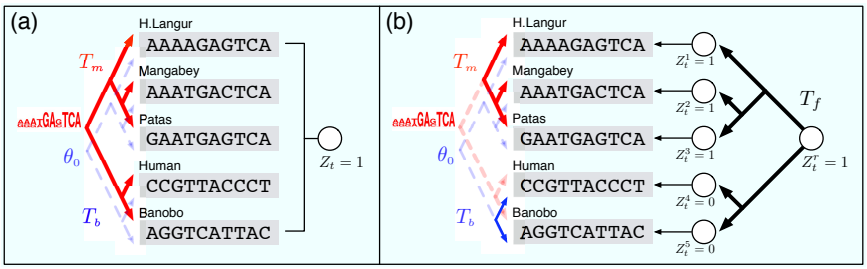


Figure 3.1: Diagrams showing the underlying generative models underlying basic phylogenetic shadowing approaches and the CSMET approach. (a) The basic mixture of full-phylogeny model underlying PhyloHMM and EMnEM, where functional homogeneity across aligned sequences is assumed, and all aligned taxa (i.e., rows) are either under a full motif phylogeny (when $Z_t = 1$) or a full background phylogeny (when $Z_t = 0$). (b) The conditional shadowing model underlying CSMET, with an explicit evolutionary model T_f for species-specific functional turnover, and partial motif or background phylogenies over subsets of taxa according to the turnover status. From [149].

3.3 Results

3.3.1 The CSMET model

Model for Phylogenetically Related Motif Sequences

To motivate and explain the statistical foundation and biological rationale underlying the CSMET model, we begin with a brief description of a conventional model for phylogenetically related sequences based on the classical molecular substitution process, where functional turnover of motifs is not explicitly modeled. This model will be used as a component in our proposed model. Consider a multiple alignment of M instances of a motif of length L. Let $\mathbf{A}$ denote an $M \times L$ matrix containing M rows $a_1, \ldots, a_M$, each representing an instance of this motif, i.e., $a_i \equiv [a_{i,1}, \ldots, a_{i,L}]$, where $a_{i,l} \in \mathbb{N} \equiv \{\mathrm{A}, \mathrm{G}, \mathrm{C}, \mathrm{T}\}$. Due to the stochastic nature of the sequence composition of TFBSs, a popular representation of a motif pattern is the *position weight matrix* (PWM), $\boldsymbol{\theta} \equiv (\theta_1, \ldots, \theta_L)$,

of which each column vector θ_l defines a *multinomial* probability distribution of the nucleotides observed at the l^{th} position of instances of this motif. That is, $P(a_{i,l}|\theta_l) = \prod_{k\in\mathbb{N}} \theta_{l,k}^{\mathbb{I}(a_{i,l},k)}$, where $\mathbb{I}(x, y)$ is an indicator function that equals to 1 when $x = y$ and 0 otherwise. Under a PWM, all sites in the motif are assumed to be mutually independent, thus the probability of a length-L instance is simply a product of the probabilities of nucleotides at every site: $P(a_i|\boldsymbol{\theta}) = \prod_{l=1}^{L} P(a_{i,l}|\theta_l)$. When the motif instances in $\mathbf{A}$ are from different genomic locations of a single species (i.e., they are phylogenetically *unrelated*), the likelihood of the aligned motifs $\mathbf{A}$ is simply a product of the likelihoods of every instance a_i, $P(\mathbf{A}) = \prod_{i=1}^{M} P(a_i|\boldsymbol{\theta}) = \prod_{l=1}^{L}\prod_{i=1}^{M} P(a_{i,l}|\theta_l)$, which means all the rows in $\mathbf{A}$ are independent of each other (although in reality, they might not evolve independently.) If $\mathbf{A}$ contains M phylogenetically related motif instances each from a different species, then a straightforward way to model the likelihood of $\mathbf{A}$ is to assume that the instances therein from different taxa are *shadowed* by a phylogenetic tree that defines a nucleotide-level substitution process from an ancestral sequence [48, 127] (Figure 3.1a). Our proposed method uses this model as a building block. Formally, a phylogenetic shadowing model T_m for a motif is a *tree-likelihood model* specified by a four-tuple $\{\boldsymbol{\theta}, \boldsymbol{\tau}, \boldsymbol{\beta}, \boldsymbol{\lambda}\}$, where $\boldsymbol{\theta} \equiv (\theta_1, \ldots, \theta_L)$ represents the equilibrium nucleotide distributions at the root of the evolutionary tree of every site within the motif; $\boldsymbol{\tau} \equiv (\tau_1, \ldots, \tau_L)$ denotes the (usually identical) topologies of the evolutionary trees of every site; $\boldsymbol{\beta} \equiv (\beta_1, \ldots, \beta_L)$ denotes the sets of branch lengths of the evolutionary trees; and $\boldsymbol{\lambda}$ represents where necessary some additional evolutionary parameters of the motif depending on the specific nucleotide substitution models. Under a phylogenetic shadowing model, the probability distribution of nucleotides in any taxon that corresponds to a leaf conditioning on its predecessor in the tree can be derived based on a continuous-time Markov model of nucleotide substitution along the tree branches [48]. We employ the F84 substitution model parameterized by a given equilibrium distribution, a transition/tranversion ratio ρ, and a total substitution rate μ that can be estimated from training data [49].

Typically, we can use the PWM of the motif as the equilibrium distribution of the motif phylogeny. For simplicity, one can also assume that all sites within the motif share the same topology τ and the same branch lengths β. This means that the evolutionary processes underlying each site within the motif are homogeneous. Similarly, we can define $T_b \equiv \{\theta_b, \tau_b, \beta_b, \lambda_b\}$ for the background. Assuming that sites within the motif evolve independently, the likelihood of M aligned L-mers can be expressed as:

$$P(\mathbf{A}|T_m) = \prod_{l=1}^{L} P_N(A_l|\theta_l, \tau, \beta, \lambda), \tag{3.1}$$

where A_l denotes the l^{th} column in $\mathbf{A}$, and $P_N(\cdot|\theta_l, \tau, \beta, \lambda)$ is the marginal likelihood of the leaves under an motif-site-specific evolutionary tree $T_m^{(l)} \equiv \{\theta_l, \tau, \beta, \lambda\}$ for nucleotide substitution, which can be computed using Felsenstein's pruning algorithm [48]. To model a multiple alignment of regulatory regions that is N base-pairs long and contains motifs at unknown positions, we can assume that every L-mer block in the alignment can correspond to either a motif sequence, or the background, specified by a hidden *functional state* Z_t, where t denotes the position of the left-most column of the block in the alignment. (For simplicity, we consider only one motif type here, but the formulation readily generalizes to multiple motif types.) The state sequence $\mathbf{Z} \equiv Z_{1:N}$ can be thought of as a functional annotation sequence of an ancestral regulatory region of

length N. In the EMnEM model [127], the Z_t's are assumed to be independently sampled from a Binomial distribution of motif and background states, similar to the classic mixture models of motif underlying MEME (Figure 3.1a). In a PhyloHMM originally proposed in [173] for comparative gene finding, which can be easily extended for motif search, $Z_{1:N}$ can follow a hidden Markov model that captures the transition probabilities between background and motifs.

Model for Motif Turnover

A caveat of the phylogenetic shadowing model described above is that, at every location t, the functionality indicator Z_t must apply to all the taxa (i.e., rows) in the alignment (as illustrated in (Figure 3.1a)), meaning that the aligned substrings from all taxa at this position are derived from the same evolutionary tree (either the motif or the background tree, depending on the value of Z_t; when Z_t is hidden, this results in a mixture of two complete trees). This is a strong orthologous assumption which insists that every row in the alignment block must have evolved from the same most recent common ancestor (MRCA) according to the same molecular evolution model. This assumption might not be valid for every region in the alignment due to abrupt functional turnover such as whole motif insertion/deletion, or due to imperfect alignment that fails to identify the true sequence orthology. We assume that every sequence segment in an alignment block, generically referred as $\mathbf{A}_t$ where t denotes the left-most position of the alignment, has its own functionality indicator Z_t^i. Generalizing the molecular evolution model for base substitution, we posit that the functional annotation vector $Z_t \equiv [Z_t^1, \ldots, Z_t^M]'$ of a block of aligned segments are themselves governed by a *coarser-grained evolutionary tree* that models the evolution of the functionalities of the attendant segments in different taxa (Figure 3.1b). We refer to this evolutionary tree as a (functional) *annotation tree* (or, interchangeably, a functional phylogeny), denoted by $T_f \equiv \{\alpha, \tau_f, b_f, \lambda_f\}$. In such a tree model, each leaf represents a random variable Z_t^i whose value reveals the functional status (i.e., being a motif, background, or more detailed function information such as motif types, etc.) of the segment from taxon i, and the root is characterized by a hypothetical ancestral functionality indicator Z_t^r and an equilibrium distribution α. Along the branches of this tree, the functional states evolve according to a *functionality substitution* model, in much the same way the nucleotides do under a *molecular substitution* model, except that now the model parameters T_f are fitted differently (we will return to this point in the methodology section) and the evolutionary dynamics can also have richer structures. For example, in the model proposed by [46] for protein function evolution, the evolutionary dynamics were captured by a logistic regression rather than a constant-rate continuous-time Markov process used in standard molecular substitution models. For simplicity,here we adopt a simple JC69 model for functionality substitution, which is denoted as $P_F(Z_t|T_f)$. In summary, the functional phylogeny T_f models the quantum changes of functional elements (rather than the fine-grained changes at the nucleotide level) during evolution in terms of whether an entire functional element is preserved, lost, or emerged, during the course of speciation.

Conditional Shadowing Under Motif Turnover

To capture the effect of motif turnover, we assume that, *conditioning* on the functional states of all rows (i.e., species), which are represented as a random column vector $Z_t \equiv [Z_t^1, \ldots, Z_t^M]'$ distributed according to the functional phylogeny specified by T_f, the sequences in alignment block t admit either a *marginal* motif phylogeny or a *marginal* background phylogeny. As shown in Figure 3.1b, typically, for a given block, only a subset of the rows $\mathbf{A}_t'$ correspond to conserved instances of a motif (e.g., rows 1, 2, and 3), and therefore their joint probability is defined by a *marginal phylogeny* T_m' of the full motif phylogeny (i.e., the subtree highlighted by solid red lines in Figure 3.1b). The remaining part of the motif phylogeny (represented by the subtree in dotted red lines in Figure 3.1b), which corresponds to taxa where the corresponding motifs had turned-over to background sequences, needs to be marginalized out. We can efficiently compute the likelihood of the preserved motif instances $\mathbf{A}_t' \equiv \{a_i(t) : s.t.\ Z_t^i = 1\}$ under the marginal motif phylogeny T_m', expressed as $P(\mathbf{A}'|T_m')$ using the standard pruning algorithm. Similarly, the subset of rows $\mathbf{A}_t'' \equiv \{a_i(t) : s.t.\ Z_t^i = 0\}$ corresponding to the background or merely gaps admit a *marginal background phylogeny* T_b' (e.g., the blue tree with leaves only correspond to rows 4 and 5 in Figure 3.1b). Putting these two parts together, now for every position t in the input alignment, we have the following joint probability (i.e., the complete likelihood) of the observed alignment block $\mathbf{A}_t$, the vector of instantiated extant functional states $\mathbf{z}_t$, and an instantiated ancestral functional state z_t^r under a conditional shadowing model with multiple evolutionary trees (aka, CSMET):

$$\begin{aligned} P(\mathbf{A}_t, \mathbf{z}_t, z_t^r) &= P(\mathbf{A}_t|Z_t = \mathbf{z}_t, T_m, T_b)P(Z_t = \mathbf{z}_t|Z_t^r = z_t^r, T_f)P(Z_t^r = z_t^r) \\ &= P(\mathbf{A}_t'|T_m')P(\mathbf{A}_t''|T_b')P(\mathbf{z}_t|z_t^r, T_a)P(z_t^r). \end{aligned} \tag{3.2}$$

In practice, the leaf functional states $\mathbf{z}_t$ of an alignment block starting at position t, and the ancestral functional state z_t^r are not observed. Thus the likelihood score of $\mathbf{A}_t$ follows a complex mixture of *marginal* phylogenies defined by all possible joint configurations of functional states $\mathbf{z}_t \equiv [z_t^1, \ldots, z_t^M]'$ and the ancestral state z^r, rather than a simple motif/background mixture as in extant models. The typical tasks in motif detection involves either computing the marginal conditional likelihood $P(\mathbf{A}_t|z_t^r)$ for all possible states of z_t^r, which will be used as the emission probability in an HMM of the ancestral functional states over the entire alignment (to be detailed in the next section); or the marginal posterior $P(\mathbf{z}_t|\mathbf{A}_{1:T})$, which will be used to extract the maximum *a posteriori* (MAP) motif annotation of the alignment. Both tasks involve a marginalization step that sums over all joint configurations of the internal tree nodes, z_r's, and $\mathbf{z}_t$'s. This leads to an inference problem in a state space defined by the product of multiple trees and therefore can be computationally intensive. Since in practice it is unusual to encounter more than 20 or so taxa in the comparative genomic setting, inference is still feasible. In this case, one can apply a *coupled-pruning algorithm* or a standard junction tree algorithm [31] for exact inference. For an alignment of a large number of species and/or for a problem which involves searching for a large number of motifs simultaneously, marginalization of the product space of trees can be prohibitive. In these circumstances, we can apply an approximate inference method such as the generalized mean field algorithm [205], which decomposes the coupled trees in CSMET into disjoint trees and applies iterative message-passing across these trees to obtain an approximate posterior of $\mathbf{z}_t$ or the conditional likelihood of $\mathbf{A}_t$. Alternatively we can replace some or all of the full phylogenetic trees for motif, background and functional evolution by star-topology phylogenies as in PhyloGibbs [171].

Tree- and Rate-Transition Along Alignment

Different sites in the genome are subject to different evolutionary constraints and therefore follow phylogenetic trees with different equilibriums, topologies and rates. The conditional phylogenetic shadowing model described above couples multiple site-specific trees of all sites *within* a moving window of alignment block via a functional phylogeny; but it does not explicitly model transitions between possibly different evolutionary processes as the window scans over different functional entities along the aligned sequences, for example, transitions between motifs and different background regions, and among different motifs. We introduce a hidden Markov model to model the transitions between functional annotations along the alignment. In principle, this HMM can employ highly structured transition models such as the global HMMs used in LOGOS [207] or CISTER [56], which intend to capture sophisticated "motif grammars" underlying higher eukaryotic CRMs. We adopt a simplistic 3-state HMM that models the length of the spacer between motifs as a geometric distribution, and allows the motifs to be on either strand of the DNA. We define the HMM over the sequence of ancestral functional states $\mathbf{Z}^r_{1:N}$, modeling the spatial transitions of functionalities along a hypothetical ancestral regulatory sequence underlying the aligned sequences from the study species. To model TFBS on either DNA strand with opposite orientations, two functional states are needed for each type of motifs, which determine the appropriate orientation for the PWM employed by the motif tree T_m for defining the likelihood of a selected DNA substring; but these two functional states correspond to a degenerated motif state (i.e., $Z^r_t = 1$) at the root of the functional tree T_f in CSMET, and follow the same turnover process. Details of such an HMM is provided. Unlike the standard HMM for mono-genomic motif detection where the emission probability uses a simple conditional multinomial distribution of a single nucleotide, or a PhyloHMM for comparative-genomic motif detection ignoring motif turnover where the emission probability is defined by a conditional likelihood of a column of aligned nucleotides under a single phylogeny, to accommodate functional turnover of segments in certain species in the alignment, we define the emission model to be the CSMET conditional likelihood of an alignment block, $P_c(\mathbf{A}_t|T_m, T_b, T_f, z^r_t) = \sum_{\mathbf{z}_t} P(\mathbf{A}_t|\mathbf{z}_t, T_f, T_b)P(\mathbf{z}_t|z^r_t, T_f)$, and thereby enable conditional shadowing over the taxa at each site. A technical issue arising from this construction is that unlike the PhyloHMM, which is still a standard 1st-order HMM, in our case we have a higher-order HMM due to the contex-dependent coupling of all the sites within a motif by the functional phylogeny T_f, which models the whole sequence segment within a window of length L as a unit. In the next section, we outline statistical inference strategies that address this technical issue.

3.3.2 Strategy

Posterior Inference

The incorporation of the functional phylogeny T_f to explicitly model functional turnover of entire segments (rather than individual sites) of DNA sequences in different taxa in a multiple alignment introduces not only higher-order Markov dependencies among sites, but also context-dependent dependencies among taxa. Thus CSMET is essentially a probabilistic model with *context-specific independencies*, which is well-known to be intractable in general [19]. Figure 3.2a and 3.2b show

an example of the context-specific relationships among variables due to two different possible value-configurations of the hidden variables corresponding to ancestral and taxa-specific functional annotations (of a small chunk of the alignment). Computing the likelihood of the entire alignment requires a summation of all joint configurations of all of these hidden variables, for which no efficient exact algorithm resembling the dynamic programming algorithms applied to HMMs is available.

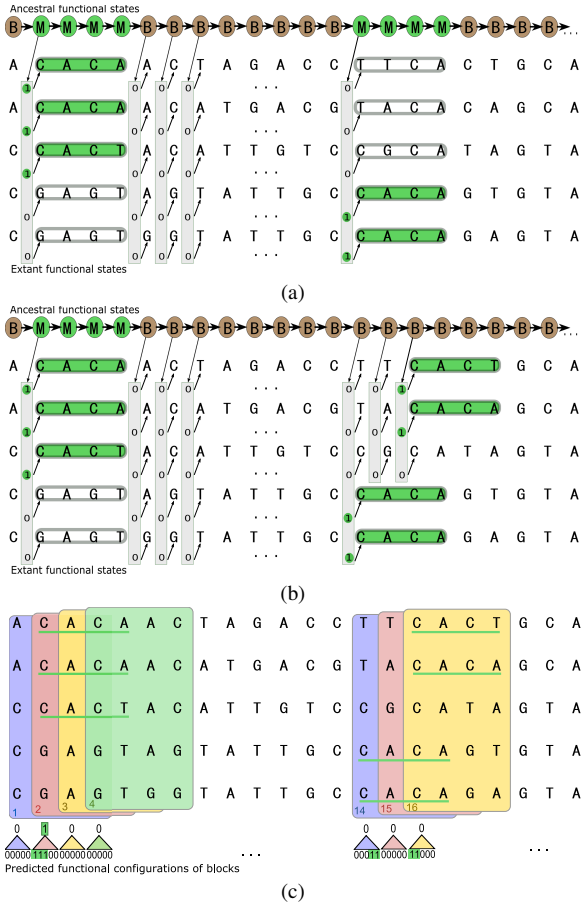


Figure 3.2: Context-specific relationships among variables due to two possible value configurations of hidden variables shown in (a) and (b). Note that when the ancestral state is a motif, the segment corresponding to the TBFS evolves as a unit (as shown by the arrow from an extant functional state pointing to a multi-column segment), either retaining its functionality as a motif, or turning-over to a background segment, as illustrated in (a). When the ancestral state is a background, then every position can evolve independently as long as it is still in the background (as shown by the arrow from a functional state pointing to a single column). But when a motif emerges out of the background, as shown in (b), the segments corresponding to the TFBS start to evolve as a unit, causing even the aligned nucleotide positions to evolve under different positional constraints. Panel (c) outlines the idea of a block approximation of the CSMET emission probability. From [149].

While it is possible to implement a Monte Carlo algorithm that performs sampling over the functional annotation space of $\{\mathbf{Z}^{1:M}_{1:N}\} \cup \{\mathbf{Z}^{r}_{1:N}\}$ conditioning on the observed multiple alignment, we propose an approximate algorithm for posterior inference. As illustrated in Figure 3.2c, we can treat an N-column alignment as a sequence of $(N - L + 1)$ consecutive L-column aligned blocks.

We assume each such block $\mathbf{A}_t$ is either generated from a CSMET emission model conditioning on the ancestral function of this segment being a background, i.e., $P(\mathbf{A}_t|Z_t^r = 0)$, or it can be generated from a CSMET conditioning on the ancestral function being a motif, say, of type k, expressed as $P(\mathbf{A}_t|Z_t^r = k)$. We can pre-compute the emission probabilities for all the aligned blocks, plug them back into an equivalent HMM of Z_t^r's on blocks rather than on columns, and then compute the posterior probabilities or Viterbi-sequence of the labels of each block using the standard dynamic programming algorithms (e.g., forward-backward) for HMMs (see the method section for details). The approximation introduced here lies in the approximate computing of the emission probabilities for the blocks, specifically at the boundary between motifs and background. For these blocks the likelihood of the aligned sequences should be defined by two different emissions, one on the background sub-block and the other on the motif sub-block, whereas our approximation employs only a single emission—either an entirely background-derived CSMET $P(\mathbf{A}_t|Z_t^r = 0)$ or an entirely motif-derived CSMET $P(\mathbf{A}_t|Z_t^r = k)$. But since our approximation results in a poorer fit only for the boundary regions, we expect that the overall posterior indication of the motifs, which is primarily driven by the emission probabilities of the motif blocks, will only suffer moderate weakening of contrast at the boundaries. We refer to this approximation method as *block-approximation* (BA). Another more subtle approximation due to BA is the ignoring of different turnover behaviors within a block $\mathbf{A}_t$ conditioning on the ancestral function of this segment(being a motif or a background), as exemplified in Figure 3.2b. Unlike a motif block derived from an ancestral motif, a segment of ancestral background sites do not evolve as a whole block, thus a block $\mathbf{A}_t$ entirely originated from a ancestral background can contain rows (descendants) that are either entirely non-motif, or partially non-motif and partially motif (i.e., starting from an arbitrary position t' in window $t : t + L$, the segment $t' : t' + L$, part of which extends out of $\mathbf{A}_t$, in an arbitrary taxon can evolve into a motif), whereas a block $\mathbf{A}_t$ entirely originated from a motif can only contain either fully preserved motif rows or turned-over non-motif rows. BA simply treats each entire row in $\mathbf{A}_t$ as a homogeneous functional evolutionary unit. The computational time for BA is linear in the length of the input, with a multiplicative factor determined by the length of the motif and the number of species concerned in the alignment. In case of multiple motifs, the emission probabilities of the blocks should be computed under the unique CSMET of each motif. Since motifs can have different lengths, bookkeeping of all the emissions can be slightly more complicated due to the need to handle blocks of different lengths. But the computational cost is only increased by the order of the number of the motifs in question.

With the BA strategy, we arrive at an approximation to the posterior distribution of motif annotation at every position given the entire alignment, $P(Z_t^{1:M}|\mathbf{A}_{1:N})$, and the posterior of the sequence of ancestral functions, $P(Z_t^r|\mathbf{A}_{1:N})$. For an alignment block of which only a few taxa correspond to motifs and others are merely background, under the CSMET model, the Z_t^r of this block can be either motif or background. In the first case, it means that absence of motifs in some taxa is interpreted as the result of loss of ancestral motifs, whereas in the second case, the presence of motifs in some taxa is interpreted as the result of emergence of nascent motifs out of the background. As far as we are aware of, CSMET is the only motif-finding algorithm that rigorously offers a closed-form deterministic solution to the posterior probability distribution of motif annotations both in the alignment and in the ancestral sequence over the entire space of binding site configurations. Phy-

loGibbs [171] offers a sample-based solution to the posterior of $Z_t^{1:M}, t = 1, \ldots, N$ given $\mathbf{A}_{1:N}$, but as mentioned earlier, it is not based on an explicit model of binding site turnover, and thus does not have a closed-form expression that can motivate efficient deterministic approximation.

Maximum Likelihood Training

The CSMET can be trained on annotated CRM alignments. We need to learn the nucleotide phylogenetic trees for motifs and backgrounds, and the phylogenetic tree that describes the evolution of functional annotation. We use the F84 model for nucleotide substitution on the motif and background trees; for evolution of functional annotation, we use the simpler JC69 model. As detailed later, for a given tree topology, for the JC69 model all we need to estimate is the branch length on the tree, which relates to total substitution probability. For the F84 model, besides the tree topology, we need to estimate the stationary distribution, which we set to be the PWM for motif phylogenies or the background nucleotide frequencies for background phylogeny; and also two additional evolutionary parameters: the overall substitution rate per site μ and the transition/transversion ratio ρ. Given a multiple alignment, the ground truth of functional annotation, the PWMs for motifs, and nucleotide frequency for the background, we use the following strategy for estimating the trees and the evolutionary parameters.

- Find a tree topology τ and the branch lengths β by running fastDNAml [137] over the entire alignment.
- Find a scaling factor r_f over branch lengths β_f of the functional tree T_f, by maximizing the likelihood of aligned functional annotations under T_f via a line-search in parameter space.
- Find a scaling factor r_m over branch lengths β_m of the motif tree T_m, and the Felsenstein rate μ_m, by maximizing the likelihood of aligned motif sequence under T_m with the F84 model.
- Find a tree topology τ_b and branch lengths b_0 for background tree T_b by running fastDNAml directly over only the background sequences. The Felsenstein rate μ_b is then estimated by maximizing the likelihood under T_b with a simple line-search.

To compute the Felsenstein substitution rate μ, we use a fixed transition-transversion ratio of 2. If the stationary nucleotide distribution defined by the motif PWM is incompatible with this value of the transition-transversion ratio, we set it to the smallest value that is compatible with the stationary distribution as in [123].

3.4 Functional turnover in the Drosophila clade

3.4.1 Performance on Synthetic Data

At present, biologically validated orthologous motifs and CRMs across multiple taxa are extremely rare in the literature. In most cases, motifs and CRMs are only known in some well-studied *reference taxa* such as the *Drosophila melanogaster*; and their orthologs in other species are deduced from multiple alignments of the corresponding regulatory sequences from these species according to the positions and PWMs of the "reference motifs" in the reference taxon. This is a process that

demands substantial manual curation and biological expertise; rarely are the outcomes from such analysis validated *in vivo* (but see [113] for a few such validations in some selected *Drosophila* species where the transgenic platforms have been successfully developed). At best, these real annotations would give us a limited number of true positives across taxa, but they are not suitable for a systematic performance evaluation based on precision and recall over true motif instances. Thus we first compare CSMET with a carefully chosen collection of competing methods on simulated CRM sequences, where the motif profiles across all taxa are completely known. We choose to compare CSMET with 3 representative algorithms for comparative genomic motif search, PhyloGibbs, EMnEM, PhyloHMM; and the program Stubb, which is specialized for motif search in eukaryotic CRMs, and is set to operate in mono-genomic mode.

Multi-specific CRM simulator

We developed a simulator of multi-specific CRMs with flexible TFBS turnover dynamics across taxa and realistic TFBS arrangement within CRM. Specifically, the input of the simulator includes: 1) topologies of the phylogenetic trees for nucleotide (e.g., in motif sites and background) and functionality substitutions; 2) prior distributions of the stationary distribution of states (i.e., nucleotide or functionalities) at the roots of the trees; 3) prior distributions of the branch lengths of the trees and the substitution rates, and other evolutionary parameters where necessary (e.g., the Felsenstein rate μ and ρ in F84 model); 4) a global HMM encoding the motif grammar in the CRMs. As detailed in the Material and Methods, during simulation, all building blocks of a CRM, such as the motif instances, background sequences, functionality states (that determines motif turnover) in different taxa, and positions of the motifs in the CRM are sampled separately, and put together to synthesize an artificial CRM. This simulator can be used to simulate realistic multi-specific CRMs resulting from various nontrivial evolutionary dynamics. It is useful in its own right for consistence/robustness analysis of motif evolution models and performance evaluation of comparative genomic motif-finding programs.

Below, we report results of four experiments based on simulated datasets. Each experiment was based upon varying one parameter of the model, keeping all the others fixed, in order to analyze robustness of CSMET and various other methods under different conditions. Every simulated CRM alignment contained 10 taxa, and for each experiment we simulated 50 datasets. The simulated data is available at the CSMET website to allow external comparisons. Performance of all the tested programs were based on the precision, recall and their F1 score (i.e., the harmonic mean of precision and recall) [194].

Performance under varying degrees of motif turnover

To examine the effect of motif turnover (i.e., functional conservation) in aligned regions across taxa on the motif-detection performance, we simulated CRM alignments with differing magnitudes of the evolutionary rate along the functional phylogeny. Since known motifs in the *Drosophila* species we are working with usually have around 75% conservation, we chose our evolutionary rates so as to achieve conservation percentages between $64 - 75\%$ (or equivalently, turnover percentages between $25 - 36\%$) at the species-specific motif-instance level.

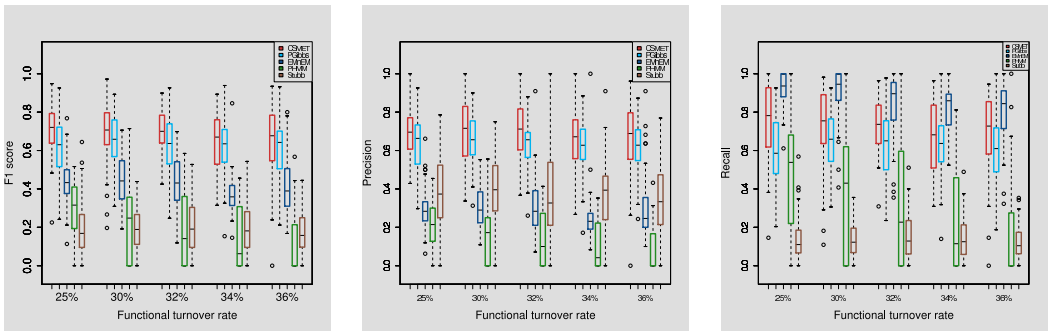


Figure 3.3: Performance under varying degrees of functional conservation.

We find that even with increasing rates of functional turnover, the performance of CSMET and Phylogibbs remain largely stable, with CSMET consistently dominating PhyloGibbs in F1 score with a modest margin (Figure 3.3). The margin is statistically significant with $p = 2.48 \times 10^{-7}$ under a paired t-test. EMnEM has a high recall score, but overall its F1-scores are well below CSMET and PhyloGibbs, also it appears to be affected more by the increased turnover rates. PhyloHMM shows an interesting trend, it performs better than its non-phylogenetic cousin Stubb on data with low turnover rates, but its performance worsens when compared to Stubb on data with increasing turnover rate. This shows that a naive application of phylogenetic shadowing in multi-species alignment with high functional divergence can actually result in degraded performance compared even to just single species analysis.

Performance under varying degree of motif/background contrast

The difference in conservation between the motif and background sequences will have an impact on the performance of the model. However, this experiment can be performed in two different ways: changing the degree of similarity between motif and background stationary distributions; and changing the evolutionary rates of one or the other. We choose the second method and conduct the simulation as follows: we attribute the motif phylogeny with a low entropy stationary distribution resembling a PWM, and with a fixed evolutionary rate; and we let the background to have a stationary distribution similar to but with higher entropy than that of the motif, and have a variable evolutionary rate. The evolutionary rate in the background tree is changed gradually from low values to high values, by varying the scaling factor applied to the background tree from 1 to 8. This is to check how well the CSMET model may detect motifs emerging out of the background with differing degrees of sequence-level conservation with respect to the background caused by their relative evolutionary rates. The corresponding performances are shown in Figure 3.4. We found that even under low variation between the motif and background, i.e., both following an evolutionary tree with similar stationary distribution, and the same branch lengths and scaling parameters, CSMET outperforms all the other methods. CSMET steadily improves in performance upto the scaling factor of 4, after which its performance roughly plateaus. PhyloGibbs behaviors similarly, but overall with lower F1 scores that is statistically significant ($p = 1.41 \times 10^{-14}$). EMnEM, on the other hand outperforms all other methods for scaling factors of 6 or more; meaning that when

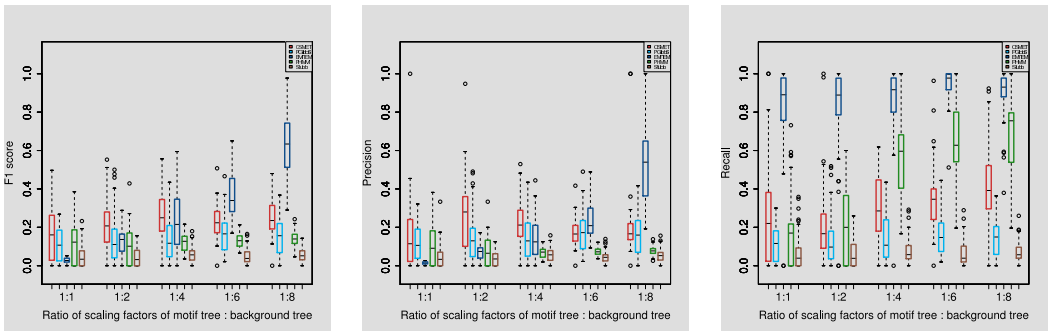


Figure 3.4: Performance under varying degree of motif/background contrast.

motifs are extremely highly conserved compared to the background, the advantage of modeling their turnover as in CSMET and PhyloGibbs over using a basic phylogenetic model diminishes, which is well expected. Since in real CRMs, the evolutionary rates of of the non-functional regions with respect to that of the functional regions (e.g., coding regions, TFBSs) in eukaryotes have been shown to be lie between 1.2 and 2.5 [18], we can claim that CSMET outperforms all other software in the region of biologically relevant parameter settings.

Robustness on data violating CSMET model assumptions

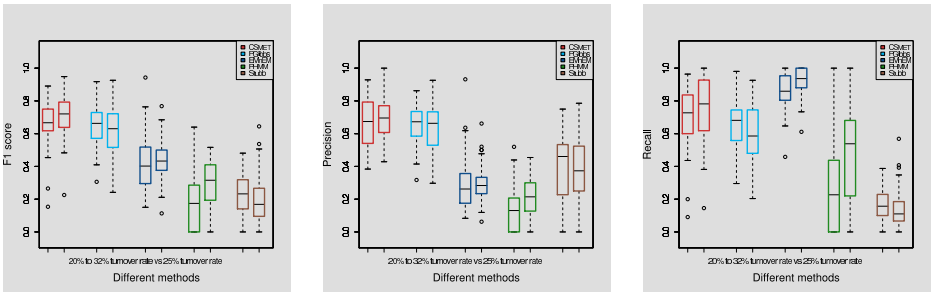


Figure 3.5: Effect of varying motif turnover rates across sequence. In the pair of barplots of each method, the left bar corresponds to performance with varying turnover rates ranging from 20% to 32%; the right bar corresponds to performance under a fixed turnover rate at 25%. From [149].

Effect of non-uniform functional evolution rates We analyzed the robustness of CSMET (compared to other algorithms) in the face of a breakdown of a key CSMET model assumption — that the motif turnover rates are allowed to vary along the simulated CRM sequences instead of staying constant, which is possible in real regulatory sequences. The CSMET model does not explicitly address this dynamics and simply assumes an invariant turnover rate throughout the sequence. We simulated a dataset where the motif turnover rates are chosen uniformly from 4 pre-specified categories, corresponding to branch scaling factors of 1.00, 1.50, 2,00 and 2.50, respectively, over the baseline phylogeny. The corresponding motif turnover rates were 20%, 25%, 30% and 32%,

respectively. As shown in Figure 3.5, we found that while performance of CSMET on such data declines compared to its performance on data simulated with a invariant turnover rate, it still performs no worse than any of the other software even though a primary assumption it adopts (that of a constant functional turnover rate) is violated.

Effect of different generative model To examine the robustness of CSMET under the violation of many of its model assumptions all at the same time, we then performed an experiment using an external simulator PSPE [82], which is based on an entirely different generative model with respect to CSMET (in terms of nucleotide substitution, motif placement, motif turnover, etc.) to synthesize multi-specific CRM sequences. However, at times PSPE generates motifs in some species with some lateral displacements, which appears to be an empirical operation not universal to evolutionary mechanisms that lead to functional turnover in *aligned* motifs (e.g., see [129]), but similar to an assumption underlying PhyloGibbs. To obtain a fair comparison, we suppress the lateral displacements by a post-processing of the sequences simulated by PSPE. In the post-processing step, we remove any motif instances that are laterally displaced in the multiple sequence alignment that is generated. This leaves us with a multiple sequence alignment with all the motif instances perfectly aligned.

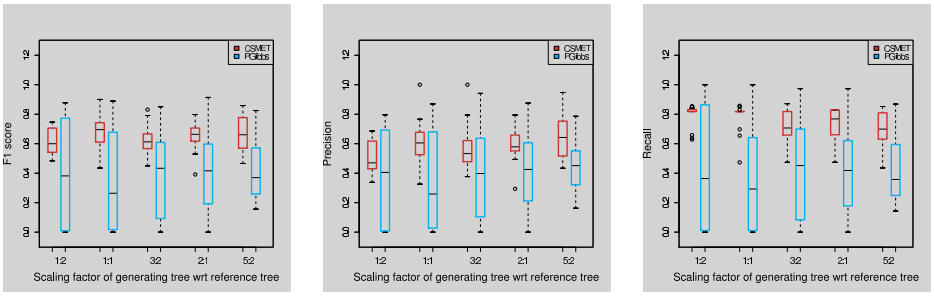


Figure 3.6: Performance on modified PSPE data.The label on the X-axis denotes the scaling factor used by the PSPE tree with respect to a reference *Drosophila* phylogeny. From [149].

We used PSPE driven by five different scaled versions of the phylogenetic tree on the 11 *Drosophila* species to simulate different degrees of motif evolution, and test CSMET and PhyloGibbs on simulations under each scaled tree. For sequence evolution, an HKY nucleotide substitution model with parameter set to 0.05 was used; for the gap distribution, a negative binomial distribution with parameters $\{1, 0.5\}$ was used (note that none of these assumptions are used in CSMET). The motif sequence was generated by PSPE from the default constraints provided. We generated sequences of length 1000 for training, each with about 7-10 motifs; and we test on sequences of length 500 containing 4-5 motifs. For each tested simulation condition (i.e., tree scaling factor), 50 samples were generated, and the performance of CSMET and PhyloGibbs are shown in Figure 3.4.1. We can see that the F1 scores of CSMET are quite stable under different tested conditions and with low variance, and in all conditions CSMET outperforms Phylogibbs on F1 scores, and the margins are statistically significant ($p = 1.875 \times 10^{-13}$). This suggests that CSMET is reasonably robust with respect to violations of its model assumptions.

3.4.2 Performance on Aligned Drosophila CRMs

We applied CSMET and competing methods to a multi-specific dataset of *Drosophila* early developmental CRMs and motifs compiled from the literature [142]. However, in this situation, we score accuracy only on the motifs annotated in *Drosophila melanogaster* (rather than in all taxa), because they are the only available gold-standard. Upon concluding this section, we also report some interesting findings by CSMET of putative motifs, some of which only exist in other taxa and do not have known counterparts in *melanogaster*.

Real CRMs from 11 Drosophila taxa

To evaluate CSMET on real sequence data, we use a pre-aligned benchmark data set containing multiple alignments of orthologous CRMs from 11 related *Drosophila* species, whose divergence time with respect to the most recent common ancestor is roughly 50 million years. The species included are: *melanogaster*, *simulans*, *sechellia*, *yakuba*, *erecta*, *ananassae*, *persimilis*, *pseudoobscura*, *mojavensis*, *virilis*, and *grimshawi*. Our data set contains 14 different multiple-alignments ranging from 3640-bp to 5316 bp long; each alignment corresponds to a DNA segment containing a CRM (Table 3.4.2) that has been annotated in *Drosophila melanogaster* [12, 142] plus 1000bp flanking regions on both ends, and its putative orthologs in the other 10 taxa identified using the precompiled *Drosophila* genome data from the UCSC Genome browser website [52]. Overall, our data set contains 250 instances of motifs in a total of 14 CRMs. To our knowledge, it represents one of the most complete multi-genomic collection of *Drosophila* CRM/motifs. This dataset, along with a full graphical representation of the CRMs and TFBSs, are available at the CSMET website.

Name of CRM	Length	Motif types
Abdominal A	1745	Hunchback, Kruppel
Buttonhead	1429	Bicoid, Hunchback
Engrailed	900	Caudal
Eve Str 2	730	Bicoid, Hunchback, Kruppel
Eve Str 3+7	512	Hunchback, Knirps
Eve Str 4+6	602	Hunchback, Knirps
FushiTarazu Zebra	653	Caudal
Hairy Str 5	1574	Kruppel
Hairy Str 6	556	Caudal, Hunchback, Knirps, Kruppel
Hairy Str 7	1471	Bicoid, Hunchback, Kruppel
Kruppel 730	1158	Bicoid, Hunchback, Knirps
Runt	1335	Bicoid, Hunchback, Knirps, Kruppel
Spalt	721	Bicoid, Caudal, Hunchback, Kruppel
Tailless	635	Caudal, Bicoid

Table 3.1: A short summary of the nature of the annotated CRMs. From [149].

Results on real CRM data sets

Using a 1 versus $K-1$ cross validation scheme, where K is the total number of CRMs in which a motif in question is present, we tested all algorithms on five motifs, *Bicoid*, *Caudal*, *Hunchback*, *Kruppel* and *Knirps*, one motif type at a time, and the results are summarized in Figure 3.7. We used posterior decoding for CSMET and PhyloHMM, since even motifs of the same type can overlap on opposite strands or even on the same strand. For the other three algorithms, we explored their optimum parameter configuration to get meanful results. The five algorithms were compared on the basis of precision, recall, and their F1 score only on the *melanogaster* motifs they manage to identify within the CRMs. Overall, CSMET outperforms all other methods in all motifs except for Kruppel. For Kruppel, all methods perform poorly because the quality of the PWM that can be obtained from training data has very high entropy. Figure 3.7b and 3.7c also show that CSMET gives a much higher recall score than other softwares in most cases while maintaining a precision comparable to them (except in some cases where Stubb has very high precision but very low recall). It is worth mentioning that in these real CRMs, biological annotations tend to be conservative because they are only based on existing footprinting experiments performed in a non-exhaustive fashion in most of the CRMs. Thus a high recall is not very surprising. Since real CRM data are

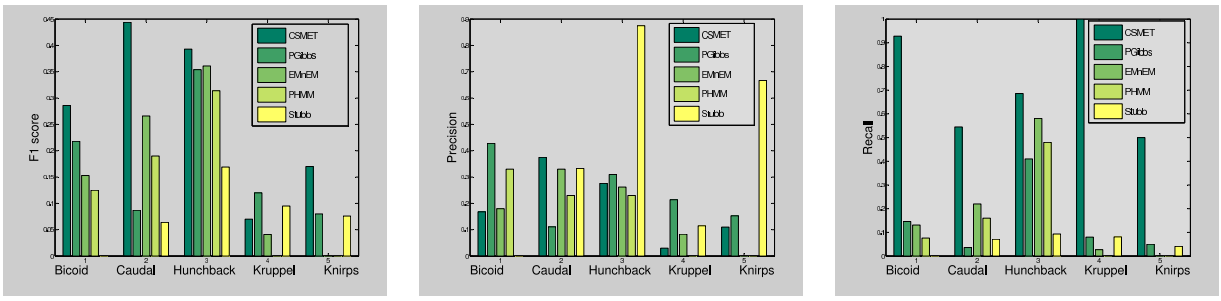


Figure 3.7: Comparison of algorithms on motif search performance over 5 motifs on real CRMs.

more complex than simulated data due to the presence of a significant number of gaps, broken motifs etc., there is a significant variance in the performances across different motifs by CSMET, as well as by all other algorithms; on the other hand, training data for fitting the model parameters needed in a CSMET is extremely limited. We found that the performance of CSMET can be improved over its maximum likelihood configuration (determined from training data) by adjusting the values of the evolutionary parameters. The evolutionary parameters that are estimated from the training data are: the tree evolutionary rates (represented as the scaling coefficients of the tree branches) for the motif and annotation tree, and the Felsenstein rates for the motif and background nucleotide substitution models. Of these parameters, we found that the predictive power of the model is most significantly affected by the evolutionary rate of the functional tree. Figure 3.8 shows the ROC curve of CSMET performance under various values of the evolutionary rate r ranging from a half to 4 times the maximum likelihood estimator of r, along with the scores of 3 competing softwares at a working parameterization adjusted based on their default setting. From Figure 3.8, it is noteworthy that the performance of all programs on the Hunchback motif is generally good. This is probably because the *Hunchback* motif instances are generally very well

conserved, and thus the quality of our training annotation based upon visual inspection is relatively more reliable.

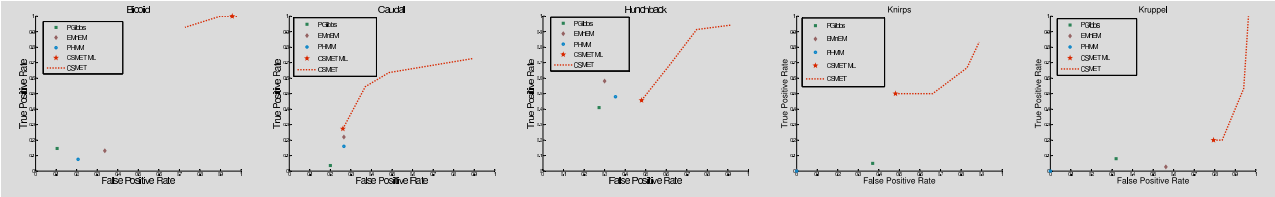


Figure 3.8: ROC of CSMET with different values of functional evolutionary (i.e., TFBS turnover) rates on *Drosophila* CRMs. From [149].

Findings on real CRM data sets

CSMET has correctly retrieved a significant portion of previously known TFBSs within the 14 CRMs in the *melanogaster* taxon, along with their putative conserved orthologs in other taxa, or in some cases, apparent site turnovers in other taxa. Furthermore it has also found numerous interesting instances of alignment blocks of putative TFBSs not known before, both inside CRMs as well as in CRM flanking regions, where TFBS turnovers are apparent in some taxa. A database containing the complete summary of our predictions is available at: `http://www.sailing.cs.cmu.edu/csmet/`, where the positions and taxonomic-identities of all predicted TFBSs and turnovers are documented graphically with appropriate color highlights for each of the 14 CRM alignments we analyzed. Some interesting examples of the predicted TFBSs are presented in Figure 3.9.

Due to the functional heterogeneity across taxa in many of these alignment blocks of putative TFBSs, these motifs can be difficult for other algorithms to detect. Some of these instances correspond to putative TFBSs appearing in non-melanogaster taxa, such as the putative Knirps motif block in the Kruppel 730 CRM (Fig. 3.9h), and the putative Hunchback motif in the flanking region of Spalt CRM (Fig. 3.9f). Another interesting observation is that numerous putative TFBS blocks were identified not just inside the developmental CRMs but also in the flanking regions of the CRMs we analyzed. We had chosen 1000 bp of flanking region from *D. melanogaster*, and found that while some putative sites are located within 100 bp of established CRM boundaries (e.g., Fig. 3.9h), others may lie as far away as 1000 bp (our limit of analysis) and possibly further away from established CRM boundaries (e.g., Fig. 3.9f). We also noted several interesting patterns in examples of functional turnover. These include single species loss of TFBSs, as for the *Caudal* motif in the Tailless CRM region (Fig. 3.9c) and for the *Knirps* motif in Even Skipped Stripes 4+6 CRM region (Fig. 3.9g); and subclade specific loss or gain of binding sites, as in the *Hunchback* motif block in the Abdominal A CRM region (Fig. 3.9d) and the *Hunchback* motif block in the Hairy Stripe 7 CRM region (Fig. 3.9e). A common form of subclade specific loss or gain is that they take place in closely related sister taxa, like *D. pseudoobscura* and *D. persimilis* as in the *Caudal* motif in the Fushi Tarazu Zebra CRM (Fig. 3.9b) and the *Hunchback* motif in the Spalt CRM (Fig. 3.9f). To assess whether CSMET predicts TFBSs of biological significance, we tried validating our findings by checking which of our predicted motif blocks with

functional turnover had been biologically validated. While this is not possible for motifs predicted only in non-melanogaster taxa, or for motifs predicted in CRM flanking regions, we found numerous examples of conserved motif blocks which were biologically validated for the ortholog in *D.melanogaster*. For example, based on the binding site database of [142], the *Caudal* motif block in Tailless CRM (Fig. 3.9c) and the *Hunchback* block in Abdominal A CRM (Fig. 3.9d) were both biologically validated. We further used two recently available large public TF databases – Oreganno [124] and the RegFly [13] – to check if we could find biologically validated binding sites outside those listed in [142]. Of the 8 motifs displayed, 2 additional cases were confirmed in this independent dataset - the *Caudal* motif in the Fushi Tarazu Zebra enhancer (Fig. 3.9b) region, and the *Hunchback* in the Hairy Stripe 7 (Fig. 3.9e) region. Even though we did not perform an exhaustive search to examine whether the validated binding sites (with functional turnover in other species) predicted by CSMET were also predicted by other programs, our results include several non-conserved biologically validated binding sites which are predicted by CSMET but not by PhyloGibbs, including the *Hunchback* motif in Abdominal A CRM (Fig. 3.9d), and the *Hunchback* motif in Hairy Stripe 7 CRM (Fig. 3.9e). Other such binding sites like mel3L+:8639083 were also noted.

3.5 Discussion

CSMET is a novel phylogenetic shadowing method that can model biological sequence evolution at both nucleotide level at each individual site, and functional level of a whole TFBS. It offers a principled way of addressing the problem that can seriously compromise the performance of many extant conservation-based motif finding algorithms: motif turnover in aligned CRM sequences from different species, an evolutionary event that results in functional heterogeneity across aligned sequence entities and shatters the basis of conventional alignment scoring methods based on a single function-specific phylogeny. CSMET defines a new evolution-based score that explicitly models functional substitution along the phylogeny that causes motif turnover, and nucleotide divergence of aligned sites in each taxa under possibly different function-specific phylogenies conditioning on the turnover status of the site in each taxon.

In principle, CSMET can be used to estimate the rate of turnover of different motifs, which can elucidate the history and dynamics of functional diversification of regulatory binding sites. But we notice that experimentally validated multi-species CRM/TFBS annotations that support an unbiased estimate of turnover rates are yet to be generated, as currently almost all biologically validated motifs only exist in a small number of representative species in each clade of the tree of life, such as *melanogaster* in the *Drosophila* clade. Manual annotation on CRM alignments, as we used in this work, tends to bias the model toward conserved motifs. Thus, at this time, the biological interpretation of evolutionary parameters on the functional phylogeny remains preliminary. Nevertheless, these estimated parameters do offer important utility from a statistical and algorithmic point of view, by elegantly controlling the trade-off between two competing molecular substitution processes — that of the motif sequence and of the background sequence — at every aligned site across all taxa beyond what is offered in any existing motif evolution model. Empirically, we find that such modelling is useful in motif detection.

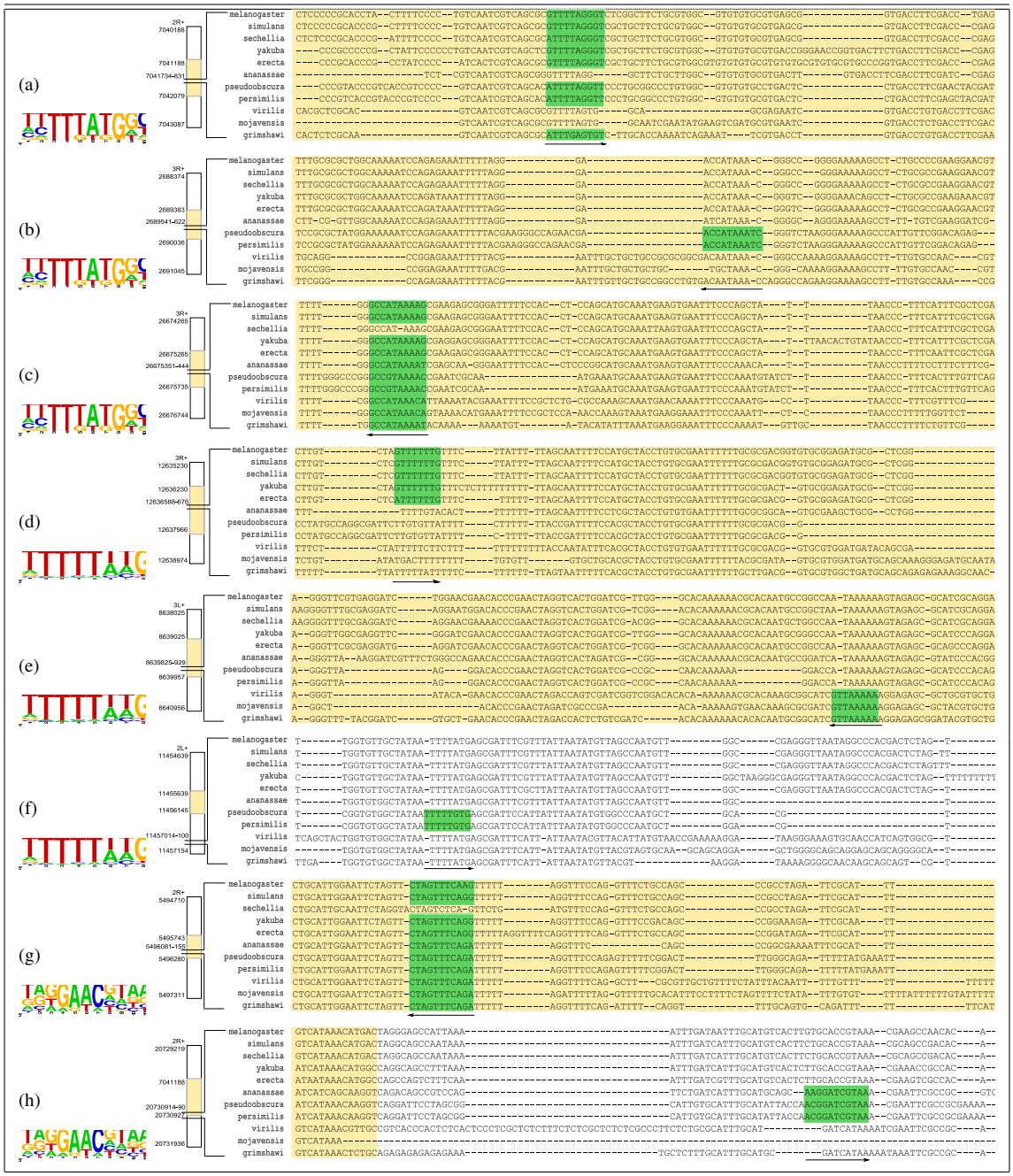


Figure 3.9: Example of previously unknown or biologically validated motif instances uncovered by CSMET in the presence of functional turnover or misalignment. CRM regions are shown in yellow in the alignment. The genomic loci for the flanking region borders, CRM borders and display snippet borders for *melanogaster* assembly 4 are shown on the immediate left of the alignment; with the logos of the identified motifs shown on the far left [32]. (a) A *Caudal* motif in *Engrailed* CRM Alignment. (b) A *Caudal* in *FushiTarazu Zebra* CRM. (c) A *Caudal* in *Tailless*. (d) A *Hunchback* in the *Abda* CRM. (e) A *Hunchback* in *Hairy stripe7* CRM. (f) A *Hunchback* in *Spalt* CRM flanking region about 1000 bp apart from the CRM. (g) A *Knirps* in *Even skipped stripe 4/6* CRM. (h) A *Knirps* in *Kruppel 730* CRM flanking region 38bp apart from the CRM. From [149].

On both synthetic data and 14 CRMs from 11 Drosophila taxa, we find that the CSMET performs competitively against the state-of-the-art comparative genomic motif finding algorithm, PhyloGibbs, and significantly outperforms other methods such as EMnEM, PhyloHMM and Stubb. In particular, CSMET demonstrates superior performance in certain important scenarios, such as cases where aligned sequences display significant divergence and motif functionalities are apparently not conserved across taxa or over multiple adjacent sites. We also find that both CSMET and PhyloGibbs significantly outperform Stubb when the latter is naively applied to sequences of all taxa without exploiting their evolutionary relationships. Our results suggest that a careful exploration of various levels of biological sequence evolution can significantly improve the performance of comparative genomic motif detection.

Recently, some alignment-free methods [89] have emerged which search for conserved TFBS rich regions across species based on a common scoring function, e.g., distribution of word frequencies (which in some ways mirrors the PWM of a reference species). One may ask, given perhaps in the future a perfect search algorithm (in terms of only computational efficiency), do we still need explicit model-based methods such as CSMET? We believe that even if exhaustive search of arbitrary string patterns becomes possible, models such as CSMET still offer important advantage not only in terms of interpretability and evolutionary insight as discussed above, but possibly also in terms of performance because of the more plausible scoring schemes they use. This is because it is impractical to obtain the PWM of a motif in species other than a few reference taxa, thus the scores of putative motif instances in species where their own versions of the PWM are not available can be highly inaccurate under the PWM from the reference species due to evolution of the PWM itself in these study species with respect to the PWM in the reference species. The CSMET places the reference PWM only at the tree root as an equilibrium distribution; for the tree leaves where all study species are placed, the nucleotide substitution model along tree branches allows sequences in each species to be appropriately scored under a species-specific distribution that is different from the reference PWM, thereby increasing its sensitivity to species-specific instantiations of motifs.

A possible future direction for this work lies in developing better approximate inference techniques for posterior inference under the CSMET model, especially under the scenarios of studying sequences from a large clade with many taxa, and/or searching for multiple motifs simultaneously. It is noteworthy that our methods can be readily extended for *de novo* motif detection, for which an EM or a Monte Carlo algorithm can be applied for model-estimation based on the maximum likelihood principle. Currently we are exploring such extensions. Also we intend to develop a semi-supervised training algorithm that does not need manual annotation of motifs in other species on the training CRM alignment, so that we can obtain a less biased estimate of the evolutionary parameters of the CSMET model.

A problem with most of the extant motif finders, including the proposed CSMET, is that the length variation of aligned motifs (e.g., alignments with gaps) cannot be accommodated. In our model, while deletion events may be captured as gaps in the motif alignment, insertion events cannot be captured as the length of the motif is fixed. This is because in a typical HMM sequence model the state transitions between sites within motifs are designed to be deterministic. Thus stochastically accommodating gaps (insertion events) within motifs is not feasible. Hence, some of the actual motifs missed by the competing algorithms were "gapped" motifs. These issues

deserve further investigation.

3.6 Materials and Methods

3.6.1 The Molecular and Functional Substitution Model

We use the Felsenstein 1984 model (F84) [49], which is similar to the Hasegawa - Kishino - Yano's 1985 model (HKY85) [77] and widely used in the phylogenetic inference and footprinting literature [49, 123], for nucleotide substitution in our motif and background phylogeny. Formally, F84 is a five-parameter model, based on a stationary distribution $\pi \equiv [\pi_A, \pi_T, \pi_G, \pi_C]'$ (which constitutes three free parameters as the equilibrium frequencies sum to 1) and the additional parameters κ and ι which impose the transition/transversion bias. According to this model, the nucleotide-substitution probability from an internal node c to its descendant c' along a tree branch of length b can be expressed as follows:

$$P_N(V_{c'} = j | V_c = i, \beta) = e^{-(\kappa+\iota)b}\delta_{ij} + e^{-\iota\beta}(1 - e^{-\kappa\beta})\Big(\frac{\pi_j}{\sum_h(\pi_h\epsilon_{jh})}\Big)\epsilon_{ij} + (1 - e^{-\iota\beta})\pi_j, \quad (3.3)$$

where i and j denote nucleotides, δ_{ij} represents the Kronecker delta function, and ϵ_{ij} is a function similar to the Kronecker delta function which is 1 if i and j are both pyrimidines or both purines, but 0 otherwise. The summation in the denominator concisely computes purine frequency or pyrimidine frequency.

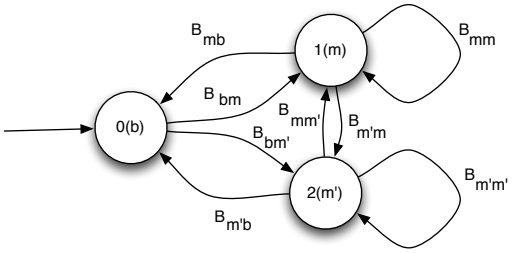


Figure 3.10: A 3-state HMM for a single motif.

To model functional turnover of aligned substrings along functional phylogeny T_f, we additionally define a substitution process over two characters (0 and 1) corresponding to presence or absence of functionality. Now we use the single parameter Jukes-Cantor 1969 model (JC69) [86] for functional turnover due to its simplicity and straightforward adaptability to an alphabet of size 2. The transition probability along a tree branch of length β (which represents the product of substitution rate μ and evolution time t, which are not identifiable independently,) is defined by:

$$\mathbf{P}_F = \begin{pmatrix} \frac{1}{2} + \frac{1}{2}e^{-2\beta} & \frac{1}{2} - \frac{1}{2}e^{-2\beta} \\ \frac{1}{2} - \frac{1}{2}e^{-2\beta} & \frac{1}{2} + \frac{1}{2}e^{-2\beta} \end{pmatrix}. \quad (3.4)$$

We perform maximum likelihood estimates of the phylogeny parameters, for details refer to Appendix.

3.6.2 Computing Complete- and Partial-Alignment Likelihood

A complete phylogenetic tree $T \equiv \{\tau, \pi, \beta, \lambda\}$ with internal nodes $\{V_i\ ;\ i = 1 : K'\}$ and leaf nodes $\{V_i\ ;\ i = K' + 1 : K\}$, where K denotes the total number of nodes (i.e., current and ancestral species) instantiated in the tree and the node indexing follows a breath-first traversal from the root, defines a joint probability distribution of all-node configurations (i.e., the nucleotide contents at an aligned site in all species instantiated in the tree), which can be written as the following product of nt-substitution probabilities along tree branches:

$$P(V_1, \ldots, V_K) = P(V_1) \prod_{i=2}^{K} P_N(V_i | V_{pa(i)}), \tag{3.5}$$

where $V_{pa(i)}$ denotes the parent-node of the node i in the tree, and the substitution probability $P_N()$ is defined by Eq. (8.2). For each position l of the multiple alignment, computing the probability of the entire column denoted by A_l of aligned nucleotides from species corresponding to the leaves of a phylogenetic tree $T^{(l)}$ defined on position l, i.e., $P(A_l|T^{(l)})$, where A_l correspond to an instantiation of the leaf nodes $\{V_i\ ;\ i = K' + 1 : K\}$, takes exponential time if performed naively, since it involves the marginalization of all the internal nodes in the tree, i.e.,

$$P(A_l|T^{(l)}) = \sum_{\mathbf{v}_{1:K'}} P(\mathbf{V}_{1:K'} = \mathbf{v}_{1:K'}, \mathbf{V}_{K'+1:K} = A_l). \tag{3.6}$$

We use the Felsenstein pruning algorithm [48], which is a dynamic programming method that computes the probability of a leaf-configuration under a tree from the bottom up. At each node of the tree, we store the probability of the subtree rooted at that node, for each possible nucleotide at that node. At the leaves, only the probability for the particular nucleotide instantiated in the corresponding taxon is non-zero, and for all the other nucleotides, it is zero. Unlike the naive algorithm, the pruning algorithm requires an amount of time that is proportional to the number of leaves in the tree.

We use a simple extension of this algorithm to compute the probabilities of a partial-alignment A'_l defined earlier under a marginal phylogeny, which is required in the coupled-pruning algorithm for CSMET, by considering only the leaves instantiated in A'_l (but not in $A''_l \equiv A_l \setminus A'_l$) that is under a subtree $T'^{(l)}$ that forms the marginal phylogeny we are interested in. Specifically, let A''_l correspond to possible instantiations of the subset of nodes we need to marginalized out. Since we already how to compute $P(A_l|T^{(l)})$ via marginalization over internal nodes $\mathbf{V}_{1:K'}$, we simply further this marginalization over leaf nodes $\mathbf{V}''$ that corresponds to taxa instantiated in A''_l, i.e.,

$$P(A'_l|T'^{(l)}) = \sum_{A''_l} P(A'_l, A''_l|T^{(l)}) = \sum_{A''_l} \sum_{\mathbf{v}_{1:K'}} P(\mathbf{V}_{1:K'} = \mathbf{v}_{1:K'}, \mathbf{V}'' = A''_l, \mathbf{V}' = A'_l), \tag{3.7}$$

where $\mathbf{V}' \equiv \mathbf{V}_{K'+1:K} \setminus \mathbf{V}''$ denotes the leaves instantiated in A'_l. This amounts to replacing the leaf-instantiation step, which was originally operated on all leaves in the Felsenstein pruning algorithm, by a node-summation step over those leaves in $\mathbf{V}''$. In fact, in can be easily shown that this is equivalent to performing the Felsenstein pruning only on the partial tree $T'^{(l)}$ that directly shadows A'_l, which is a smaller tree than the original $T^{(l)}$, and only requires time $O(|A'_l|)$.

3.6.3 Computing the Block-Emission Probabilities

Under the CSMET model, to perform the forward-backward algorithm for either motif prediction or unsupervised model training, we need to compute the emission probability given each functional state at every alignment site. This is nontrivial because a CSMET is defined on an alignment block containing whole motifs across taxa rather than on a single alignment-column. We adopt a "block-approximation" scheme, where the emission probability of each state at a sequence position, say, t, is defined on an alignment block of length L started at t, i.e., $P(\mathbf{A}_t|z_t^r)$, where $\mathbf{A}_t \equiv (A_1(t), A_2(t), \ldots, A_L(t))$, and $A_l(t)$ denotes the lth column in an alignment block started from position t.

The conditional likelihood $\mathbf{A}_t$ given the nucleotide-evolutionary trees T and T_b coupled by the annotation tree T_a under a particular HMM state s_t is also hard to calculate directly, because the leaves of the two nucleotide trees are connected by the leaves of the annotation tree (Fig. 3.1b). However, if the leaf-states of the annotation tree are known, the probability components coming from the two trees become conditionally independent and factor out (see Eq. (3.2)). Recall that for a motif of length L, the motif tree actually contains L site-specific trees, i.e., $T_m \equiv (T_m^{(1)}, \ldots, T_m^{(L)})$, and the the choice of these trees for every site in the same row (i.e., taxon), say, a_i^t in the alignment block $\mathbf{A}_t$, is coupled by a common annotation state Z_t^i. Hence, given an annotation vector Z_t for all rows of $\mathbf{A}_t$, we actually calculate the probability of two subset of the rows given two subtrees (i.e., marginal phylogenies) of the original phylogenetic trees for motif and backgrounds, respectively (Fig. 3.1b).

The subset $\mathbf{A}'_t \equiv \{a_i(t) : s.t.\ Z_t^i = 1\}$ is constructed by simply stacking the DNA bases of those taxon for which the annotation variables indicate that they were generated from the motif tree. The subtree T'_m is constructed by simply retaining the set of nodes which correspond to the chosen subset, and the ancestors thereof. Similarly we have $\mathbf{A}''_t$ and T'_b. Hence, we obtain

$$P(\mathbf{A}_t|Z_t = \mathbf{z}_t, T_m, T_b) = P(\mathbf{A}'_t|T'_m)P(\mathbf{A}''_t|T'_b) = \prod_{l=1}^{L} P(A'_l(t)|T_m^{\prime(l)})P(A''_l(t)|T'_b). \tag{3.8}$$

The probability of a particular leaf-configuration of a tree, be it a partial or complete nucleotide tree, or an annotation tree, can be computed efficiently using the pruning algorithm. Thus for each configuration of $\mathbf{z}_t$, we can readily compute $P(\mathbf{A}_t|Z_t = \mathbf{z}_t, T_m, T_b)$ and $P(\mathbf{z}_t|T_f, Z_t^r = z_t^r)$. The block emission probability $P(\mathbf{A}_t|z_t^r)$ under CSMET can be expressed as:

$$P(\mathbf{A}_t|z_t^r) = \sum_{\mathbf{z}_t} P(\mathbf{A}_t, \mathbf{z}_t|z_t^r) = \sum_{\mathbf{z}_t} P(\mathbf{A}'_t(\mathbf{z}_t)|T'_m(\mathbf{z}_t))P(\mathbf{A}''_t(\mathbf{z}_t)|T'_b(\mathbf{z}_t))P(\mathbf{z}_t|T_a, z_t^r) \tag{3.9}$$

where we use $\mathbf{A}'_t(\mathbf{z}_t)$, $\mathbf{A}''_t(\mathbf{z}_t)$, $T'_m(\mathbf{z}_t)$ and $T'_b(\mathbf{z}_t)$ to make explicit the dependence of the partial blocks and marginal trees on functional indicator vector $\mathbf{z}_t$. We call this algorithm a *coupled-pruning algorithm*.

Note that in this algorithm we need to sum over a total number of 2^M configurations of $\mathbf{z}_t$ where M is the total number of taxa (i.e., rows) in matrix $\mathbf{A}_t$. It is possible to reduce the computational complexity using a full junction tree algorithm on CSMET, which will turn the graphical model underlying CSMET into a clique tree of width (i.e., maximum clique size) possibly smaller than M.

But this algorithm is complicated and breaks the modularity of the tree-likelihood calculation by the coupled-pruning algorithm. In typical comparative genomic analysis, we expect that M will not be prohibitively large, so our algorithm may still be a convenient and easy-to-implement alternative to the junction-tree algorithm. Also this computation can be done off-line and in parallel.

3.6.4 Posterior Inference Under CSMET

Given the emission probabilities for each ancestral functional state at each site, we use the forward-backward algorithm for posterior decoding of the sequence of ancestral functional states $Z^r_{1:N}$ along the input CRM alignment of length N. The procedure is the same as in a standard HMM applied to a single sequence, except that now the emission probability at each site, say with index t, is defined by the CSMET probability over an alignment block $\mathbf{A}_t$ at that position under an ancestral functional state Z^r_t, rather than the conditional probability of a single nucleotide observed at position t as in the standard HMM. The complexity of this FB-algorithm is $O(Nk^2)$ where k denotes the total number of functional states. In this work, we only implemented a simple HMM with one type motif allowed on either strand, so that $k = 3$. We defer a more elaborate implementation that allows multiple motifs and encodes sophisticated CRM architecture as in LOGOS [207] to a future extension.

Given an estimate of $\mathbf{Z}^r_{1:N}$, we can infer the MAP estimates of Z^i_t – the functional annotation of every site t in every taxon i of the alignment. Specifically, the posterior probability of a column of functional states Z_t under ancestral functional state z^r_t can be expressed as:

$$P(Z_t|\mathbf{A}_t, Z^r_t = z^r_t) = \frac{P(Z_t, \mathbf{A}_t|Z^r_t = z^r_t)}{P(\mathbf{A}_t|Z^r_t = z^r_t)} = \frac{P(\mathbf{A}_t|Z_t)P(Z_t|Z^r_t = z^r_t)}{P(\mathbf{A}_t|Z^r_t = z^r_t)}. \tag{3.10}$$

Recall that in the coupled-pruning algorithm, we can readily compute all the three conditional probability terms in the above equation.

Performing posterior inference allows us to make motif predictions in two ways. A simple way is look at blocks in the alignment at which the posterior inference produces ones, and predict those to be motifs. Alternatively, we can also use the inferred state of the alignment block together with the inferred ancestral state to compute a probability score (as a heuristic) based on the functional annotation tree. The score for the block is the sum of probabilities of each block element being one.

3.6.5 Tree Estimation

Given blocks of aligned substrings $\{\mathbf{A}_t\}$ containing motif instances in at least one of the aligned taxa, in principle we can estimate both the *annotation tree* $T_f \equiv \{\alpha, \tau_f, \beta_f\}$ and the *motif trees* $T_m \equiv \{\theta, \tau_m, \beta_m, \lambda_m\}$ based on a maximum likelihood principle. But since in our case most training CRM sequences do not have enough motif data to warrant correct estimation of the motif and function tree, we use the topology and branch lengths of a tree estimated by fastDNAml [137] from the entire CRM sequence alignment (containing both motif and background) as the common basis to build the T_f and T_m. Specifically, fastDNAml estimates a maximum likelihood tree under the F84 model from the entire CRM alignment; we then scale the branch lengths of this tree to

get the sets of branch lengths for T_f and T_m by doing a simple linear search (see below) of the scaling coefficient that maximize the likelihood of aligned motif sequences and aligned annotation sequences, under the T_m and T_f (scaled based on the coefficients) respectively.

For simplicity, we estimate the background tree $T_b \equiv \{\theta_b, \tau_b, \beta_b, \lambda_b\}$ separately from only aligned background sequences that are completely orthologous (i.e., containing no motifs in any taxon).

For both motifs and background phylogenies, the Felsenstein rate parameter μ for the corresponding nucleotide substitution models must also be estimated from the training data. Ideally, the optimal value of the rate parameter should be obtained by performing a gradient descent on the likelihood under the corresponding phylogeny with respect to the Felsenstein rate parameter μ. However, due to the phylogenetic tree likelihood terms involved in the likelihood computation, there is no closed form expression for the gradient that can be evaluated for a specific value of the rate parameter to determine the direction to choose for optimization. Therefore, to find an approximation to the optimal value of μ, we again perform a simple linear search in the space of μ. For example, to find the Felsenstein rate parameter for motif evolution: μ_{minl} and μ_{maxl} are lower and

---

for $\mu = \mu_{minl}$ to $\mu = \mu_{maxl}$ in steps of δ **do**
  $L(\mu)$= Training motif likelihood under motif phylogeny T with Felsenstein rate μ
end for
Choose μ that gives maximum likelihood: $\mu_{best} = \text{argmax}_\mu\, L(\mu)$

---

upper bounds respectively on the space of μ that is searched, and are heuristically chosen based on observation. The step δ can be chosen to be as small as desired or is allowable, since having a smaller δ increases the number of values of μ that must be tested and hence increases computation, but gives a more accurate optimum.

More technically, for the motif phylogeny, the scaling coefficient and the Felsenstein rate parameter should be optimized jointly, for example via a gradient ascent in 2-d parameter space. However, that is impractical since there is no closed form expression for the gradient of the likelihood with respect to either parameter. So we chose to optimize each parameter separately by a heuristic iterative linear search. At convergence, this gives an approximation to the optimal values of the parameter.

3.6.6 Estimation of HMM parameters

For prediction of motifs and non-motifs on test sequences, we use an HMM to find the highest probability state (forward or reverse motif/ background) at each site. The parameters for the HMM are the initial probability vector π and the transition probability matrix $\mathbf{B}$. Figure 3.10 shows the state space of the HMM.

The initial probabilities are fixed by assuming that the HMM always starts in the background state. Thus $\pi_0 = 1$ and $\pi_1 = \pi_2 = 0$. For the transition matrix, we use the maximum likelihood estimator for transition from state i to state j (which has probability $B_{i,j}$), this is given by the count of the number of such events in the training data divided by the total number of sites in state i. We follow the no-strand-bias assumption, and allow equal transition probabilities from the background

state to both the forward-motif and reverse-motif states. Also, in the case where we do not have annotated training alignments, we can use the Baum-Welch algorithm for unsupervised estimation of the transition probability matrix.

3.6.7 Comparison of CSMET to available software

We compare CSMET with four other software - PhyloGibbs, EMnEM, PhyloHMM and Stubb.

PhyloGibbs is chosen as it is presently a state of the art in multi-species motif detection [171] and it handles motif turnover. PhyloGibbs is an unsupervised algorithm for *de novo* motif detection, and it can also optionally run in supervised mode given PWM for motif search. For a fair comparison, we run PhyloGibbs by specifying the motif PWM based on a maximum likelihood estimation from training data. We run PhyloGibbs with the default set of parameters. We approximately specify the number of motifs expected to be seen, as needed by PhyloGibbs, since the actual number of conserved motifs can vary a lot in both our simulated data and in real biological data.

EMnEM is chosen as it is another popular multi-species motif detection algorithm based on a different phylogenetic model that does not handle motif turnover and evolutionary-rate autocorrelation. EMnEM performs *de novo* motif detection, but also has a supervised motif search mode, which we choose to operate on. Again, we also approximately specify the number of motifs expected to be seen, and run EMnEM with the default set of parameters.

PhyloHMM is chosen since it is a direct analog of CSMET, which assumes functional homogeniety across aligned sites. Available PhyloHMM-based tools are implemented for detecting genes [123] and conserved regions [121, 172], but no PhyloHMM implementations were available for motif finding. Hence, we implemented our own in-house PhyloHMM for the purpose of supervised motif detection.

Finally, Stubb is chosen as a representative single-species HMM based motif finder to investigate the advantage of comparative-genomic motif detection over traditional approaches that treat each species independently. Stubb can be run both as a single species or as an aligned two species model. Since we are interested in comparing our performance with single species motif detector, we use the single species mode. Also, it might not always be apparent as to which two species to compare in order to get the most meaningful contrast for separating functional sites and non-functional sites. Stubb was run individually on all the aligned sequences, with all the results collated for analysis.

3.6.8 Drosophila CRM data processing and experimental setup

Our dataset was created based on the motif database in [12, 142], from which we chose to predict TFBS of TF which have at least 10 or more biologically validated training instances. The five TFs which met this requirement were Bicoid, Caudal, Kruppel, Knirps and Hunchback motifs. Motif finding was performed on 14 CRMs listed in Table 1 which contained instances for these 5 binding sites. The multiple sequence alignment corresponding to the CRMs were obtained by using the UCSC Genome Browser pre-compiled alignments [52]. The sequence corresponding to D. willistoni was left out due to poor alignment quality and missing contigs. Flanking regions of

1000 bp on each side of the CRMs were also analyzed. For each CRM alignment, we use the motifs identified in *melanogaster* as references to mark all alignment blocks that contain at least one instance of motifs among the 11 taxa to be analyzed. [1] The *melanogaster* CRMs contain both biologically validated motifs and computationally identified but plausible motifs, as documented in [12, 142].

To train the CSMET, we manually annotated the functional states (i.e., Z_t) across all taxa in all alignment blocks (i.e., $\mathbf{A}_t$) containing the *melanogaster* motif. We employ a 1 versus $K-1$ cross-validation scheme for testing on each motif type, where K is the total number of CRMs where a motif type is present. Specifically, for each motif type we trained all programs on $K-1$ out of the K CRMs hosting the motif, and tested them on the remaining one, and we iterated this until all K CRMs had been tested. Recall that the test accuracy is accessed only for reported motifs in *melanogaster*, but not on those manually annotated ones in other taxa.

To avoid overfitting the motif and functional phylogenies of CSMET under limited training data, for all our experiments, we used a single phylogenetic tree estimated from the entire training sequence alignment dataset as the un-scaled version of the motif and functional trees. We assumed that the T_f's of every type of motif share the same topology and branch lengths, but different equilibriums. Thus T_f can be fitted from a concatenation of motif-instance alignments of all types of motifs. For the motif sequence phylogenies, we enforced the trees at every site in the same motif have the same topology, branch length, and the Felsenstein total substitution rate, but different equilibriums. A second tree was estimated on background sites only, and was used as the background phylogeny.

To handle real data which contains gaps and other complexities, it is necessary to change some settings of the competing software from their defaults to ensure proper behavior. EMnEM was run with default parameters, but with the threshold set to 0.999 to reduce false positives; as for the suggested threshold of 0.5, virtually every location was being classified as a motif. PhyloGibbs was run with default parameters, but for handling gaps, the modes of using the full alignment, as well as using partial alignments were tried, and the pre-estimated phylogeny on all species for the entire sequence was given to it. PhyloHMM was run naively using posterior decoding. Stubb was run with default settings with a slightly reduced threshold of 6.0. At the suggested threshold of 10.0 for a window size of 500, Stubb predicts no true positives.

[1] As a result our benchmark is biased toward *melanogaster*, because annotations in other taxa are not available to mark motifs that are present in other Drosophila taxa but not in *melanogaster*.

Chapter 4

CRFs for correlating genetic and epigenetic features with binding sites

4.1 Related work

Discriminative models make it easier to incorporate various sources of evidence for predicting TFBS locations, while at the same time keeping the estimation and inference procedures simple. We developed DISCOVER, a discriminative method for motif detection in higher eukaryotic genomes that enjoys the dual advantage of modelling CRM architecture of sequences and features of individual motifs. It is a Conditional Random Field (CRF) model [96], which incorporates a wide range of both CRM structure-based and individual motif-based features. CRFs have previously been used in sequence analysis, most notably in gene prediction [37, 71], since coding regions are much better characterized in terms of sequence level features with respect to regulatory regions. Craven *et al* [17] has applied a similar scheme to identify regulatory signals in prokaryotic sequences; but their model employs a simple feature set to resolve the motif sequence overlap problem, and also requires a prescreening of motif scores via basic PWM-based models. Our method is important in several respects in the context of the literature. Firstly, it is a discriminative model explicitly tailored towards maximizing the likelihood of predicting motifs, rather than maximizing the joint likelihood - which often confounds the analysis in the case of generative models. Secondly, it employs a comprehensive set of features carefully selected from the literature designed to capture a variety of characteristics of the motif and CRM patterns. Thirdly, it is an integrative model which allows sequence specific features to be added at will to enhance the prediction scheme. Further, since feature scores are computed offline, it is easier to incorporate scores involving complicated computation and long computation times as well as long range dependencies. We evaluate the CRF model on both simulated CRMs and actual biologically validated transcription regulatory sequences of *Drosophila melanogaster*, in comparison with a wide spectrum of existing models including, Cister [56], Cluster-Buster [55], BayCis [107], MSCAN [1], Ahab [148] and Stubb [175]. The results suggest that our proposed method significantly outperforms others on real *Drosophila* sequences.

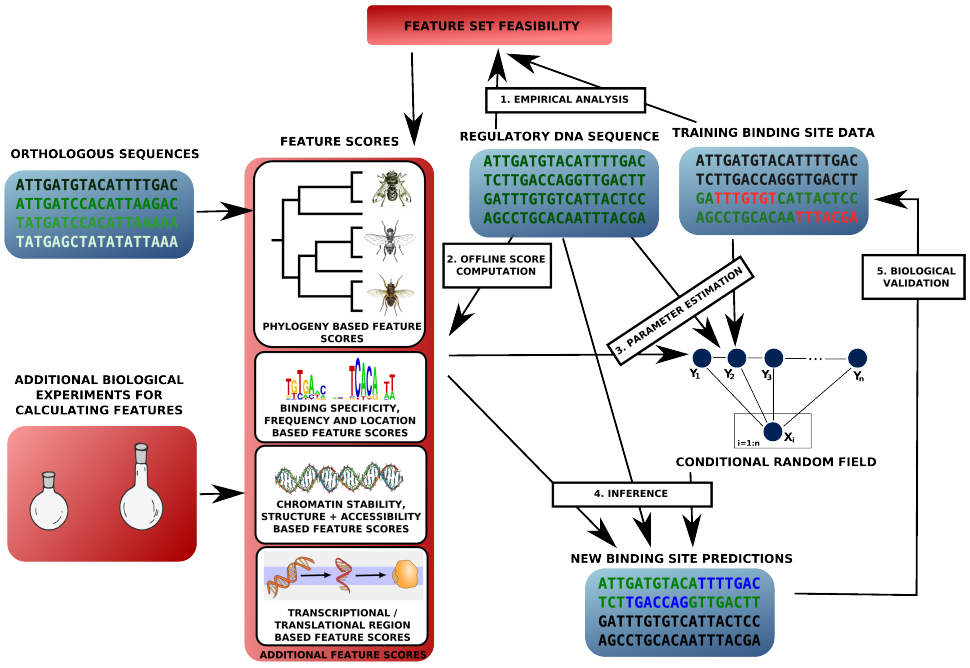


Figure 4.1: A schematic of the workflow. From [57].

4.2 The discriminative model

The conventional PWM representation for transcription factor binding sites is not discriminative enough to distinguish true binding sites from false binding sites. We desire a model for TFBSs and genomic sequence that supports a more complex motif representation without losing the ability to characterize sequence-wide properties, which means a flexible feature design. The CRF model - a feature-based log-linear model in which features are easily incorporated - is an appropriate model choice under the circumstances. The basic inputs to such a computational model is a set of genetic sequences, a set of feature values corresponding to every nucleotide in the sequences, and the PWMs of TFs which are being predicted. The output of the model is a prediction of a set of TFBSs which are being predicted, ranked in order of decreasing likelihood. The CRM boundaries can also be similarly predicted, but we focus on the analysis of the TFBS predictions. A CRF model that describes a conditional probability distribution of a genomic sequence is defined as:

$$P(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{Z} \exp\left\{\boldsymbol{\lambda} \cdot \mathbf{F}(\mathbf{y}, \mathbf{x})\right\} \tag{4.1}$$

$$\text{where} \qquad Z = \sum_{\mathbf{y}} \exp\left\{\boldsymbol{\lambda} \cdot \mathbf{F}(\mathbf{y}, \mathbf{x})\right\} \tag{4.2}$$

where we use x_i to represent the type of the observed nucleotide at site i in a sequence, and y_i to represent the hidden state associated with x_i, which corresponds to the functionality of the site in the genomic sequence. The value of a hidden state is also called a state label. Vector

$\mathbf{x} = \{x_i : i = 1, 2, \ldots, L\}$, and vector $\mathbf{y} = \{y_i : i = 1, 2, \ldots, L\}$, where L is the length of the sequence. Vector $\mathbf{F}$ is the set of features, each element F of which is the sum of feature scores of a particular feature category (where feature scores refer to the numerical value of the feature). Vector $\boldsymbol{\lambda}$ corresponds to the feature weights assigned to the set of features, and is learnt from data to decide which features may be more important in predicting TFBSs. Z is a partition function that normalizes the pdf and is a function of $\mathbf{x}$ and $\boldsymbol{\lambda}$. The value space for each x_i is {A,C,G,T}. The values represent the four types of nucleotide in DNA, *adenine*, *cytosine*, *guanine* and *thymine* respectively. The value space for hidden states y_i, however, is not so straightforward, and it will be defined in next subsection.

State Design: We design a set of hidden states based on the possible functionality of each nucleotide in the genomic sequence being analyzed.We incorporate each motif type as a state since this is our prediction goal. We number the types of motifs and name the state for the m-th motif type $\mathbf{M}^{(m)}$. Representation-wise, a hidden state y_i being state $\mathbf{M}^{(m)}$ implies that a motif of the m-th type is located starting at site i of the sequence. Those states are all that we need to represent binding sites. Next, we know that transcription factors are usually working together to regulate genes, especially in genomes of higher organisms. In order to work together, different types of TFBSs often lie close to each other in the range of hundreds of base pairs forming a so-called *cis*-Regulatory Module [35]. We use state **C** to represent all nucleotides in the CRM regions except those binding sites which have already been labeled as Ms. The nucleotides which are still unlabeled after the first two rounds are set to state **G**, which represents a global background in the genomic sequence. Hence the set of hidden states for modelling the functionality at a nucleotide position is given by **S={G,C,M**$^{(1)}$**,...,M**$^{(N_M)}$**}**, where N_M is the number of motif types. We do not allow two motifs to share the same starting position, but such occurrences are infrequent. It is still an improvement on HMM-based approaches where modelling even partial overlap of motifs causes a combinatorial increase in the state space. Overlapping of starting positions of TFBSs can be accommodated in our model by using marginal probabilities in the prediction step.

Feature Design: Each element $F(\mathbf{y}, \mathbf{x})$ of vector $\mathbf{F}(\mathbf{y}, \mathbf{x})$ in Eq 4.1 is the sum of feature scores of a particular feature category, where feature score simply refers to the numerical value of the feature. It sums up feature function f's over the sequence, which have a common meaning and share the same weight. An example is shown in Eq 9.5, after we see some concrete features. The design of fs is a critical part of CRF models. We include a rich set of features, most of which are introduced in the Results section. Features with a one-to-one correspondence with nucleotide base pairs can be easily integrated into the framework by defining as:

$$f(y_i, \mathbf{x}) = \left(\sum_m \delta(y_i, \mathbf{M}^{(m)}) \right) S(i, \mathbf{x}) \tag{4.3}$$

where $S(i, \mathbf{x})$ is the feature score, All features are in the form of $f(\mathbf{y}, \mathbf{x})$, but as for now, they are have a simpler common form of $f(y_i, y_{i+1}, \mathbf{x})$, which we called a chain-structure CRF model.

Model Parameters: Feature weights constitute the set of model parameters, some of which are fixed and some are free to be estimated. More free parameters make the CRF model more complex, which might be harder to learn. The set of free parameters are modelled to avoid redundant parameters, which will not make any contribution. Also, parameters that are not likely to be properly

estimated from training data should never be included, because including them will only increase the chance of over-fitting the model. Our focus is on the weight of state transition features, because they account for a large proportion of the whole parameter set and good estimation of the weights are critical for successfully predicting TFBSs. In the CRF model, we assign a parameter as a weight to each of the features defined previously which are collectively the vector $\boldsymbol{\lambda}$ in Eq 4.1. Not all of these parameters are free parameters. Among state transition parameters, we constrain an M state to be only directly reachable from a C state, and not from a G state, since motifs are not present outside CRMs. Thus, state transition features corresponding to taboo transitions have a weight $-\infty$ (a low enough number in practice), meaning that the transitions never occur in the CRF model. However, we want to have a reasonable number of free model parameters as more free parameters increase the expressibility of the model. With increase in the number of free parameters, the hardness of estimating model parameters increase, the running time of the learning algorithm also rises, and some parameters may overfit due to data scarcity for corresponding features.

4.2.1 Model Training and Inference

In this section, we briefly describe the model training and inference procedures in which feature weights of the CRF model are learnt from training data and subsequently used to make TFBS predictions.

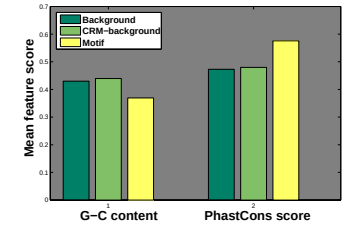
Model training: Firstly, a learning criterion is set up, which can either be to maximize likelihood or maximize posterior probability. It is then converted to a convex optimization problem, and finally a Quasi-Newton method is applied. Our goal here is to learn the best setting for $\boldsymbol{\lambda}$, the weights of features in the CRF model given a set of sequences as training data with their nucleotide types $\mathbf{x}$ and state labels $\mathbf{y}$. The value of feature functions $\mathbf{f}$ can be computed given necessary hyper-parameters.

A reasonable criteria to learn the feature weights $\boldsymbol{\lambda}$ from nucleotide types $\mathbf{x}$ and state labels $\mathbf{y}$ (or more precisely from feature values $\mathbf{f}$) in a CRF model is to maximize likelihood of $\boldsymbol{\lambda}$ wrt $\mathbf{y}$ conditioned on $\mathbf{x}$, which equals the probability of state labels $\mathbf{y}$ given feature weights $\boldsymbol{\lambda}$ conditioned on nucleotide types $\mathbf{x}$, because the probability model itself is defined in this conditional scheme. The max likelihood estimator of $\boldsymbol{\lambda}$ can be expressed as:

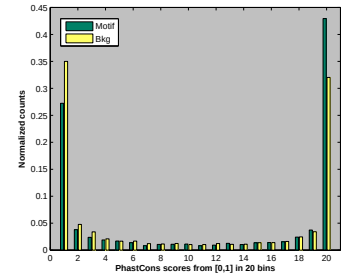
$$\widehat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} L(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x})$$

$$\text{where} \qquad L(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x}) = P(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\lambda})$$

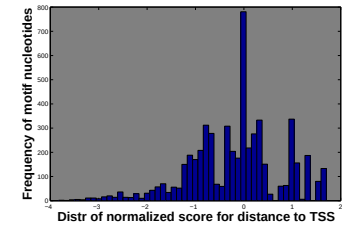
Inference: The learnt feature weights of the CRF model are used to predict TFBSs on a new genomic sequence - the inference step. There are two categories of prediction schemes analogous to the popular inference schemes for HMMs: sequence decoding



(a)



(b)



(c)

Figure 4.2: (a) Means of two discriminative features plotted for G-C content and PhastCons score for Motifs, CRMs, and background nucleotides ,(b) Distribution of PhastCons scores in motifs vs non-motifs (c) Multimodal empirical distribution of feature values for the transformed Distance to TSS feature. From [57].

by *Viterbi* algorithm and marginal decoding by Forward-Backward algorithm. We chose the marginal probability rank scheme as it enables us to predict overlapping TFBSs. Marginal decoding considers one hidden state at a time, making predictions based on the marginal probability, $P(y_i \mid \mathbf{x}, \boldsymbol{\lambda})$, which can be computed by the dynamic programming Forward-Backward algorithm in a chain-structure CRF model [96, 167]. Variants on the marginal decoding scheme include **maximum a posteriori decoding** (MAP) where we predict a TFBS if the marginal probability of it is the highest among all state labels

$$\widehat{y_i} = \arg\max_{y_i} P(y_i \mid \mathbf{x}, \boldsymbol{\lambda}) \tag{4.4}$$

Alternatively, we make a positive prediction whenever the marginal probability is above a threshold, known as **threshold decoding**. It is a flexible method, but a good threshold is hard to set in practice. We use a similar scheme that takes advantage of thresholding by choosing a threshold automatically by limiting the number of predictions. Thus we calculate a list of TFBS and marginal probability pairs, sort them by probability in descending order, and output the top P ones as predictions, P being the number of desired predictions. We make P for each sequence proportional to its length L, as a longer sequence tends to contain more TFBSs. The coefficient $k = P/L$ is called *prediction factor*. We call this **rank decoding**.

4.3 Framework and experiments using genetic and epigenetic data

We evaluate our method of TFBS prediction on a set of real genomic Transcription Regulatory Sequences (TRSs) of *Drosophila melanogaster*, as well as a set of synthetic TRSs. The prediction performance is compared with 6 popular published methods for supervised discovery of motifs/CRMs based on a wide spectrum of models: Cister [56], Cluster-Buster [55], BayCis [107], Stubb [175], Ahab [148] and MSCAN [84]. In general, the prediction performance of the CRF model is superior or competitive wrt all the chosen benchmark methods on this comprehensive selection of real *D. melanogaster* dataset. The semi-synthetic dataset was generated by artificially simulated CRM structures with a 3rd-order Markov model for background sequences and planting real TFBSs from the TRANSFAC database [202] into the simulated background sequences based on the generative model for the HMM-based TFBS prediction tool Baycis and published in Lin *et al* [107]. It involves 30 20kbp-long sequences, containing 887 TFBSs of 10 types. The real *D. melanogaster* binding site data was obtained from the *Drosophila cis*-regulatory Database at National University of Singapore [132]. The PWM and CRM boundary data were obtained independently of the binding site database from the REDfly CRM database [61]. This TRS dataset was previously published in Lin *et al* [107]. The dataset contains 97 CRMs pertaining to 35 early developmental genes of *Drosophila melanogaster* (in 35 sequences). Each of the 35 sequences contains 1 to 4 CRMs. The lengths of sequences range from 10 thousand base pairs to 16 thousand base pairs, except two extremely long sequences whose length are around 40k bps and 79k bps respectively. There are 700 TFBSs of 44 types labeled in the dataset in all. It is worthwhile noticing that 12 out of the 44 types appear in only one sequence, which account for 10 percent of

the binding sites. A visualization of the dataset illustrating the locations of TFBSs and CRMs is presented in Fig 4.3.

4.3.1 Input features

We include a rich set of features in our model, based on previous findings in the literature as well as some derived features which empirical evidence suggests are more discriminative than the original features from which they were derived. Most of the feature scores are accurately or heuristically calculated based solely on the sequence data, but some require external annotation (like translated and transcribed regions, and transcription start site). It is also easy to change feature values from sequence-derived heuristic values to actual experimental results should they become available. See the work schematic (Fig 5.1) for a visual schema of feature calculation.

CRFs adjust feature weights based on training data, so it is also interesting to try new features to check if they improve the predictive power of the model. Binding site positioning and characterization of the nucleotide content of binding sites in terms of binding site specificity have been the most standard features which have been used in motif finding, especially in generative models like HMMs. This is based on sound biological validation of the fact that specificity of binding sites and CRM "architecture"s are pervasive in regulatory regions [35].

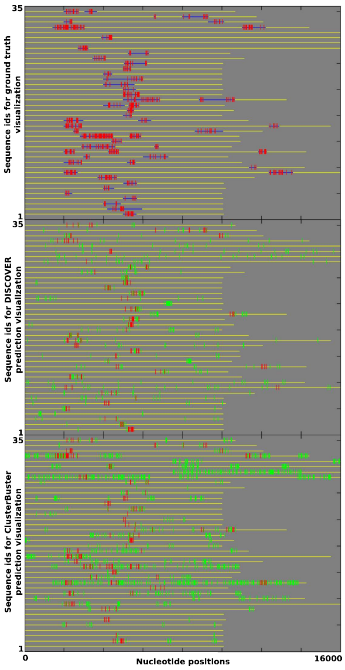


Figure 4.3: Aligned data and prediction visualizations with CRMs in blue, ground truth and true positive TFBSs in red and false positive TFBSs in green. Very long sequences are broken in two for ease of depiction. From [57].

PWM Constraints: The basic feature we use is the PWM constraint, which implements the information present in the PWM of a motif. It represents the binding specificities of the DNA binding domain(s) of the TF in question as an ordered set of multinomials, and is an indicator of the level of evolutionary constraint, and hence selection each nucleotide is under. Some PWMs tend to be more constrained (under greater purifying selection) than others. Some PWMs also tend to suffer from noisy data. Because of this, the discriminative power of the PWM constraints feature varies from PWM to PWM. For PWMs with poor discriminative power, additional features are critical for improving predictability. The PWM score provides a good baseline measure for the CRF model in motif prediction, though it is not an essential feature in our model.

State Transition: State transition features are an effort to model the architecture of the regulatory region. The state transition feature models the relationship between the functionality of neighboring nucleotides, which correspond to neighboring states in the CRF and is based on the differing likelihoods of the hidden CRF states transitioning from one to the other. Evolutionary conservation and presence or absence of evolutionary events like duplication and repeats can also play a role in identifying TFBS, as evidenced by the large body of work in phylogenetic motif finding. The basic premise in such cases is that functionally relevant nucleotides like TFBS would be under selection, and would hence be distinguishable from

surrounding sequence on the basis of evolutionary parameters. While we do not explicitly use multiple species sequence data, we implicitly use evolutionary data in terms of feature data.

Presence of Repeats: Interspersed repeats and low complexity DNA sequences are common elements in the genome, often near coding regions and inside regulatory sequences. The repeat feature is a simple single nucleotide based feature indicative of whether that nucleotide is part of a repeat as predicted by RepeatMasker using the repeat database RepBase [87]. On one hand, repeats with motif-like patterns may lead to a large number of false positive results, but repeats have also been reported to have been under purifying selection [21] and to have been harnessed into the regulatory machinery [88]. Thus, instead of masking out repeats to lower the false positive rate, we choose to identify repeats in the sequence in a bid to find locational correlations with TFBSs.

PhastCons Score & related features: We use the PhastCons score as an evolutionary score based feature. PhastCons [121] is a phylogenetic 2-state HMM which predicts if nucleotide positions in a multiple alignment are in an evolutionarily conserved state or not. The PhastCons score at a nucleotide position is merely the posterior probability that the nucleotide was generated from the conserved state based on the 15-way Multiz [14] alignment of the *Drosophilae* species, *A. mellifera*, *A. gambiae* and *T. castaneum*. We also use two other derived binary features which we feel to be discriminative based on an empirical analysis of PhastCons score distributions (see Fig 4.2): " Is PhastCons score < 0.05" and "Is PhastCons score > 0.95". We also keep an additional feature indicating whether PhastCons data is available or not for bookkeeping purposes. It is well established in the literature that the distance of the TFBS to the transcription start site (TSS) plays an important role of the efficacy of the TFBS in regulating the gene [38], and of the nature of function of the TFBS [45]. We therefore incorporate several features which contain information of the distance to the TSS, the locations of the transcribed and translated regions, and the positioning of binding site with respect to the gene transcription-translational direction.

Distance to TSS & Translated: TFBS are typically present near coding sequences, and we utilize two features indicative of that fact. The binary feature "Translated" indicates at each nucleotide position whether it is translated or not by the gene translation/transcription machinery. It has also been shown that TFBSs are not uniformly distributed wrt their distance from the TSS [38], and the Distance to TSS feature is a score of the distance of each nucleotide from the transcription start site in question.

5'UTR & 3'UTR: The position of the TFBS wrt directionality of the gene being coded has been shown to be a discriminative feature for identifying TFBS. We use 2 binary features indicative of this fact, the "5'UTR" feature indicates for each nucleotide if it is located in the 5'prime untranslated region, and the "3'UTR" feature indicates likewise for the 3' prime untranslated region. Recent work in the literature has approached the TFBS prediction problem as a non-binary classification problem, instead choosing to model the affinity of a TF to bind to a particular oligonucleotide sequence with an affinity score [200]. This has led to the realization that TFBS may also be effective gene regulators in cases of low binding affinity but high chromatin stability and accessibility [141]. While we model our TFBS prediction as a sort of classification problem, we still incorporate the notions of chromatin accessibility and stability.

G-C Content & Melting Temperature: The G-C content feature of a genomic sequence or the fraction of G+C bases in a sequence is a simple heuristic which can be used to estimate several

factors reflective of the stability of the chromatin structure like the melting temperature and in higher eukaryotes is a determining factor for identifying CpG islands [212], thus being indicative of how easy it might be for a TF to actually bind in the locality. The window size w for the genomic neighborhood over which to estimate the G-C content is a hyper-parameter that must be determined ahead of time, and is usually chosen to be of the order of magnitude of the binding site. The Melting temperature feature is defined as the temperature for which half the DNA strands of an oligonucleotide are in the double helical structure, while the other half are in a random coil formation. It corresponds strongly to chromatin stability, and has been shown as a feature to correlate well with TFBS [145].

Nucleosome Occupancy: Recent research has suggested that nucleosome occupancy has a strong correlation with binding preference of TFs [165]. This is due to the non-feasibility of access to the chromatin by the TF when a nucleosome is already bound there. Some research has successfully used nucleosome occupancy scores to improve TFBS predictions [134].

We also tried several other features directly computable from sequence information, and found that the following features can help in discriminating between TFBS and non-TFBS. The cause of the discriminative power of these tracks may stem from the nature of the binding specificities of the TFs in question, and a closer investigation is warranted.

Reverse Complementarity & Conservation Symmetry: We also try two additional features for the CRF based on symmetry of the oligonucleotide in question. The Reverse Complementarity feature indicates as a fraction between 0 and 1 how similar a nucleotide sequence is to its reverse complement. It is exactly 1 only when an oligonucleotide sequence is identical to its reverse complement. The Conservation Symmetry feature models how symmetric the degree of conservation in the PWM is wrt the center of the binding site. This is based on the empirical observation that DNA binding domain binding specificities often have symmetric sequence conservation profiles. As a working example, we show how the feature is defined:

$$f_{CS}(y_i, \mathbf{x}) = \sum_m \delta(y_i, \mathbf{M}^{(m)}) \left(cs(\theta^{(m)}, x_{i:i+l^{(m)}-1}) - cs_0 \right) \tag{4.5}$$

$$\begin{aligned} cs(\theta^{(m)}, x_{i:i+l^{(m)}-1}) = & \frac{1}{\lfloor l^{(m)}/2 \rfloor} \sum_{j=1}^{\lfloor l^{(m)}/2 \rfloor} \Big| \beta(\theta_j^{(m)}, x_{i+j-1}) - \\ & \beta(\theta_{l^{(m)}+1-j}^{(m)}, x_{i+l^{(m)}-j}) \Big| \end{aligned} \tag{4.6}$$

where cs averages the conservation symmetry score over a potential binding site, cs_0 is an offset value of choice, $l^{(m)}$ is the length of the motif, and β function is the conservation score of a single base. As an example of summing the feature scores, the sum of conservation symmetry features can be computed as:

$$F_{CS}(\mathbf{y}, \mathbf{x}) = \sum_{i=1}^{L} f_{CS}(y_i, \mathbf{x}) \tag{4.7}$$

where f_{CS} is defined in Eq 9.8 and L is the length of the sequence. $F_{CS}(\mathbf{y}, \mathbf{x})$ is one of the elements in function vector $\mathbf{F}(\mathbf{y}, \mathbf{x})$ used in a CRF model in Eq 4.1.

The design of new features has exciting new possibilities. Long range regulatory effects have been reported in the literature [23]. The CRF model also readily enables us to model long range dependencies if we deviate from the chain structured CRF structure. It can also be used as a form of ensemble learning by incorporating predictions by other independent tools as features. Other features which have been shown in the literature to correlate well with the data and which are candidates for future inclusion on this and other datasets include the presence of the nucleotide in the first intron of the regulated gene, and presence of the nucleotide in the neighborhood of a CpG island. We tested the discriminative nature of these features on the dataset in Figure 4.2. 4.2(a) shows the difference in mean values for background, CRM and motif nucleotides for two of the most discriminative features : G-C content and PhastCons score. 4.2(b) shows the distribution of PhastCons scores in motif versus non-motif nucleotides, with the most discriminative bins being at either end of the score range, which offered us some insight as to how to define a derived feature which is more discriminative than the original one. 4.2(c) shows the interesting multimodal distribution of the normalized and transformed values of the feature Distance to the TSS, suggesting a complicated, non-uniform distribution worth additional investigation.

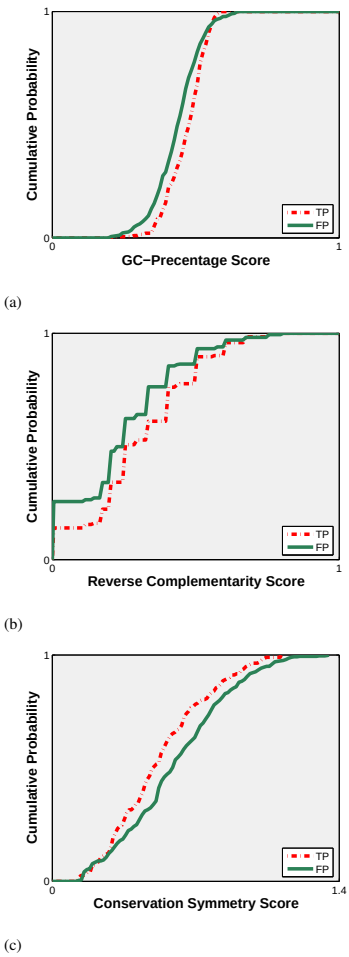


Figure 4.4: On (a) GC-percentage score, (b) reverse complementary score and (c) conservation symmetry score, a comparison of cumulative distribution function curves between TP group and FP group. From [57].

4.3.2 Experimental setup

In this part, we include biological and empirical bases for selection of some features, data preparation, hyper-parameter setting, test scheme, and evaluation scheme. For training data, we use a part of the sequences with ground truth labels. For testing, the required hyper-parameters in the CRF model are the window size used in GC-percentage calculation and pseudo-counts used to smooth the probabilities in PWMs to allow for greater tolerance in motif discovery. We set the window size of GC-percentage to 8 bps (approximately the average length of a motif) and pseudo-count for smoothing PWM probabilities to 0.5.

Our evaluation is based on a leave-one-out cross validation (LOOCV) scheme. Each time we take all but one sequences as training data, and predict on the remaining sequence by the model with parameters learnt from the training data. We use the rank decoding scheme with the prediction factor k set to 0.0015 by default. This threshold is obtained by analyzing the empirical density of TFBSs in training data. Varying the value of the threshold results in increasing one of the performance metrics of precision or recall at the cost of the other. For evaluating performance, we

use the standard definitions of precision, recall and the F1 score using counts of true positive (TP), false positive (FP) and false negative (FN) prediction instances. Specificity scores and ROC-curves are not shown as these evaluation schemes are inappropriate in the context of motif detection. True negative (TN) instances in ground truth for motif data is rare as instances labelled as negatives in the ground truth may be discovered to contain motifs in the future.Also, the number of positive instances and number of predictions are much smaller than the number of total instances, causing the specificity to be very close to 1 almost always.

4.3.3 Tests on features

We have empirically established the discriminative nature of our feature set, but we also examine the soundness of the designed features in the context of the CRF model after incorporating some basic features, before including all of them in the model to test for feature redundancy and compatibility in the CRF framework. The state transition features and sequence conservation features are fundamental, so we check the validity of the other features based on predictions made by a basic model consisting of only state transition features and sequence conservation features. The soundness of additional feature is shown by comparing the distributions of the set of TPs and the set of FPs as predicted by the basic model. We learn a CRF model using the two kinds of fundamental features, and use it to get a set of predictions of TFBSs, which contains both TP predictions and FP predictions. We split the predictions into two groups, TP group and FP group, and compute the GC-percentage score, reverse complementary score and conservation symmetry score for each of the instances in the two groups. We can show the soundness of a feature by a statistical analysis on the difference between scores of the two groups.

There are 193 instances in TP group and 499 instances in FP group. Comparisons of cumulative distribution function (CDF) curves between TP group and FP group on GC-percentage scores, reverse complementary scores and conservation symmetry scores are shown in Figure 4.4. The scores plotted are raw scores without an offset. We can see that the CDF curve of TP group is almost always lower than that of FP group in GC-percentage score and reverse complementary score, while the CDF curve of TP group is almost always higher than that of FP group in conservation symmetry score. For the feature of GC-percentage, the scores in TP group have a mean at 0.4641 and sample variance at 0.0043, and the scores in FP group have a mean at 0.4323 and sample variance at 0.0065. Assuming that they both follow Gaussian distributions, we have a difference between means at 0.0318 with a standard deviation at 0.0059, which gives us a confidence value at $1 - 4 * 10^{-8}$ that the mean of TP group is bigger than the mean of FP group.

It is credible that GC-percentage feature is informative. Following a similar analysis, for the feature of reverse complementarity, the mean TP score is 0.3041 and sample variance 0.0349, and the mean FP score is 0.2413 and sample variance 0.0360. With a difference between means at 0.0159 with a standard deviation at 0.0059, we have a confidence value at $1 - 4 * 10^{-5}$ that the mean of TP group is bigger than the mean of FP group. For the feature of conservation symmetry, the TP scores have mean 0.5215 and sample variance 0.0541, and the FP scores have a mean 0.5950 and sample variance 0.0666. The confidence value that TP group has a smaller average score than FP group is $1 - 1.5 * 10^{-4}$.

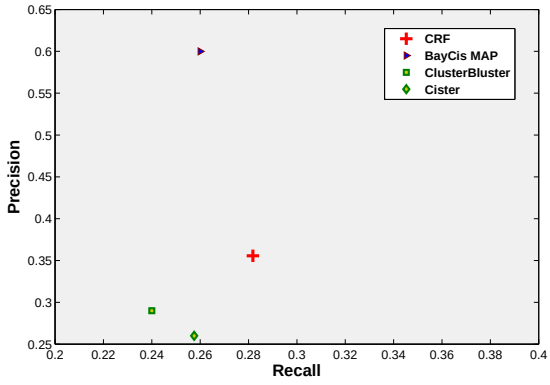
4.3.4 Performances on TFBS prediction

Synthetic dataset

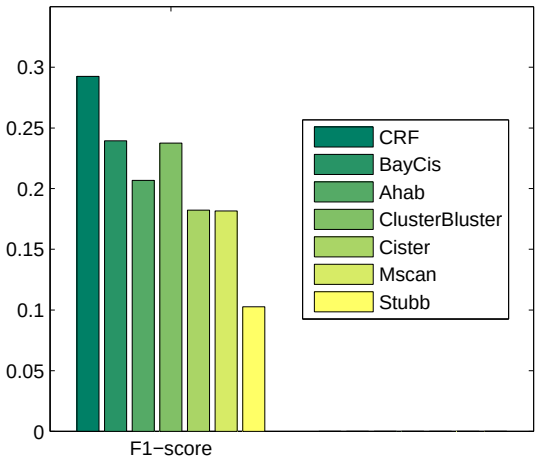
We compare the CRF model with BayCis, ClusterBuster, and Cister on the synthetic TRS dataset. CRF model outperforms ClusterBuster and Cister but not BayCis (Fig 4.5a) on the synthetic dataset. BayCis has an advantage over the other tools having the same background model as the simulation scheme, but we outperform Baycis on the real dataset.

Drosophila dataset

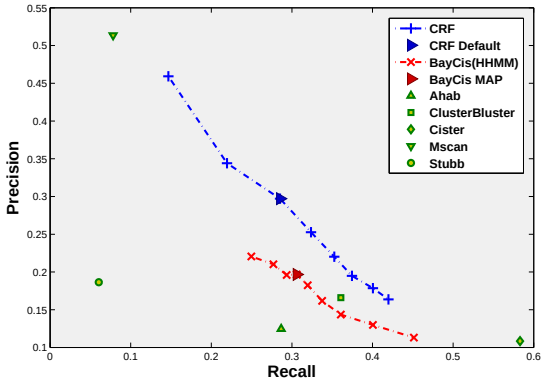
We compare the CRF model with BayCis, Ahab, Cluster-Buster, Cister, Mscan and Stubb on the real *D. melanogaster* TRS dataset. The overall F1-scores of the CRF model and six comparing methods are shown in Fig 4.5. All the algorithms are set to default configurations. The feature-based CRF model outperforms all other methods on the F1-score measure. It is 22% higher than the best competing tool. We also show the precision-recall (P-R) curves of the our methods and BayCis, as well as points in the P-R landscape for other tools in Fig 4.5. We plot P-R curves of the CRF model by varying the prediction factor k (from 0.0005 to 0.0040). For BayCis, we plot a P-R curve resulting from different thresholds for predictions, in addition to its default MAP setting. The CRF model outperforms BayCis, Ahab, ClusterBuster and Stubb in their default settings. The other two methods strike extremely different balances between precision and recall in their default output. MSCAN focuses on very high precision predictions, while Cister is geared towards high values of recall. It is noticeable that Stubb's performance is much below the rest, possibly because it uses distinct motif-to-motif transition probabilities, which can only be properly learned without over-fitting from datasets richer in scope than the present one. Addition of further non-redundant features like other epigenetic feature scores is expected to improve performance further. A set of predictions by the CRF model with default setting comparing with that of Cluster-Buster is shown in Fig 4.3. While they have comparable TP predictions, CRF model makes much less FP predictions than Cluster-Buster does. In a way, the performance gap between the CRF model and the HMM based models may be looked upon as a combination of two factors : the discriminative nature of the analysis, and the availability of features besides PWM and transition data.



(a)



(b)



(c)

Figure 4.5: (a) Precision-recall performance of CRF, BayCis, Cluster-Buster and Cister on the synthetic dataset (b) F1-score and (c) P-R curve of the CRF model in comparison with other algorithms at their default settings on the real *D. melanogaster* TRS dataset

Chapter 5

Admixture of Dictionaries Analysis of the Regulatory Genome

5.1 Related work

Research to date suggests that a large proportion of important phenotypic differences within or across species may originate from changes in gene expression rather than changes in coding sequences of the genes themselves. As a good example, Hox genes related to body segmentation are highly conserved across arthropods, including diverse groups of species ranging from insects to crustaceans. More interestingly, the determination of different body plans during embryogenesis are controlled by the spatio-temporal differential expression of these Hox genes, rather than by the utilization of any new Hox genes with no counterpart in other species [35]. Apparently, this is a consequence of the changes in the transcription regulation system. Therefore, understanding the rate, pattern and driving force of regulatory changes within and across species is critical for a comprehensive understanding of biological evolution, and is essential for more accurate characterization and prediction of the functional state of important biological processes.

The spatio-temporal expression of genes is controlled by numerous interacting elements known either as the *trans*-elements (e.g., transcription factors (TFs)) or the *cis*-elements (e.g., TF binding sites (TFBS)) [35]. Of particular importance to the transcription regulation of higher eukaryotes are the *cis*-regulatory modules (CRMs). CRMs are genetically hardwired information processors encoded by clusters of TFBSs. There is a large and growing body of work on predicting CRMs and TFBSs [194].

The occurrence of a specific TFBS in a CRM indicates that the CRM will listen to the signals from a corresponding TF; and the overall architecture of the CRM determines what combinatorial regulatory signals encoded by multiple TFs it can interpret and process to influence the behavior of its downstream gene. One important characteristic of CRM function in higher organisms is that they are often *multi-functional*: under different conditions and times, CRMs can drive very different biological regulatory functions via the differential recruitment of a combination of transcription regulatory proteins. Furthermore, functional selection acting on CRMs causes differential enrichment of nucleotide contents across evolutionarily related orgamisms. The exact impact of such

selection on gene regulatory mechanisms has not been fully understood.

Despite the availability of whole genome sequences from several model organisms, the lack of methodologies for modeling the structural and functional evolution of these regulatory elements has hindered an in-depth investigation of the mechanism and process of gene regulation and its evolution. Existing models for transcription factor binding site include the PWM [182], which models binding sites as fixed length oligomers where every position in the oligomer is modelled by a multinomial over nucleotides. Fratkin *et al* [53] performed min-cut on a similarity matrix over the set of all oligomers (called words, and their realization in the sequence called motifs henceforth) in a regulatory region to obtain strongly connected components corresponding to sets of motifs in a regulatory region. Sonnenburg *et al* [181] treat sequence based function prediction as a classification problem solved by training support vector machines with complex string kernels. More generic and model-based analyses like those of [73] and [22] model regulatory sequences using a single stochastic *dictionary* : which is a set of oligomers (the vocabulary) and a distribution over them. However, none of these methods can capture the multi-functionality of the CRM, and offer limited insight into the organizational and evolutionary mechanism of this phenomenon. Here we present a probabilistic graphical model called Admixture of Stochastic Dictionaries (ASD) for compactly extracting and exposing the sequence compositional information of CRMs.

More specifically, our generative model analyzes a collection of CRMs or regulatory sequences as being generated from a set of stochastic dictionaries with vocabularies of fixed length oligomers (see Fig 5.1 for a broad work schema). One crucial advantage of ASDs compared to earlier works [22, 53, 73] is that it recruits multiple dictionaries for modeling a variety of combinatorial usage of TFBSs, and hence models multi-functionality of CRMs. The functional differences across CRMs are compactly represented as the differential and proportional composition of these stochastic dictionaries, which we call *function composition vectors*. As we show in our results, the learnt stochastic dictionaries and the function composition vectors indeed succinctly capture functionally discriminative sequence infomation, and can be used for predicting regulatory regions. We note that similar modeling ideas are established in the literature : beginning with the introduction of Latent Dirichlet Allocation (LDA) models to model words and topics in text documents in the field of information retrieval [135]. Such models have been pursued in evolution and SNP variation [146] using "admixture model"s to explain the SNP variation in human beings as a mixture model over the genetic variations in different populations, to identify mRNA - microRNA modules from gene expression data [108], and for the purposes of functionally annotating coding sequences in *E. coli* [26] . However, we first aim to study gene regulation at the sequence level, for which our algorithm ASD is novel. We then establish a novel evolutionary extension of our model (EASD) to show how such stochastic dictionaries may evolve across species, in order to study regulatory evolution. Our algorithms are unsupervised methods and require no training based on binding site annotation, which allows us to avoid overfitting or false positive predictions that plague other regulatory genomic analyses [57].

Another key advantage of our method is that it can readily incorporate evolutionary information from multiple species. Naive applications of our ASD model to multiple species will either analyze each species sparately or model all regulatory sequences using a single ASD by throwing away the information that these sequences are evolutionarily related. Here we also propose

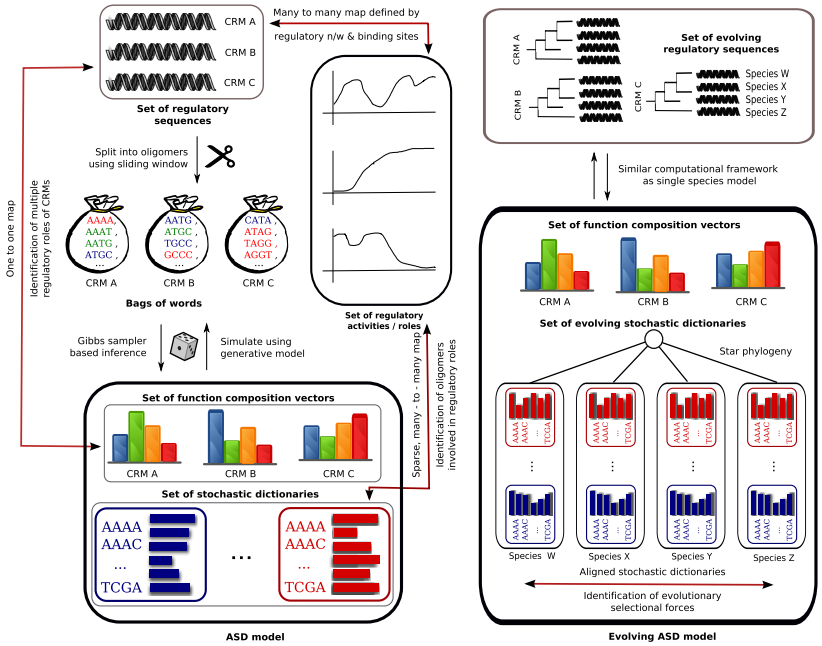


Figure 5.1: Schema of our single species and multiple species model, showing how the model parameters correspond to the different regulatory functions of CRMs

a sophisticated extension of ASD, an Evolutionary Admixture of Stochastic Dictionaries (EASD) model, which employs a star topology to integrate a set of species specific as well as function specific dictionaries. Such construction achieves the effect of "killing two birds with one stone": the species specific dictionaries allows us to model differential enrichment of nucleotide contents due to multiple functional selectional forces acting on CRMs; while the star topology tying these component dictionaries together captures the evolutionary relatedness of these species. Interestingly, in our later experiments, we find that the function specific dictionaries are nicely aligned across species, which suggests that annotated regulatory information from one species can be used to infer un-annotated regulatory sequences in other species.

It is worth noting that ASD can also be augmented with richer topologies such as a tree topology according to known phylogenetic information, though with increased computational costs. For simplicity, we will focus on the EASD with star topology here, and present an efficient sampling algorithm for learning the parameters of the model. From here onwards, we will start by presenting the modelling principles, and then show how to construct an evolutionary ASD by combining multiple basic ASD model. In the results section, we will show how the stochastic dictionaries and function composition vectors obtained from our model can be interpreted and used to predict regulatory regions and to analyze the regulatory regions of multiple species simultaneously. We will conclude the analysis with a discussion and future work.

5.2 Methods

Modeling principle: Over the course of a biological process, such as yeast cell cycles or Drosophila embryonic development, there may exist multiple underlying "themes" that determine the functions of each gene and their relationships with each other, and such themes are dynamic and stochastic. To tailor a gene's expression to a diverse range of internal and extenal conditions, multiple control elements need to be coded in the regulatory region of a gene, and signals received in these multiple loci are then integrated for the control of gene expression [35]. The two key organizational principles of regulatory sequences are: **(1)** The existence of cis-regulatory modules suggests that the control of gene expression is achieved by combinatorial use of multiple sequence elements. **(2)** Experimental observations on eukaryotic genomes further indicate that sets of functionally related genes typically share transcription factor binding motifs.

Based on these biological observations, we model the regulatory sequences as a conglomerate of motifs drawn from a collection of stochastic dictionaries. More specifically: **(1)** Each stochastic dictionary is a probability distribution over oligomers of a particular length. Each dictionary captures one way of combinatorial use of sequence motifs. Within each dictionary, motifs with high probability are those that tend to occur together often, while the low probability ones are those that rarely co-occur with high probability ones. Each stochastic dictionary is thus loosely correlated with one or more regulatory roles. **(2)** Multiple stochastic dictionaries are used to model the observation that regulatory modules tend to have multiple combinatorial usage of its sequence motifs (ie. their involvement in multiple regulatory activities). A particular regulatory role uses some motifs more than others (as dictated by the corresponding stochastic dictionary probabilities) in a regulatory sequence, and is loosely dictated by the transcription factors that are recruited for that regulatory activity. **(3)** Different dictionaries share the same motifs, but the probability assigned to each motif is different from dictionary to dictionary. Such a model is chosen based on the observation that each gene can have multiple functions, and depending on the biological needs, this gene may be controlled by different regulatory modules at different points of time and in different conditions. Hence the dynamic usage of the set of binding motifs can be very different. Each regulatory sequence is modelled as an unordered set of motifs, each potentially associated with different stochastic dictionaries. Each motif can thus participate in multiple regulatory activity. **(4)** The set of stochastic dictio-

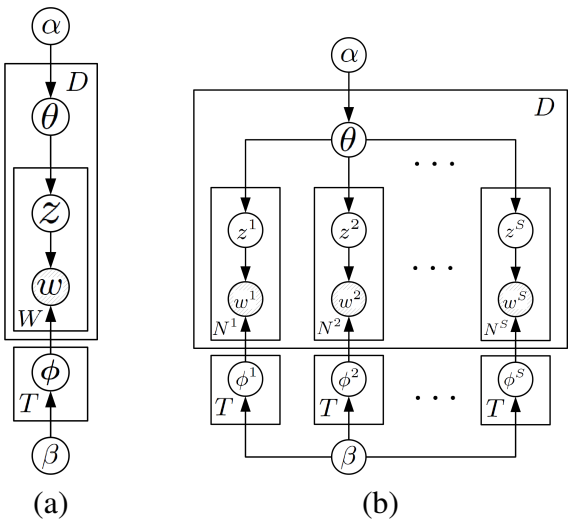


Figure 5.2: (a) Probabilistic graphical model for discovering a set of T stochastic dictionaries $\boldsymbol{\phi}$ for a collection of D regulatory sequences. (b) Probabilistic graphical model for discovering S sets of T stochastic dictionaries $\boldsymbol{\phi}^s$ for S species based on D collection of aligned regulatory sequences across species.

naries are shared across the whole collection of regulatory sequences. The sharing of stochastic dictionaries across regulatory regions models the observation that regulatory regions of functionally related genes share common regulatory roles and are often co-regulated by the same TFs. **(5)** The proportional usage of these stochastic dictionaries varies from regulatory sequence to regulatory sequence. Such proportional usage reflects the differential contribution of different regulatory regions in a certain regulatory role (as measured by the number of binding motifs associated with each stochastic dictionary). **(6)** We do not explicitly model genomic distances between motifs or the distance from motif to the Transcription Start Site (TSS), factors known to play a role in gene regulation. This is due to the fact that we aim to analyze the regulatory role of nucleotide content in a regulatory region, without being confounded by the regulatory role played by the positioning of individual motifs in a regulatory region. **(7)** Epigenetic regulatory forces are not directly modelled, since the nature of selection on epigenetic forces is unclear. However, we analyze CRMs which are part of the promoter and proximal enhancers (and not distal enhancers), hence less prone to cell-type specific regulation by changing chromatin accessibility through histone modifications [78]. Regulation by methylation of CRMs is also possible, but it has been demonstrated in the literature that a bag-of-oligomers model is sufficient to predict methylation status upto 85% accuracy [34], hence our stochastic dictionaries should be informative with respect to genomic features like CpG islands, that have a bearing on methylation status.

Besides regulatory genomics, our work is also well suited to be applied to metagenomics [155], which is used to analyze next-gen DNA sequence data of all bacteria or other organisms in a particular biome simultaneously. Each bag of words (see Fig 5.1) would correspond to a replicate experiment. The stochastic dictionaries would thus correspond to oligomer distributions in specific species, and the function composition vectors would correspond to the proportion of genetic data from each species obtained in different replicate experiments.

Motif likelihood: If we have T stochastic dictionaries, we can write the probability of the ith motif in regulatory sequence d as

$$\mathbb{P}(w_i) = \sum_{t=1}^{T} \mathbb{P}(w_i|z_i = t)\mathbb{P}(z_i = t) \tag{5.1}$$

where z_i is a latent variable indicating which stochastic dictionary is associated with motif w_i, and $\mathbb{P}(w_i|z_i = t)$ is the probability of motif w_i occurring in stochastic dictionary t. $\mathbb{P}(z_i = t)$ gives the probability of motifs in regulatory sequence d being associated with stochastic dictionary t. In other words, $\mathbb{P}(w|z)$ indicates which motifs are more likelily to occur in a particular stochastic dictionary, whereas $\mathbb{P}(z)$ implies the prevalence of various stochastic dictionaries in particular regulatory sequences.

5.2.1 Illustrative example of the ASD model

In the following, we give a simplistic example of this model. Suppose there are two regulatory sequences $d_1 = GCTCTG$ and $d_2 = AGCTAG$. Then there are 7 dimers appearing in the two sequences

$$\mathcal{W} = \{AG, GA, GC, CT, TA, TG, TC\}.$$

which are generated from $T = 3$ stochastic dictionaries

	AG	GA	GC	CT	TA	TG	TC
$\boldsymbol{\phi}_1$	0.94	0.06	0	0	0	0	0
$\boldsymbol{\phi}_2$	0	0	0	0.99	0	0	0.01
$\boldsymbol{\phi}_3$	0	0	0.5	0	0.25	0.25	0

where we have defined the vector $\boldsymbol{\phi}_t := (\mathbb{P}(w|z = t))_{w \in \mathcal{W}}$ for $t = 1 \dots T$. Then the vectors of dimers $\mathbf{w}_{d_1}$ and $\mathbf{w}_{d_2}$ for d_1 and d_2 and their associated dictionary indicator are

$$\mathbf{w}_{d_1} = \{GC, CT, TC, CT, TG\}, \quad \mathbf{z}_{d_1} = \{3, 2, 2, 2, 3\}$$
$$\mathbf{w}_{d_2} = \{AG, GC, CT, TA, AG\}, \quad \mathbf{z}_{d_2} = \{1, 3, 2, 3, 1\}$$

respectively indicating which stochastic dictionary is used to generate the corresponding dimers. Then the prevalence of the 3 stochastic dictionaries in the two sequences are

$$\boldsymbol{\theta}_{d_1} = \{0, 0.6, 0.4\}, \quad \boldsymbol{\theta}_{d_2} = \{0.4, 0.4, 0.2\},$$

where we have defined the vector $\boldsymbol{\theta} := (\mathbb{P}(z = t))_{t=1}^{T}$. The fact that multiple stochastic dictionaries can contribute to the regulatory sequence is a key feature of our model.

Mixture model of regulatory sequences: Viewing regulatory regions as mixtures of multiple stochastic dictionaries makes it possible to automatically discover these stochastic dictionaries given a collection of regulatory sequences.

Formally, given D regulatory sequences containing T stochastic dictionaries based on W unique motifs, we can represent $\mathbb{P}(w|z)$ with a set of T multinomial distributions $\boldsymbol{\phi}_t := \{\phi_{tv}\}_{v=1}^{W}$ $(t = 1 \dots T)$ over these W motifs, such that

$$\mathbb{P}(w|z, \boldsymbol{\phi}_z) = \prod_{v=1}^{W} (\phi_{zv})^{\delta_v(w)} \tag{5.2}$$

where $\delta_v(w)$ is an indicator function which returns 1 if $w = v$ and 0 otherwise. Since $\boldsymbol{\phi}_t$ is a probability distribution, we require that its entries are normalized, *i.e.* $\sum_{v=1}^{W} \phi_{tv} = 1, \ \forall t$.

We represent $\mathbb{P}(z)$ with a set of D multinomial distributions $\boldsymbol{\theta}_d := \{\theta_{dt}\}_{t=1}^{T}$ $(d = 1 \dots D)$ over the set of T stochastic dictionaries, such that for regulatory sequence d

$$\mathbb{P}(z|\boldsymbol{\theta}_d) = \prod_{t=1}^{T} (\theta_{dt})^{\delta_t(z)}. \tag{5.3}$$

where $\delta_t(z)$ is also an indicator function which returns 1 if $z = t$ and 0 otherwise.

For convenience, we denote the vector of motifs occurring in regulatory sequence d as $\mathbf{w}_d := \{w_{di}\}_{i=1}^{N_d}$, where N_d is the total number of motifs in sequence d; the associated vector of stochastic dictionary indicator as $\mathbf{z}_d := \{z_{di}\}_{i=1}^{N_d}$. Furthermore, we denote the aggregation of these $\mathbf{w}_d$ as $\mathbf{w} := \{\mathbf{w}_d\}_{d=1}^{D}$; likewise $\mathbf{z} := \{\mathbf{z}_d\}_{d=1}^{D}$, $\boldsymbol{\phi} := \{\boldsymbol{\phi}_t\}_{t=1}^{T}$, and $\boldsymbol{\theta} := \{\boldsymbol{\theta}_d\}_{d=1}^{D}$. We also provide a prior to the set of stochastic dictionaries $\boldsymbol{\phi}_t$ and $\boldsymbol{\theta}_d$. To faciliate subsequent Bayesian inference

on these parameters, we will use Dirichlet distributions which are conjugate to the multinomial distribution, *i.e.*

$$\mathbb{P}(\boldsymbol{\theta}_d) = \frac{\Gamma(T\alpha)}{(\Gamma(\alpha))^T} \prod_{t=1}^{T} (\theta_{dt})^{\alpha-1}, \quad d = 1 \dots D \tag{5.4}$$

$$\mathbb{P}(\boldsymbol{\phi}_t) = \frac{\Gamma(W\beta)}{(\Gamma(\beta))^W} \prod_{v=1}^{W} (\phi_{tv})^{\beta-1}, \quad t = 1 \dots T \tag{5.5}$$

where $\Gamma(\cdot)$ is a gamma function defined as $\Gamma(n) := (n-1)!$, and α and β are hyperparameters specifying the nature of the prior on $\boldsymbol{\theta}_d$ and $\boldsymbol{\phi}_t$. Therefore, the complete generative process of the probabilistic graphical model can be summarized below (and the corresponding plate diagram can be found in Figure 5.2(a))

for each stochastic dictionary $t = 1 \dots T$ **do**
  $\boldsymbol{\phi}_t \sim$ Dirichlet(β) according to (5.4)
end for
for each regulatory sequence $d = 1 \dots D$ **do**
  $\boldsymbol{\theta}_d \sim$ Dirichlet(α) according to (5.5)
  for each motif $i = 1 \dots N_d$ **do**
    $z_i \sim$ Multinomial($\boldsymbol{\theta}_d$) according to (5.3)
    $w_i | z_i \sim$ Multinomial($\boldsymbol{\phi}_{z_i}$) according to (5.2)
  end for
end for

Inferring model parameters: To discover the set of stochastic dictionaries $\boldsymbol{\phi}$ and the dictionary mixture proportion $\boldsymbol{\theta}$, we want to obtain an estimate of them that gives high probability to the motifs that actually appear in the collection of regulatory sequences. One strategy for such inference tasks is to use a collapsed Gibbs sampling techique. This approach does not explicitly represent $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ as parameters to be estimated, but instead considers the posterior distribution over the assignments of motifs to stochastic dictionaries, $\mathbb{P}(\mathbf{z}|\mathbf{w})$. We then obtain estimates of $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ by examining this posterior distribution. We evaluate $\mathbb{P}(\mathbf{z}|\mathbf{w})$ using a Gibbs sampling technique, resulting in a simple algorithm that only requires the counts of motifs assigned to the stochastic dictionaries.

Inference of ASD model parameters: To discover the set of stochastic dictionaries $\boldsymbol{\phi}$ and the dictionary mixture proportion $\boldsymbol{\theta}$, we want to obtain an estimate of them that gives high probability to the motifs that actually appear in the collection of regulatory sequences. One strategy for such inference tasks is to use a collapsed Gibbs sampling techique. This approach does not explicitly represent $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ as parameters to be estimated, but instead considers the posterior distribution over the assignments of motifs to stochastic dictionaries, $\mathbb{P}(\mathbf{z}|\mathbf{w})$. We then obtain estimates of $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ by examining this posterior distribution. We evaluate $\mathbb{P}(\mathbf{z}|\mathbf{w})$ using a Gibbs sampling technique, resulting in a simple algorithm that only requires the counts of motifs assigned to the stochastic dictionaries.

More formally, since $\mathbb{P}(\mathbf{w}, \mathbf{z}) = \mathbb{P}(\mathbf{w}|\mathbf{z})\mathbb{P}(\mathbf{z})$ and $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ only appear in the first and the second terms, respectively, we can integrate out $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ by performing two separate operations.

First, integrating out $\boldsymbol{\phi}$ from $\mathbb{P}(\mathbf{w}|\mathbf{z}, \boldsymbol{\phi})$ gives

$$\begin{aligned}\mathbb{P}(\mathbf{w}|\mathbf{z}) &= \int_{\boldsymbol{\phi}} \mathbb{P}(\mathbf{w}|\mathbf{z}, \boldsymbol{\phi})\mathbb{P}(\boldsymbol{\phi})\, d\boldsymbol{\phi} \\ &= \left(\frac{\Gamma(W\beta)}{(\Gamma(\beta))^W}\right)^T \prod_{t=1}^{T} \frac{\prod_{v=1}^{W}\Gamma(n_{tv}+\beta)}{\Gamma(n_{t*}+W\beta)}\end{aligned} \tag{5.6}$$

where n_{tv} is the number of times motif v assigned to stochastic dictionary t in the vector of assignment $\mathbf{z}$, and $n_{t*} := \sum_{v=1}^{W} n_{tv}$. Second, integrating out $\boldsymbol{\theta}$ from $\mathbb{P}(\mathbf{z}|\boldsymbol{\theta})$ gives

$$\begin{aligned}\mathbb{P}(\mathbf{z}) &= \int_{\boldsymbol{\theta}} \mathbb{P}(\mathbf{z}|\boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})\, d\boldsymbol{\theta} \\ &= \left(\frac{\Gamma(T\alpha)}{(\Gamma(\alpha))^T}\right)^D \prod_{d=1}^{D} \frac{\prod_{t=1}^{T}\Gamma(m_{dt}+\alpha)}{\Gamma(m_{d*}+T\alpha)}\end{aligned} \tag{5.7}$$

where m_{dt} is the number of times a motif from regulatory sequence d assigned to stochastic dictionary t, and $m_{d*} := \sum_{t=1}^{T} m_{dt}$. Our goal is then to evaluate the posterior distribution of the dictionary assignment vector $\mathbf{z}$ given the observed sequence motifs $\mathbf{w}$, *i.e.* $\mathbb{P}(\mathbf{z}|\mathbf{w}) = \frac{\mathbb{P}(\mathbf{w},\mathbf{z})}{\sum_{\mathbf{z}}\mathbb{P}(\mathbf{w},\mathbf{z})}$. However, computing $\mathbb{P}(\mathbf{z}|\mathbf{w})$ requires the normalization factor $\mathbb{P}(\mathbf{w}) = \sum_{\mathbf{z}} \mathbb{P}(\mathbf{w}, \mathbf{z})$ which is an integration problem on a large discrete state space. We deal with this problem of evaluating the posterior using a Gibbs sampling techique.

More specifically, in a Gibbs sampler, a simple Markov chain is constructed to converge to the target distribution $\mathbb{P}(\mathbf{z}|\mathbf{w})$. Each state of the chain is an assignment of values to the variables being sampled, in this case $\mathbf{z}$, and transitions between states follow a simple rule where the next state is reached by sequentially sampling each individual variable z_i conditioned on the current values of all other variables $\mathbf{z} \setminus z_i$ and the observations $\mathbf{w}$. Therefore, we first compute the full conditional distribution $\mathbb{P}(\bar{z}_i|\mathbf{z} \setminus z_i, \mathbf{w})$

$$\begin{aligned}&\mathbb{P}(\bar{z}_i = t|\mathbf{z} \setminus z_i, \mathbf{w}) \propto \\ &\qquad \frac{n_{tw_i} - \delta_t(z_i) + \beta}{n_{t*} - \delta_t(z_i) + W\beta} \frac{m_{d_i t} - \delta_t(z_i) + \alpha}{m_{d_i *} - \delta_t(z_i) + T\alpha}\end{aligned} \tag{5.8}$$

where we have used $\bar{z}_i$ and z_i to distinguish the current value and previous value of the variable being sampled; w_i and d_i are the motif and regulatory sequence corresponding to the dictionary indicator variable z_i. This sampling formula is quite intuitive: the first ratio estimates the likelihood of motifs equal to w_i being generated from stochastic dictionary t, and the second ratio estimates the proportion of contribution of stochastic dictionary t to regulatory sequence d_i. Importantly, the four counts used to estimate the ratio are the only information necessary for computing the full conditional distribution, allowing the algorithm to be implemented efficiently.

With a set of samples from the posterior distribution $\mathbb{P}(\mathbf{z}|\mathbf{w})$, we can estimate $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ from the value $\mathbf{z}$ by

$$\phi_{tv} = \frac{n_{tv} + \beta}{n_{t*} + W\beta}, \quad t = 1 \ldots T, \; v = 1 \ldots W \tag{5.9}$$

$$\theta_{dt} = \frac{m_{dt} + \alpha}{m_{d*} + T\alpha}, \quad t = 1 \ldots T, \; d = 1 \ldots D \tag{5.10}$$

Joint modeling of multiple species: In principle, one can use the mixture model discussed in the last section to build separate models for each individual species. However, there may be better way of making use of the information that these species are closely related. In this case, although the genomes of these species have evolved away from each other in order to accomplish slightly different biological functions, one would expect that the regulatory sequences could still be better conserved than intergenic regions. One way to accommodate such evolutionary conservation is to assume that the high level organizational principle of homologous regulatory sequences remain the same, but the stochastic dictionaries of these species can be different. In this section, we will present a model that augments the mixture model from the last section for simultaneously modeling multiple species.

Suppose the total number of related species be S, and we obtain a set $\mathcal{D}^s = \{d_i^s\}_{i=1}^D$ of size D regulatory sequences for each species s. (We will use the convention that the superscript s is used to denote index of species.) These S collection of regulatory sequences are aligned such that sequences with the same subscript i are homologous. The joint model for multiple species uses the mixture model for single species as building blocks, but there are two important augmentations to accommodation the evolutionary changes and conservation: **(1)** Each species has its own set of stochastic dictionaries $\boldsymbol{\phi}^s = \{\boldsymbol{\phi}_t^s\}_{t=1}^T$ of size T. These S collections of stochastic dictionaries allow the model to adapt to evolutionary changes. **(2)** Each set of homologous regulatory sequences $\{d_i^s\}_{s=1}^S$ compiled from different species share the same mixture vector $\boldsymbol{\theta}_{d_i}$. By sharing the same mixture vector, the model takes into account conservation in homologous regulatory sequences.

In the above two augmentations, the sharing of θ_{d_i} is essential for borrowing information across species. At first glance, the model does not specify the correspondence between the stochastic dictionaries across species; and it seems that allowing each species to have their own stochastic dictionaries could lead to arbitarily different dictionaries for different species. By forcing homologous sequences to have the same mixture vector actually implicitly requires that the stochastic dictionaries across species to be aligned as well. Aligned stochastic dictionaries can be examined quantatively to see the evolutionary change and conservation which we will illutrate in later experiments.

As a summary, the complete probabilistic generative model for multiple species is given below (and the corresponding plate diagram can be found in Figure 5.2(b)):

1: **for** each species $s = 1 \ldots S$ **do**
2: **for** each stochastic dictionary $t = 1 \ldots T$ **do**
3: $\boldsymbol{\phi}_t^s \sim$ Dirichlet(β) according to (5.5)
4: **end for**
5: **end for**

for each set of aligned regulatory sequences $d = 1 \dots D$ **do**
  $\boldsymbol{\theta}_d \sim$ Dirichlet(α) according to (5.4)
  for each species $s = 1 \dots S$ **do**
    for each motif $i = 1 \dots N_d^s$ **do**
      $z_i^s \sim$ Multinomial($\boldsymbol{\theta}_d$) according to (5.3)
      $w_i^s | z_i^s \sim$ Multinomial($\boldsymbol{\phi}_{z_i^s}^s$) according to (5.2)
    end for
  end for
end for

Here, we have used N_d^s to denote the number of motifs contained in the regulatory sequence d from species s. We will again use the same Gibbs sampling technique for estimating the model parameter $\boldsymbol{\phi} := \{\boldsymbol{\phi}^s\}_{s=1}^S$ and $\boldsymbol{\theta} := \{\boldsymbol{\theta}_d\}_{d=1}^D$. Therefore we first need to integrate out $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ in $\mathbb{P}(\mathbf{w}|\mathbf{z}, \boldsymbol{\phi})$ and $\mathbb{P}(\mathbf{z}|\boldsymbol{\theta})$ respectively.

For jointly modelling multiple species, we use the algorithm corresponding to the inference procedure of the EASD model. We have used N_d^s to denote the number of motifs contained in the regulatory sequence d from species s. We will again use the same Gibbs sampling technique for estimating the model parameter $\boldsymbol{\phi} := \{\boldsymbol{\phi}^s\}_{s=1}^S$ and $\boldsymbol{\theta} := \{\boldsymbol{\theta}_d\}_{d=1}^D$. Therefore we first need to integrate out $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ in $\mathbb{P}(\mathbf{w}|\mathbf{z}, \boldsymbol{\phi})$ and $\mathbb{P}(\mathbf{z}|\boldsymbol{\theta})$ respectively.

Note that the vector $\mathbf{w} := \{\mathbf{w}^s\}_{s=1}^S$ collects motifs from all sequences and all species, and $\mathbf{z} := \{\mathbf{z}^s\}_{s=1}^S$ is the corresponding vector of dictionary indicator. Integrating out all $\boldsymbol{\phi}$ gives

$$\mathbb{P}(\mathbf{w}|\mathbf{z}) = \int_{\boldsymbol{\phi}} \mathbb{P}(\mathbf{w}|\mathbf{z}, \boldsymbol{\phi}) \mathbb{P}(\boldsymbol{\phi}) \, d\boldsymbol{\phi} \tag{5.11}$$

$$= \prod_{s=1}^{S} \left(\int_{\boldsymbol{\phi}^s} \mathbb{P}(\mathbf{w}^s|\mathbf{z}^s, \boldsymbol{\phi}^s) \mathbb{P}(\boldsymbol{\phi}^s) \, d\boldsymbol{\phi}^s \right) \tag{5.12}$$

$$= \prod_{s=1}^{S} \left(\left(\frac{\Gamma(W\beta)}{(\Gamma(\beta))^W} \right)^T \prod_{t=1}^{T} \frac{\prod_{v=1}^{W} \Gamma(n_{tv}^s + \beta)}{\Gamma(n_{t*}^s + W\beta)} \right) \tag{5.13}$$

where n_{tv}^s is the number of times motif w has been assigned to stochastic dictionary t in the species s, and $n_{t*}^s := \sum_w n_{tw}^s$. Note that the counts n_{tw}^s are computed within each species. Integrating out $\boldsymbol{\theta}$ gives

$$\mathbb{P}(\mathbf{z}) = \int_{\boldsymbol{\theta}} \mathbb{P}(\mathbf{z}|\boldsymbol{\theta}) \mathbb{P}(\boldsymbol{\theta}) \, d\boldsymbol{\theta} \tag{5.14}$$

$$= \left(\frac{\Gamma(T\alpha)}{\Gamma(\alpha)^T} \right)^D \prod_{d=1}^{D} \frac{\prod_{t=1}^{T} \Gamma(l_{dt} + \alpha)}{\Gamma(l_{d*} + T\alpha)} \tag{5.15}$$

where l_{dt} is the number of times a motif from the dth set of aligned regulatory sequences being assigned to stochastic dictionary t no matter which species it is coming from, and $l_{d*} := \sum_{t=1}^{T} l_{dt}$. Note that the count l_{dt} here is computed across multiple species, and the sum of counts from each individual species.

Having $\mathbb{P}(\mathbf{w}|\mathbf{z})$ and $\mathbb{P}(\mathbf{z})$ allows us to compute the posterior $\mathbb{P}(\mathbf{z}|\mathbf{w})$ and derive the Gibbs sampling formula below

$$\mathbb{P}(\bar{z}_i^s = t|\mathbf{z} \setminus z_i^s, \mathbf{w}) \propto \frac{n^s_{tw_i^s} - \delta_t(z_i) + \beta}{n^s_{t*} - \delta_t(z_i) + W\beta} \frac{l_{d_i^s t} - \delta_t(z_i) + \alpha}{l_{d_i^s *} - \delta_t(z_i) + T\alpha} \tag{5.16}$$

where we have used $\bar{z}_i^s$ and z_i^s to distinguish the current value and previous value of the variable being sampled; w_i^s and d_i^s are the motif and regulatory sequence corresponding to the dictionary indicatory variable z_i. Note that this formula is different from the sampling formula (5.8) for a single species. Here the first ratio uses information within a single species, but the second ratio integrates information across species.

Similarly, with a set of samples from the posterior distribution $\mathbb{P}(\mathbf{z}|\mathbf{w})$, we can estimate $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ from the value of $\mathbf{z}$ by

$$\phi^s_{tv} = \frac{n^s_{tv} + \beta}{n^s_{t*} + W\beta}, \;\; s = 1 \ldots S, \; t = 1 \ldots T, \; v = 1 \ldots W \tag{5.17}$$

$$\theta_{dt} = \frac{l_{dt} + \alpha}{l_{d*} + T\alpha}, \;\; t = 1 \ldots T, \; d = 1 \ldots D \tag{5.18}$$

5.3 Results

In the previous section, we developed algorithms for learning the Admixture of Stochastic Dictionaries (ASD) model from regulatory sequences in one species as well for learning the more sophisticated evolving ASD (EASD) model across multiple species. Experiments using the single species ASD model allows us to carefully examine the multi-functionality of the analyzed CRMs. Experiments using the EASD model for multiple related species allows us to analyze how CRMs and their multi-functionalities evolve by analyzing the extent and nature of change of functionality-specific dictionaries across organisms.

Datasets: Our primary dataset of regulatory data consists of *cis*-regulatory regions from 21 early developmental genes in *Drosophila melanogaster*, along with TFBS positional information and independently estimated binding specificities for 75 TFs and positional information of the constituent CRM boundaries. However, many of these TFs have too few binding sites or have poorly estimated PWMs and are not amenable for purposes of regulatory analysis. Typically, in our experiments we use data from 17 TFs which have sufficient robustly estimated PWMs in the dataset for our analyses. The dataset is from [57], with the positions of TFBSs and CRM boundaries obtained from the REDfly database [62] and the Drosophila Cis-regulatory Database at the National University of Singapore [133]. Orthologous sequences in 9 other related *Drosophila* species - *simulans, sechellia, yakuba, erecta, pseudoobscura, ananassae, persimilis, virilis, mojavensis* was obtained from precomputed BLAT alignments [91] from UCSC Genome Browser [80]. We also perform regulatory region prediction on a secondary yeast dataset (details present in the experimental description) of regulatory regions [98] showing our methods are not species- or clade-specific and can work over a wide range of eukaryotic sequences.

Model selection: We perform model selection by specifying the number of stochastic dictionaries to be estimated and the length of the words in the dictionary. For the purposes of model selection, we can use an information theoretic criteria like Bayesian Information Criteria, or choose the model based on biological insight. We use a vocabulary of fixed length words, which avoids the problem of determining the size of words adaptively, but requires us to choose the length of words in the dictionary. Conserved cores of TFBSs in most eukaryotes are between 6 and 10 bps long [124]. We choose a word length in this range, since this size offers an appropriate level of granularity for studying regulatory organization and evolution and the analysis remains stable for slightly smaller or larger choice of word length. Flanking regions of conserved cores are also somewhat conserved in TFBS if our choice is slightly larger than a binding site in the sequence. If it is slightly smaller, the method will analyze subsequences of the actual motifs which are subject to the same organizational and selectional forces in the genome. However, a significantly larger choice of word length (say 20) will cause difficulties in analysis as the stochastic dictionaries will become ultra-sparse. A significantly shorter choice of word length (say 2) will be able to analyze genomic phenomena like variation of GC content which operate at such resolutions but not regulatory evolution. The choice of number of the number of dictionaries is explained in the experimental details. We perform subsequent experiments using 20 stochastic dictionaries and oligomers of length 9 if not specified otherwise. The evolutionary analysis is performed with oligomers of length 7 and 10 dictionaries, to re-iterate that such analyses can be performed with oligomer sizes between 5 and 10, and number of dictionaries between 10 and 40; and tradeoff in terms of computational efficiency and biological interpretability is smooth.

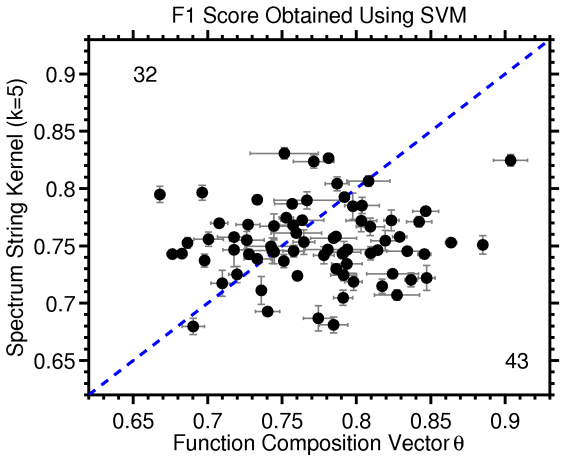


Figure 5.3: F1 score obtained using SVM and function vectors as features versus using SVM and spectrum string kernels. The horizontal axis is F1 score using function vectors, and the vertical axis is F1 score using spectrum string kernels. Dots lying on the diagonal indicates a tie. Dots lying in the lower half triangle indicates that function vectors leads wrt classifers with higher F1 score. In 43 datasets, functions vectors lead with better results. The horizontal and vertical error bars are standard error for classifiers trained with function vectors and spectrum string kernel respectively.

Functional discrimination using ASD: Our model can be viewed as an unsupervised method for extracting information from a collection of regulatory sequences. Applying our model results in a set of stochastic dictionaries and a function indication vector for each sequence. A natural question is that whether the CRM-specific function composition vectors $\boldsymbol{\theta}_i$ capture any information for discriminating transcription binding target.

In this experiment, we used yeast ChIP-ChIP data to create classification datasets, and use the function vector as features for classification [98]. We view TF binding as a classification problem, where those regulatory sequences with a p-value smaller than 0.01 will be treated as positive

sequences and otherwise as negative sequences. This thresholding results in highly unbalanced binary classification problem for each transcription factor. Typically, at most a few hundred regulatory sequences have p-value smaller than 0.01 (potential binding targets for a TF) while the remaining a few thousands have p-value larger than 0.01. We only use those datasets where the number of positive examples are larger than 30. This result in 75 binary classification datasets with the number of positive examples varying from 30 to 348. To avoid an unbalanced classification problem, we create training and test subsets for each dataset by randomly sampling $3/4$ of the positive examples and an equal number of negative examples. We repeat such random samplings of training and test pairs 10 time for each dataset, and the classification results using these pairs are used to estimate the mean and standard error. We evaluate the estimation procedures using an $\mathsf{F1}$ score, which is the harmonic mean of precision (Pre) and recall (Rec).

We evaluate the estimation procedures using an $\mathsf{F1}$ score, which is the harmonic mean of precision (Pre) and recall (Rec), *i.e.* $\mathsf{F1} := \frac{2*\mathsf{Pre}*\mathsf{Rec}}{\mathsf{Pre}+\mathsf{Rec}}$. Precision is calculated as $\mathsf{Pre} := \frac{|\hat{\mathcal{P}} \cap \mathcal{P}|}{|\hat{\mathcal{P}}|}$, and recall as $\mathsf{Rec} := \frac{|\hat{\mathcal{P}} \cap \mathcal{P}|}{|\mathcal{P}|}$, where $\mathcal{P}$ and $\hat{\mathcal{P}}$ are the set of actual positive examples and predicted positive examples in a test set, and $|\cdot|$ compute the number of elements in a set. The $\mathsf{F1}$ score is a natural choice of the performance measure as it tries to balance between precision and recall; only when both precision and recall are high can $\mathsf{F1}$ be high.

We compare our classification using the function composition vectors to those using spectrum string kernel [99]. Particularly, we use oligomers of length 5 to learn the function composition vector and corresponding oligomers are also used for the spectrum kernel. We use an SVM classifier where the regularization parameters are chosen from $\{0.01, 0.1, 1, 10, 100\}$ using cross-validation. The classification results on the 75 datasets are summarized in Figure 5.3. Although standard errors show that our method tends to have larger standard error, it better captures disciminative information than the spectrum string kernel in more datasets.

Discriminative parameters for predicting CRMs : We first learn an ASD for regulatory sequences from *Drosophila melanogaster*, and show that the learnt parameters can be more discriminative than PWMs when prediciting CRMs. The goal of this experiment is not to produce a high-quality, supervised CRM predictor but to show potential usage of the stochastic dictionaries for predicting CRM in new regulatory sequences by choosing the right discriminative parameters.

For predicting the regulatory region, we use a sliding window based approach and a local score is computed from the window to predict the location of the CRM. We use PWM scores in the sliding window and utilize those PWMs which are known to regulate the gene in question. We also use stochastic dictionary based scores using those stochastic dictionaries which were known to have good correlation with TFBSs of the TFs known to regulate the gene.

The results (see Fig 5.4) for predicting the regulatory region of hunchback gene is depicted here as it is particularly illustrative, using both PWM scores of the regulating TFs as well as the parameters of the stochastic dictionaries of the ASD model. We use a sliding window to score the genomic sequence around the regulatory region. We find that while the PWM scores in the sliding window find the CRM, they also find numerous other false positives. The stochastic dictionary probability scores in the sliding window, on the other hand has a few non-zero scores (due to the sparse nature of dictionaries over words of length 9).

However, the few non-zero scores the sliding window has are on the regulatory region or near it.

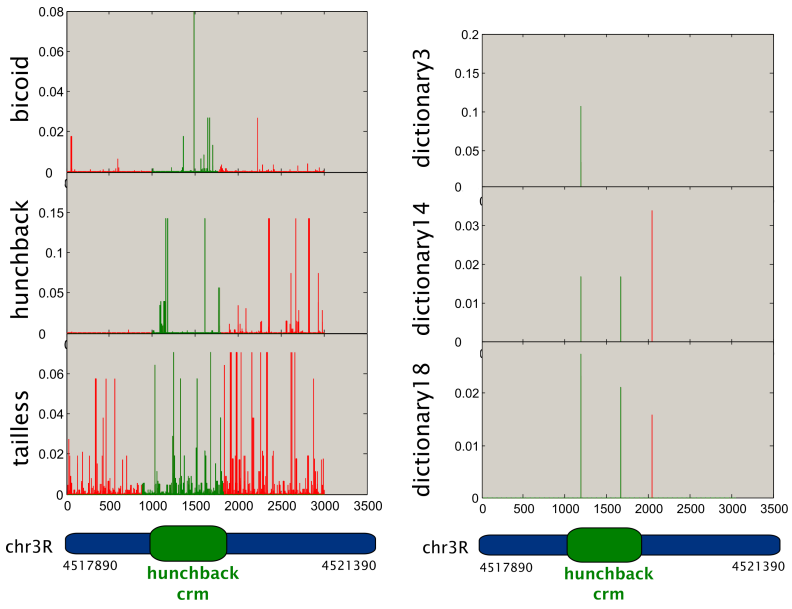


Figure 5.4: (a) PWM scores on words of length 9 from known regulatory TFs used to predict the location of the hunchback CRM. (b) Dictionary parameters from stochastic dictionaries known to have good correlation with TFBSs of the regulatory TFs used to predict the location of the hunchback CRM

The significantly richer representation of the stochastic dictionary enables more accurate prediction of the CRM, while the PWM based score has a severe limitation in the fact that it cannot recognize nucleotide composition changes as a result of selection due to several different functions, as in the case of the CRM. We looked into these ultra-discriminative stochastic dictionaries, and found them to be sparse which also explains the higher standard errors (albeit in a different experiment) in Fig 5.3. Further, we found that the some members of the sets of words with non-zero entries in the three discriminative stochastic dictionaries that we depict have overlaps with each other.

Understanding the model parameters: Each stochastic dictionary potentially plays a role in the regulatory process. The function of a stochastic dictionary is determined by the distribution of the oligomers insider it. The multi-functionality of each CRM arises due to the influence of the different stochastic dictionaries and the precise nature of the multi-functionality is based on the values of the function composition vector corresponding to the CRM. We first analyze the function composition vectors for 21 CRMs in *Drosophila melanogaster*, which clearly show that different CRMs are generated from very different proportions of the 20 different function-specific stochastic dictionaries (see Fig 5.5). This effectively causes most CRMs to be able to uniquely perform a variety of regulatory functions.

Transcription factor binding affinity of a word can be well approximated by the Position Weight Matrix (PWM) score: the likelihood of the word given the PWM (ordered set of multinomials) model. The best correlated PWMs and stochastic dictionaries are depicted in a bipartite graph over

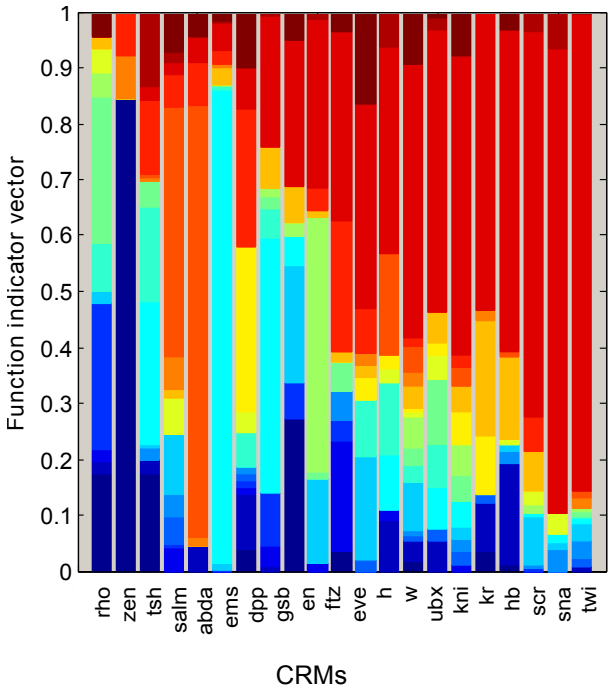


Figure 5.5: This figure shows the varying values of the function composition vector for each of the 21 CRMs we analyzed. Each color corresponds to a stochastic dictionary with a specific regulatory function.

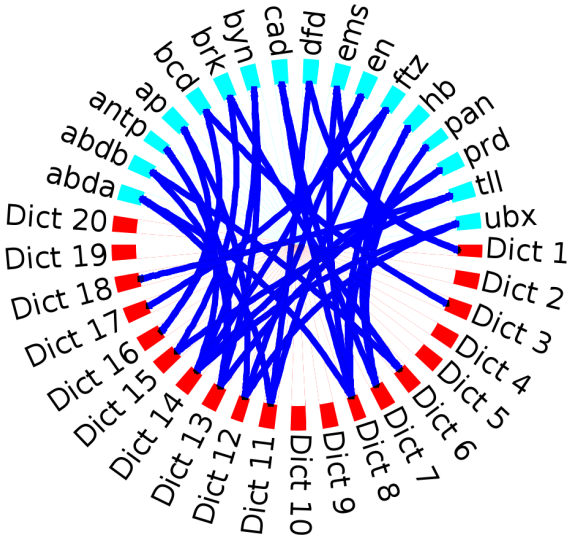


Figure 5.6: This bipartite graph with nodes arranged in a circle shows the association between 17 transcription factors (TF) and 20 stochastic dictionaries as measured by the correlation between the TF PWM expected binding affinities and the 20 dictionaries estimated from the Drosophila CRMs.

PWMs and stochastic dictionaries in Fig 5.6, depicting a relatively sparse graph. The expected value of the PWM score with the expectation calculated over the distribution of the oligomer in the dictionary is the Expected Binding Affinity (EBA) score. We also find that only a handful of stochastic dictionaries account for the highest EBA scores of most of the PWMs. Three particular stochastic dictionaries generate the highest EBA scores for 10 out of the 17 PWMs we analyzed. This can be explained by some stochastic dictionaries having disproportionately high probabilities of certain words which have high binding affinity scores. These stochastic dictionaries thus model regulatory functions controlled by high affinity binding events.

In order to check whether there were functionally redundant stochastic dictionaries in the set of 20 stochastic dictionaries we generated, we obtained correlation coefficients between every pair of stochastic dictionary estimated for every species (see Fig 5.7). It was found that stochastic dictionaries within the same species in general correlated much worse than stochastic dictionaries across species confirming that only minimal functional redundancy by having two stochastic dictionaries model similar distributions over words.

Typically, the number of dictionaries is chosen to be comparable to the number of transcription factors binding in the CRMs. The number of dictionaries can be systematically increased in experiments and stopped after the analysis becomes stable, when the number of dictionaries is overestimated. We tried obtaining an even broader spectrum of regulatory functions by increasing the number of stochastic dictionaries upto size 50, but the outcome of the experiment with 20 stochastic dictionaries is indicative of the results obtained with higher numbers of stochastic dictionaries. The choice of the number of stochastic dictionaries is thus not critical, as long as it is not trivially low or high. One may draw an analogy with cladistics where a choice of too few clades would cause related taxa to be merged into one, and a choice of too many clades cause some taxa to be split into two or more closely related sub-taxa.

Analyzing the regulatory genome using EASD: In this section, we learn our model with regulatory sequences from all 10 Drosophila species using the EASD model. We would typically like to investigate the nature of functional evolution in the regulatory genome. Since each stochastic dictionary has a potential regulatory role, it is interesting to investigate whether functionally aligned stochastic dictionaries are similar across species. Such similarity indicates the conservation of the constituent composition of the stochastic dictionaries, and hence its function. Alternatively, if we find that functionally aligned stochastic dictionaries are very different, it would emphasize that the specific function of the stochastic dictionary is undergoing rapid change, possibly due to positive selection. We compute the correlation coefficient for these 10 sets of stochastic dictionaries, and visualize them in Figure 5.7. We find that functionally aligned stochastic dictionaries have significantly higher correlations, though they are different from each other as well, implying some functional evolution is undergoing. The correlation coefficients of functionally aligned stochastic dictionaries can be taken to be noisy indicators of the evolutionary rate at the locations of oligomers with high probabilities in these stochastic dictionaries. Assuming functional alignment, stochastic dictionaries can be used to detect selection by choosing a range of correlation coefficients corresponding to stochastic dictionaries learnt from genomic regions known to be under neutral selection [93]. Correlation coefficients of aligned stochastic dictionaries above the neutral range thus correspond to negative selection (oligomers evolving slower than the neutral rate due to

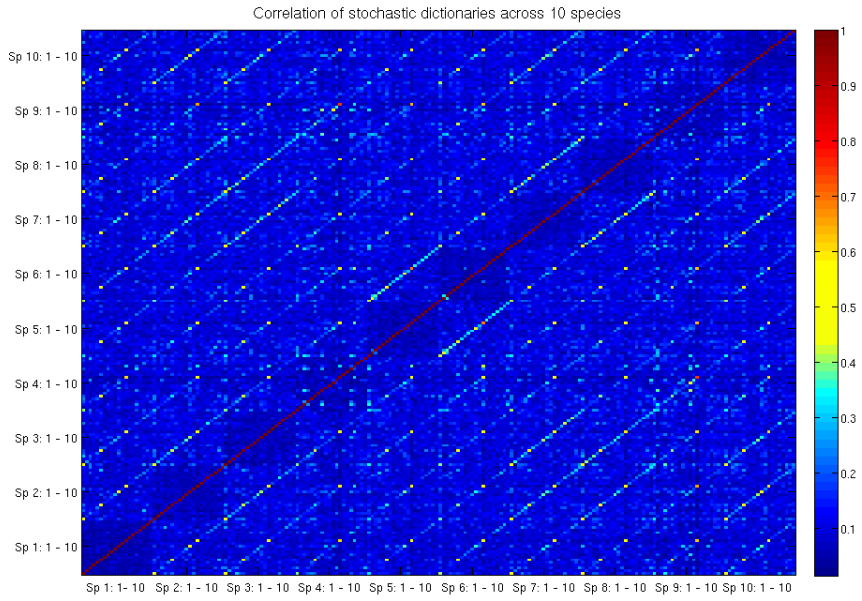


Figure 5.7: Correlations of stochastic dictionaries across species. Each of the 10 species have their own set of 20 stochastic dictionaries over observed 7-mers. Heat map shows correlations between these $200 = 20 \times 10$ stochastic dictionaries. It can be seen that diagonal blocks have low correlation compared to other blocks, while the clear diagonal pattern within other blocks indicates that functionally aligned stochastic dictionaries are highly correlated across species.

functional constraints), while correlation coefficients below the neutral range correspond to positive selection. Regulatory regions are notoriously hard to align, due to high turnover in nucleotide content [114] and presence of repeats, hence this would be an alignment-free way to test selection. Biological validation of binding sites implicated in regulation is a labor and cost intensive process, since ChIP-Seq experiments merely confirm biochemical and not neccessarily regulatory function. Our algorithm provides a convenient alternative to algorithms which detect selection in regulatory sequences based on nucleotide level alignment and binding site annotation [126].

5.4 Discussion

As future work, we may perform experiments allowing us to transfer TFBS and CRM annotation information from one species to help perform regulatory region prediction or other kinds of regulatory analysis on unannotated species. This may be achieved by finding stochastic dictionaries that achieve functional discrimination in an annotated species, and using the corresponding aligned stochastic dictionary in unannotated species to perform regulatory analysis.

Here we present an Admixture of Stochastic Dictionaries (ASD) modeling regulatory sequences. We have developed sophisticated algorithms for learning the Admixture of Stochastic Dictionaries within one organism, and across multiple evolutionarily related organisms, which allow us to examine multi-functionality of CRMs, and the way it evolves by analyzing the extend of change of every functional specific dictionary in the ASD models across organisms. We show that the learnt component dictionaries in our model are indeed functionally discriminative, and can be used for predicting regulatory regions. We further show that such discriminative ability is based on their TF binding affinity scores. We find that the corresponding function specific dictionaries across species have similar (but non-identical) distributions over oligomers, such that regulatory information from one species can be used to predict regulatory regions in other species. We conclude that our model is easy to estimate and interpret, and serves as a good platform for modeling functional evolution of the regulatory genome, and a useful tool to identify regulatory function based on these properties. Our current model can be extended in many ways to model richer aspects of regulatory sequence evolution. At present, our model performs analysis based on a predefined set of oligomers of fixed length and may be extended to automatically discover the set of most important oligomers from the data. Integrating additional discriminative information for well-studied species such as *Drosophila melanogaster* into our model can potentially improve TFBSs prediction in other species.

Chapter 6

The changing face of DNA binding motif finding and cis-regulatory module analysis

6.1 Development of the motif model

DNA binding motif finding has been very successfully used ever since the discovery of the *lac* operator [40, 65, 117] and the *lambda* operators [118, 120, 198] provided evidence that DNA binding protein factors could regulate gene expression. Advances went hand in hand with improvement in technologies for effecient sequencing of DNA [122, 162]. With the advent of detection of binding specificities of protein - DNA binding using DNAse-footprinting [60], and the ability of DNA synthesis technologies to synthesize arbitrary sequences of nucleotides and place them in promoter regions to gauge the effect of specific oligonucleotides on protein binding and hence gene regulation [69].

In modelling the binding site, initially simplistic models like modelling binding specificity as a fixed recognition site of constant length or a small set of such sites; and as consensus sequences (a combinatorial way of combining position specific information at the binding site) were tried for transcription factor binding sites [119], which had proved successful for modelling cleavage sites of restriction enzymes [2, 90, 95, 179], ribosomal binding sites in prokaryotes [170] , and translation initiation in eukaryotes [24, 94]. As these methods did not scale well to sequence motifs with more variability and other constraints, various other computational approaches began to be tried starting in the late 1970s : switch sites and regulatory sequence patterns modelled as palindromes to capture molecular conformations [36, 138], quantitative estimates of binding specificity and calculation of dissociation constants [106], and models of distance between a cis-acting site and the coding sequence [44]. Use of weight matrices to characterize diversity in different positions of fixed length binding sites was first successfully formulated by Stormo *et al* [190] to characterize translation start sites. However, the weight matrices were used to compute the decision function for a perceptron, and the model was not a probabilistic one - returning a decision but no likelihoods for ranking candidate sequences. The work of Rodger Staden at Cambridge [182, 183, 184, 185, 186, 187, 188] firmly established the Position Weight Matrix (PWM) as the primary model for sequence motifs in DNA, proteins and RNA in a probabilistic framework by generating a likelihood based score for

each oligomer based on motif and background models, and convert the scores to p-value. Later algorithms which have built on top of this basic model also apply False Discovery Rate control or multiple testing correction to generate a q-value [70].

The placement of the PWM model in an information theoretic framework [163], and in the perspective of binding energies in a statistical mechanics framework [11], the ability of the model to act as a discriminative feature to distinguish one set of sequences against another [152, 177], along with Occam's Razor of being the simplest model to accurately model binding specificities for transcription factors has ensured that it remains the *de facto* sequence motif model 25 years later [194]. The large number of variations on this model and the accompanying algorithms to perform estimation and inference in both supervised and unsupervised frameworks have been thoroughly reviewed in the earlier chapters of this thesis.

6.2 Traditional approaches to binding site detection

Sequence motif finding algorithms, which became popular with the rise in the number of sequenced regulatory regions, aimed at *in silico* characterization and detection of transcription factor binding sites in genomic DNA. They proceed in their exploration in an iterative manner, starting from a handful of biologically validated binding sites, by learning a binding motif model (position weight matrix) from them in a supervised fashion. If such an initial set is not readily available, unsupervised approaches are used to identify statistically significantly over-represented motifs [5]. The algorithms then perform a whole genome scan based on the learnt motif, to predict a novel set of putative binding sites. Because of the potential of high rates of false positives in the predicted set, these algorithms typically filter the set of predictions using features like evolutionary conservation. Finally, a new, high-confidence set of putative binding sites are selected for biological validation, and successfully validated binding sites are fed back into the model as training data, taking care to avoid overfitting issues.

Motif finding algorithms have been at the heart of this binding site discovery framework. Whilst being immensely successful [194] at predicting binding events in regulatory sequences, this traditional motif finding approach suffers from several shortcomings :

- This approach initially suffered from experiential bias, where the inductive bias of the learning algorithm was based on the initial set of binding sites used. If the original set of binding sites were not representative enough (did not sample the space of oligomers with respect to binding specificity in a faithful way), this approach would get stuck by identifying only a subset of real binding events. However, with the increase in sequenced and biologically validated regulatory sequences, as well as bayesian frameworks allowing priors over motif models [204], this is no longer a problem.
- Even though putative motif discovery can be performed using whole genome scans *in silico* in a high throughput way, the biological validation of predicted binding sites is still low throughput, and remains a bottleneck in the discovery procedure.
- This approach identifies all linear DNA sequences for which the binding specificity is high. However, typically these methods suffer from high rates of false positives [194], primar-

ily due to two reasons. Firstly, whole genome scans will tend to throw up lots of oligonucleotides which have sequence specificity scores marginally above empirically chosen threshold for long motifs (¿ 10 bp) with multiple high entropy positions due to the combinatorially large search space with few constraints. Potential false positives may be filtered by carefully modelling the sequence by autoregressing nucleotide distributions and modelling eukaryotic regulatory architecture using Hidden Markov Models or Conditional Random Fields, using evolutionary conservation and other genomic cues in a systematic and model based way, as explained in Chapters 2, 3 and 4 of this thesis. Secondly, traditional approaches primarily use only genomic sequence for predicting DNA binding events, and are agnostic to the cell type and cell type specific chromatin accessibility, and cell type specific regulatory mechanisms like epigenetic modifications that can impact transcription factor binding. Hence, *in vivo* validation in a particular cell type at a fixed time point will only validate a subset of the predicted instances, without shedding light on which binding events are cell type specific. High throughput ways of biologically validating events are thus required to identify cell type specific binding events by comparing binding profiles in multiple cell types.

It was primarily to overcome the above shortcomings that high throughput chromatin immunoprecipitation based techniques were developed for identifying binding sites.

6.3 Chromatin Immunoprecipitation based techniques and motif finding

In 1984, Gilmour and Lis developed a protocol to use ultra-violet irradiation to cross-link DNA with proteins in contact with the DNA in an *in vivo* [67, 68] fashion. This was the birth of chromatin immunoprecipitation (ChIP) studies for *in vivo* studies of protein - DNA binding, where proteins in contact with DNA are first cross-linked, the DNA is then sheared by sonication, followed by cell lysis, addition of bead-attached antibodies specific to the protein of interest for the purpose of immunoprecipitation, after which the DNA fragments are purified. This basic protocol has been improved in various ways [30], and has been used in combination Polymerase Chain Reaction [159] (ChIP - PCR [79]) , microarrays [25, 210] (ChIP on Chip or ChIP-Chip [105, 153]) and next-generation sequencing [29, 76] (ChIP - Seq [85, 156]). One of the biggest advantages of ChIP techniques is that they provide an *in vivo* snapshot of the binding of the protein of interest at a particular time point, and hence experiments across cell types can identify cell type specific binding [177]. Another aspect of ChIP technologies is that it is blind to sequence specificity, and both sequence specific and non-specific binding is captured, and such signals thus need to be deconvoluted.

The first popular incarnation of ChIP technologies was chromatin immunoprecipitation followed by a tiling array experiment, also known as ChIP on Chip or ChIP-Chip [105, 153]. Typically, the resolution of the tiling array is of the magnitude of 100s of bps, while individual binding sites typically vary in the range of 5 - 10 bps. As a result, the outcome of the experiment typically provides a set of bound regions [105, 153] and not specific binding sites. If binding specificities for the protein of interest are known, these “bound” regions are used as a filter for motif scans

of the genome to obtain individual binding sites. If binding specificities are unknown, these regions are subject to motif finding, typically by finding the most discriminative motifs between bound and unbound regions of the tiling array [177]. However, since hybridization based techniques suffer from noise and batch effects and required multiple biological replicates for reliable inference, ChIP-Chip turned out to be a costly platform, and was soon replaced by next-generation sequencing alternatives for identifying the bound DNA fragments.

ChIP-Seq [85, 156] platforms provided chromatin immunoprecipitation, followed by next generation sequencing. It provides an economical (compared to ChIP-Chip), and more robust estimate of the whole genome binding profile for a protein of interest. Typically, ChIP-Seq analysis focusses on identifying high coverage regions of the genome based on mapping the DNA fragments isolated from the ChIP process. These regions are called "peak"s and the algorithms to identify the bound regions are referred to as peak-callers. Several model based peak callers have been well referenced in the literature [125, 178, 211], and several approaches go further to differentiate between sequence specific versus non-specific peaks, histone versus transcription factor peaks, and characterize the shape of the peaks [208].

Peak calling typically aims to identify regions of specific binding for transcription factors, but again regions identified (peaks) have a larger resolution (around 100 bps) as opposed to a DNA binding event (5 - 10 bp). Hence, in order to obtain binding specificities and exact binding sites, we still need to perform motif analysis on the ChIP-Seq data. Again, for known binding motifs, peaks calls act as a filter to weed out false positives, while for unknown binding motifs, novel motifs are obtained by sampling oligomers from the whole genome or the called peaks, using coverage depth as a guide for the sampling distribution, as well as by using Expectation Maximization approaches for *de novo* motif finding [4, 115, 116, 211]. Motif finding may or may not use called peaks when sampling motifs, peak indicators are at best a filter, and at worst nuisance variables with respect to the motif finding algorithm. Newer generations of ChIP-Seq technology, like ChIP-Exo [154], aim to lower resolution of ChIP-Seq to that of DNA binding [154], reducing motif finding primarily to a multiple sequence alignment problem. Chip-Exo uses an exonuclease that degrades unbound DNA (double stranded) in the 5' to 3' direction completely and specifically, providing theoretically single base pair resolution.

Similar technologies have been developed for identifying regions of RNA bound by protein : ChIP-Chip technologies have an anologue in RIP - Chip (Ribonucleoprotein Immunoprecipitation on a Chip / microarray)[196], and ChIP-Seq technologies have an analogue in ClIP-Seq (Cross-linking Immunoprecipitation High Throughput Sequencing) [161]. All motif analysis techniques devised for identifying motifs for DNA binding events can be readily ported to explore RNA-binding events.

However, Chip-Seq technology, and the accompanying motif detection algorithms, also suffers from certain problems :

- Since ChIP-Seq uses next generation sequencing technology, it suffers from all the known problems suffered in next generation sequencing approaches. Primarily, these are :
  - Mapping to novel genomes is difficult, since in situ *de novo* genome assembly of novel genomes is typically not possible from ChIP-Seq data, which only binds a fraction of the genome. In such situations, motif sampling can be performed directly from the

ChIP-Seq fragments or from partial assemblies.

  - Mapping reads in repeat regions is problematic, and this is typically solved by either generating single end reads of longer length, or by the usage of paired end tags (paired end sequencing [59]) to generate uniquely mappable DNA "fragments". Chip-Sequencing with paired-end tags (PET) is known as ChIP-PET [27]. However, this is in direct contrast to generate shorter reads to achieve higher resolution. Thus, in principle, there is a tradeoff in Chip-Seq techniques between resolution of the binding data and mappability of reads, with motif finding gaining more importance towards the end of the spectrum of lower resolution. However, Chip-Exo [154] techniques combined with paired end tags are able to circumvent this inherent trade-off.
  - Distribution of mapped read coverage given the set of original fragments is inevitably a Poisson distribution, whereas ideally for sampling the fragments one prefers a uniform distribution : this bias effectively lowers the probability of detection of low affinity binding events. Motif finding algorithms, however, can rescue some low affinity binding events by changing their sampling strategy.
  - Mapping artifacts due to incorrect reference genomes (especially in the presence of Copy Number Variations / CNVs), or from fragment distribution bias or mapping problems can lead to false positives in Chip-Seq peaks. Typically, negative controls are performed by the process of input control, by sequencing fragmented DNA with non-specific antibody like Immunoglobulin G, allows for normalization or calibration of the ChIP - Seq signal against the control [103].

- A drawback of all immunoprecipitation based methods is the requirement of a specific antibody with respect to the protein of interest. Obtaining such an antibody, and performing controls to validate its specificity is non-trivial. If such an antibody is not available, the only recourse is to fall back on the traditional binding site discovery paradigm using traditional motif detection techniques. New techniques, like DamID aims to identify binding sites without requiring protein specific antibodies, by expressing the protein of interest as a fusion protein with DNA methyltransferase [197]. The shortcoming of DAMID basd binding site detection is that the resolution of the method is variable across the genome, and is a function of the oligomer content of the genome, typically at 100s of bps. Further, it profiles a trace of where the protein interacted with DNA over a time period, rather than a fixed time snapshot. Typically, for transcription factors, there is low temporal variability.
- Another problem with ChIP-Seq technology is the typically large number of cells required for sequencing (typically in the millions) in order to obtain DNA material in the magnitude of nanograms for chromatin immunoprecipitation. Often, especially with samples that require a long protocol stretching into weeks or field samples, such large number of cells may not be available, and the only recourse is to fall back on traditional techniques of binding site analysis. However, recent improvements in ChIP-Seq protocols enable ChIP-Seq experiments on as few as one hundred cells [66]. Thus, all ChIP-Seq experiment outcomes are really based on populations of cells, and the mixture components are reflected in the statistics. They are especially sensitive to contamination or mixtures of cell types, which may

cause convolution of multiple ChIP-Seq signals : such convoluted signals can however, be teased apart by motif models, which can be made to work with inferencing mixture models.

However, it is not possible to harness single cell sequencing technology to obtain a single cell snapshot of the binding profile in order to understand competitive binding and steric hindrance effects among pairs of binding sites. For such situations, Competition-ChIP protocols [104] aims to profile kinetics of transcription factor binding, since binding site prediction by itself is a poor predictor of transcription factor function.

- Since ChIP-Seq techniques create a high-throughput, genome-wide profile of binding of the protein of interest, it is especially difficult to assign function to specific binding events. Typically, identification of peaks and validated motif instances in promoters can predict a regulatory role in the downstream coding region, but identification of peaks and motif instances in enhancers typically lead to the question : disambiguating which enhancer - promoter unit(s) is / are linked to the enhancer. Identifying function therefore translates to identifying linkages between the promoter and enhancer by mapping long range interaction using techniques like Chromatin Interaction Analysis by Paired-End Tag Sequencing (Chia-PET) [58] or Chromosome Conformation Capture (3C) [39], or by predicting enhancer - promoter units (EPUs) [169].
- Another aspect of ChIP-Seq technologies is that it will profile both direct and indirect binding. As a result, it is possible to obtain signal from both direct and co-factor mediated binding events. Typically, in motif search terms, this translates to finding multiple motifs in the bound regions, an important cue which may lead to clues about which co-factors may be at work. If the transcription factor - co-factor interaction is already known, it is possible to perform sequential ChIP-Seq protocols [63] to obtain a simultaneous snapshot of two binding profiles : in such situations, one way to understand whether the binding is co-operative, competitive or indirect is to resort to binding site analysis using motif detection. Co-binding studies is becoming one of the major areas of ChIP-Seq sequence analysis. Various algorithms have been developed for identifying direct versus indirect binding [6], identification of transcription factor complexes [201], and identifying co-binding transcription factors [209].

  However, for a gene of interest with unknown specificity, if the motifs found in a particular region of bound genome are novel, there is no way to distinguish between the facts of whether it is sequence specificity of the gene of interest or a co-factor. This has led to interest in *in vitro* methods of capturing sequence specificity of proteins. Both well-known methods of capturing sequence specificity in an *in vitro* fashion require the learning of sequence motifs in a model-based way in order to analyze the specificity in *in vivo* studies.

  - The traditional approach to sequence specificity and position weight matrix determination was based on evolving oligomers with high binding affinity to the protein in question, and is named Systematic Evolution of Ligands by Exponential Enrichment (SELEX) [136]. DNA motifs may then be learnt by sampling from the sequenced fragments. However, given that the probability of success in a SELEX experiment depends on a host of factors [100], including the number of rounds of the protocol, an alternative, easier-to-use on-chip protocol has become more popular recently.

- Martha Bulyk *et al* have developed an on-chip methodology for analyzing the binding specificity of a protein of interest in an *in vitro* fashion. The method of Protein Binding Microarray (PBM) [143] allows quantification of binding intensities of the protein in question to a large set of oligomers. The binding specificities can then be estimated in a systematic model-based manner and the motif extracted : multiple model based methods have been developed for this purpose [139].
- Another aspect of identification of binding specificities *in vitro* is to be able to compare them against a database of known binding specificities, with the goal of being able to identify the co-factor of family of co-factors for a protein of interest. Several such algorithms based on the motif model exist [164, 191].
- Finally, models of sequence specificity (like binding motifs or nucleotide content) is essential for deconvoluting sequence-specific versus non-specific binding [110].

- However, the most dominant use of motif analysis in ChIP-Seq studies is in the generalization of results across cell types, conditions and time-points. ChIP-Seq experiments are *in vivo* and provide only a single binding profile for the gene of interest at a particular time point under specific conditions, for a population of cells. The binding profile is inevitably cell-type and condition-specific. In order to infer binding profiles in other cell types, or at other conditions or time points in a model based fashion, either multiple ChIP-Seq experiments need to be carried out (not always feasible due to economic constraints and due to the fact that ChIP-Seq destroys the cell population) or integrative analysis can be performed to predict binding sites using various available cues for the data in question, like epigenetic marks, chromatin accessibility (experiments like DNAse-Seq [180] and FAIRE-Seq [131]) and the presence or absence of other binding factors, as well as temporal information (if relevant and available). Integrating multiple information sources for binding site prediction was the major focus of Chapter 4 of this thesis. The ability to identify discriminative evolutionary features was a focus of Chapter 5 of this thesis. The best genetic feature for identifying binding sites which remains the simplest to estimate is the Position Weight Matrix, which is why motif-based analysis has remained so valid for such a long time in such transfer learning scenarios, even in the face of rapidly changing technologies. There has been significant work in the literature on modelling such integrative analysis as multifeature classifiers [57], bayesian priors integrating epigenetic information [33], and in the perspective of integrative analysis for transcriptional activity prediction using multiple transcription factor binding data [28, 140].

In conclusion, it is best to understand transcription factor binding site analysis and the analysis of the evolution and architecture of regulatory regions not just as a standalone problem, but as the critical first step towards building up a theoretical, model-based understanding of the process of gene regulation as utilized in some preliminary work in the literature [28, 140].

Chapter 7

Appendix A: Details on the BayCis model and algorithm

A1. Modeling spacer length distribution via GhHMM

Consider the actual spacer length histogram in *D. melanogaster* in Figure 7.1. Smoothed distribution fitted by maximum likelihood estimation according to geometric, normal, and negative binomial distribution are also shown. The normal distribution is definitely a very poor approximation. In the tail, the exponential and the negative binomial is not very different but in the shorter region, the negative binomial provides a better fit to the distribution. Furthermore, the peak lies between 5 and 10, not lying between 0 and 5.

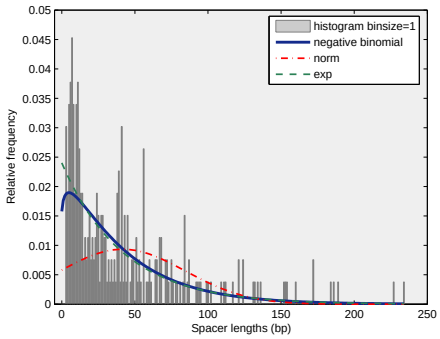


Figure 7.1: The histogram of spacer length distribution with known standard distributions superimposed.

Generalized hidden Markov models (GHMM) have been proposed for the explicit modeling of the state durations in an HMM [50, 101, 147]. A state in a GHMM does not generate one character at a time but instead a region of arbitrary length. The length of the regions is determined according to an explicit duration distribution

The explicit duration models accurately models the state durations at the cost of computation. Alternatively, the negative binomial distributions can be modeled by using instead of one self-transiting state, several externally indistinguishable but internally distinguishable states joined together, as shown in Figure 7.2. This allows approximation of the GHMM functionality in a HMM [43], where the efficient forward-backward and posterior decoding algorithms can be reused.

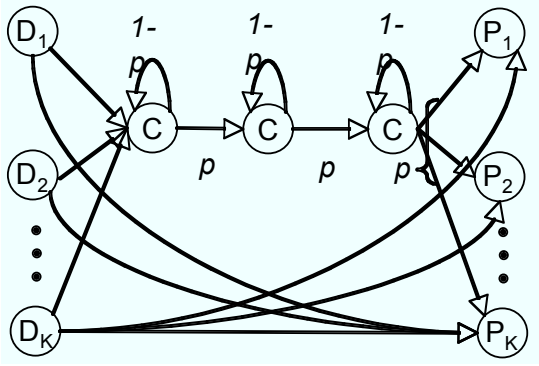


Figure 7.2: The state-transition diagram of a gHMM.

In the GhHMM version of BayCis, we model the cluster background as negative binomial distribution, but leave the global, proximal and distal background as geometric distribution. Unlike the Poisson distribution, the negative binomial distribution can model different mean and variance, allowing a better fit to the empirical distribution shown in Figure 7.1. This scenario has been used to model exon length distribution by EasyGene to achieve better accuracy [97]. To control computation cost, we approximate the negative binomial distribution by joining several geometrically distributed states. This also makes assigning conjugate priors possible, which will be explained in detail shortly. For the global background, the length distribution has a heavy tail, and in practical usage of BayCis system its length is dependent on how the user cuts the upstream sequence. For the proximal and distal background, the lengths tend to be very short, and the joining of a distal and then a proximal background already provides better expressive power.

A2. Details on Flattening hHMM and the modified FB-algorithm

When a hHMM is flattened to a HMM, if there are re-used models in the hHMM, these models must be duplicated, and the heirarchical structure will be lost under unsupervised learning of the parameters [130]. If the hierarchy is a tree, as in BayCis hHMM, the hHMM can be converted to a HMM without losing the hierarchical structure. The HMM state space is exactly the production states in the hHMM, denoted as $\mathbb{Q} = \{b_g, b_c\} \cup \mathbb{B} \cup \left(\cup_k \mathbb{M}_k \right)$.

Due to the sparsity of our transition probability matrix, as shown in Figure 2, we can further reduce the time complexity of inference for obtaining the probability of a hidden state given the sequence, i.e. the forward-backward algorithm, which is a subroutine in the Bayesian learning algorithm. For notational simplicity, we assume the number of cluster background states is 3. The state space consists of a global background, 3 cluster backgrounds, K proximal and distal backgrounds, and $2L_k$ motif states for each motif k (including sense and antisense), so the total size of the state space N is

$$N = 4 + 2K + 2\sum_{k=1}^{K} L_k.$$

Following Rabiner's notation [147], let $\alpha_t(j)$ be the probability of the partial sequence $Y_1 \cdots Y_t$ and state s_j at location t, or $\alpha_t(j) = p(Y_1 \cdots Y_t, X_t = s_j)$. Let $\beta_t(j)$ be the probability of the partial sequence $Y_{t+1} \cdots Y_T$ given the state s_j at location t, or $\beta_t(j) = p(Y_{t+1} \cdots Y_T | X_t = s_j)$ (in this section the term $\beta_t(j)$ is used in backward algorithm for convention, not to be confused with the parameters $\beta_{g,k}, \beta_{c,k}$, etc.) The induction step in the forward and backward algorithm are thus

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) A_{ij}] B_j(Y_{t+1}), \qquad t = 1, 2, \cdots, T-1, 1 \le j \le N, \tag{7.1}$$

$$\beta_t(i) = \sum_{j=1}^{N} A_{ij} B_j(Y_{t+1}) \beta_{t+1}(j), \qquad t = T-1, T-2, \cdots, 1, 1 \le j \le N, \tag{7.2}$$

It is known that the standard forward and backward algorithm both take $O(N^2 T) = O(K^2 \bar{L}^2 T)$, where $\bar{L}$ is the averaged motif length, $\bar{L} = \frac{1}{K} \sum_{k=1}^{K} L_k$. If there are many motifs, the amount of calculations in the forward algorithm may still be large. Our modified forward-backward algorithm further reduces the amount of calculations in the matrix multiplication in (7.2), based on the fact that "non-trivial" transitions, i.e. transitions whose probability is not 0 nor 1, are restricted to transitions from any of the background states going to either any background state or to the first sense/ last antisense motif position. These transitions correspond to a smaller block of size $(4 + 2K)$ by $(4 + 4K)$ in the transition probability matrix, marked as "'non-trivial transitions"' in Figure 2. With this observation, the modified induction step in the forward algorithm is described here. The vector $\tilde{\alpha}$ is a holder for temporary values.

1. Let $\tilde{\mathbb{Q}}_1$ and $\tilde{\mathbb{Q}}_2$ be the sets of source and target states of the non-trivial transitions, repectively. Formally speaking, if $0 < A_{ij} < 1$, we know $i \in \tilde{\mathbb{Q}}_1$ and $j \in \tilde{\mathbb{Q}}_1$, where

$$\begin{aligned} \tilde{\mathbb{Q}}_1 &= \{b_g, b_c, b_p^{(1)}, \cdots, b_p^{(K)}, b_d^{(1)}, \cdots, b_d^{(K)}\}, \\ \tilde{\mathbb{Q}}_2 &= \tilde{\mathbb{Q}}_1 \cup \{1^{(1)}, 1^{(2)}, \cdots, 1^{(K)}, L^{(1')}, L^{(2')}, \cdots, L^{(K')}\} \end{aligned}$$

2. Forward induction: for each $t = 1, 2, \cdots, T-1$,

$$\begin{aligned} \tilde{\alpha}(j) &\leftarrow \sum_{i \in \tilde{\mathbb{Q}}_1} \alpha_t(i) A_{ij}, \quad j \in \tilde{\mathbb{Q}}_2, \\ \tilde{\alpha}(l^{(k)}) &\leftarrow \alpha_t((l-1)^{(k)}), \quad 2 \le l \le L_k, 1 \le k \le K, \\ \tilde{\alpha}(l^{(k')}) &\leftarrow \alpha_t((l+1)^{(k')}), \quad 1 \le l \le L_k - 1, 1 \le k \le K, \\ \tilde{\alpha}(b_d^k) &\leftarrow \tilde{\alpha}(b_d^k) + \alpha_t(L_k^{(k)}) + \alpha_t(1^{(k')}), \quad 1 \le k \le K, \\ \alpha_{t+1}(j) &\leftarrow \tilde{\alpha}(j) B_j(Y_{t+1}), \quad j \in \mathbb{Q} \end{aligned}$$

3. Backward induction: for each $t = T-1, T-2, \cdots, 1$,

$$
\begin{aligned}
\beta_t(i) &\leftarrow \sum_{j=1}^{N} A_{ij} B_j(Y_{t+1}) \beta_{t+1}(j), \quad i \in \tilde{\mathbb{Q}}_1, j \in \tilde{\mathbb{Q}}_2 \\
\beta_t(l^{(k)}) &\leftarrow B_{(l+1)^{(k)}}(Y_{t+1}) \beta_{t+1}((l+1)^{(k)}), \quad 1 \le l \le L_k - 1, 1 \le k \le K, \\
\beta_t(l^{(k')}) &\leftarrow B_{(l-1)^{(k')}}(Y_{t+1}) \beta_{t+1}((l-1)^{(k')}), \quad 2 \le l \le L_k, 1 \le k \le K, \\
\beta_t(L_k^{(k)}) &\leftarrow B_{b_d^k}(Y_{t+1}) \beta_{t+1}(b_d^k), \quad 1 \le k \le K, \\
\beta_t(1^{(k')}) &\leftarrow B_{b_d^k}(Y_{t+1}) \beta_{t+1}(b_d^k), \quad 1 \le k \le K,
\end{aligned}
$$

The time complexity of the modified forward-backward algorithm is $O((K^2 + K\bar{L})T)$. Since the motif length is typically short, we can assume $\bar{L} < K$ and the time complexity of the modified forward-backward algorithm will be $O(K^2 T)$, instead of $O(K^2 \bar{L}^2 T)$ of the standard forward-backward algorithm.

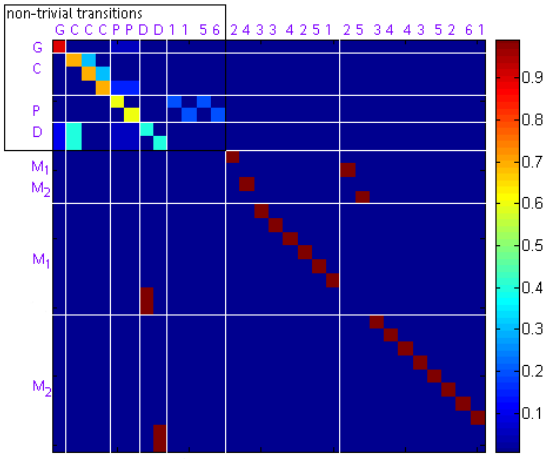


Figure 7.3: The transition probability matrix of the flattened HMM, shown as a heat map. G, C, P, D, and the numbers correspond to global, cluster, proximal and distal background, and the motif states. The motif states are ordered as: $1^{(1)}, 1^{(2)}, \cdots, 1^{(K)}, L_1^{(1')}, L_2^{(2')}, \cdots, L_K^{(K')}, 2^{(1)}, (L_1-1)^{(1')}, 3^{(1)}, (L_1-2)^{(1')}, \cdots, L_1^{(1)}, 1^{(1')}, \cdots, 2^{(K)}, (L_K-1)^{(K')}, 3^{(K)}, (L_K-2)^{(K')}, \cdots, L_K^{(K)}, 1^{(K')}$.

A3. Posterior decoding of DNA binding sites

We can read off the functional annotation (or segmentation) of the input sequences from the posterior probability distribution of the functional states at each position of the sequences according to a *maximal a posteriori* (MAP) scheme. In this scheme, the predicted functional state X_t^* of position t is: $X_t^* = \arg\max_{s \in \mathbb{S}} p(X_t = s|Y)$, where S is the set of functional states (motifs and different kinds of background) and Y is the observed (genomic) sequence.

Note that by using such a posterior decoding scheme (rather than a Viterbi), we integrate the contributions of all possible functional-state-paths for the input sequence (rather than a single "most probable" path), into the posterior probability of each position. Therefore, although in the

HMM architecture we do not explicitly model overlapping motifs, our inference procedure does take into account possible contributions of DNA binding sites interacts with competing TFs.

A4. Bayesian inference and learning

Under the Bayesian framework described in the main chapter, the parameters in the HMM are treated as continuous random variables (collectively referred as Ξ) with a prior distribution. Now to compute the posterior probability of functional states, we need to marginalize out these parameter variables:

$$p(X_t|Y) = \int p(X_t = s|Y, \Xi)p(\Xi|Y)d\Xi \tag{7.3}$$

This computation is intractable in closed form. One approach to obtain an approximate solution is to use Markov chain Monte Carlo methods (e.g., a Gibbs sampling scheme). Here we use a more efficient, deterministic approximation scheme based on *Generalized Mean Field* inference [206], also referred to as *variational Bayesian learning* [64] in the special scenario applied to our problem setting. Omitting theoretical and technical details, our algorithm can be understood as replacing the single-round posterior decoding with an iterative procedure consisting of the following two step:

- Compute the expected counts for all state-transition events (formally called sufficient statistics) using the forward-background algorithm, using **current** values of the HMM parameters.
- Compute the Bayesian estimation (to be detailed shortly) of the HMM parameters based on its prior distribution and the expected sufficient statistics from last step. **Update** the HMM parameters with these estimations.

This procedure is different from the standard EM algorithm which alternates between inference about the hidden variables (the E step) and maximal likelihood estimation of the model parameters (the M step). In our algorithm, the "M" step is a Bayesian estimation step, in which we compute the posterior expectation of the HMM parameters.

Now we outline the formulas for Bayesian estimation of the HMM parameters. Note that since the state-transition probability distributions (which are multinomial) and the prior distributions (which are either beta or gamma) of the transitioning parameters are conjugate-exponential [9] [1], we have to compute the Bayesian estimation of the logarithm of the transitioning parameters (referred to as the *natural parameterizations*) rather than of the parameters themselves. For example,

[1] Strictly speaking, this claim is only partially true. Because the conjugacy only applies to the transition probability between a pair of states, but not to the total transition probability mass from a state of interest to all motif-buffer states, $\sum_{k\in\mathbb{B}_p} \beta_{[\cdot,k]}$, which is treated as a single "motif-buffer-going" probability in our beta or gamma prior models. (Defining priors for each individual $\beta_{[\cdot,k]}, k \in \mathbb{B}_p$ would require too many hyper-parameters.) As a heuristic surrogate, in certain computational step, we split the *prior mass* (total pseudocounts) corresponding to the total "motif-buffer-going" probability equally among all individual "motif-buffer-going" probabilities as if each has its own pseudocounts, and install strict conjugacy. Since each prior distribution involves at most one such "motif-buffer-going" probability, and that the state-transition probabilities are multinomial parameters subject to a normalization constrain, we only need to use the installed conjugate-exponential property for Bayesian parameter estimation for each "non-motif-going" transition probability, and then obtain the Bayesian estimation of the total "motif-buffer-going" probability indirectly, by subtracting all newly estimated "non-motif-going" transition probabilities from 1.

for the state-transitioning parameter $\beta_{g,g}$, we have:

$$E[\ln(\beta_{g,g})] \int_{\beta_{g,g}} \ln \beta_{g,g} p(\beta_{g,g}|\xi_{g,1}, \xi_{g,2}, E[n_{g,g}]) d\beta_{g,g}$$
$$= \Psi(\xi_{g,1} + E[n_{g,g}]) - \Psi(\sum_j \xi_{g,j} + \sum_{k \in \mathbb{B}_p} E[n_{g,k}]), \tag{7.4}$$

where $\Psi(x) = \frac{\partial \log \Gamma(x)}{\partial x} \frac{\Gamma'(x)}{\Gamma(x)}$ is the digamma function; $E[\cdot]$ denotes the expectation with respect to the posterior distribution of the argument; and $n_{g,g}$ refers to the sufficient statistic of parameter $\beta_{g,g}$ (i.e., counts of transitioning event $g \rightarrow g$). The Bayesian estimate of the original parameter is simply $\beta^*_{g,g} = \exp(E[\ln(\beta_{g,g})])$. (In fact we will keep using the natural parameterization in the actual forward-background inference algorithm to avoid numerical underflow caused by long products of probability terms.)

The total "motif-buffer-going" probability is estimated as described in footnote 1, e.g., $\beta^*_{g,\bar{g}} = \sum_{k \in \mathbb{B}_p} \beta^*_{g,k} = 1 - \beta^*_{g,g}$. To estimate each individual "motif-buffer-going" probability, we use the standard Baum-Welch update based on expected sufficient statistics computed from the matrix of co-occurrence probabilities $p(X_t, X_{t+1}|Y)$, scaled by the Bayesian estimation of the total "motif-buffer-going" probability, for example:

$$\beta_{g,i} \quad = \quad \beta^*_{g,\bar{g}} \frac{\sum_t p(X_t = g, X_{t+1} = i|Y)}{\sum_{t,k} p(X_t = g, X_{t+1} = k|Y)} \tag{7.5}$$

The initial state probability of the the *BayCis* HMM is not important for CRM prediction as it only directly determine the functional state of the first position of the input sequences and its influence diminishes quickly along the sequence. We simply fix the initial state to be a global background with probability 1.

A5. Bayesian learning of the GHMM parameters

The Bayesian estimation of the GHMM parameters is similar to the estimation of the HMM parameters, with some modifications. Note that although we use HMM state space to simulate a negative binomial duration distribution, the self-transition probability of all the cluster background state must remain the same. Otherwise, the duration distribution will no longer be negative binomial. Hence the averaged number of self-transitions and transitions to the next state is used.

Let c^j denotes the j-th cluster background states, n_{c^j,c^j} denotes the number of self transition on state c^j, $n_{c^j,c^{j+1}}$ denotes the number of transition from state c^j to c^{j+1}. Let $E[n_{c,c}]$ denotes the average of expected number of self-transitions from every cluster background states, and $E[n_{c,c1}]$ denotes the average of expected number of transitions out of every cluster background states, defined as:

$$E[n_{c,c}] = \frac{1}{\xi_{cr}} \sum_{j=1}^{\xi_{cr}} E[n_{c^j,c^j}], \tag{7.6}$$

$$E[n_{c,c1}] = \frac{1}{\xi_{cr}} \Big(\sum_{j=1}^{\xi_{cr}-1} E[n_{c^j,c^{j+1}}] + \sum_{k \in \mathbb{B}_p} E[n_{c^{\xi_{cr}},k}] \Big) \tag{7.7}$$

Bayesian estimation of the expected value of (log) self-transition probability, with respect to the posterior distribution, would be

$$E[\ln(\beta_{c^j,c^j})]\Psi(\xi_{c,1} + E[n_{c,c}]) - \Psi(\xi_{c,1} + \xi_{c,2} + E[n_{c,c}] + E[n_{c,c1}]) \quad 1 \leq j \leq \xi_{cr}. \tag{7.8}$$

As in other parameters, the natural parameterization $\ln(\beta_{c^j,c^j})$ is used, but when the Bayesian estimation of the original parameter is preferred, we use $\beta^*_{c^j,c^j} = \exp(E[\ln(\beta_{c^j,c^j})])$.

A6. The Drosophila TRS dataset

We tested our model on a selective dataset consisting of transcriptional regulatory regions regulating the *Drosophila melanogaster* developmental genes. Each TRS in the dataset consists of the CRMs pertinent to a particular gene, any intra-CRM background inbetween, with flanking regions on either side of the extremally located CRMs such that the entire sequence is at least 10K bp long, and the boundaries of the dataset are at least 2K bp from the extremal CRMs. We included the exonic regions of the genes only when they fell in the aformentioned selected region, and not otherwise.

Selection of the datasets was based on the REDfly CRM database and the Drosophila Cis-regulatory Database at the National University of Singapore [61, 132]. We initially chose 89 CRMs pertaining to 34 early developmental genes. This selection was based on a filtering of CRMs, through which we only chose CRMs which were at least 200 bp long, and contained at least 5 motif instances (2 CRMs with a borderline count of 4 motif instances were also included).

All motif instances used were based on biological curation, and motif instances of the same type in the database often correspond to varying lengths of nucleotide sequences. This is at odds with most computational models of the motifs, which assume a fixed length of the motif in terms of nucleotides. We overcome this issue by searching a 10 bp neighborhood of the annotated location for a fixed width nucleotide sequence which has a high log odds probability of being a motif over background (based on the PWM counts of the motif). Since both our motif algorithm and most competing motif search algorithms assume a PWM based model of the motif, this curation provides more accurate annotation data without placing any competing algorithm at a disadvantage. A short summary of our input sequences is provided in this section.

This database is available online at `http://www.sailing.cs.cmu.edu/BayCis`. Each TRS is graphically depicted with color coded CRM and motif regions, and is extensively hyperlinked so that the corresponding sequences may be obtained by clicking on a relevant gene dataset or CRM. A snapshot of the front page of the online database is shown in the main chapter.

$Gene(Length)$	$CRM/Length$	$Motif$	$Gene/Length$	$CRM(Length)$	$Motif$
1.28 (10072)	1.28_DRE / 664	DEAF1 / 8 DFD / 4	abd-a (10045	abd-A)_iab-2(1.7) / 1745	EVE / 4 KR / 1 GT / 1 HB / 5
alphaTub84B (10055)	alphaTub84B_alpha1-tubulin_promoter / 855	TRL / 5	ap (10050)	ap_ApME680 / 680	ANTP / 5
bap(10000)	bap_baplac4.5 / 4957	MAD / 4	betatub60D (10181)	betaTub60D_beta3-14/vm1 / 524	BAP / 1 UBX / 2
ct(10068)	ct_wing_margin_enhancer / 2692 wingmargin_Guss / 668	SD / 7	dfd (11658)	Dfd_EAE / 2658 Dfd_EAE-D / 833 Dfd_EAE-F9 / 329 EAE-F2 / 392	DEAF1 / 2 DFD / 13 EXD / 1
dpp (30199)	dpp_dpp813 / 812 dpp_dpp261 / 256 dpp_dpp419 / 419 dpp_intron2 / 1983 dpp_dl_mel / 539 dpp_BS1.0 / 8801 dpp_BS1.1 / 1738	ABD-A / 9 BIN / 3 DL / 14 EN / 5 EXD / 5 GRH / 1 UBX / 13	en (11004)	en_stripe_enhancer_intron_1 / 900 en_intron / 720 en_upstream_enhancer / 2401	EN / 6 EVE / 3 FTZ / 12 FTZ-F1 / 2 HB / 2 KR / 1 ZEN / 3
ems (10304)	ems_elementIV / 304 ems_ARFE / 1244	ABD-B / 7 TLL / 2 BCD / 2 EMS / 3	twi (10415)	twi_dl_mel / 1415	DL / 7
ftz (10487)	ftz_upstream_enhancer / 2562 ftz_proxA / 580 ftz_Prox-323 / 324 ftz_neurogenic_enhancer / 2250 ftz_zebra_element / 745	CAD / 2 FTZ / 21 FTZ-F1 / 1 GRH / 4 TTK / 4 HR39 / 1 SLP1 / 1	salm (10144)	salm_salE/Pv / 1078 salm_wingpouch_Guss / 328 salm_blastoderm_early_enhancer / 512 salm_sal242S/P / 242 salm_sal272P/P / 276	BCD / 7 CAD / 4 HB / 1 HKB / 2 SD / 2 KR / 3 UBX / 5
h (10867)	h_stripe_3+4_ET22 / 1745 h_h7_element / 932 h_stripe_6+2 / 1081 h_stripe_6 / 547	BCD / 10 HB / 29 KNI / 22 KR / 13 TLL / 7	hb (12055)	hb_0.7 / 730 hb_anterior_activator / 245 hb_HZ1.4 / 1421 hb_upstream_enhancer / 1424 hb_HZ526 / 528	BCD / 8 HB / 1 TLL / 9
kni (15498)	kni_KD / 870 kni_L2_enhancer / 1360	BCD / 2 CAD / 1 GT / 2 TLL / 6 HB / 8 KR / 4 HIS2B / 5 SD / 5	kr (11348)	Kr_CD1 / 1159 Kr_StBg1.2HZ / 1130 Kr_StH0.6HZ / 540 Kr_H/I / 950 Kr_Kr/F / 1587	BCD / 4 GT / 1 HB / 6 KNI / 1 TRL / 7 TLL / 7
otp (10000)	otp_C / 441	BYN / 4	rho (10589)	rho_NEE-600 / 590 rho_NEE-300 / 328 rho_NEE / 299	DL / 4 SNA / 4 TWI / 2
gsb (10916)	gsb_fragIV / 516	EVE / 3 FTZ / 3 PRD / 7	ser (10000)	Ser_minimal_wing_enhancer / 812	AP / 14 SUH / 2 PAN / 9
scr (13258)	Scr_5.5HH / 5653 Scr_3.0XX / 2953 Scr_6.5KS / 6985	CAD / 2 SLP1 / 1 FTZ / 21 GRH / 4 FTZ-F1 / 1 HR39 / 1 TTK / 4	tsh (11144)	tsh_enhancer / 2144 tsh_del-1-5 / 463 tsh_220bp / 221	ABD-A / 4 ANTP / 4 FTZ / 4 UBX / 4
slp1 (10000)	slp1_5-2 / 1554	PAN / 9	sna (10013)	sna_2.8kb / 2913 sna_VA / 612	DL / 10 TWI / 2
so (10012)	so_so10 / 428 so_so7 / 1612	EY / 3 TOY / 5	tll (10063)	tll_P2 / 2764 tll_P3 / 1725	BCD / 8 TRL / 1 GRH / 1 TTK / 1
tin (10000)	tin_tinD / 350	MAD / 7 MED / 3 TIN / 2	sim (10065)	sim_mesectoderm / 631	SNA / 3 TWI / 2
eve (14256)	eve_stripe_3+7 / 511 eve_stripe2 / 484 eve_MHE / 312 eve_EME-B / 395 eve_EME-B5 / 233 eve_eme2 / 300 eve_EME-B3 / 262	BCD / 5 GT / 3 HB / 12 KNI / 5 KR / 10 MED / 5 TIN / 4 PAN / 6 ZFH1 / 1	ubx (78414)	Ubx_bx1 / 1705 Ubx_BRE / 502 Ubx_basal_promoter / 1189 Ubx_PRE_polycomb_response_element / 1556 Ubx_PBX_enhancer / 1378 Ubx_pbxPB / 297 Ubx_pbxSB / 623 Ubx_pbxAS / 584	EN / 5 EVE / 2 ZEN / 2 FTZ / 10 TLL / 5 GRH / 1 TRL / 17 HB / 27 KNI / 3 TWI / 6 KR / 1 UBX / 2 PHO / 5 Z / 20
vg (12096)	vg_boundary_enhancer / 754 vg_minimal_boundary_enhancer / 360 vg_quadrant_enhancer / 798	MAD / 2 NUB / 4 SUH / 1 SD / 4 VVL / 1	w (11737)	w_Bmdel-W / 6628 w_HPst-W / 7737 w_H-del-BgRVdel-W / 770	Z / 11
zen (10662)	zen_0.7 / 726 zen_1.4 / 1513 zen_dorsal_ectoderm / 624	BRK / 6 DL / 3 GRH / 1 MAD / 10			

Table 7.1: Summary of the Drosophila TRS dataset used for in performance comparison.

A7. Hyperparameter selection scheme

Choosing hyperparameters for transition probabilities can be a difficult problem and has significant impact on the performance of the model. As discussed in the Methods section, the hyperparameters of the BayCis model reflect prior beliefs about the architectural features of the CRM structure, such as rough spans of the inter- or intra-module background and distances between motif instances.

A standard way of specifying hyperparameters would be to see which parameter settings work best for datasets with known TFBS, and apply the same on all datasets on which TFBS discovery is to be performed. This is somewhat similar to the supervised learning setup of "training" and "test" sets. The basic assumption here is that in CRMs regulating genes of similar functionality, the CRM architecture would be somewhat similar causing the same set of hyperparameters to work well. More formally, the hyperparameters can be also estimated in the maximal likelihood fashion based on the empirical Bayes principle. We chose to use a representative dataset based on the CRMs of the *even-skipped* gene to choose our hyperparameters for the hHMM and GhHMM.

Based on our observations, the most important hyperparameters governing precision and recall are those regulating transition probabilities into and out of the CRM background state(s). The CRM background state(s) and motif specific states are the only states from where one can enter the motif specific states of the HMM. Hence, hyperparameters which cause the HMM to stay in the CRM background states more frequently than usual risk a low precision, high recall performance while hyperparameters which cause the CRM background states to be rarely visited risk a high precision, low recall scenario. Accurate prediction of CRMs cause the HMM to obtain acceptable values of precision and recall.

We specify the hyperparameters as follows: for the global background, $\omega_g = 0.002$; for the inter-module background, $\omega_c = 0.05$; for the proximal motif buffer, $\omega_p = 0.25$; for the distal buffer hyperparameters, $\omega_{d,1} = 0.125$ (distal to global background) $\omega_{d,2} = 0.125$ (distal to clustal background) and $\omega_{d,3} = 0.25$ (distal to proximal buffer), and the strength of the hyperparameters are set to $1/10$ of the expected counts of the transitions on a 15 kbp dataset with the exception of ω_g which is set to $10,000$. The background probability of the nucleotide at each position was computed locally using a 2nd-order Markov model from a sliding window of 1100 bp centered at the corresponding position. For the GhHMM, based on visual inspection of spacer length distributions between motifs, we choose the parameter as $r = 2$.

A8. More on F1 and CC scores

The nucleotide-based prediction error is used in the Nature Biotechnology benchmark paper by Tompa et al. [195]. The formulas for the F1 and CC scores are as follows:

$$CC = \frac{nTP \times nTN - nFN \times nFP}{\sqrt{(nTP + nFN)(nTN + nFP)(nTP + nFP)(nTN + nFN)}}, \tag{7.9}$$

$$F1 = \frac{2 \times Pr \times Re}{Pr + Re}, \tag{7.10}$$

where $Pr = \frac{nTP}{nTP+nFP}$ (Precision) and $Re = \frac{nTP}{nTP+nFN}$ (Recall).

Both CC and F1 are calculated from the number of nucleotides (single positions) that are correcly/wrongly predicted as positives/negatives. The value range of CC is in principle between -1 and +1 (as it is a correlation), but in practice it would lie between 0 (random predictions) and 1 (perfect predictions). As F-1 measure is also a value between 0 and 1, we use the same numerical units in the plot.

Chapter 8

Appendix B: Details on the CSMET model and algorithm

B1. The Molecular and Functional Substitution Model

We use the Felsenstein 1984 model (F84) [49], which is similar to the Hasegawa - Kishino - Yano's 1985 model (HKY85) [77] and widely used in the phylogenetic inference and footprinting literature [49, 123], for nucleotide substitution in our motif and background phylogeny. Formally, F84 is a five-parameter model, based on a stationary distribution $\pi \equiv [\pi_A, \pi_T, \pi_G, \pi_C]'$ (which constitutes three free parameters as the equilibrium frequencies sum to 1) and the additional parameters κ and ι which impose the transition/transversion bias. Using concise notation for the purine frequency $\pi_R = \pi_A + \pi_G$ and pyrimidine frequency $\pi_Y = \pi_T + \pi_C$, the instantaneous rate matrix can be written as:

$$\mathbf{Q}_N = \begin{pmatrix} * & (1+\kappa/\pi_Y)\iota\pi_C & \iota\pi_A & \iota\pi_G \\ (1+\kappa/\pi_Y)\iota\pi_T & * & \iota\pi_A & \iota\pi_G \\ \iota\pi_T & \iota\pi_C & * & (1+\kappa/\pi_R)\iota\pi_G \\ \iota\pi_T & \iota\pi_C & (1+\kappa/\pi_R)\iota\pi_A & * \end{pmatrix} \tag{8.1}$$

Since rows of the instantaneous rate matrix must sum to zero, the starred elements of the matrix are determined from the other 3 elements of the row, and not shown for clarity. According to the continuous-time Markov process theory, the corresponding nucleotide-substitution probability matrix over a period of time t is given by $P_N(t) = e^{Q_N t}$. To apply this model to a motif or a background phylogeny, we set the stationary distribution π to be the empirical nucleotide-frequency in the corresponding sequence entity that the phylogeny is defined on (e.g., for phylogeny $T_m^{(l)}$ defined on site l of a motif, we let $\pi \equiv \theta_l$, the l-th column of the PWM of the motif), and the nucleotide-substitution probability from an internal node c to its descendant c' along a tree branch of length b can be expressed as follows:

$$P_N(V_{c'} = j | V_c = i, \beta) = e^{-(\kappa+\iota)b}\delta_{ij} + e^{-\iota\beta}(1 - e^{-\kappa\beta})\Big(\frac{\pi_j}{\sum_h(\pi_h\epsilon_{jh})}\Big)\epsilon_{ij} + (1 - e^{-\iota\beta})\pi_j, \tag{8.2}$$

where i and j denote nucleotides, δ_{ij} represents the Kronecker delta function, and ϵ_{ij} is a function similar to the Kronecker delta function which is 1 if i and j are both pyrimidines or both purines,

but 0 otherwise. The summation in the denominator concisely computes π_R or π_Y.

A less concise, but more intuitive parameterization involves the overall substitution rate per site μ and the transition/transversion ratio ρ, which can be easily estimated or specified. We can compute the transition matrix P_N from μ and ρ using Eq. (8.2) based on the following relationship between (κ, ι) and (μ, ρ):

$$\kappa = \frac{2\pi_R\pi_T\rho - (2\pi_A\pi_G + 2\pi_C\pi_T)}{(2\pi_A\pi_G/\pi_R + 2\pi_C\pi_T/\pi_Y)} \frac{\mu}{1+\rho}, \qquad \iota = \frac{1}{2\pi_R\pi_Y} \frac{\mu}{1+\rho}.$$

To model functional turnover of aligned substrings along functional phylogeny T_f, we additionally define a substitution process over two characters (0 and 1) corresponding to presence or absence of functionality. Now we use the Jukes-Cantor 1969 model (JC69) [86] for functional turnover due to its simplicity and straightforward adaptability to an alphabet of size 2. The JC69 model is a single parameter model, using an instantaneous substitution rate μ which is confounded with the time variable. The instantaneous rate matrix under JC 69 is:

$$\mathbf{Q}_F = \begin{pmatrix} -\mu & \mu \\ \mu & -\mu \end{pmatrix}. \tag{8.3}$$

And the transition probability along a tree branch of length β (which now represents the product of substitution rate μ and evolution time t, which are not identifiable independently,) is defined by:

$$\mathbf{P}_F = \begin{pmatrix} \frac{1}{2} + \frac{1}{2}e^{-2\beta} & \frac{1}{2} - \frac{1}{2}e^{-2\beta} \\ \frac{1}{2} - \frac{1}{2}e^{-2\beta} & \frac{1}{2} + \frac{1}{2}e^{-2\beta} \end{pmatrix}. \tag{8.4}$$

From Eqs. (8.2) and (8.4), we can see that the likelihood of aligned nucleotides and functional states can be expressed as a function of the evolutionary parameters, based on which a maximum likelihood estimation of these parameters can be obtained from training data.

B2. Multi-specific CRM simulation and experimental setup

The synthetic CRMs where true TFBS annotations are known for evaluating CSMET are generated as follows. First, the simulator stochastically samples the evolutionary trees of motif, background, and functional-annotation, T_m, T_b and T_f, from the prior distributions (recall that each tree is a three-tuple including the stationary distribution, the tree topology, and the branch lengths). The Felsenstein transition/transversion coefficient can in principle be also sampled, but for simplicity and biological validity we pre-specify it to be 2. Then it simulates motif instances, background sequences, and functionality states (that determine motif turnover) in different taxa from their respective evolutionary trees under certain substitution rates. It can also simulate motifs with changing substitution rates according to a scheduling along a sequence, or in random order. Then it uses the global HMM to generate positional organization of the motifs and backgrounds in the CRM. Finally these building blocks are put together to synthesize an artificial CRM. This simulator can be used to simulate realistic multi-specific CRMs resulting from various nontrivial evolutionary dynamics. It is useful in its own right for consistency/robustness analysis of motif evolution models and performance evaluation of comparative genomic motif-finding programs.

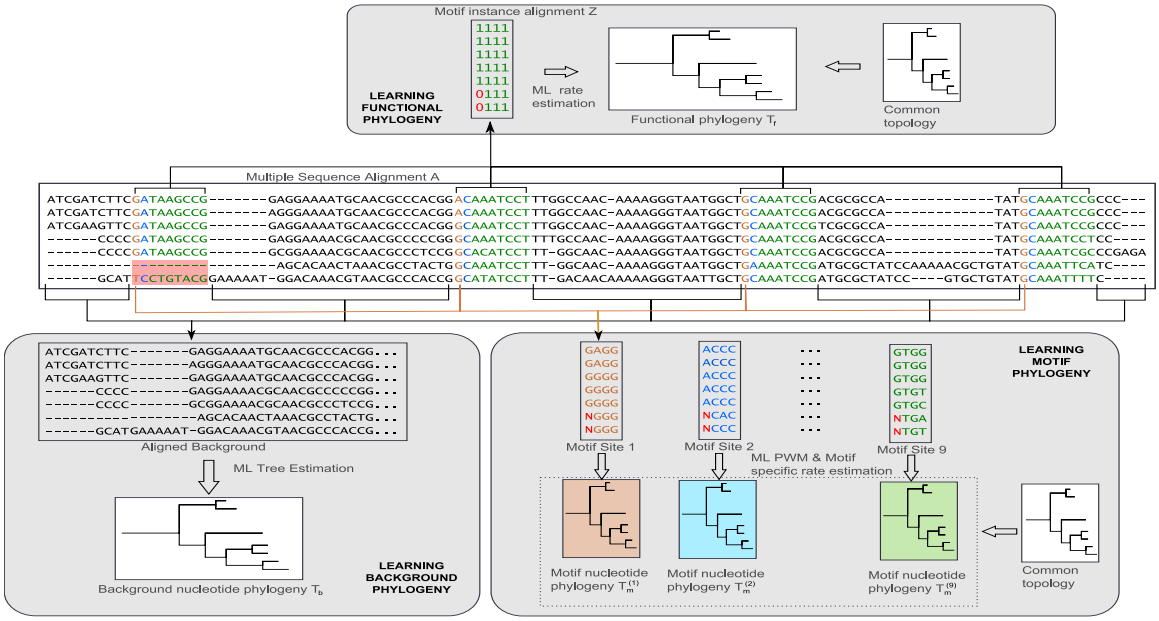


Figure 8.1: A schematic diagram of CSMET training. For the functional phylogeny, motif-instance alignments were generated by concatenating columns of indicators of motif presence/loss along the sequence alignment; and the scaling factor was fitted using the common topology. For the motif phylogeny, the nt-alignment of only each attendant site was generated by concatenating all columns of aligned nucleotides from that site and the corresponding multinomial estimated from them; the common topology was used for all sites. The motif specific mutation rate and scaling factor were estimated using the common topology from aligned nucleotides corresponding to all motif sites. For the background phylogeny, all segments of inter-motif sequences and flanking regions of CRMs were used.

We performed three sets of simulation experiments based on simulated datasets. In each case, we generate a data set of CRM alignments from the simulator that is simulating a pre-specified coupled functional and molecular evolution processes unknown to the programs used in the test phase. Each data set contains 50 simulated alignments, each of which is 1500 basepairs in length and includes 10 taxa whose divergence is controlled by the topologies and the branch lengths of the functional and molecular phylogenies being used. Each alignment contains instances of a single type of motif, whose length is set to be 8-bp. The parameters of the generative model used for the simulations are chosen to be representative of such parameters estimated from real biological data.

The density of motif instances is subject to a systematic adjustment for each data set over a wide range to generate problems of different degrees of difficulty.

The experiment for evaluating performance of CSMET under varying TFBS turnover rates was performed by using a different annotation tree for each experimental point. An initial benchmark evolutionary tree was chosen with branch lengths and topology based on estimation from actual nucleotide alignments on 11 aligned fly species. All parameters of the Jukes Cantor model based evolutionary tree were kept fixed across experimental data points, except for the fact that the branch lengths were scaled by a constant factor at each data point with respect to the initially chosen tree. The scaling factors correspondingly used for the data points were respectively: 1.50, 2.00, 2.50, 3.00, and 3.50. With increasing branch lengths, the amount of turnover per site in the simulated data increases - for a scaling factor tending to infinity the turnover model becomes random and approximates 50% For our data points, the estimated turnover rates corresponding to the chosen scaling factors were : 25%, 30%, 32%, 34% and 36%.

The simulated sequences with non-uniform TFBS turnover rates were generated by allowing the annotation tree scaling factor to vary for each motif block inside every simulated sequence. The scaling factor for each instance of a generated motif was equiprobably picked from the values of 1.00, 1.50, 2.00 and 2.50 . The corresponding turnover rates were 20%, 25%, 30% and 32%.

B3. Evaluation

Given each 1500bp multiple alignment, we use 1000 bp for training, and the remaining 500 for testing the performance of the trained models. We base our evaluation of every program on three commonly used evaluation metrics - precision, recall and the F1 score (i.e., the harmonic mean) based on precision and recall [194]. The precision is defined as the ratio of number of true predicted positives over number of all predicted instances; and recall is defined as the number of true predicted positives to the number of all positives in the gold-standard annotation. (By this choice of evaluation score we avoided trivial specificity measure due to very large number of both predicted and true negatives.) We also allow a little leeway in the prediction of the motif location – a predicted hit falling within a tolerance window of size 5bp on either side of the actual starting location of the motif is also counted as a correct hit. When an algorithm fails to make any predictions, both precision and recall are taken to be zero. F1 score in such cases is also taken to be zero. For simulation-based evaluation, since the ground-truth of motif locations is known in all taxa, the numbers of true and false predictions are counted over motif instances in all taxa. For each experiment, we report summary statistics of performance scores over all 50 alignments for each algorithm.

Chapter 9

Appendix C: Details on the DISCOVER model and algorithm

C1.Formal definitions of some features

Sequence Conservation

This feature captures the degree of conservation of a potential motif binding site i given the position weight matrix of the motif, $\theta^{(m)}$. The feature function is defined as:

$$\begin{aligned} f_{SC}^{(m)}(y_i, \mathbf{x}) &= f_{SC}^{(m)}(y_i, x_{i:i+l^{(m)}-1}) \\ &= \delta(y_i, \mathrm{M}^{(m)}) \sum_{j=1}^{l^{(m)}} \beta(\theta_j^{(m)}, x_{i+j-1}) \end{aligned} \tag{9.1}$$

$$\beta(\theta_j^{(m)}, k) = \log \theta_{jk}^{(m)} - \log \theta_{0k}; \tag{9.2}$$

where $\theta^{(m)} = \{\theta_{jk}^{(m)} : j = 1, ..., l^{(m)}, k \in \{\mathrm{A,C,G,T}\}\}$ is the PWM of motif type m, $l^{(m)}$ is the length of the motif, and $\theta_0 = \{\theta_{0k} : k \in \{\mathrm{A,C,G,T}\}\}$ is the nucleotide frequency in background. The δ function equals 1 when y_i is assigned to state $\mathrm{M}^{(m)}$ and 0 otherwise.

GC-Content

A high percentage of nucleotide *guanine* (G) and *cytosine* (C) may indicate a region containing regulatory elements. The feature function is defined as:

$$f_{GC}(y_i, \mathbf{x}) = \delta(y_i, \mathrm{M}) \left(p(x_{i-w/2:i+w/2}) - p_0 \right) \tag{9.3}$$

$$p(x_{left:right}) = \frac{1}{right - left + 1} \sum_{i=left}^{right} \Big(\delta(x_i, \mathrm{G}) + \delta(x_i, \mathrm{C}) \Big) \tag{9.4}$$

where w is the window size, p is the GC-percentage inside the window whose value lies in [0,1], and p_0 is the average GC-percentage over the dataset. The $\delta(y_i, \mathrm{M})$ function equals 1 when y_i is assigned to any motif state and 0 otherwise.

As an example, the sum of conservation symmetry features can be computed as:

$$F_{CS}(\mathbf{y}, \mathbf{x}) = \sum_{i=1}^{L} f_{CS}(y_i, \mathbf{x}) \tag{9.5}$$

where f_{CS} is defined in Eq 9.8 and L is the length of the sequence. $F_{CS}(\mathbf{y}, \mathbf{x})$ is one of the elements in function vector $\mathbf{F}(\mathbf{y}, \mathbf{x})$ used in a CRF model in Eq 4.1.

Reverse Complementarity

This feature assesses how likely a potential binding site i is reverse complementary with itself. In other words, that is how similar the site is to its counterpart on the other genomic strand. The higher similarity may suggest a true motif. The feature function is defined as:

$$f_{RC}(y_i, \mathbf{x}) = \sum_{m} \delta(y_i, \mathbf{M}^{(m)}) \left(s(x_{i:i+l^{(m)}-1}) - s_0 \right) \tag{9.6}$$

$$s(x_{i:i+l-1}) = \frac{1}{\lfloor l/2 \rfloor} \sum_{j=1}^{\lfloor l/2 \rfloor} \delta_{pair}(x_{i+j-1}, x_{i+l-j}) \tag{9.7}$$

where s is the reverse complementary score of a potential binding site whose value lies in [0,1], s_0 is an offset value that is set at the mean, and l is the length of the motif. The $\delta(y_i, \mathbf{M}^{(m)})$ function equals 1 when y_i is the state of motif type m and 0 otherwise. The $\delta_{pair}(a, b)$ function equals 1 if and only if a and b are a Watson-Crick pair.

Conservation Symmetry

This feature captures the symmetry of the degree of sequence conservation given motif PWM within a motif binding site with respect to the center. The feature is defined as:

$$f_{CS}(y_i, \mathbf{x}) = \sum_{m} \delta(y_i, \mathbf{M}^{(m)}) \left(cs(\theta^{(m)}, x_{i:i+l^{(m)}-1}) - cs_0 \right) \tag{9.8}$$

$$\begin{aligned} cs(\theta^{(m)}, x_{i:i+l^{(m)}-1}) = & \frac{1}{\lfloor l^{(m)}/2 \rfloor} \sum_{j=1}^{\lfloor l^{(m)}/2 \rfloor} \Big| \beta(\theta_j^{(m)}, x_{i+j-1}) - \\ & \beta(\theta_{l^{(m)}+1-j}^{(m)}, x_{i+l^{(m)}-j}) \Big| \end{aligned} \tag{9.9}$$

where cs averages the conservation symmetry score over a potential binding site, cs_0 is an offset value of choice, $l^{(m)}$ is the length of the motif, and β function is the conservation score of a single base defined in Eq 9.2.

Melting Temperature

This feature provides an estimated melting temperature of sequences within a certain size of window by a formular:

$$f_{MT}(y_{i:i+w-1}, \mathbf{x}) = 64.9 + \frac{41 * (G + C - 16.4)}{A + T + G + C} \tag{9.10}$$

where w is the window size, and A, T, G and C are the counts of the four types of nucleotides within the window. We set the window size to 15, which is about the length of a long TFBS.

Distance to Transcription Start Site

Sites closer to a transcription start site are more likely to be TFBSs, so we adopt this feature to assess how close each site is to a nearest transcription start site. It is easy to understand that a distance change from 0-bp to 1k-bp makes more difference than a distance change from 10k-bp to 11k-bp though both of them are shifted by 1k-bp, so the feature score should not be linear on distance. We apply a logarithm function and a small constant to avoid logarithm going to negative infinity. The feature scores are calculated as:

$$f(z) = \log(z + 5) \tag{9.11}$$

where z is the distance in base-pair.

C2. Model Parameters

Feature weights constitute the set of model parameters. Some of them can be fixed and the others are free. More free parameters make the CRF model more complex, which might be harder to learn. As a guide line, we want to avoid redundant free parameters, since they will not make any contribution. On the other hand, parameters that are not likely to be properly learned from training data should never be included, because including them will only increase the chance of over-fitting. In this part, our main focus is on the weight of state transition features, because they account for a large portion in the whole parameter set.

In the CRF model, we assign a parameter as a weight to each of the features defined in the previous subsection. Those are the vector $\boldsymbol{\lambda}$ in Eq 4.1. However, some of them are not free parameters because of the context. In state transition, it is not allowed to reach an M state directly from a G state, since it is enforced that state M's representing TFBSs are surrounded by state C representing *cis*-Regulatory Module region. Thus, the corresponding state transition features have a weight being *-inf*, which means that the transitions will never happen in the CRF model. In practice, we set the weights to a small enough number.

For the sake of a good performance, we want to have a reasonable number of free model parameters. More free parameters will promote the expressing ability of the model, but at the same time the hardness of model learning will increase, the running time of learning algorithm will rise, and some parameters may be overfitting due to the lack of data describing the related features. In our case, the state transitions from a motif state to a motif state are rare, if they ever happened, which will make those transition features an inevitable overfit if we set them free. Our solution is

banning the transition between motif states and setting the matching weights to $-inf$. As a result, the number of all possible state transitions reduces dramatically.

A close look at the remaining set of state transitions will reveal redundancy. Assuming that no CRM region is on the edge, the sequence of hidden states will start with a global background state and end with a global background state. In that case, the number of transition from state G to state C will be exactly the same as the number of transition from state C to state G along the sequence of states. The models are identical to each other as long as the sum of the weight of transition feature G-C and the weight of transition feature C-G is a constant, given all the other parameters unchanged. Only one of the two weights need be a free parameter, leaving the other one to be fixed at any finite value. For simplicity, we set the weight of C-G to zero. Similar situations happen to the pair of state transition C-M$^{(m)}$ and M$^{(m)}$-C, so we fix the weight of M$^{(m)}$-C at zero.

The free parameters of state transition features left so far are G-G, C-C, G-C and C-M's. The number of state transitions along the sequence is unchanging given the sequence, so there is one more degree of redundancy, a common offset within the weights of state transition features. We get rid of the common offset by fixing the weight of G-G at zero. The final free parameters of state transition features are those of C-C, G-C and C-M's.

For those free parameters, it is not a good idea to let them be totally free. A prior can be imposed on each of them, as a way to encode prior knowledge on them. This may help in the attempt to avoid over-fitting issues. For example, we can make a prior be a normal distribution of mean 0 and variance σ^2.

C3. Model training

In this section, we briefly describe the model training procedure in which feature weights of the CRF model are learned from training data. A more thorough exposition is presented in the Appendix.Firstly, a learning criterion is set up, which can be either to maximize likelihood or to maximize posterior probability. Then, it is turned into a convex optimization problem, and finally a Quasi-Newton method is applied.

Our goal in the model learning task is to learn the best setting for $\boldsymbol{\lambda}$, the weights of features in the CRF model. What we have are a group of sequences as training data with their nucleotide types $\mathbf{x}$ and state labels $\mathbf{y}$, so the value of feature functions $\mathbf{f}$ can be computed given necessary hyper-parameters.

A criterion is needed to learn the feature weights $\boldsymbol{\lambda}$ from nucleotide types $\mathbf{x}$ and state labels $\mathbf{y}$, or more precisely from feature values $\mathbf{f}$. In the CRF model, a reasonable criterion is to maximize the likelihood of $\boldsymbol{\lambda}$ with respect to $\mathbf{y}$ conditioned on $\mathbf{x}$, which equals the probability of state labels $\mathbf{y}$ given feature weights $\boldsymbol{\lambda}$ conditioned on nucleotide types $\mathbf{x}$, because the probability model itself is defined in this conditional scheme. The max likelihood estimator of $\boldsymbol{\lambda}$ can be expressed as:

$$\widehat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} L(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x})$$

$$\text{where} \qquad L(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x}) = P(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\lambda})$$

For the simplicity of notation, we just showed likelihood function in a one-training-sequence circumstance. When multiple (for example, m) training sequences are used, as we do in our

experiment, the likelihood function will be:

$$L(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x}) = P(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\lambda}) = \prod_{k=1}^{m} P(\mathbf{y}^{(k)} \mid \mathbf{x}^{(k)}, \boldsymbol{\lambda})$$

where $\mathbf{x}^{(k)}$ and $\mathbf{y}^{(k)}$ represent the vector of nucleotide types and a vector of state labels of the k-th sequence, respectively.

Getting the maximum point of a likelihood function is equivalent to getting the maximum point of a log-likelihood function $L_\lambda = \log L(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x})$, since logarithm function is monotonically increase.

$$L_\lambda = \sum_{k=1}^{m} \Big[\boldsymbol{\lambda} \cdot \mathbf{F}(\mathbf{y}^{(k)}, \mathbf{x}^{(k)}) - \log Z(\mathbf{x}^{(k)}, \boldsymbol{\lambda}) \Big]$$

We can prove that the function of L_λ is concave with respect to $\boldsymbol{\lambda}$, so it turns into a typical convex optimization problem to find the maximum point [20]. Gradient method or Newton's method can be adopted, and convergence is assured in theory. Both of them are iterative methods which first get a search direction and then find a proper step length in each iteration. The update scheme is:

$$\boldsymbol{\lambda}^{(n+1)} = \boldsymbol{\lambda}^{(n)} + t \triangle \boldsymbol{\lambda}$$

where n is the iteration round, $\triangle \boldsymbol{\lambda}$ is the search direction, and t is the step length. The search direction is set to the negative of the first derivative of log-likelihood function $-\nabla L_\lambda$ in Gradient method, and $-\nabla L_\lambda / \nabla^2 L_\lambda$ in Newton's method. The step length is determined by a Back-track Search method. The initial point $\boldsymbol{\lambda}^{(0)}$ can be picked by experience.

It can be shown that the first derivative of log-likelihood function with respect to $\boldsymbol{\lambda}$ is:

$$\nabla \boldsymbol{\lambda} = \sum_{k=1}^{m} \left\{ \mathbf{F}(\mathbf{y}^{(k)}, \mathbf{x}^{(k)}) - E\Big[\mathbf{F}(\mathbf{y}, \mathbf{x}^{(k)}) \,\Big|\, \mathbf{x}^{(k)}, \boldsymbol{\lambda} \Big] \right\}$$

The derivative is tractable, because the conditional expectation of feature sums $\mathbf{F}(\mathbf{y}, \mathbf{x}^{(k)})$ given genomic sequence $\mathbf{x}^{(k)}$ and feature weights $\boldsymbol{\lambda}$ is computational feasible.

In practice, however, gradient method is likely to converge slowly, and the second derivative term in Newton's method is hard to compute efficiently. A Quasi-Newton method [3] is more practical, in which an approximation is applied to the inverse of the second derivative of log-likelihood with respect to feature weights $\boldsymbol{\lambda}$ and the rest parts are the same as Newton's method. More specifically we use BFGS approximation method.

Besides choosing the likelihood of $\boldsymbol{\lambda}$ as the target function to maximize, we can instead use the posterior probability:

$$P(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x}) = \frac{P(\boldsymbol{\lambda}, \mathbf{y}, \mathbf{x})}{P(\mathbf{y}, \mathbf{x})} = \frac{P(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\lambda})\, P(\mathbf{x} \mid \boldsymbol{\lambda})\, P(\boldsymbol{\lambda})}{P(\mathbf{y}, \mathbf{x})}$$

As long as feature weights are independent of genomic sequences, $P(\mathbf{x} \mid \boldsymbol{\lambda}) = P(\mathbf{x})$, which is constant. So,

$$P(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x}) \propto P(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\lambda})\, P(\boldsymbol{\lambda})$$

The full version of posterior probability for multiple (m) training sequences is:

$$P(\boldsymbol{\lambda} \mid \mathbf{y}, \mathbf{x}) \propto P(\boldsymbol{\lambda}) \prod_{k=1}^{m} P(\mathbf{y}^{(k)} \mid \mathbf{x}^{(k)}, \boldsymbol{\lambda})$$

assuming state labels of different sequences $\mathbf{y}^{(k)}$ are independent of each other given $\mathbf{x}^{(k)}$ and $\mathbf{y}^{(k)}$ is independent of $\mathbf{x}^{(j)}$ given $\mathbf{x}^{(k)}$ when $j \neq k$.

The new target function is concave, as long as the prior distribution function of $\boldsymbol{\lambda}$ is log-concave. We keep using L_λ to represent the logarithm of posterior probability. As an example, the full version for multiple (m) training sequences is:

$$L_\lambda = \sum_{k=1}^{m} \left[\boldsymbol{\lambda} \cdot \mathbf{F}(\mathbf{y}^{(k)}, \mathbf{x}^{(k)}) - \log Z(\mathbf{x}^{(k)}, \boldsymbol{\lambda}) \right] - \frac{\boldsymbol{\lambda} \cdot \boldsymbol{\lambda}}{2\sigma^2} + C$$

if each λ follows a $\mathcal{N}(0, \sigma^2)$ as a prior. C is a constant in the equation. The equation has a similar form to a regularized log-likelihood.

C4. Facets of present work and scope of future work

We have proposed a new method based on *Conditional Random Fields* for transcription factor binding site prediction in genomic sequences. Our approach takes advantage of the CRF models, which can overcome label bias problems that often happen in HMM models. The CRF model is a discriminative method that is based on a set of feature designs. The flexible forms of feature designs make it possible and easier to encode current knowledge in the field as well as to incorporate new information on TFBS when they are available. For example, we have made use of the knowledge about *cis*-Regulatory Module architecture as well as the abundance level of *guanine* and *cytosine* in nearby region in our predictor for TFBS. A feature weight, a parameter in the CRF model, determines the degree to which the feature influences the probability model. Priors can be put on the parameters, as long as they do not break the concavity of the target function. The concavity (or convexity) is such a good characteristic that we no longer need to worry about the annoying local maximum (or minimum) issues in iterative methods, and convergence is guaranteed theoretically. As expected, our method outperforms window-based methods and HMM-based methods in the experiment.

The CRF model also allow us to put together more than necessary features, because the feature weights that we got from the learning step will decide whether they are in use or not in the final model. However, as for now, the limited data size we got may prevent us from learning out the actual value of some under-represented features, and may result in severe over-fitting if we introduce too many features at a time. On the other hand, the iterative methods in the learning step may have a higher difficulty in convergence as more and more free feature parameters are added into the model, because an approximation is being used. Sometimes, singularity may occur in the approximation to the Hessian matrix [1]. In such case, we used the identity matrix to replace it,

[1]The second derivative matrix of target function, log-likelihood or log-posterior-probability, with respect to the variable vector, $\boldsymbol{\lambda}$.

which is the same as its initial setting. The analysis and improving of convergence speed regarding various free parameter set could be a future work.

As for now, our feature functions are limited to containing only neighboring hidden states. More variety of features, such as long distance features between two hidden states that are away from each other and features involving more than two hidden states, are desired when trying to encode some knowledge. For example, we will need long distance features to encode motif co-occurrence, some other kind to directly describe motif spacing and CRM length, etc. However, complex feature functions could make the algorithms currently used in the learning step invalid, therefore alternative algorithms need be studied. There is a (hidden) trade-off between the express power of feature functions and the efficiency of learning. This will be one of the future directions to work on.

It is noticeable that an offset is presented in Eq (9.3) (9.6) (9.8), which tries to move the mean value of a feature to 0. The motivation is trying to minimize the impact of adding/removing the feature to other weights. It is helpful in practice.

A special prediction scheme, rank decoding, is used here. We control the number of positive predictions made rather than a common threshold for probability values. This can strike a good balance between sequences, because longer sequences tend to fit into a model worse when it is different from the (unknown) real model. On the other hand, this scheme is reasonable in the sense of working load when we want to verify the predictions in biology experiments. Sequence decoding, another prediction scheme, does not work at most time, which barely output positive predictions, because of the modeling error accumulated along the long sequence. MAP decoding may sometimes work well.

Bibliography

[1] W. B. Alkema, O. Johansson, J. Lagergren, and W. W. Wasserman. Mscan: identification of functional clusters of transcription factor binding sites. *Nucleic Acids Res*, 32(Web Server issue):W195–8, 2004.

[2] W. Arber and S. Linn. DNA modification and restriction. *Annu. Rev. Biochem.*, 38:467–500, 1969.

[3] M. Avriel. *Nonlinear Programming: Analysis and Methods*. Dover Publishing, 2003.

[4] T. L. Bailey. DREME: motif discovery in transcription factor ChIP-seq data. *Bioinformatics*, 27(12):1653–1659, Jun 2011.

[5] T. L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc Int Conf Intell Syst Mol Biol*, 2:28–36, 1994.

[6] T. L. Bailey and P. Machanick. Inferring direct DNA binding from ChIP-seq. *Nucleic Acids Res.*, 40(17):e128, Sep 2012.

[7] T. L. Bailey, N. Williams, C. Misleh, and W. W. Li. MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Res*, 34(Web Server issue):369–373, Jul 2006.

[8] Y. Barash, G. Elidan, N. Friedman, and T. Kaplan. Modeling dependencies in protein-dna binding sites. In *Proceedings of the Seventh International Conference on Research in Computational Molecular Biology*, Berlin, Germany, 2003.

[9] M. J. Beal, Z. Ghahramani, and C. E. Rasmussen. The infinite hidden Markov model. In *Advances in Neural Information Processing Systems 13*, 2001.

[10] P. V. Benos, D. L. Corcoran, and E. Feingold. Web-Based Identification of Evolutionary Conserved DNA cis-Regulatory Elements. *Methods Mol Biol*, 395:425–436, 2007.

[11] O. G. Berg and P. H. von Hippel. Selection of DNA binding sites by regulatory proteins. Statistical-mechanical theory and application to operators and promoters. *J. Mol. Biol.*, 193(4):723–750, Feb 1987.

[12] C. Bergman. Drosophila DNase I footprint database: a systematic genome annotation of transcription factor binding sites in the fruitfly, Drosophila melanogaster. *Bioinformatics*, 21(8):1747–1749, 2005.

[13] C. M. Bergman and S. E. Carlson, Joseph Wand Celniker. Drosophila dnase i footprint database: a systematic genome annotation of transcription factor binding sites in the fruitfly,

drosophila melanogaster. *Bioinformatics*, 21(8):1747–1749, 2005.

[14] M. Blanchette, W. Kent, C. Riemer, L. Elnitski, A. Smit, K. Roskin, R. Baertsch, K. Rosenbloom, H. Clawson, E. Green, D. Haussler, and W. Miller. Aligning multiple genomic sequences with the threaded blocksand etaligner. *Genome Res.*, 14:708–715, Apr 2004.

[15] M. Blanchette, S. Kwong, and M. Tompa. An empirical comparison of tools for phylogenetic footprinting. In *BIBE '03: Proceedings of the 3rd IEEE Symposium on BioInformatics and BioEngineering*, page 69, Washington, DC, USA, 2003. IEEE Computer Society.

[16] M. Blanchette and M. Tompa. FootPrinter: A program designed for phylogenetic footprinting. *Nucleic Acids Res*, 31(13):3840–3842, Jul 2003.

[17] J. Bockhurst and M. Craven. Markov networks for detecting overlapping elements in sequence data. *Proc of Advances in Neural Information Processing Systems*, 17:193–200, 2005.

[18] D. Boffelli, J. McAuliffe, D. Ovcharenko, K. D. Lewis, I. Ovcharenko, L. Pachter, and E. M. Rubin. Phylogenetic shadowing of primate sequences to find functional regions of the human genome. *Science*, 299(5611):1391–1394, February 2003.

[19] C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller. Context-specific independence in Bayesian networks. In *Uncertainty in Artificial Intelligence*, pages 115–123, 1996.

[20] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[21] R. Britten. Evolutionary selection against change in many Alu repeat sequences interspersed through primate genomes. *Proc. Natl. Acad. Sci. U.S.A.*, 91:5992–5996, Jun 1994.

[22] H. J. Bussemaker, H. Li, and E. D. Siggia. Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. *Proc. Natl. Acad. Sci. U.S.A.*, 97:10096–10100, Aug 2000.

[23] J. Carroll, X. Liu, A. Brodsky, W. Li, C. Meyer, A. Szary, J. Eeckhoute, W. Shao, E. Hestermann, T. Geistlinger, E. Fox, P. Silver, and M. Brown. Chromosome-wide mapping of estrogen receptor binding reveals long-range regulation requiring the forkhead protein FoxA1. *Cell*, 122:33–43, Jul 2005.

[24] D. R. Cavener. Comparison of the consensus sequence flanking translational start sites in Drosophila and vertebrates. *Nucleic Acids Res.*, 15(4):1353–1361, Feb 1987.

[25] T. W. Chang. Binding of cells to matrixes of distinct antibodies coated on solid surface. *J. Immunol. Methods*, 65(1-2):217–223, Dec 1983.

[26] X. Chen, X. Hu, T. Y. Lim, X. Shen, E. K. Park, and G. L. Rosen. Exploiting the functional and taxonomic structure of genomic data by probabilistic topic modeling. *IEEE/ACM Trans Comput Biol Bioinform*, 9(4):980–991, 2012.

[27] Y. Chen, N. Negre, Q. Li, J. O. Mieczkowska, M. Slattery, T. Liu, Y. Zhang, T. K. Kim, H. H. He, J. Zieba, Y. Ruan, P. J. Bickel, R. M. Myers, B. J. Wold, K. P. White, J. D. Lieb, and X. S. Liu. Systematic evaluation of factors influencing ChIP-seq fidelity. *Nat. Methods*, 9(6):609–614, Jun 2012.

[28] C. Cheng, R. Alexander, R. Min, J. Leng, K. Y. Yip, J. Rozowsky, K. K. Yan, X. Dong, S. Djebali, Y. Ruan, C. A. Davis, P. Carninci, T. Lassman, T. R. Gingeras, R. Guigo, E. Birney, Z. Weng, M. Snyder, and M. Gerstein. Understanding transcriptional regulation by integrative analysis of transcription factor binding data. *Genome Res.*, 22(9):1658–1667, Sep 2012.

[29] G. M. Church. Genomes for all. *Sci. Am.*, 294(1):46–54, Jan 2006.

[30] P. Collas and J. A. Dahl. Chop it, ChIP it, check it: the current status of chromatin immunoprecipitation. *Front. Biosci.*, 13:929–943, 2008.

[31] R. G. Cowell, S. L. Lauritzen, and D. J. Spiegelhater. *Probabilistic Networks and Expert Systems*. Springer, 2005.

[32] G. E. Crooks, G. Hon, J.-M. Chandonia, and S. E. Brenner. WebLogo: A Sequence Logo Generator. *Genome Res.*, 14(6):1188–1190, 2004.

[33] G. Cuellar-Partida, F. A. Buske, R. C. McLeay, T. Whitington, W. S. Noble, and T. L. Bailey. Epigenetic priors for identifying active transcription factor binding sites. *Bioinformatics*, 28(1):56–62, Jan 2012.

[34] R. Das, N. Dimitrova, Z. Xuan, R. A. Rollins, F. Haghighi, J. R. Edwards, J. Ju, T. H. Bestor, and M. Q. Zhang. Computational prediction of methylation status in human genomic sequences. *Proc. Natl. Acad. Sci. U.S.A.*, 103(28):10713–10716, Jul 2006.

[35] E. H. Davidson. *Genomic Regulatory Systems: Development and Evolution*. Academic Press, San Diego, CA, 2001.

[36] M. M. Davis, S. K. Kim, and L. E. Hood. DNA sequences mediating class switching in alpha-immunoglobulins. *Science*, 209(4463):1360–1365, Sep 1980.

[37] D. DeCaprio, J. Vinson, M. Pearson, P. Montgomery, M. Doherty, and J. Galagan. Conrad: gene prediction using conditional random fields. *Genome Res.*, 17:1389–1398, Sep 2007.

[38] M. Defrance and H. Touzet. Predicting transcription factor binding sites using local over-representation and comparative genomics. *BMC Bioinformatics*, 7:396, 2006.

[39] J. Dekker, K. Rippe, M. Dekker, and N. Kleckner. Capturing chromosome conformation. *Science*, 295(5558):1306–1311, Feb 2002.

[40] R. C. Dickson, J. Abelson, W. M. Barnes, and W. S. Reznikoff. Genetic regulation: the Lac control region. *Science*, 187(4171):27–35, Jan 1975.

[41] I. J. Donaldson, M. Chapman, and B. Gottgens. Tfbscluster: a resource for the characterization of transcriptional regulatory networks. *Bioinformatics*, 21(13):3058–3059, 2005.

[42] I. Dubchak and D. V. Ryaboy. VISTA family of computational tools for comparative analysis of DNA sequences and whole genomes. *Methods Mol Biol*, 338:69–89, 2006.

[43] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1988.

[44] H. Echols, D. Court, and L. Green. On the nature of cis-acting regulatory proteins and genetic organization in bacteriophage: the example of gene Q of bacteriophage lambda.

Genetics, 83(1):5–10, May 1976.

[45] L. Elnitski, V. Jin, P. Farnham, and S. Jones. Locating mammalian transcription factor binding sites: a survey of computational and experimental techniques. *Genome Res.*, 16:1455–1464, Dec 2006.

[46] B. E. Engelhardt, M. I. Jordan, and S. E. Brenner. A graphical model for predicting protein molecular function. In W. W. Cohen and A. Moore, editors, *Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, pages 297–304. ACM, 2006.

[47] J. Felsenstein. Evolutionary trees from dna sequences: a maximum likelihood approach. *J Mol Evol*, 17(6):368–76, 1981.

[48] J. Felsenstein. *Inferring Phylogenies.* Sinauer, 2001.

[49] J. Felsenstein and G. A. Churchill. A Hidden Markov Model approach to variation among sites in rate of evolution. *Mol Biol Evol*, 13(1):93–104, 1996.

[50] J. D. Ferguson. Variable duration models for speech. *Proc. of the Symposium on the Application of HMM to Text and Speech*, pages 143–179, 1980.

[51] S. Fine, Y. Singer, and N. Tishby. The hierarchical hidden Markov model: Analysis and applications. *Machine Learning*, 32:41–62, 1998.

[52] FlybaseConsortium. The FlyBase Database of the Drosophila Genome Projects and community literature. *Nucleic Acids Research*, 87(1):85–88.

[53] E. Fratkin, B. T. Naughton, D. L. Brutlag, and S. Batzoglou. MotifCut: regulatory motifs finding with maximum density subgraphs. *Bioinformatics*, 22:e150–157, Jul 2006.

[54] M. C. Frith, U. Hansen, and Z. Weng. Detection of cis-element clusters in higher eukaryotic DNA. *Bioinformatics*, 17:878–889, 2001.

[55] M. C. Frith, M. C. Li, and Z. Weng. Cluster-buster: Finding dense clusters of motifs in dna sequences. *Nucleic Acids Res*, 31(13):3666–3668, 2003.

[56] M. C. Frith, J. L. Spouge, U. Hansen, and Z. Weng. Statistical significance of clusters of motifs represented by position specific scoring matrices in nucleotide sequences. *Nucleic Acids Research*, 30(14):3214–3224, 2002.

[57] W. Fu, , P. Ray, and E. P. Xing. Discover: A feature-based discriminative method for motif search in complex genomes. In *Proceedings of the 16th International Conference on Intelligent Systems for Molecular Biology*, 2009.

[58] M. J. Fullwood and Y. Ruan. ChIP-based methods for the identification of long-range chromatin interactions. *J. Cell. Biochem.*, 107(1):30–39, May 2009.

[59] M. J. Fullwood, C. L. Wei, E. T. Liu, and Y. Ruan. Next-generation DNA sequencing of paired-end tags (PET) for transcriptome and genome analyses. *Genome Res.*, 19(4):521–532, Apr 2009.

[60] D. J. Galas and A. Schmitz. DNAse footprinting: a simple method for the detection of protein-DNA binding specificity. *Nucleic Acids Res.*, 5(9):3157–3170, Sep 1978.

[61] S. M. Gallo, L. Li, Z. Hu, and M. S. Halfon. Redfly: a regulatory element database for drosophila. *Bioinformatics*, 22(3):381–383, 2006.

[62] S. M. Gallo, L. Li, Z. Hu, and M. S. Halfon. Redfly: a regulatory element database for drosophila. *Bioinformatics*, 22(3):381–383, 2006.

[63] J. V. Geisberg and K. Struhl. Quantitative sequential chromatin immunoprecipitation, a method for analyzing co-occupancy of proteins at genomic regions in vivo. *Nucleic Acids Res.*, 32(19):e151, 2004.

[64] Z. Ghahramani and M. Beal. Propagation algorithms for variational Bayesian learning. In *Advances in Neural Information Processing Systems 13*, 2001.

[65] W. Gilbert and A. Maxam. The nucleotide sequence of the lac operator. *Proc. Natl. Acad. Sci. U.S.A.*, 70(12):3581–3584, Dec 1973.

[66] G. D. Gilfillan, T. Hughes, Y. Sheng, H. S. Hjorthaug, T. Straub, K. Gervin, J. R. Harris, D. E. Undlien, and R. Lyle. Limitations and possibilities of low cell number ChIP-seq. *BMC Genomics*, 13:645, 2012.

[67] D. S. Gilmour and J. T. Lis. Detecting protein-DNA interactions in vivo: distribution of RNA polymerase on specific bacterial genes. *Proc. Natl. Acad. Sci. U.S.A.*, 81(14):4275–4279, Jul 1984.

[68] D. S. Gilmour and J. T. Lis. In vivo interactions of RNA polymerase II with genes of Drosophila melanogaster. *Mol. Cell. Biol.*, 5(8):2009–2018, Aug 1985.

[69] D. V. Goeddel, D. G. Yansura, and M. H. Caruthers. Binding of synthetic lactose operator DNAs to lactose represessors. *Proc. Natl. Acad. Sci. U.S.A.*, 74(8):3292–3296, Aug 1977.

[70] C. E. Grant, T. L. Bailey, and W. S. Noble. FIMO: scanning for occurrences of a given motif. *Bioinformatics*, 27(7):1017–1018, Apr 2011.

[71] S. Gros, C. Dc, M. Sirota, and S. Batzoglou. CONTRAST: a discriminative, phylogeny-free approach to multiple informant de novo gene prediction. *Genome Biol.*, 8(12):R269, 2007.

[72] Y. Guan, M. J. Dunham, and O. G. Troyanskaya. Functional analysis of gene duplications in Saccharomyces cerevisiae. *Genetics*, 175:933–943, Feb 2007.

[73] M. Gupta and J. S. Liu. Discovery of conserved sequence patterns using a stochastic dictionary model. *Journal of the American Statistical Association*, 98(461):55–66, 2003.

[74] M. Gupta and J. S. Liu. De novo cis-regulatory module elicitation for eukaryotic genomes. *Proc Natl Acad Sci U S A*, 102(20):7079–7084, 2005.

[75] M. W. Hahn, M. V. Rockman, N. Soranzo, D. B. Goldstein, and G. A. Wray. Population genetic and phylogenetic evidence for positive selection on regulatory mutations at the factor vii locus in humans. *Genetics*, 167(2):867–77, 2004.

[76] N. Hall. Advanced sequencing technologies and their wider impact in microbiology. *J. Exp. Biol.*, 210(Pt 9):1518–1525, May 2007.

[77] M. Hasegawa, H. Kishino, and T. Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J Mol Evol*, 22(2):160–74, 1985.

[78] N. D. Heintzman, G. C. Hon, R. D. Hawkins, P. Kheradpour, A. Stark, L. F. Harp, Z. Ye, L. K. Lee, R. K. Stuart, C. W. Ching, K. A. Ching, J. E. Antosiewicz-Bourget, H. Liu, X. Zhang, R. D. Green, V. V. Lobanenkov, R. Stewart, J. A. Thomson, G. E. Crawford, M. Kellis, and B. Ren. Histone modifications at human enhancers reflect global cell-type-specific gene expression. *Nature*, 459(7243):108–112, May 2009.

[79] N. D. Heintzman, R. K. Stuart, G. Hon, Y. Fu, C. W. Ching, R. D. Hawkins, L. O. Barrera, S. Van Calcar, C. Qu, K. A. Ching, W. Wang, Z. Weng, R. D. Green, G. E. Crawford, and B. Ren. Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome. *Nat. Genet.*, 39(3):311–318, Mar 2007.

[80] A. S. Hinrichs and etal. The ucsc genome browser database: update 2006. *Nucleic Acids Res.*, 34:D590–8, 2006.

[81] H. Huang, M. Kao, X. Zhou, J. S. Liu, and W. H. Wong. Determination of local statistical significance of patterns in Markov sequences with application to promoter element identification. *Journal of Computational Biology*, 11 (1), 2004.

[82] W. Huang, J. R. Nevins, and U. Ohler. Phylogenetic Simulation of Promoter Evolution: Estimation and Modeling of Binding Site Turnover Events and Assessing Their Impact on Alignment Tools. *Genome Biol*, 8(10):R225, 2007.

[83] V. R. Iyer, C. E. Horak, C. S. Scafe, D. Botstein, M. Snyder, and P. O. Brown. Genomic binding sites of the yeast cell-cycle transcription factors SBF and MBF. *Nature*, 409:533–538, Jan 2001.

[84] O. Johansson, W. Alkema, W. W. Wasserman, and J. Lagergren. Identification of functional clusters of transcription factor binding motifs in genome sequences: the mscan algorithm. *Bioinformatics*, 19 Suppl 1:i169–76, 2003.

[85] D. S. Johnson, A. Mortazavi, R. M. Myers, and B. Wold. Genome-wide mapping of in vivo protein-DNA interactions. *Science*, 316:1497–1502, Jun 2007.

[86] T. H. Jukes and C. R. Cantor. Evolution of protein molecules. In H. N. Munro, editor, *Mammalian Protein Metabolism*, pages 21–32. Academic Press, New York, 1969.

[87] J. Jurka, V. Kapitonov, A. Pavlicek, P. Klonowski, O. Kohany, and J. Walichiewicz. Repbase Update, a database of eukaryotic repetitive elements. *Cytogenet. Genome Res.*, 110:462–467, 2005.

[88] M. Kamal, X. Xie, and E. Lander. A large family of ancient repeat elements in the human genome is under strong selection. *Proc. Natl. Acad. Sci. U.S.A.*, 103:2740–2745, Feb 2006.

[89] M. R. Kantorovitz, G. E. Robinson, and S. Sinha. A statistical method for alignment-free comparison of regulatory sequences. *Bioinformatics*, 23(13):i249–55, 2007.

[90] T. J. Kelly and H. O. Smith. A restriction enzyme from Hemophilus influenzae. II. *J. Mol. Biol.*, 51(2):393–409, Jul 1970.

[91] W. J. Kent. BLAT–the BLAST-like alignment tool. *Genome Res*, 12(4):656–664, Apr 2002.

[92] S. Kim and J. Pritchard. Adaptive evolution of conserved noncoding elements in mammals. *PLoS Genet.*, 3:1572–1586, Sep 2007.

[93] M. Kimura. The neutral theory of molecular evolution: a review of recent evidence. *Jpn J Genet*, 66(4):367–86, 1991.

[94] M. Kozak. An analysis of 5'-noncoding sequences from 699 vertebrate messenger RNAs. *Nucleic Acids Res.*, 15(20):8125–8148, Oct 1987.

[95] U. Kuhnlein, S. Linn, and W. Arber. Host specificity of DNA produced by Escherichia coli. XI. In vitro modification of phage fd replicative form. *Proc. Natl. Acad. Sci. U.S.A.*, 63(2):556–562, Jun 1969.

[96] J. Lafferty, F. Pereira, and A. McCallum. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning*, 2001.

[97] T. S. Larsen and A. Krogh. Easygene–a prokaryotic gene finder that ranks orfs by statistical significance. *BMC Bioinformatics*, 4:21, Jun 2003.

[98] T. Lee, , and etal. Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 5594(298):799–804, 2002.

[99] C. Leslie, E. Eskin, and W. Noble. The spectrum kernel: a string kernel for svm protein classification. In *Pac Symp Biocomput.*, pages 564–75, 2002.

[100] H. A. Levine and M. Nilsen-Hamilton. A mathematical analysis of SELEX. *Comput Biol Chem*, 31(1):11–35, Feb 2007.

[101] S. E. Levinson. Continuously variable duration hidden markov models for automatic speech recognition. *Comput. Speech Lang.*, 1(1):29–45, 1986.

[102] H. Li and W. Stephan. Inferring the demographic history and rate of adaptive substitution in Drosophila. *PLoS Genet.*, 2:e166, Oct 2006.

[103] K. Liang and S. Keles. Normalization of ChIP-seq data with control. *BMC Bioinformatics*, 13:199, 2012.

[104] C. R. Lickwar, F. Mueller, S. E. Hanlon, J. G. McNally, and J. D. Lieb. Genome-wide protein-DNA binding dynamics suggest a molecular clutch for transcription factor function. *Nature*, 484(7393):251–255, Apr 2012.

[105] J. D. Lieb, X. Liu, D. Botstein, and P. O. Brown. Promoter-specific binding of Rap1 revealed by genome-wide maps of protein-DNA association. *Nat. Genet.*, 28:327–334, Aug 2001.

[106] S. Lin and A. D. Riggs. The general affinity of lac repressor for E. coli DNA: implications for gene regulation in procaryotes and eucaryotes. *Cell*, 4(2):107–111, Feb 1975.

[107] T.-H. Lin, P. Ray, G. K. Sandve, S. Uguroglu, and E. P. Xing. Baycis: a bayesian hierarchical hmm for cis-regulatory module decoding in metazoan genomes. In *Proceedings of RECOMB 2008*, 2008.

[108] B. Liu, L. Liu, A. Tsykin, G. J. Goodall, J. E. Green, M. Zhu, C. H. Kim, and J. Li. Identifying functional miRNA-mRNA regulatory modules with correspondence latent dirichlet allocation. *Bioinformatics*, 26(24):3105–3111, Dec 2010.

[109] X. Liu, D. L. Brutlag, and J. Liu. Bioprospector: Discovering conserved DNA motifs in

upstream regulatory regions of co-expressed genes. In *Proc. of Pac Symp Biocomput*, pages 127–138, 2001.

[110] G. Locke, D. Tolkunov, Z. Moqtaderi, K. Struhl, and A. V. Morozov. High-throughput sequencing reveals a simple model of nucleosome energetics. *Proc. Natl. Acad. Sci. U.S.A.*, 107(49):20998–21003, Dec 2010.

[111] G. G. Loots, I. Ovcharenko, L. Pachter, I. Dubchak, and E. M. Rubin. rVista for comparative sequence-based discovery of functional transcription factor binding sites. *Genome Res*, 12 (5):832–839, 2002.

[112] M. Z. Ludwig, C. M. Bergman, N. H. Patel, and M. Kreitman. Evidence for stabilizing selection in a eukaryotic enhancer element. *Nature*, 403:564–567, 2000.

[113] M. Z. Ludwig, A. Palsson, E. Alekseeva, C. M. Bergman, J. Nathan, and M. Kreitman. Functional evolution of a cis-regulatory module. *PLoS Biol*, 3(4):e93, Apr 2005.

[114] M. Z. Ludwig, N. H. Patel, and M. Kreitman. Functional analysis of eve stripe 2 enhancer evolution in Drosophila: rules governing conservation and change. *Development*, 125(5):949–958, Mar 1998.

[115] X. Ma, A. Kulkarni, Z. Zhang, Z. Xuan, R. Serfling, and M. Q. Zhang. A highly efficient and effective motif discovery method for ChIP-seq/ChIP-chip data using positional information. *Nucleic Acids Res.*, 40(7):e50, Apr 2012.

[116] P. Machanick and T. L. Bailey. MEME-ChIP: motif analysis of large DNA datasets. *Bioinformatics*, 27(12):1696–1697, Jun 2011.

[117] N. M. Maizels. The nucleotide sequence of the lactose messenger ribonucleic acid transcribed from the UV5 promoter mutant of Escherichia coli. *Proc. Natl. Acad. Sci. U.S.A.*, 70(12):3585–3589, Dec 1973.

[118] T. Maniatis, A. Jeffrey, and D. G. Kleid. Nucleotide sequence of the rightward operator of phage lambda. *Proc. Natl. Acad. Sci. U.S.A.*, 72(3):1184–1188, Mar 1975.

[119] T. Maniatis, M. Ptashne, K. Backman, D. Kield, S. Flashman, A. Jeffrey, and R. Maurer. Recognition sequences of repressor and polymerase in the operators of bacteriophage lambda. *Cell*, 5(2):109–113, Jun 1975.

[120] T. Maniatis, M. Ptashne, B. G. Barrell, and J. Donelson. Sequence of a repressor-binding site in the DNA of bacteriophage lamda. *Nature*, 250(465):394–397, Aug 1974.

[121] E. H. Margulies, M. Blanchette, , D. Haussler, and E. D. Green. Identification and characterization of multi-species conserved sequences. *Genome research*, 13(12):2507–2518, 2003.

[122] A. M. Maxam and W. Gilbert. A new method for sequencing DNA. *Proc. Natl. Acad. Sci. U.S.A.*, 74(2):560–564, Feb 1977.

[123] J. D. McAuliffe, L. Pachter, and M. Jordan. Multiple-sequence functional annotation and the generalized hidden markov phylogeny. *Bioinformatics*, 20:1850–1860, 2004.

[124] S. B. Montgomery and etal. Oreganno: an open access database and curation system

for literature-derived promoters, transcription factor binding sites and regulatory variation. *Bioinformatics*, 22(5):637–640, 2006.

[125] A. Mortazavi, B. A. Williams, K. McCue, L. Schaeffer, and B. Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Methods*, 5(7):621–628, Jul 2008.

[126] A. M. Moses. Statistical tests for natural selection on regulatory regions based on the strength of transcription factor binding sites. *BMC Evol. Biol.*, 9:286, 2009.

[127] A. M. Moses, D. Y. Chiang, and M. B. Eisen. Phylogenetic motif detection by expectation-maximization on evolutionary mixtures. *Pac Symp Biocomput*, pages 324–35, 2004.

[128] A. M. Moses, D. Y. Chiang, D. A. Pollard, I. V. N., and M. B. Eisen. Monkey: Identifying conserved transcription-factor binding sites in multiple alignments using a binding site-specific evolutionary model. *Genome Biol*, 5:R98, 2004.

[129] A. M. Moses, D. A. Pollard, D. A. Nix, V. N. Iyer, X.-Y. Li, M. D. Biggin, and M. B. Eisen. Large-scale turnover of functional transcription factor binding sites in Drosophila. *PLoS Comput Biol*, 2(10):e130, 2006.

[130] K. Murphy and M. Paskin. Linear time inference in hierarchical hmms. In *Advances in Neural Information Processing Systems 14*, 2002.

[131] T. Nammo, S. A. Rodriguez-Segui, and J. Ferrer. Mapping open chromatin with formaldehyde-assisted isolation of regulatory elements. *Methods Mol. Biol.*, 791:287–296, 2011.

[132] V. Narang, W. K. Sung, and A. Mittal. Computational annotation of transcription factor binding sites in D. melanogaster developmental genes. In *Proceedings of The 17th International Conference on Genome Informatics*, 2006.

[133] V. Narang, W. K. Sung, and A. Mittal. Computational annotation of transcription factor binding sites in D. melanogaster developmental genes. In *Proceedings of The 17th International Conference on Genome Informatics*, 2006.

[134] L. Narlikar, R. Gordan, and A. J. Hartemink. A nucleosome-guided map of transcription factor binding sites in yeast. *PLoS Comput Biol*, 3(11):e215, 2007.

[135] D. B. A. Ng and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[136] A. R. Oliphant, C. J. Brandl, and K. Struhl. Defining the sequence specificity of DNA-binding proteins by selecting binding sites from random-sequence oligonucleotides: analysis of yeast GCN4 protein. *Mol. Cell. Biol.*, 9(7):2944–2949, Jul 1989.

[137] G. J. Olsen, H. Matsuda, R. Hagstrom, and R. Overbeek. fastDNAmL: a tool for construction of phylogenetic trees of DNA sequences using maximum likelihood. *Comput Appl Biosci*, 10(1):41–8, 1994.

[138] M. C. O'Neill. Symmetry, homology, and phrasing in the recognition of helical regulatory sequences in DNA. *Nucleic Acids Res.*, 4(12):4439–4463, Dec 1977.

[139] Y. Orenstein, C. Linhart, and R. Shamir. Assessment of algorithms for inferring positional

weight matrix motifs of transcription factor binding sites using protein binding microarray data. *PLoS ONE*, 7(9):e46145, 2012.

[140] Z. Ouyang, Q. Zhou, and W. H. Wong. ChIP-Seq of transcription factors predicts absolute and differential gene expression in embryonic stem cells. *Proc. Natl. Acad. Sci. U.S.A.*, 106(51):21521–21526, Dec 2009.

[141] F. Ozsolak, J. Song, X. Liu, and D. Fisher. High-throughput mapping of the chromatin structure of human promoters. *Nat. Biotechnol.*, 25:244–248, Feb 2007.

[142] D. A. Papatsenko, V. J. Makeev, A. P. Lifanov, M. Regnier, A. G. Nazina, and C. Desplan. Extraction of functional binding sites from unique regulatory regions: the Drosophila early developmental enhancers. *Genome Res*, 12(3):470–481, Mar 2002.

[143] A. A. Philippakis, A. M. Qureshi, M. F. Berger, and M. L. Bulyk. Design of compact, universal DNA microarrays for protein binding microarray experiments. *J. Comput. Biol.*, 15(7):655–665, Sep 2008.

[144] N. Polavarapu, L. Mario-Ramrez, D. Landsman, J. McDonald, and I. Jordan. Evolutionary rates and patterns for human transcription factor binding sites derived from repetitive DNA. *BMC Genomics*, 9:226, 2008.

[145] J. Ponomarenko, M. Ponomarenko, A. Frolov, D. Vorobyev, G. Overton, and N. Kolchanov. Conformational and physicochemical DNA features specific for transcription factor binding sites. *Bioinformatics*, 15:654–668, 1999.

[146] J. K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155:945–959, Jun 2000.

[147] L. R. Rabiner and B. H. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, pages 4–15, January 1986.

[148] N. Rajewsky, M. Vergassola, U. Gaul, and E. D. Siggia. Computational detection of genomic cis-regulatory modules applied to body patterning in the early drosophila embryo. *BMC bioinformatics*, 3:30, 2002.

[149] P. Ray, S. Shringarpure, M. Kolar, and E. P. Xing. Csmet: Comparative genomic motif detection via multi-resolution phylogenetic shadowing. *Public Library of Science Computational Biology*, 4(6), June 2008.

[150] P. Ray and E. Xing. Analysis of co-evolution in drosophila regulatory genome. In *Recomb Regulatory Genomics Satellite 2008*, 2008.

[151] M. Rebeiz, N. L. Reeves, and J. W. Posakony. Score: a computational approach to the identification of cis-regulatory modules and target genes in whole-genome sequence data. site clustering over random expectation. *Proc Natl Acad Sci U S A*, 99(15):9888–9893, 2002.

[152] E. Redhead and T. L. Bailey. Discriminative motif discovery in DNA and protein sequences using the DEME algorithm. *BMC Bioinformatics*, 8:385, 2007.

[153] B. Ren, F. Robert, J. J. Wyrick, O. Aparicio, E. G. Jennings, I. Simon, J. Zeitlinger, J. Schreiber, N. Hannett, E. Kanin, T. L. Volkert, C. J. Wilson, S. P. Bell, and R. A. Young.

Genome-wide location and function of DNA binding proteins. *Science*, 290:2306–2309, Dec 2000.

[154] H. S. Rhee and B. F. Pugh. Comprehensive genome-wide protein-DNA interactions detected at single-nucleotide resolution. *Cell*, 147(6):1408–1419, Dec 2011.

[155] C. S. Riesenfeld, P. D. Schloss, and J. Handelsman. Metagenomics: genomic analysis of microbial communities. *Annu. Rev. Genet.*, 38:525–552, 2004.

[156] G. Robertson, M. Hirst, M. Bainbridge, M. Bilenky, Y. Zhao, T. Zeng, G. Euskirchen, B. Bernier, R. Varhol, A. Delaney, N. Thiessen, O. L. Griffith, A. He, M. Marra, M. Snyder, and S. Jones. Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nat. Methods*, 4:651–657, Aug 2007.

[157] M. V. Rockman, M. W. Hahn, N. Soranzo, D. B. Goldstein, and G. A. Wray. Positive selection on a human-specific transcription factor binding site regulating il4 expression. *Curr Bio*, 13(23):2118–2123, 2003.

[158] M. V. Rockman, M. W. Hahn, N. Soranzo, D. A. Loisel, D. B. Goldstein, and G. A. Wray. Positive selection on mmp3 regulation has shaped heart disease risk. *Curr Bio*, 14(17):1531–1539, 2004.

[159] R. K. Saiki, S. Scharf, F. Faloona, K. B. Mullis, G. T. Horn, H. A. Erlich, and N. Arnheim. Enzymatic amplification of beta-globin genomic sequences and restriction site analysis for diagnosis of sickle cell anemia. *Science*, 230(4732):1350–1354, Dec 1985.

[160] A. Sandelin, W. W. Wasserman, and B. Lenhard. ConSite: web-based prediction of regulatory elements using cross-species comparison. *Nucleic Acids Res*, 32(Web Server issue):249–252, Jul 2004.

[161] J. R. Sanford, X. Wang, M. Mort, N. Vanduyn, D. N. Cooper, S. D. Mooney, H. J. Edenberg, and Y. Liu. Splicing factor SFRS1 recognizes a functionally diverse landscape of RNA transcripts. *Genome Res.*, 19(3):381–394, Mar 2009.

[162] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proc. Natl. Acad. Sci. U.S.A.*, 74(12):5463–5467, Dec 1977.

[163] T. D. Schneider, G. D. Stormo, L. Gold, and A. Ehrenfeucht. Information content of binding sites on nucleotide sequences. *J. Mol. Biol.*, 188(3):415–431, Apr 1986.

[164] D. E. Schones, P. Sumazin, and M. Q. Zhang. Similarity of position frequency matrices for transcription factor binding sites. *Bioinformatics*, 21(3):307–313, Feb 2005.

[165] E. Segal, Y. Fondufe-Mittendorf, L. Chen, A. Thastrom, Y. Field, Y. K. Moore, J.-P. Wang, and J. Widom. A genomic code for nucleosome positioning. *Nature*, 442:772–778, 2006.

[166] P. Sethupathy, H. Giang, J. Plotkin, and S. Hannenhalli. Genome-wide analysis of natural selection on human cis-elements. *PLoS ONE*, 3:e3137, 2008.

[167] F. Sha and F. Pereira. Shallow parsing with conditional random fields. *Proceedings of Human Language Technology-NAACL*, 1:134–141, 2003.

[168] R. Sharan, I. Ovcharenko, A. Ben-Hur, and R. M. Karp. Creme: a framework for identifying

cis-regulatory modules in human-mouse conserved segments. *Bioinformatics*, 19 Suppl 1:i283–91, 2003.

[169] Y. Shen, F. Yue, D. F. McCleary, Z. Ye, L. Edsall, S. Kuan, U. Wagner, J. Dixon, L. Lee, V. V. Lobanenkov, and B. Ren. A map of the cis-regulatory sequences in the mouse genome. *Nature*, 488(7409):116–120, Aug 2012.

[170] J. Shine and L. Dalgarno. Determinant of cistron specificity in bacterial ribosomes. *Nature*, 254(5495):34–38, Mar 1975.

[171] R. Siddharthan, E. van Nimwegen, and E. D. Siggia. Phylogibbs: A gibbs sampler incorporating phylogenetic information. In E. Eskin and C. Workman, editors, *Regulatory Genomics*, volume 3318 of *Lecture Notes in Computer Science*, pages 30–41. Springer, 2004.

[172] A. Siepel, K. S. Pollard, and D. Haussler. New methods for detecting lineage-specific selection. *Lecture Notes in Computer Science*, 3909, 2006.

[173] A. C. Siepel and D. Haussler. Combining phylogenetic and hidden Markov models in biosequence analysis. In *Proceedings of the seventh annual international conference on Research in computational molecular biology (RECOMB)*, pages 277–286, 2003.

[174] S. Sinha, M. Blanchette, and M. Tompa. PhyME: a probabilistic algorithm for finding motifs in sets of orthologous sequences. *BMC Bioinformatics*, 5:170, 2004.

[175] S. Sinha, Y. Liang, and E. Siggia. Stubb: a program for discovery and analysis of cis-regulatory modules. *Nucleic acids research*, Web Server issue:W555–9, 2006.

[176] S. Sinha, E. van Nimwegen, and E. D. Siggia. A probabilistic method to detect regulatory modules. *Bioinformatics*, 19 Suppl 1:292–301, 2003.

[177] A. D. Smith, P. Sumazin, D. Das, and M. Q. Zhang. Mining ChIP-chip data for transcription factor and cofactor binding sites. *Bioinformatics*, 21 Suppl 1:i403–412, Jun 2005.

[178] A. D. Smith, P. Sumazin, D. Das, and M. Q. Zhang. Mining ChIP-chip data for transcription factor and cofactor binding sites. *Bioinformatics*, 21 Suppl 1:i403–412, Jun 2005.

[179] H. O. Smith and K. W. Wilcox. A restriction enzyme from Hemophilus influenzae. I. Purification and general properties. *J. Mol. Biol.*, 51(2):379–391, Jul 1970.

[180] L. Song and G. E. Crawford. DNase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harb Protoc*, 2010(2):pdb.prot5384, Feb 2010.

[181] S. Sonnenburg, A. Zien, P. Philips, and G. Rtsch. POIMs: positional oligomer importance matrices–understanding support vector machine-based signal detectors. *Bioinformatics*, 24:6–14, Jul 2008.

[182] R. Staden. Computer methods to locate signals in nucleic acid sequences. *Nucleic Acids Res*, 12(1 Pt 2):505–19, 1984.

[183] R. Staden. Methods to define and locate patterns of motifs in sequences. *Comput. Appl. Biosci.*, 4(1):53–60, Mar 1988.

[184] R. Staden. Methods for calculating the probabilities of finding patterns in sequences. *Comput. Appl. Biosci.*, 5(2):89–96, Apr 1989.

[185] R. Staden. Methods for discovering novel motifs in nucleic acid sequences. *Comput. Appl. Biosci.*, 5(4):293–298, Oct 1989.

[186] R. Staden. Screening protein and nucleic acid sequences against libraries of patterns. *DNA Seq.*, 1(6):369–374, 1991.

[187] R. Staden. Staden: searching for motifs in nucleic acid sequences. *Methods Mol. Biol.*, 25:93–102, 1994.

[188] R. Staden. Staden: searching for motifs in protein sequences. *Methods Mol. Biol.*, 25:131–139, 1994.

[189] A. Stark, M. Lin, P. Kheradpour, J. Pedersen, L. Parts, J. Carlson, M. Crosby, M. Rasmussen, S. Roy, A. Deoras, J. Ruby, J. Brennecke, E. Hodges, A. Hinrichs, A. Caspi, B. Paten, S. Park, M. Han, M. Maeder, B. Polansky, B. Robson, S. Aerts, J. van Helden, B. Hassan, D. Gilbert, D. Eastman, M. Rice, M. Weir, M. Hahn, Y. Park, C. Dewey, L. Pachter, W. Kent, D. Haussler, E. Lai, D. Bartel, G. Hannon, T. Kaufman, M. Eisen, A. Clark, D. Smith, S. Celniker, W. Gelbart, and M. Kellis. Discovery of functional elements in 12 Drosophila genomes using evolutionary signatures. *Nature*, 450:219–232, Nov 2007.

[190] G. D. Stormo, T. D. Schneider, L. Gold, and A. Ehrenfeucht. Use of the 'Perceptron' algorithm to distinguish translational initiation sites in E. coli. *Nucleic Acids Res.*, 10(9):2997–3011, May 1982.

[191] E. Tanaka, T. Bailey, C. E. Grant, W. S. Noble, and U. Keich. Improved similarity scores for comparing motifs. *Bioinformatics*, 27(12):1603–1609, Jun 2011.

[192] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. De Moor, P. Rouze, and Y. Moreau. A higher-order background model improves the detection of promoter regulatory elements by gibbs sampling. *Bioinformatics*, 17(12):1113–22, 2001.

[193] W. Thompson, M. J. Palumbo, W. W. Wasserman, J. S. Liu, and C. E. Lawrence. Decoding human regulatory circuits. *Genome Res*, 14(10A):1967–1974, 2004.

[194] M. Tompa and etal. Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotechnol*, 23(1):137–44, 2005.

[195] M. Tompa and etal. Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotechnol*, 23(1):137–44, 2005.

[196] W. H. Townley-Tilson, S. A. Pendergrass, W. F. Marzluff, and M. L. Whitfield. Genome-wide analysis of mRNAs bound to the histone stem-loop binding protein. *RNA*, 12(10):1853–1867, Oct 2006.

[197] M. J. Vogel, D. Peric-Hupkes, and B. van Steensel. Detection of in vivo protein-DNA interactions using DamID in mammalian cells. *Nat Protoc*, 2(6):1467–1478, 2007.

[198] A. Walz and V. Pirrotta. Sequence of the PR promoter of phage lambda. *Nature*, 254(5496):118–121, Mar 1975.

[199] T. Wang and G. D. Stormo. Combining phylogenetic data with co-regulated genes to identify regulatory motifs. *Bioinformatics*, 19(18):2369–80, 2003.

[200] L. Ward and H. Bussemaker. Predicting functional transcription factor binding through alignment-free and affinity-based analysis of orthologous promoter sequences. *Bioinformatics*, 24:i165–171, Jul 2008.

[201] T. Whitington, M. C. Frith, J. Johnson, and T. L. Bailey. Inferring transcription factor complexes from ChIP-seq data. *Nucleic Acids Res.*, 39(15):e98, Aug 2011.

[202] E. Wingender, X. Chen, R. Hehl, H. Karas, I. Liebich, V. Matys, T. Meinhardt, M. Pruss, I. Reuter, and F. Schacherer. TRANSFAC: An integrated system for gene expression regulation. *Nucleic Acids Res.*, 28:316–319, 2000.

[203] E. Wingender, P. Dietze, H. Karas, and R. Knuppel. TRANSFAC: a database on transcription factors and their DNA binding sites. *Nucleic Acids Res*, 24(1):238–41, 1996.

[204] E. P. Xing, M. I. Jordan, R. M. Karp, and S. Russell. A hierarchical Bayesian Markovian model for motifs in biopolymer sequences. In *Advances in Neural Information Processing Systems 15*, 2003.

[205] E. P. Xing, M. I. Jordan, and S. Russell. A generalized mean field algorithm for variational inference in exponential families. In *Proceedings of the 19th Annual Conference on Uncertainty in AI*, 2003.

[206] E. P. Xing, M. I. Jordan, and S. Russell. A generalized mean field algorithm for variational inference in exponential families. In *Proceedings of the 19th Annual Conference on Uncertainty in AI*, 2003.

[207] E. P. Xing, W. Wu, M. I. Jordan, and R. M. Karp. LOGOS: a modular Bayesian model for de novo motif detection. *Proc IEEE Comput Soc Bioinform Conf*, 2:266–276, 2003.

[208] H. Xing, Y. Mo, W. Liao, and M. Q. Zhang. Genome-wide localization of protein-DNA binding and histone modification by a Bayesian change-point method with ChIP-seq data. *PLoS Comput. Biol.*, 8(7):e1002613, 2012.

[209] M. Xu, C. R. Weinberg, D. M. Umbach, and L. Li. coMOTIF: a mixture framework for identifying transcription factor and a coregulator motif in ChIP-seq data. *Bioinformatics*, 27(19):2625–2632, Oct 2011.

[210] J. Yazaki, B. D. Gregory, and J. R. Ecker. Mapping the genome landscape using tiling array technology. *Curr. Opin. Plant Biol.*, 10(5):534–542, Oct 2007.

[211] J. A. Young, J. R. Johnson, C. Benner, S. F. Yan, K. Chen, K. G. Le Roch, Y. Zhou, and E. A. Winzeler. In silico discovery of transcription regulatory elements in Plasmodium falciparum. *BMC Genomics*, 9:70, 2008.

[212] M. Zhang. Computational analyses of eukaryotic promoters. *BMC Bioinformatics*, 8 Suppl 6:S3, 2007.

[213] Q. Zhou and W. H. Wong. Cismodule: de novo discovery of cis-regulatory modules by hierarchical mixture modeling. *Proc Natl Acad Sci U S A*, 101(33):12114–12119, 2004.