BayCis: A Bayesian Hierarchical HMM for Cis-Regulatory Module Decoding in Metazoan Genomes

Tien-ho Lin[1,⋆], Pradipta Ray[1,⋆] , Geir K. Sandve[2], Selen Uguroglu[3], and Eric P. Xing[1,⋆⋆]

[1] School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
[2] Dept of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway
[3] Dept of Computer Science and Engineering, Sabanci University, Istanbul, Turkey
epxing@cs.cmu.edu

Abstract. The transcriptional regulatory sequences in metazoan genomes often consist of multiple *cis-regulatory modules* (CRMs). Each CRM contains locally enriched occurrences of binding sites (motifs) for a certain array of regulatory proteins, capable of integrating, amplifying or attenuating multiple regulatory signals via combinatorial interaction with these proteins. The architecture of CRM organizations is reminiscent of the grammatical rules underlying a natural language, and presents a particular challenge to computational motif and CRM identification in metazoan genomes. In this paper, we present BayCis, a Bayesian hierarchical HMM that attempts to capture the stochastic syntactic rules of CRM organization. Under the BayCis model, all candidate sites are evaluated based on a posterior probability measure that takes into consideration their similarity to known BSs, their contrasts against local genomic context, their first-order dependencies on upstream sequence elements, as well as priors reflecting general knowledge of CRM structure. We compare our approach to five existing methods for the discovery of CRMs, and demonstrate competitive or superior prediction results evaluated against experimentally based annotations on a comprehensive selection of *Drosophila* regulatory regions. The software, database and Supplementary Materials will be available at http://www.sailing.cs.cmu.edu/baycis.

1 Introduction

Rules determining the spatio-temporal variations of gene expression in multi-cellular organisms are believed to be encoded as "*cis*-regulatory sequences", known to account for a large portion of a metazoan genome [15]. While recent years have seen substantial progress in *in silico* prediction of protein coding sequences from metazoan genomes, our understanding of the vocabulary and rules governing *cis*-regulatory sequences is limited, and remains a major open problem.

⋆ The first two authors in the list contributed equally to the paper and should be acknowledged as co-first authors.
⋆⋆ Correspondence should be addressed.

M. Vingron and L. Wong (Eds.): RECOMB 2008, LNBI 4955, pp. 66–81, 2008.

Unlike prokaryotes or uni-cellular organisms like yeast, metazoan transcription factor binding sites (TFBS, also known as motifs) are usually neither located immediately upstream of the proximal promoter element, nor are they distributed uniformly and independently in the extended surrounding region. Instead, the distributions of these motifs exhibit apparent general principles referred to as *modular organizations* – being organized into a series of discrete regions of roughly 200-1000 bp in length, each of which controls a distinct aspect of a gene's expression pattern [3]. Each CRM consists of a locally enriched collection of motifs of certain combination and ordering, capable of integrating, amplifying, or attenuating multiple regulatory signals via combinatorial physical interaction with multiple transcriptional regulatory proteins (i.e., TFs) [2]. Furthermore, it is believed that there also exist dependencies among CRMs so that coordinations between regulatory signals can be orchestrated.

Motif models of TFBSs 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 [25]. In recent years, focus has shifted from predicting TFBSs for a single TF towards predicting CRMs comprising several TFBSs, 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 [12,16,22,23]. 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 in this paper 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 [19,21,20,4]. 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 [21,10]. Rajewsky *et al.* addressed the issue of compensating the matching scores for co-occurring weak motif sites using an updatable word frequency measure, leading to higher scores for motifs co-occurring more frequently within a given window size [1] [19].

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. 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

[1] Their algorithm also contains an important extension for unsupervised CRM prediction, where representations of novel motifs are estimated directly from input DNA sequences. However, under a modular formulation of the CRM prediction problem (cf. the LOGOS model [30]), prediction of motif instances from given representations, and estimation of motif representations from predicted instances, can be treated as two orthogonal sub-problems to be solved separately and coupled as components of a higher-level joint model with estimates exchanged in iterative fashion. In this paper, we only focus on CRM prediction given motif representations and defer implementing the fully autonomous *de novo* motif-finding program to a later paper.

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 TFBS distributions, as well as intra-CRM and inter-CRM backgrounds, have been used in several CRM discovery methods, e.g. in Cister [7], Cluster-Buster [6], CisModule [31] and EMCModule [9]. 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 [24] and Module Sampler [27].

In this paper, we present a new method, *BayCis*, which implements a Bayesian hierarchical HMM for CRM search. *BayCis* represents a step further along the direction of HMM-based CRM models. It uses a more sophisticated HMM model that is intended to capture, to a reasonable degree, the detailed syntactic structure of CRM and cis-regulatory regions containing CRMs. By combining general intra-CRM background, motif specific background surrounding motif instances, as well as specific motif-to-motif transitions, it allows couplings between motifs to be captured. We also introduced more advanced approaches to model the background, using separate inter-CRM, intra-CRM and motif-specific higher-order Markov backgrounds. Furthermore, inter-motif distances may be modeled with more flexible distributions (rather than only simple geometric distributions). Finally, as detailed in the following sections, we treat parameters of the HMM grammar as stochastic variables for which Bayesian priors are applied, instead of regarding the state-transition parameters of the HMM grammar as fixed parameters that solely rely on empirical default values or user specification like in previous methods. This technique in principle alleviates user specification of model parameters (although advanced users could choose to decide the "strength" of the priors, or define their own priors). On the computational front, we developed an efficient variational inference algorithm for posterior inference of sequence annotation and Bayesian parameter estimation. This algorithm enjoys a desirable convergence guarantee and is much more efficient than the classical Gibbs sampling methods without compromising much accuracy.

BayCis has several advantages over existing methods for CRM discovery. The explicit model of CRMs makes architectural assumptions clear, and supports rich interpretation of results by analyzing likelihoods at states and transitions. The sophisticated modeling, including motif-to-motif specific transitions and several distinct background states should allow more specific CRM predictions at the same level of sensitivity. Finally, by relying on soft priors instead of hard specification of model parameters, the Bayesian approach adds generality and user convenience to the method.

2 Methods

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.

2.1 A Hierarchical HMM of TRS

As first proposed in [5], 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 → CRM → Motif → 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. Fig. 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

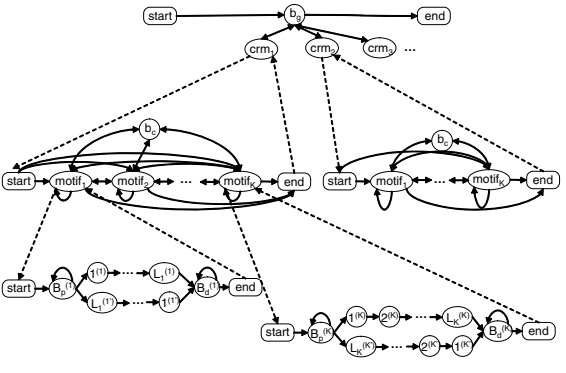

Fig. 1. The BayCis hHMM state transition diagram with 3-level hierarchy. Circular nodes represent functional states in DNA sequences, and round boxes represent start and end states in each sub-model. CRM and motif states are sub-models invoked by higher level models. Arrows between nodes represent permissible state transtions, including horizontal transitions denoted as black arrows, and verticle transitions denoted as dashed arrows.

specifically, we define: $\mathbb{M}_i \equiv \mathbb{M}_i^f \cup \mathbb{M}_i^r$, where $\mathbb{M}_i^f = \{1^{(i)} \dots L_i^{(i)}\}$ 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 $\mathbb{B}^p = \{b_p^{(1)}, \dots, b_p^{(K)}\}$ denotes the set of *proximal-buffer* states associated with each type of motif [2], and $\mathbb{B}^d = \{b_d^{(1)}, \dots, b_d^{(K)}\}$ 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 Fig. 1. (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 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 later). 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 CRMs; 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 [5] 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 [17] 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. In section 2.3 and

[2] 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.

Supplementary Materials, we exploit this strategy, and develop an efficient algorithm for inference and learning under a Bayesian extension of hHMM to be described in the sequel.

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 in this paper, 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 [3]. 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

[3] The distinction between proximal and distal buffers avoids generating empty envelops (otherwise, a single buffer state wont be able to remember if a motif has been generated beyond k positions prior to the current position under a k-th order Markov model).

range of spacer length is under selection forces according to comparative genomics data of several *Drosophila* species [13]. Empirically, we found that the distribution of spacer lengths can be approximated by a negative binomial distribution (see figure in Supplementary Materials), 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. In Supplementary Materials, we describe a generalized hierarchical hidden Markov model (GhHMM) which 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 in this paper (although extending our model for *de novo* motif detection is straightforward based on, for example, the LOGOS framework [30]). For motif model we choose the classical product-multinomial (PM) model, which can be represented by a PWM [25].

Several previous studies have stressed the importance of using a richer background model for the non-motif sequences [26,11]. 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 compute the local Markov model of each successive position, each with a constant "update cost" based on the previous one).

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 overestimation 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 [7]. 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*. 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 [7].

2.3 Inference and Learning

We have developed an efficient algorithm called *modified FB-algorithm* for inference on a "flattened" hHMM, which reduces the time complexity of the standard forward-backward algorithm from $O(K^2\bar{L}^2T)$ to $O(K^2T)$. Identification of motifs/CRMs is

based on posterior decoding. We also developed a *variational EM algorithm* for Bayesian inference and parameter estimation under our Bayesian hHMM and GhHMM, which is much more efficient than the traditional MCMC sampling approaches. Due to space limit, details of these algorithmic innovations are given in the Supplementary Materials.

3 Results

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 [7], Cluster-Buster [6], MSCAN [1], Ahab [19] and Stubb [24] (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.

3.1 Semi-realistic Synthetic TRS

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 [29] database in simulated background sequences according to model assumptions underlying the background distribution, the inter-TFBS and inter-CRM spacer length distributions for Baycis. 30 sequences of length 20,000 bp containing 0 - 3 CRMs were generated. The CRM length is uniformly distributed between 200 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 CRM appear as predefined pairs. The background sequences

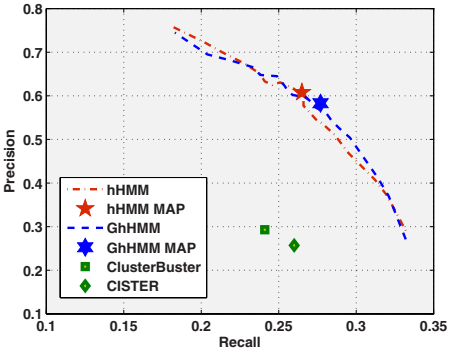

Fig. 2. The precision-recall (P/R) curves of two models of BayCis (hHMM and GhHMM) versus the P/R of default predictions by CISTER and ClusterBuster

inside/outside the CRM are simulated by a 3rd-order Markov model learnt from an intergenic region.

As shown in Fig. 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 precision 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 (see next subsection).

3.2 Real Drosophila TRS

The dataset. The synthetic TRSs are generated partially based on the same model assumptions underlying BayCis, and thus the results cannot be interpreted as conclusive. A systematic investigation of the robustness of BayCis with respect to a wide spectrum of simulation conditions can be highly interesting but is beyond the scope of this short report; we will pursue this in a later full version of the paper. 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.

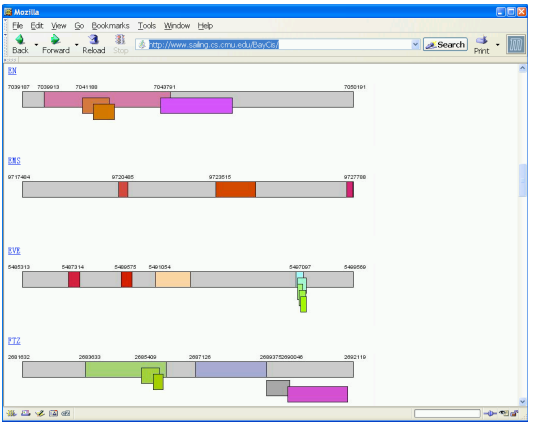

Fig. 3. Frontpage screenshot of the motif database

We created a manually curated dataset containing 97 CRMs pertaining to 35 early developmental genes (see table in Supplementary Materials for details). This collection was compiled based on a filtering of all known CRMs from a number of public databases (e.g., the REDfly CRM database [8] and the Drosophila Cis-regulatory Database at the National University of Singapore [18]), 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 software will soon be also released. A snapshot of the interface of

graphical interface of the database shown in Fig. 3, and more details are available in Supplementary Materials.

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, and will be pursued in a later paper.

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-CRM 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, in this paper 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

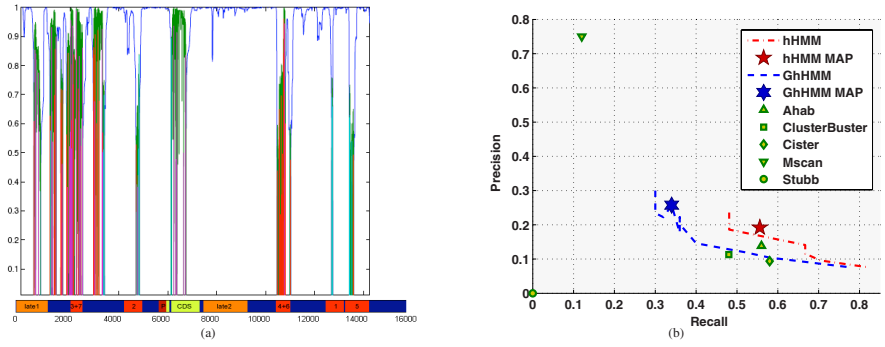

Fig. 4. Performance of BayCis (hHMM) on a representative *eve* TRS. (a) The posterior probability plot of the global background (blue), cluster background (green) and motif specific (red and other colors) states. (b) The precision versus recall performance of the MAP and thresholded predictions of the hHMM and GhHMM algorithms, as compared to those made by other methods.

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 [28] as single point measures (see Supplementary Materials B.3 for detailed definitions). 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 P/R, we plot the binding-site level P/R curve from different thresholds in extracting predictions, along with the P/R at MAP predictions.

Motif prediction performance. As an illustration, Fig. 4a 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 Fig. 4b, 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

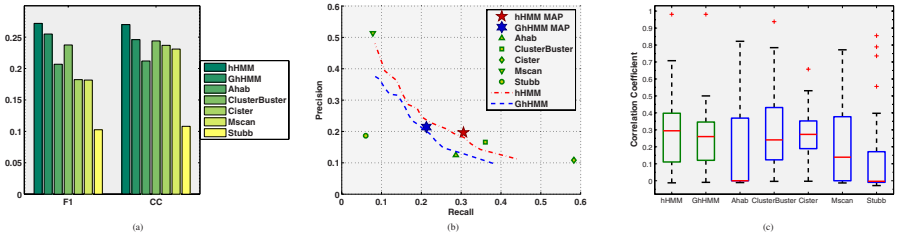

Fig. 5. (a) F1 and CC scores, and (b) P/R performances of the MAP and thresholded predictions of the hHMM and GhHMM, in comparison with other algorithms on the full *Drosophila* TRS dataset (c) A boxplot showing variation in CC across datasets

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 Fig. 5a. 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 P/R landscape on our entire dataset, we plot the P/R curve resulting from different thresholds for BayCis predictions. For other methods we provide the single points in P/R landscape corresponding to their default output. As is apparent from Fig. 5b, 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 P/R curves of both versions of BayCis span a balanced range in the P/R landscape, with MAP estimates lying in the middle of the curves. Again, in practice the P/R 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 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 Fig. 5(c) 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 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. We defer this investigation to the full paper.

4 Discussion

BayCis uses an advanced probabilistic framework to accurately model metazoan transcriptional regulatory genomic sequences — which often consist of multiple CRMs, tandemly joined by long stretches of background DNA, each containing locally enriched occurrences of binding motifs for a certain array of transcriptional regulatory proteins. Thus, we are able to detect many TFBS while avoiding too many false positives and (slightly) outperform the best of the existing methods on a comprehensive set of *Drosophila* regulatory regions. The BayCis software will soon be released on our website.

Recently, experimental results have shown that sequences immediately flanking a TFBS may contribute to the binding energy between a TF and the TFBS [14]. This suggests that sequence composition of the proximal and distal buffers of motifs may have weak type specificity, which we would like to explore in our future work. Our current TRS database for performance evaluation is still limited in size and very diverse in terms of CRM structures and complexity, which could cause BayCis to overfit certain TRS when it is applied independently to each TRS separately (as we did in this paper), using a generic set of hyperparameters that are empirically chosen. We intend to adopt a more systematic approach to fit the hyperparameters based on a small amount of labeled TRS, e.g., using a k-fold cross validation scheme. But ultimately, we believe additional TRS data will be needed to attain further performance increase. One direction of increasing input data is to combine regulatory regions of several genes that are believed to share similar CRM structure. Such gene sets should be attainable for many real scenarios where CRM discovery methods are used, could trivially be used as input to BayCis. We speculate that this could improve predictions. The limitation lies mostly in collecting such gene sets containing rich, high-quality annotations that could serve in quantitatively measuring correspondence between computational prediction and experimental determination.

Another direction is to conjoin BayCis with a phylogenetic model of motifs across species [16,22,23], and apply the integrant to orthologous TRSs. Although this limits the applicability of the approach to species where valuable orthologous sequence is available, and to the discovery of regulatory elements shared between species, we believe it could attain considerably performance gain in the cases for which it is suited.

Acknowledgements. This material is based on work supported by the Pennsylvania Dept of Health's Health Research Program under Grant No. 2001NF-Cancer Health Research Grant ME-01-739, and by an NSF CAREER Award under Grant No. DBI-054659. The authors thank Wenjie Fu for result analysis, Jostein Johansen for help with evaluating CRM predictions, and Oznur Tastan for investigating the spacer length distributions.

References

1. Alkema, W.B., Johansson, O., Lagergren, J., Wasserman, W.W.: Mscan: identification of functional clusters of transcription factor binding sites. Nucleic Acids Res. 32(Web Server issue), 195–198 (2004)
2. Berman, B.P., Nibu, Y., Pfeiffer, B.D., Tomancak, P., Celniker, S.E., Levine, M., Rubin, G.M., Eisen, M.: Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the Drosophila genome. Proc. Natl. Acad. Sci. USA 99(2), 757–762 (2002)
3. Davidson, E.H.: Genomic Regulatory Systems. Academic Press, London (2001)
4. Donaldson, I.J., Chapman, M., Gottgens, B.: Tfbscluster: a resource for the characterization of transcriptional regulatory networks. Bioinformatics 21(13), 3058–3059 (2005)
5. Fine, S., Singer, Y., Tishby, N.: The hierarchical hidden Markov model: Analysis and applications. Mach Learning 32, 41–62 (1998)
6. Frith, M., Li, M., Weng, Z.: Clusterbuster:finding dense clusters of motifs in dna seqs. Nuc. Ac. Res. 31(13), 3666–3668 (2003)
7. Frith, M.C., Hansen, U., Weng, Z.: Detection of cis-element clusters in higher eukaryotic DNA. Bioinf. 17, 878–889 (2001)
8. Gallo, S., Li, L., Hu, Z., Halfon, M.: Redfly:a regulatory element database for drosophila. Bioinf. 22(3), 381–383 (2006)
9. Gupta, M., Liu, J.S.: De novo cis-regulatory module elicitation for eukaryotic genomes. Proc. Natl. Acad. Sci. USA 102(20), 7079–7084 (2005)
10. Huang, H., Kao, M., Zhou, X., Liu, J.S., Wong, W.H.: Determination of local statistical significance of patterns in Markov sequences with application to promoter element identification. Journal of Computational Biology 11(1) (2004)
11. Liu, X., Brutlag, D.L., Liu, J.: Bioprospector: Discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. Proc. of Pac. Symp. Biocomput., 127–138 (2001)
12. Loots, G.G., Ovcharenko, I., Pachter, L., Dubchak, I., Rubin, E.M.: rVista for comparative sequence-based discovery of functional transcription factor binding sites. Genome Res. 12(5), 832–839 (2002)
13. Ludwig, M.Z., Patel, N.H., Kreitman, M.: Functional analysis of eve stripe 2 enhancer evolution in Drosophila: rules governing conservation and change. Development 125(5), 949–958 (1998)
14. Maerkl, S.J., Quake, S.R.: A systems approach to measuring the binding energy landscapes of transcription factors. Science 315, 233–237 (2007)
15. Michelson, A.: Deciphering genetic regulatory codes:a challenge for fnal genomics. Pr. Nat. Acad. Sc. USA 99, 546–548 (2002)
16. Moses, A.M., Chiang, D.Y., Eisen, M.B.: Phylogenetic motif detection by expectation-maximization on evolutionary mixtures. Pac. Symp. Biocomput., 324–335 (2004)
17. Murphy, K., Paskin, M.: Linear time inference in hierarchical hmms. Adv. in Neural Inf. Proc. Sys. 14 (2002)
18. Narang, V., Sung, W.K., Mittal, A.: Computational annotation of transcription factor binding sites in D melanogaster developmental genes. In: Proceedings of The 17th International Conference on Genome Informatics (2006)
19. Rajewsky, N., Vergassola, M., Gaul, U., Siggia, E.D.: Computational detection of genomic cis-regulatory modules, applied to body patterning in the early Drosophila embryo. BMC Bioinformatics 3(30), 1–13 (2002)
20. Rebeiz, M., Reeves, N.L., Posakony, J.W.: 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. USA 99(15), 9888–9893 (2002)

21. Sharan, R., Ovcharenko, I., Ben-Hur, A., Karp, R.M.: Creme: a framework for identifying cis-regulatory modules in human-mouse conserved segments. Bioinformatics 19(Suppl 1), i283–291 (2003)
22. Siddharthan, R., Siggia, E.D., van Nimwegen, E.: Phylogibbs: A gibbs sampling motif finder that incorporates phylogeny. PLoS Computational Biology 1(7), e67 (2005)
23. Sinha, S., Blanchette, B., Tompa, M.: Phyme: A probabilistic algorithm for finding motifs in sets of orthologous sequences. BMC Bioinformatics 5(170) (2004)
24. Sinha, S., Liang, Y., Siggia, E.: Stubb: a program for discovery and analysis of cis-regulatory modules. Nucleic Acids Res. 34(Web Server issue), W555–W559 (2006)
25. Staden, R.: Computer methods to locate signals in nucleic acid sequences. Nucleic Acids Res. 12(1 Pt 2), 505–519 (1984)
26. Thijs, G., Lescot, M., Marchal, K., Rombauts, S., DeMoor, B., Rouze, P., Moreau, Y.: A higher-order background model improves the detection of promoter regulatory elements by gibbs sampling. Bioinformatics 17(12), 1113–1122 (2001)
27. Thompson, W., Palumbo, M.J., Wasserman, W.W., Liu, J.S., Lawrence, T.E.: Decoding human regulatory circuits. Genome Res. 14(10A), 1967–1974 (2004)
28. Tompa, M., Li, N., Bailey, T., Church, G., DeMoor, B., Eskin, E., Favorov, A., Frith, M., Fu, Y., Kent, W., Makeev, V., Mironov, A., Noble, A., Pavesi, G., Pesole, G., Regnier, M., Simonis, N., Sinha, S., Thijs, G., van Helden, J., Vandenbogaert, M., Weng, Z., Workman, C., Ye, C., Zhu, Z.: Assessing computational tools for the discovery of transcription factor binding sites. Nat. Biotech. 23(1), 137–144 (2005)
29. Wingender, E., Dietze, P., Karas, H., Knuppel, R.: TRANSFAC: a database on transcription factors and their DNA binding sites. Nucleic. Acids. Res. 24(1), 238–241 (1996)
30. Xing, E.P., Wu, W., Jordan, M.I., Karp, R.M.: Logos: A modular Bayesian model for de novo motif detection. Journal of Bioinformatics and Computational Biology 2(1), 127–154 (2004)
31. Zhou, Q., Wong, W.H.: Cismodule: de novo discovery of cis-regulatory modules by hierarchical mixture modeling. Proc. Natl. Acad. Sci. USA 101(33), 12114–12119 (2004)

BayCis: A Bayesian Hierarchical HMM for Cis-regulatory Module Decoding in Metazoan Genomes

Tien-ho Lin[1], Pradipta Ray[1] , Geir K. Sandve[2], Selen Uguroglu[3], Eric P. Xing[1⋆]

[1] School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
[2] Dept of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway
[3] Dept of Computer Science and Engineering, Sabanci University, Istanbul, Turkey

APPENDIX: SUPPLIMENTARY MATERIALS

Keywords
Motif, transcription factor, Bayesian model, *cis*-regulatory module,
hierarchical hidden Markov model, generalized hidden Markov model,
Drosophila melanogaster

A Details on the BayCis model and algorithm

A.1 Modeling spacer length distribution via GhHMM

Consider the actual spacer length histogram in *D. melanogaster* in Figure 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.

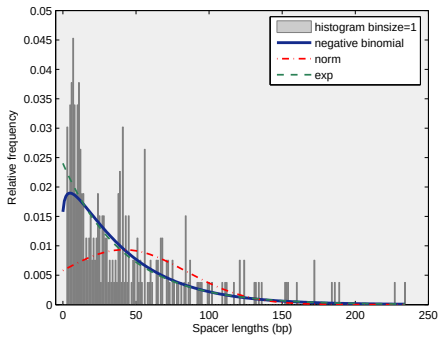

Fig. 1. The histogram of spacer length distribution with known standard distributions superimposed.

Generalized hidden Markov models (GHMM) have been proposed for the explicit modeling of the state durations in an HMM [10, 3, 7]. 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 2. This allows approximation of the GHMM functionality in a HMM [2], where the efficient forward-backward and posterior decoding algorithms can be reused.

⋆ Correspondence should be addressed to epxing@cs.cmu.edu. This material is based on work supported by the Pennsylvania Dept of Health's Health Research Program under Grant No. 2001NF-Cancer Health Research Grant ME-01-739, and by an NSF CAREER Award under Grant No. DBI-054659. The authors thank Wenjie Fu for result analysis, Jostein Johansen for help with evaluating CRM predictions, and Oznur Tastan for investigating the spacer length distributions.

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 1. This scenario has been used to model exon length distribution by EasyGene to achieve better accuracy [6]. 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.

A.2 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 [8]. 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 $\mathbb{Q} = \{b_g, b_c\} \cup \mathbb{B} \cup \big(\cup_k \mathbb{M}_k \big)$.

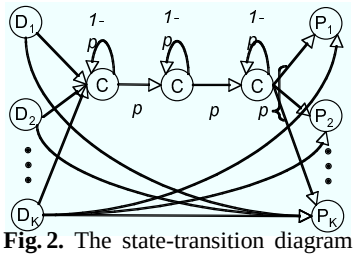

Fig. 2. The state-transition diagram of a gHMM.

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 [10], 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

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

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

It is known that the standard forward and backward algorithm both take $O(N^2T) = O(K^2\bar{L}^2T)$, 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 (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

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

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.

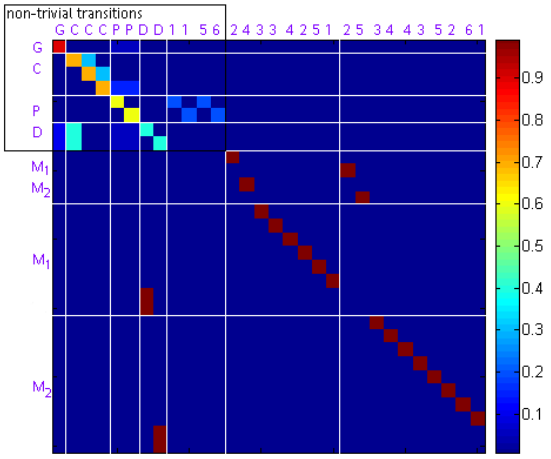

Fig. 3. The transition probability matrix of the flattened HMM, shown as a heat map. G, C, P, D, and the numbers correspond to global, cluster, proximal and distal background, and the motif states. The motif states are ordered as: $1^{(1)}, 1^{(2)}, \cdots, 1^{(K)}, L_1^{(1')}, L_2^{(2')}, \cdots, L_K^{(K')},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

A.3 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.

A.4 Bayesian inference and learning

Under the Bayesian framework described in the main paper, 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:

$$p(X_t|Y) = \int p(X_t = s|Y, \Xi)p(\Xi|Y)d\Xi \tag{3}$$

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 [12], also referred to as *variational Bayesian learning* [5] 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 [1] [4], 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, for the state-transitioning parameter $\beta_{g,g}$, we have:

$$\begin{aligned} E[\ln(\beta_{g,g})] &\int_{\beta_{g,g}} \ln \beta_{g,g} p(\beta_{g,g}|\xi_{g,1}, \xi_{g,2}, E[n_{g,g}])d\beta_{g,g} \\ &= \Psi(\xi_{g,1} + E[n_{g,g}]) - \Psi(\sum_j \xi_{g,j} + \sum_{k \in \mathbb{B}_p} E[n_{g,k}]), \end{aligned} \tag{4}$$

[4] 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.

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 4, 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:

$$\beta_{g,i} = \beta^*_{g,\bar{g}} \frac{\sum_t p(X_t = g, X_{t+1} = i|Y)}{\sum_{t,k} p(X_t = g, X_{t+1} = k|Y)} \tag{5}$$

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.

A.5 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:

$$E[n_{c,c}] = \frac{1}{\xi_{cr}} \sum_{j=1}^{\xi_{cr}} E[n_{c^j,c^j}], \tag{6}$$

$$E[n_{c,c1}] = \frac{1}{\xi_{cr}} \Big(\sum_{j=1}^{\xi_{cr}-1} E[n_{c^j,c^{j+1}}] + \sum_{k \in \mathbb{B}_p} E[n_{c^{\xi_{cr}},k}] \Big) \tag{7}$$

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

$$E[\ln(\beta_{c^j,c^j})]\Psi(\xi_{c,1} + E[n_{c,c}]) - \Psi(\xi_{c,1} + \xi_{c,2} + E[n_{c,c}] + E[n_{c,c1}]) \quad 1 \leq j \leq \xi_{cr}. \tag{8}$$

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 $\beta^*_{c^j,c^j} = \exp(E[\ln(\beta_{c^j,c^j})])$.

B Additional details on experiments

B.1 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 aforementioned 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 [4, 9]. 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 Table 1.

$Gene(Length)$	$CRM/Length$	$Motif$	$Gene/Length$	$CRM(Length)$	$Motif$
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_1.2_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 1. Summary of the Drosophila TRS dataset used for in performance comparison.

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 Fig.3 in the main paper.

B.2 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 $r = 2$.

B.3 More on F1 and CC scores

The nucleotide-based prediction error is used in the Nature Biotechnology benchmark paper by Tompa et al. [11]. The formulas for the F1 and CC scores are as follows:

$$CC = \frac{nTP \times nTN - nFN \times nFP}{\sqrt{(nTP + nFN)(nTN + nFP)(nTP + nFP)(nTN + nFN)}}, \tag{9}$$

$$F1 = \frac{2 \times Pr \times Re}{Pr + Re}, \tag{10}$$

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.

References

1. M. J. Beal, Z. Ghahramani, and C. E. Rasmussen. The infinite hidden Markov model. In *Advances in Neural Information Processing Systems 13*, 2001.
2. R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1988.

3. 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.
4. S. Gallo, L. Li, Z. Hu, and M. Halfon. Redfly:a regulatory element database for drosophila. *Bioinf*, 22(3):381–383, 2006.
5. Z. Ghahramani and M. Beal. Propagation algorithms for variational Bayesian learning. In *Advances in Neural Information Processing Systems 13*, 2001.
6. T. S. Larsen and A. Krogh. Easygene–a prokaryotic gene finder that ranks orfs by statistical significance. *BMC Bioinformatics*, 4:21, Jun 2003.
7. S. E. Levinson. Continuously variable duration hidden markov models for automatic speech recognition. *Comput. Speech Lang.*, 1(1):29–45, 1986.
8. K. Murphy and M. Paskin. Linear time inference in hierarchical hmms. In *Adv in Neural Inf Proc Sys 14*, 2002.
9. 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.
10. L. R. Rabiner and B. H. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, pages 4–15, January 1986.
11. M. Tompa, N. Li, T. Bailey, G. Church, B. DeMoor, E. Eskin, A. Favorov, M. Frith, Y. Fu, W. Kent, V. Makeev, A. Mironov, W. Noble, G. Pavesi, G. Pesole, M. Regnier, N. Simonis, S. Sinha, G. Thijs, J. van Helden, M. Vandenbogaert, Z. Weng, C. Workman, C. Ye, and Z. Zhu. Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotech*, 23(1):137–44, 2005.
12. 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.