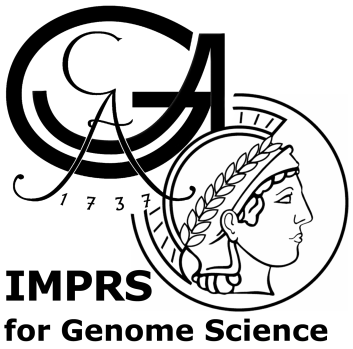
Discovery and prediction of protein binding sites in DNA and RNA sequences using Bayesian Markov models



Dissertation
for the award of the degree
"Doctor rerum naturalium"
division of Mathematics and Natural Sciences
of the Georg-August-Universität Göttingen

within the doctoral program
International Max Planck Research School for Genome Science
of the Georg-August University School of Science (GAUSS)

submitted by

Wanwan Ge

from Xuchang, China
Göttingen, April 2021

Thesis Committee

• Dr. Johannes Söding, Research Group Quantitative and Computational Biology, Max Planck Institute for Biophysical Chemistry

• Prof. Dr. Burkhard Morgenstern, Institute for Microbiology and Genetics, Department Bioinformatics, Georg-August University Göttingen

• Prof. Dr. Michael Habeck, Research Group Statistical Inverse Problems in Biophysics, Max Planck Institute for Biophysical Chemistry

Members of the Examination Board

Referee: Dr. Johannes Söding, Research Group Quantitative and Computational Biology, Max Planck Institute for Biophysical Chemistry

2nd Referee: Prof. Dr. Burkhard Morgenstern, Institute for Microbiology and Genetics, Department Bioinformatics, Georg-August University Göttingen

Further members of the Examination Board

• Prof. Dr. Michael Habeck, Research Group Statistical Inverse Problems in Biophysics, Max Planck Institute for Biophysical Chemistry

• Prof. Dr. Stephan Waack, Institute of Computer Science, Georg-August University Göttingen

• Dr. Juliane Liepe, Research Group Quantitative and Systems Biology, Max Planck Institute for Biophysical Chemistry

• Dr. Nico Posnien, Department Developmental Biology, Georg-August University Göttingen

Date of oral examination: 10th of July, 2020

Declaration

I hereby declare that the doctoral thesis entitled, "Discovery and prediction of protein binding sites in DNA and RNA sequences using Bayesian Markov models" has been written independently and with no other sources and aids than quoted.

Wanwan Ge
April 2021

Acknowledgements

The Ph.D. journey is a big leap for me, and it would not have been possible without the supports from the people I met along the way. Thank you all sincerely.

First and foremost, I would like to acknowledge my advisor Johannes for giving me the opportunity to work with him, being a great mentor, and offering me all the freedom and guidance through my Ph.D. study. I have learned a lot from him, not only about science.

I thank all my other mentors for being supportive: Prof. Dr. Burkhard Morgenstern and Dr. Michael Habeck for being members of my thesis committee and giving valuable suggestions, Prof. Dr. Stephan Waack, Dr. Juliane Liepe and Dr. Nico Posnien for being in my examination committee.

I thank my collaborator Prof. Dr. Herbert Jäckle for providing the *Drosophila* transcriptomics data and having the discussions.

I thank IMPRS-GS for creating such an interdisciplinary environment and providing financial support, Henriette and Katja for their efficient coordination, and the fellow students for all the events and discussions.

I am grateful to the current and former Soeding lab members for being supportive colleagues as well as good friends. I want to thank Matthias and Anja for guiding me into this project, Christian for all the discussions, critics and ideas, Saikat for organizing all the movie nights, game nights and nice trips, Eli for her great empathy and inner power which inspire me a lot, Niko especially for all the cross-cultural discussions, Ruoshi for sharing the barista skills, Milot for all the technical support and hosting us for game nights and dinners, Stefan, Markus, Martin, Clovis, Gonzalo, Salma, Annika, Franco as well as the master students and interns for the scientific discussions, coffee talks and overall a friendly and scientific working atmosphere.

I am very thankful for harvesting friendships with many kind people in Göttingen, especially Yehan, Yuanzi, Bingyao, Ashish and Ling for sharing the cheerful moments in this beautiful town, Yang and Le for sharing their passion for science, Karin and Wolfgang for introducing me the peaceful local life.

Last but not least, I can never thank my family members enough for their love, patience, and unlimited support.

Abstract

Transcription factors control the essential step of gene expression via recognizing the over-represented binding sites (or *motifs*) on the genome. One crucial task is to accurately predict these binding sites on the genome, to understand the regulatory mechanisms.

This thesis approaches this task in three parts.

In the first part, I introduce a tool, BaMMmotif2, that I have developed to identify motifs *de novo* from DNA sequencing data. Compared to the existing position weight matrix (*PWM*)-based motif discovery tools, the higher-order Bayesian Markov models (*BaMM*s) have the advantages of learning the interdependence of the nucleotides for transcription factor binding while being fast and having high predictive accuracy. The core of the BaMMs is that the higher-order probability is learned by combining the k-mer counts and the probability of one order lowers with a pseudo-factor α tuning the weights between the two. I optimize a position- and order-specific pseudo-factor α for higher-order BaMMs. I also introduce the method to learn the positional preferences of the transcription factors. Besides, I apply a masking step to the input sequences to train the model only with the most relevant positions, and thus it helps distinguish weak motifs when multiple binding motifs are present in the data.

In the second part, I introduced a new and better motif performance score, the average recall (*AvRec* score), to give the users some guidance on evaluating the motif quality. Besides, to validate the existing motif detection tools, I developed a full scheme including (I) N-fold cross-validation, (II) cross-platform validation, and (III) cross-cell-line validation. In 5-fold cross-validation, BaMMmotif2 outperforms the selected state-of-the-art tools in this field, with at least 13.6% and 12.2% median increase in the AvRec score using *in vivo* and *in vitro* data, respectively. In the cross-cell-line validations on 238 datasets, BaMMmotif2 gains $>11\%$ median increases in the AvRec score. BaMMs also perform the best in the cross-platform validation on 16 data sets. By applying BaMMs for the CTCF motif to scan the whole human genome, I discover 1.5 million CTCF binding sites with high accuracy. This result could lead to a better understanding of the genome 3D structure and its biological functions.

In the third part, we offer the community an interactive web server with the tool and database: bammmotif.soedinglab.org. It provides four main functionalities: (I) *de novo* predicting motifs from DNA/RNA sequences, (II) finding motif occurrences given a sequence and a motif model, (III) searching for similar known motifs in the database, given a novel motif model, and (IV) offering databases with higher-order BaMMs for different organisms.

Table of contents

Chapter 1

Introduction

To understand how life processes are happening at the cellular level, we need to learn how to interpret the genomics dictionary, which consists of simple code letters (A,C,G,T). Deciphering this book and translating it into meaningful sentences is a complex task for the cell. The central dogma of biology (Figure 1.1) illustrates the flow of genetic information from DNAs to functional proteins via RNAs. The DNA replicates itself and also transcripts its sequence information into RNAs. For eukaryotes, the precursor RNA sequences contain both exons (<10%) for being translated into proteins and introns (90%), which will be spliced after being translocated from the nucleolus to cytoplasm. After splicing, some of the mature RNAs, namely messenger RNAs (mRNAs), are translated into protein amino acid chains, which later can be folded into 3D functional proteins.

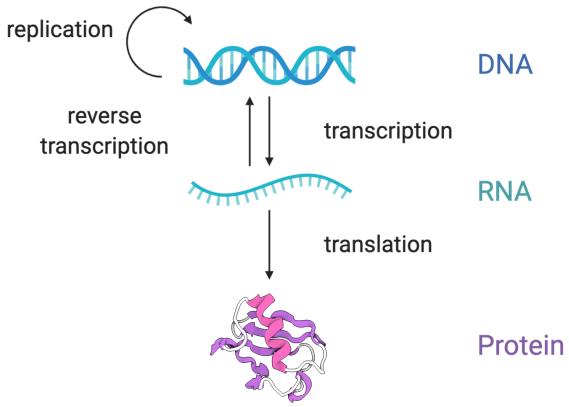


Fig. 1.1 **The central dogma of biology.**

A full set of the human genome consists of approximately 3 billion base pairs of DNA molecules, which are translated into about 20,000 functional proteins and small molecular RNAs. The transcription regulation is thus a complicated task. It is mainly controlled by

the regulatory factors, such as *transcription factors* (TFs) and non-coding RNAs, when they recognize and bind to their target binding sites.

This thesis aims to understand the transcription regulation by deciphering the genomic regulatory information with statistical models. I first introduce the transcription regulation mechanism and the key players - transcription factors. Then I focus on the genomic regulatory patterns, or *motifs*, which can be recognized by transcription factors with specific affinities. Next, I describe the experimental approaches for studying the interactions between transcription factors and motifs, followed by the computational methods for interpreting the sequencing results. Last but not least, I conclude this thesis with results and applications.

1.1 Transcriptional regulation and transcription factors

Transcription is the very first and essential step in gene expression (Figure 1.2). It is a complex process that controls how genetic information will be converted from DNA to RNA sequence. In eukaryotes, transcription starts when transcription activators recognize accessible genome regions and bind to enhancer elements. For the transcription of protein-coding genes, general transcription factors and RNA polymerase II (Pol II) are assembled into the pre-initiation complex (PIC), at the core promoter region, with the assistance of the mediator complex. RNA polymerase binds to the transcription start site (TSS) and starts transcribing DNA into RNA. During this process, there is a CCCTC-binding factor (CTCF) which is crucial for mediating the intra- and inter-chromosomal contacts [1].

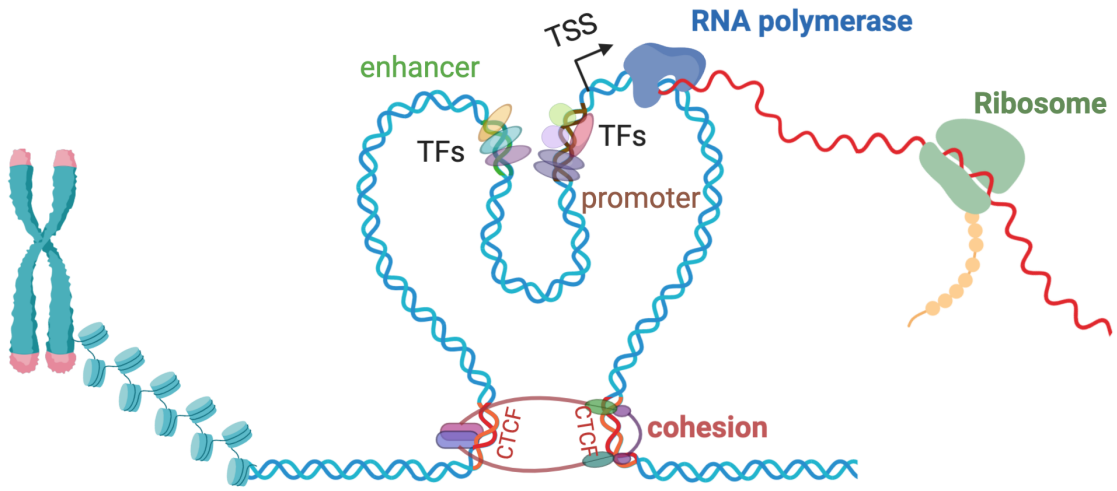


Fig. 1.2 **Transcriptional regulation in eukaryotic cells.**

In human cells, 1,600 out of approximately 2,000 total proteins with DNA binding domains, might function as transcription factors. Transcription factors are often classified according to either (1) their DNA binding domains, or (2) regulatory functions. The spatiotemporal combination of transcription factors determines when to express specific sets of genes and how much these genes will be expressed. Misregulations of transcription factor activities can lead to diseases such as cancer [2] and diabetes [3, 4]. Therefore, transcription factors can act as markers for cancer treatments [5].

1.2 Transcription factor binding motifs

The binding of transcription factors determines gene expression during cell development to accessible DNA patterns in promoter-proximal and distal regions. These regions are mostly GC-rich, nucleosome-depleted, and DNase I-accessible regions [6], and they are highly conserved during evolution. These functional DNA patterns are called motifs. They are typically 6-20 base pairs long and determines the binding sites for proteins such as transcription factors and nucleases, as well as for RNA processing such as splicing and modifications. The binding affinity of transcription factors to motifs depends primarily on hydrogen bonding between specific amino acid residues in the protein and individual bases in the DNA sequence [7]. It can also be influenced by the 3D structure of DNA and chromatin modifications [8].

Transcription factor-DNA weak binding affinities

Previous studies on the phage λ operator and the yeast Gal1 promoter identified binding sites with a range of affinities crucial for gene regulation [9, 10]. Another example is the zinc finger (ZF) family, which is a major transcription factor family with the largest portions (approximate 80%) of unknown motifs [11]. ZF proteins often have different "fingers" for binding to different DNA residues with weak binding affinities and in various combinations. The low-affinity binding sites are TF-bound DNA sites that are 10^3 fold weaker than the optimal pattern yet still can be recognized by transcription factors, compared to other sequences. It allows the modulation of the regulatory processes and makes cells adaptive to different environments. Notably, the low-affinity binding sites are crucial for precisely regulating specific gene expressions during cell development [12, 13]. To reach particular specificity with low binding sites, the cell needs a local transcription factor boost to a particularly high level, which might co-develop with the formation of transcriptional hubs in the cells [13].

Different transcription factor-motif binding modes

The same transcription factors do not always target the same motifs. One example is that dimeric transcription factors commonly bind to motifs with variable spacing residues between the two halves, such as the transcription factor family Maf [14]. Another example is the basic leucine zipper (bZIP) transcription factor family member Hac1 [15].

The determinants for transcription factor-DNA binding specificity can be classified into four groups: (I) base and shape readout, (II) effects of co-factors, (III) cooperativity between different transcription factors, and (IV) chromatin accessibility status [16] (Figure 1.3).

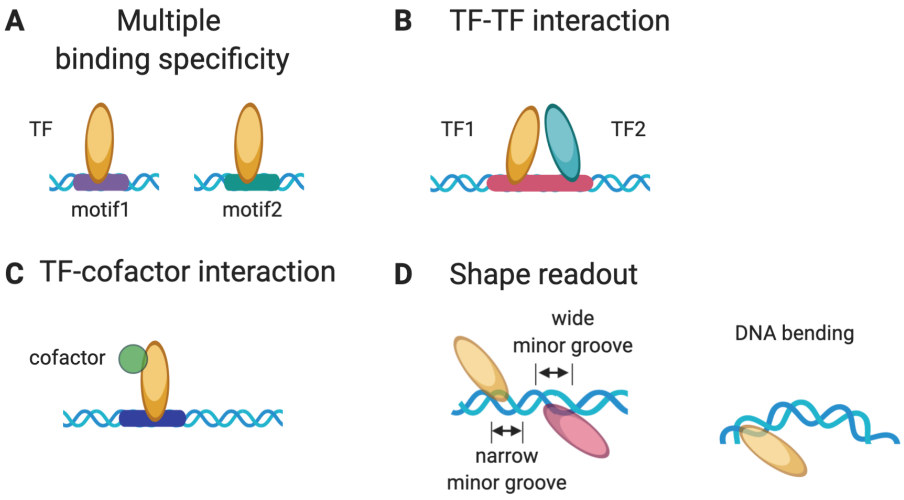


Fig. 1.3 **Multiple transcription factor-DNA binding modes.**
(**A**) Transcription factors can have multiple binding modes with various binding specificities. Thus the motifs can be different. (**B**) Transcription factors can interact with each other and bind to close regions. Thus the shared motif is different from motifs that are bound individually. One example is the Oct4-Sox2 complex. (**C**) The co-factor binds to the transcription factor and thus changes its binding affinity to its motif. (**D**) Transcription factor recognizes DNA shapes such as wide/narrow minor groove and DNA bending, thus alters its binding affinity.

The base readout (or *direct* readout) mechanism involves specific hydrogen bond formation and hydrophobic contacts between amino acid side chains and bases. The shape readout (or *indirect* readout) mechanism refers to protein binding that is influenced by the shape of a DNA molecule, which can be determined by sequence-dependent DNA bending and deformability (Figure 1.3 D). Most transcription factors bind to DNA via the interplay of the base and shape readout. Noticeably, in narrow minor grooves, arginines are most enriched and recognize enhanced electrostatic potentials [8].

Co-factors

Transcription factors can either bind directly to the genome or act as co-factors to assist other more specific transcription factor bindings [6]. These co-factors often bind to the secondary motifs *in vivo* via indirect binding. For instance, Lu et al. [17] discovered 23 co-factor motifs for 127 transcription factors in the human ENCODE project.

Pioneer factors

The binding of most transcription factors to the genome can be hurdled by nucleosomes on chromatin. Therefore, most transcription factors tend to bind to nucleosome-free DNA, instead of nucleosomal DNA. Pioneer factors are transcription factors that can directly bind to condensed chromatin and actively open up the chromatin while consuming ATPs. This leads to the rearrangement of nucleosomes and allows more space for other transcription factors to bind, and thus initiates the transcription process. Zhu et al. [18] systematically investigated the role of the nucleosome in DNA-TF binding, and found that some transcription factors actually bind to nucleosomal DNA gyres with orientation preferences.

1.3 Experimental measurements for protein-DNA interactions *in vivo* and *in vitro*

Early investigation of protein-DNA was carried out by cleavage DNA sequences with and without protein binding via a cleavage agent [19]. The DNA fragments were amplified using the polymerase chain reaction (PCR) and loaded on the polyacrylamide gel. The DNA regions with the protein of interest bound were protected and thus distinguishable from the randomly chopped DNA fragments. Large-scale detection of the DNA-protein interactions was boosted by next-generation sequencing (NGS). Assays such as MITOMI [20], ChIP-seq [21] and HT-SELEX [22], allow examining long DNA binding regions on the whole-genome level (Figure 1.4).

Approaches such as ChIP-exo [23] and MNase-seq [24] improve the detection of protein-DNA interactions to single-nucleotide resolution. CAP-SELEX [25] and NCAP-SELEX [18] were developed to examine the cooperativity of transcription factors and the role of nucleosomes in transcription factor-DNA binding. Methods such as ATAC-seq [26] provide information on DNA accessibility, which leads to a more accurate measurement of transcription factor binding. Supp. Table A.4 lists more techniques for detecting protein-DNA interactions in the past decades.

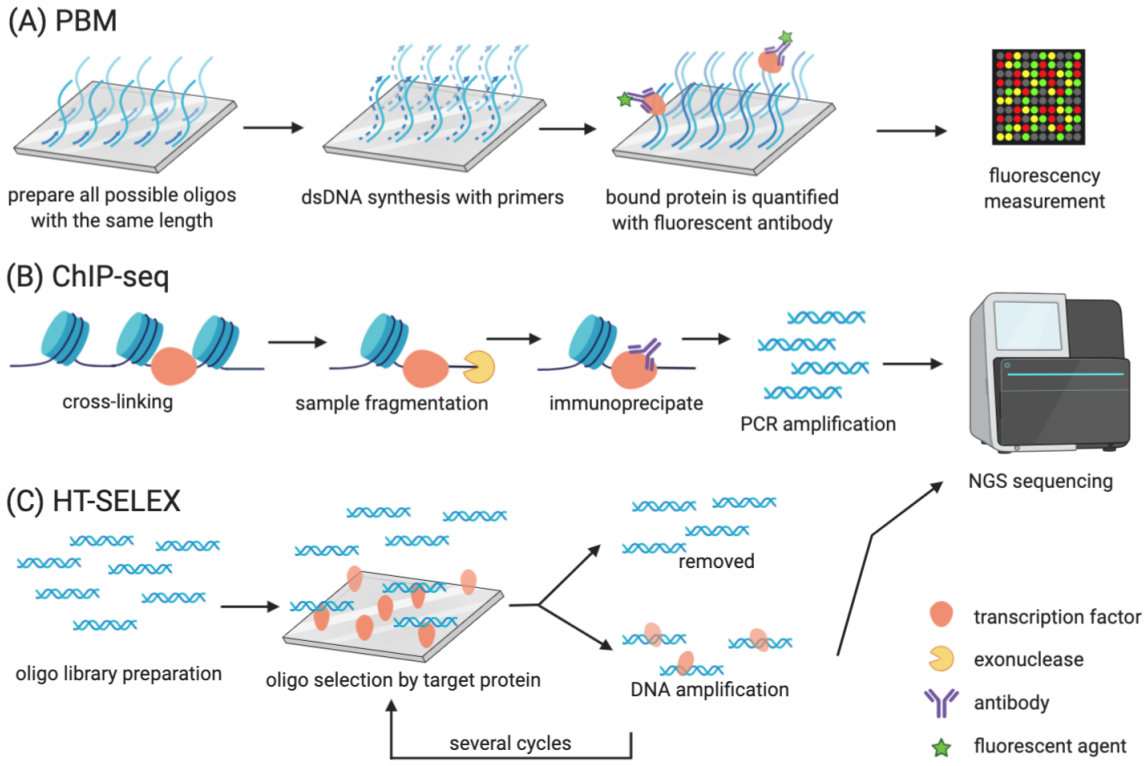


Fig. 1.4 **Representative experiments for detecting transcription factor binding.** Adapted from [27]. (**A**) An overview of protein binding microarrays (PBM) procedure. All possible sequences with the same length (e.g., ten bases long) are mounted on an array. dsDNAs are then synthesized mediated by primers on the array. Proteins of interest are added, and the nonspecific binding is washed away. The bound protein is detected and quantified with an antibody tagged with fluoresce. (**B**) An overview of chromatin immunoprecipitation followed by sequencing (ChIP-seq) design. Protein and DNA are cross-linked. The DNA molecule is fragmented randomly either by endonuclease or sonication. The bound regions are protected by proteins and separated via the immunoprecipitation of proteins and antibodies. Then the DNA is purified, amplified, and sequenced. (**C**) An overview of the high-throughput-SELEX (HT-SELEX) experiment. A random library of DNA oligomers is prepared and exposed to the transcription factor of interest. Unbound sequences flow through while the bound ones are amplified and redo the selection round for a few times. The bound oligomers are sequenced after each selection round, to determine the probability of being bound.

1.4 Computational techniques for finding motifs

Given the TF-bound sequences without knowing where motifs are, an essential computational task is to find the motifs enriched in the input set. There are three distinct approaches for *de novo* motif discovery: (1) *k*-mer based enumeration, (2) deterministic optimization (e.g., expectation-maximization), and (3) probabilistic optimization (e.g., Gibbs sampling) (see review [28]).

Figure 1.5 shows a simplified pipeline for *de novo* motif discovery workflow:

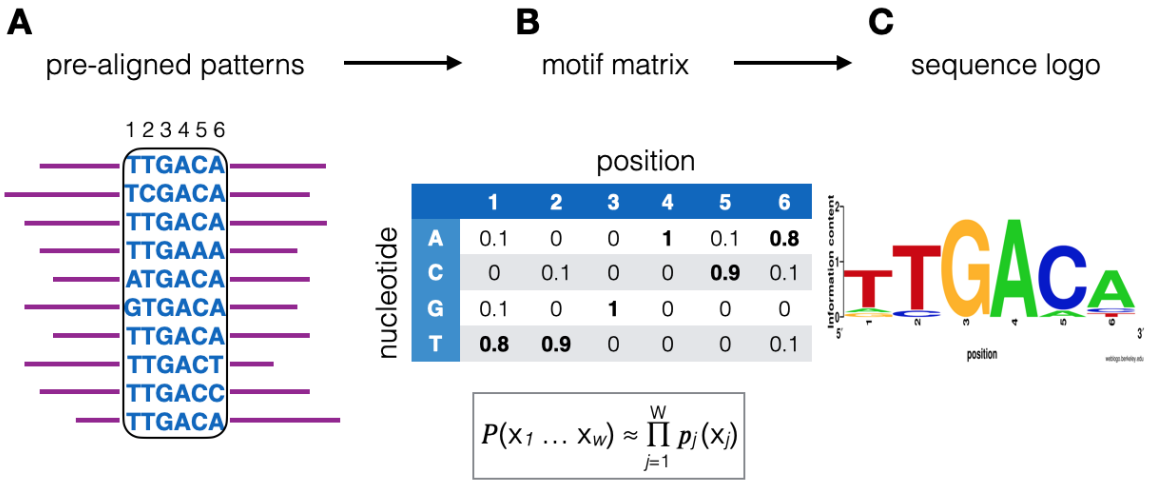


Fig. 1.5 ***De novo*** **motif discovery using position weight matrices.**
(**A**) TF-bound sequences are pre-aligned based on pattern similarities. (**B**) The frequencies of bases at each position of the enrichment patterns are calculated. The probability of a binding site X is calculated as the product of the probabilities of all its bases at their corresponding positions. (**C**) Motif logo. The information content of each base at each position, compared to a simplistic background model, which assumes that each base occurs with a probability of 0.25 at each position.

Position weight matrices (PWMs)

The specificity of DNA-protein binding was earlier represented by position weight matrices (PWMs) in 1982 [29]. PWM is defined as a matrix M, which contains scores for each position i in a motif of W base pair long, with each nucleotide $x_i \in \{A, C, G, T\}^W$ (Figure 1.5 B). By multiplying all the scores for each nucleotide within the motif sequence, it yields a motif score $S(\mathbf{x})$, which indicates the approximated TF-DNA binding specificity at position i:

$$S(\mathbf{x}) = \sum_{j=0}^{W-1} \mathrm{M}(\mathbf{x}, i)\mathrm{S_j}(\mathbf{x}) := -\frac{\Delta G(\mathbf{x})}{k_B T} + \text{const.} \tag{1.1}$$

This additive motif score $\mathrm{S}(\mathbf{x})$ reflects the ratio between Gibbs binding energy $\Delta G(\mathbf{x})$ and the product of the Boltzmann constant $k_B T$, for any potential binding site sequence $\mathbf{x} = x_{1:W} \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^W$.

The PWM matrix $M(\mathbf{x}, i)$ can be determined by the following:

$$\mathrm{M}(\mathbf{x}, i) = -\log \frac{f(\mathbf{x}, i)}{p_{\mathrm{bg}}(\mathbf{x})}, \tag{1.2}$$

where $f(\mathbf{x}, i)$ represents the frequency of nucleotide $\mathbf{x}$ at position i, and $p_{\mathrm{bg}}(\mathbf{x})$ is the frequency of nucleotide x in the background sequences.

PWM is widely applied to represent motif binding preferences. To cope with various new types of approaches and data for protein-DNA binding detection (see the summary A.4), numerous new algorithms have been developed to optimize parameters of PWM in order to get better estimates (see review [30]).

However, PWM has its limitations and may not accurately capture the real binding specificity of transcription factors. One major limit of PWM is that this model assumes the independence of neighboring nucleotides for transcription factor binding. In reality, *in vitro* approaches such as SELEX [31] and protein binding microarrays (PBMs) [32] show that the dependency of nucleotides within the motif does contribute to the binding of some transcription factors since single mutations at one position impact on the interactions at other positions. Besides, the nucleotide dependencies can be partially supported by co-crystal structures of TF-DNA complexes [33]. Therefore, accounting for the nucleotide dependencies within the motif leads to better prediction of the TF binding events, and more comprehensive models are in need to better describe the binding events of TFs and motifs.

Motif sequence logo

PWM can be graphically represented by a sequence logo [34], where all the four bases (A, C, G, T) within the motif are illustrated (Figure 1.5 C). In a motif logo, the height (H_i) of every base at each position i is determined by the product of the frequency of that base $f_i(x)$ and the information content IC_i, or $\log_2 f_i(x)$ at that position i [34]:

$$H_i = \sum_i f_i \times IC_i. \tag{1.3}$$

The information content is measured in bits (maximum $\log_2 4 = 2$) and reflects how much a base diverges from a background distribution of mononucleotide frequencies of the background sequences. However, this simplistic model does not capture the complexity of real sequences such as the GC contents, poly-A, or poly-T repeats.

Till now, the mononucleotide sequence logo is the most popular representation of the DNA/RNA motif. Higher-order logos have been designed to depict higher-order correlations among neighboring bases [35–37].

Shape models

The DNA shape feature is one example of the nucleotide dependencies because the inter-dependent DNA residues initially determine the properties of the DNA structure in the minor groove [8]. Instead of studying the TF-DNA binding using the sequence-specific features purely, some technologies utilize the distinct DNA shape features for motif discovery [38, 39]. The 3D shape features of DNA include minor groove width (MGW), propeller twist (ProT), helix twist (HelT) and roll, etc. Comparing to *k*-mer features (k>3), DNA shape features can assist motif prediction by lowering the dimensionality of feature space for optimization, especially when *k*-mer gets larger (Figure 1.6).

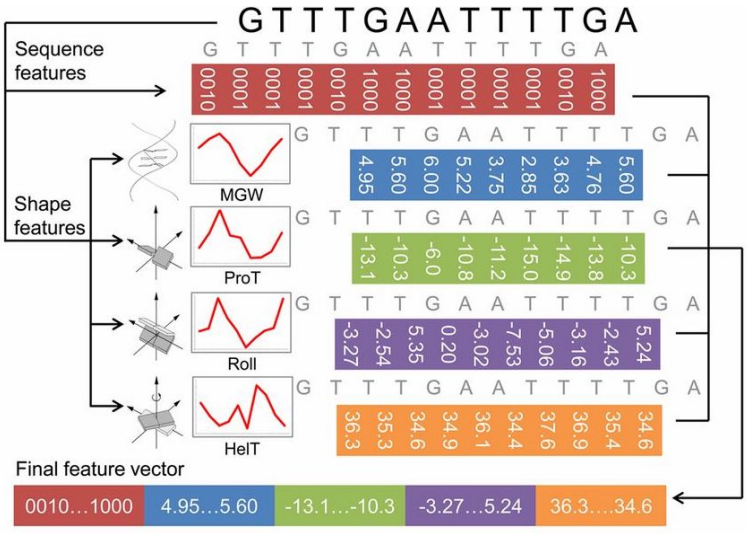


Fig. 1.6 **Design of the shape and sequence feature vector.**
Taken from [39]. In this design, the monomer sequence feature and four shape features (MGW, ProT, HelT and Roll) are combined to predict TF-DNA binding specificity.

Till now, some tools predict DNA shape features [40], while others adopt these predicted shape features, together with PWMs or 1- to 3-mer sequence features, to predict DNA motifs

[41, 42], or predicting motifs from *de novo* [43]. One bottleneck of shape-based models is that they are limited by the pre-defined shape types.

Deep learning models

As more and more genomics data are available, the prediction of TF-DNA binding sites also benefits from the deep neural networks, which can improve prediction accuracy by learning the relevant and complicated features, such as nucleotide correlations in this case. Deep learning was first applied to DNA sequencing data by DeepSea [44], DeepBind [45] and Basset [46]. As an illustration, Figure 1.7 shows how a convolutional neural network (CNN) can be applied for predicting the binding affinity of a motif pair complex and the spacing between them. Since then, deep learning techniques have been applied to various tasks for predictions on genomics data, such as predicting chromatin accessibility, DNA modifications, and genetic variants (see review [47]).

Apart from its ability to learn complex information and to have good predictive accuracy, another significant advantage of deep learning models is that the trained models can be used for rapidly developed as new models on new data via transfer learning. To reuse the trained models, a model repertoire *Kipoi* [49] has been established. However, two major challenges are remaining for deep learning models: (1) the interpretability of the model parameters, and (2) how to avoid biases in training sets.

Higher-order Markov models

Higher-order Markov models can be good candidates for learning the adjacent nucleotide dependencies within the motif. Note that a zeroth-order Markov model (MM) is equivalent to a PWM. For instance, in a Markov model of fixed-order K, the information of K prior bases can be used to predict the probability of the base on $K+1$ position. Then a motif model of length W can be represented by:

$$p_i^{\mathrm{MM}}(x_{1:W}) = \prod_{i=1}^{W} p_i^{\mathrm{MM}}(x_i | x_{i-K:\,i-1}). \tag{1.4}$$

Methods based on higher-order Markov models have been developed [50–53]. However, the bottleneck of fixed higher-order Markov models is that the number of parameters increases exponentially with the order K. For a model of order K, it requires 4^{K+1} parameters to optimize. When the sequences are not sufficient for training, the model is prone to over-fitting.

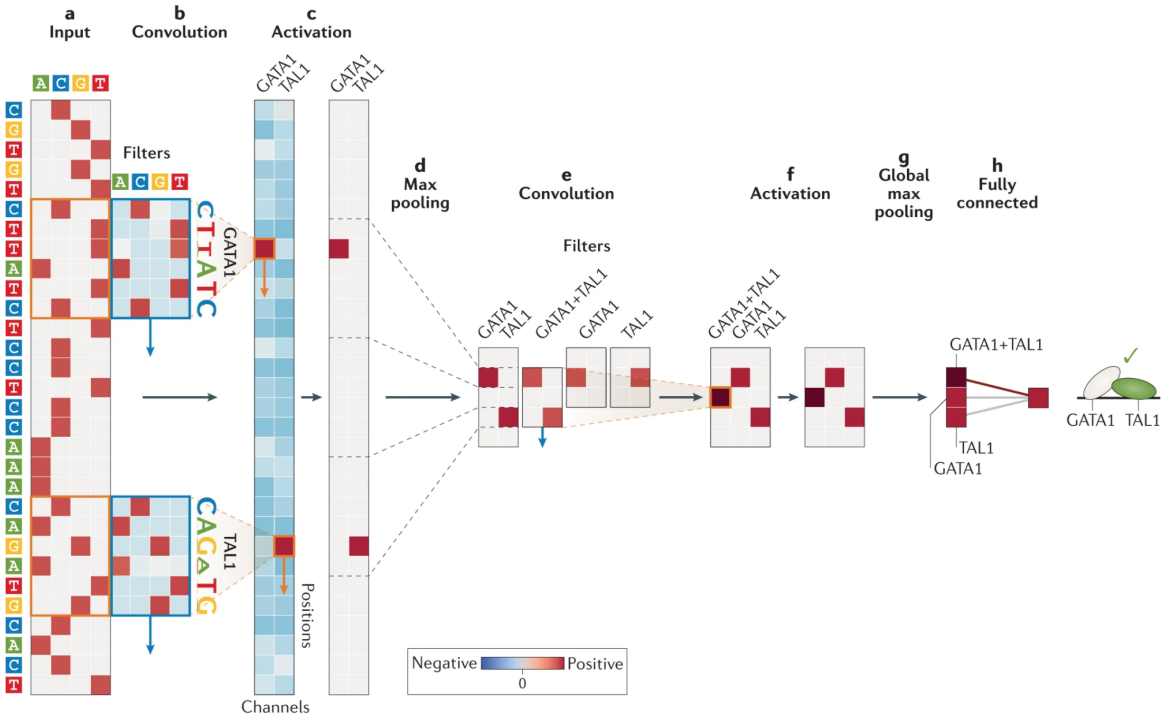


Fig. 1.7 **Training motif models using a convolutional neural network.** Taken from [48]. This shows an example for predicting the binding specificity of GATA1 and TAL1 complex. (**a**) Input DNA sequences are represented by one-hot encoding. (**b**) The sequences are scanned with PWMs of GATA1 and TAL1 motifs. (**c**) Positions with negative values are truncated to 0 using ReLU activation function. (**d**) Matrices are condensed by taking the maximum value in each channel using max-pooling function. (**e**) The sequences are scanned again for the GATA1/TAL1 pair and individual occurrences. (**f**) Similar to (**c**), the ReLU activation function is used. (**g**) Similar to (**d**), positions with the maximum value is chosen for each channel using max-pooling function. (**h**) Finally, a fully connected layer can be applied to make predictions on genomics data.

For overcoming this problem, GLIMMER introduced an *interpolated Markov models* (IMMs) [54, 55]. In contrast to fixed-order Markov models, IMMs are various-order Markov models for which the nucleotide dependencies are not limited to adjacent bases, but can be extended to bases that are a few bases away.

$$
\begin{aligned}
p_i^{\mathrm{IMM}}(x_i|x_{i-K:\,i-1}) = & \lambda_0 \times p(x_i) + \\
& \lambda_1 \times p(x_i|x_{i-1}) + \\
& \ldots + \\
& \lambda_K \times p(x_i|x_{i-K,\ldots,i-1}). \qquad (1.5) \\
\text{where} \sum_i \lambda_i = 1. &
\end{aligned}
$$

For each k-mer ($k \leqslant K$), there is a weight parameter λ_k to control how much different lower-order models contribute to it.

A significant advantage of using an IMM is that it allows counts of oligomers of mixed lengths as lower-order information. For example, there is not enough K-mer information in the data (i.e. K-mer counts are low), the probability $p_{\mathrm{IMM}}(x_i|x_{i-K:\,i-1})$ falls back to lower to zeroth order models, whereas if there is sufficient K-mer information present, $p(x_i|x_{i-K,\ldots,i-1})$ will dominate and thus it will tend to be a fixed Kth-order model.

Our previous in-house tool BaMMmotif expanded the IMMs to the *inhomogenous interpolated Markov models* (iIMMs) [35]. For a Kth-order iIMM, the conditional probability of base x at the position i is calculated by combining the counts of K-mers with pseudo-counts estimated from lower-order probabilities by

$$
p_i^{\mathrm{iIMM}}(x_i|x_{i-K:\,i-1}) = \frac{n_i(x_{i-K:\,i}) + \alpha_K \times p_i^{\mathrm{iIMM}}(x_i|x_{i-K+1:\,i-1})}{n_{i-1}(x_{i-K:\,i-1}) + \alpha_K}, \qquad (1.6)
$$

with the hyper-parameter α_K determining how much weight to assign to the lower-order $p_{\mathrm{iIMM}}(x_i|x_{i-K+1:\,i-1})$.

The advantage of an iIMM is that by interpolating a hyper-parameter α between oligomer counts and pseudo-counts, it does not require prior information about the nucleotide dependencies. Moreover, this probabilistic model can be optimized by applying the expectation-maximization (EM) algorithm. Higher-order BaMMs showed robust performance over PWMs on *in vivo* data in the previous benchmark [35].

1.5 Applications of predictive models

There are four main scenarios where the motif models can be applied to improve our understanding of gene regulation. First, given a set of sequences from DNA-protein binding assays, the *de novo* motif discovery tools can be applied to capture the enriched patterns in

the sequences. Second, given a motif model trained on sequences from one cell line or *in vitro* data, it can be used as seed for initializing the motif discovery in other related data sets. Third, given a set of known motif models and a sequence set, we can scan the sequences for the occurrences of the motifs and identify potential functional transcription factor networks. Forth, by comparing motif models learned from different cell lines, we can interpret the different binding modes of the same transcription factors. There are several online tools and databases developed for these purposes, including our BaMM web server (see Table A.6).

1.6 Aims and contents of this thesis

This project is a continuous work of the previous project [35] on developing higher-order Markov models and optimizing its parameters using Bayesian approaches for predicting the transcription factor binding motifs from high-throughput transcriptomics data. Its major contributions to the further interpretation of the gene regulatory process are presented in three sections:

1. Optimization of parameters and hyper-parameters of the higher-order models.
2. Benchmark state-of-art tools on large-scale *in vivo* and *in vitro* data sets.
3. Construct web server, databases, and develop visualizations for motif models.

Chapter 2

Algorithm and benchmark

Given a set of sequences that are measured by protein-DNA binding assay, e.g. *in vivo* by a ChIP-seq experiment or *in vitro* using SELEX-seq, the goal is to find the binding sites to which the protein of the interest (mostly a transcription factor) has bound. These binding sites are usually enriched in the sequences, compared to those sequences without performing any selection with the target protein binding. These binding sites have common bases $\in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}$ with certain degree of degeneracy (e.g. mismatches) and can be summarized by a probabilistic model. Our task is to derive such a model that can accurately describe the binding preferences of proteins. We also need a *null model* to describe the scenario where no protein-DNA binding is present in the sequences. The challenge for *de novo* motif discovery is that the positions where the motifs occur in the sequence set are not known beforehand. Therefore, training a motif model involves searching for these positions as well as optimizing the model parameters to accurately describe the binding preferences.

In this chapter, I first explain the theory how the TF-DNA binding affinity is approximated by models from simple to complex structures (Section 2.1). Then I show how the Bayesian Markov models are built and how the parameters and hyperparameters are optimized (Section 2.2). Finally, I summarize how the data are prepared (Section 2.3) and how the benchmark are implemented (Section 2.4).

2.1 How to model protein-DNA binding energies?

To deduce motifs from the TF-bound sequences of much longer regions than the actual binding sites without knowing where the bindings are located, we derive a probabilistic model based on the Gibbs free energy $\Delta G(\mathbf{x})$ for any potential binding site $\mathbf{x} = x_{1:W} \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}^W$. This model allows us to make predictions for any arbitrary DNA sequences about where and how strong the TF binds.

Let us assume the DNA sequences of length W, with all DNA bases $x_{1:W} \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}^W$. According to Boltzmann's law, the probability of a genomic site with sequence $\mathbf{x}$ to be bound $p(\text{bound}|\mathbf{x})$ by the transcription factor divided by the probability of $\mathbf{x}$ not to be bound $p(\text{not bound}|\mathbf{x})$ is

$$\frac{p(\text{bound}|\mathbf{x})}{p(\text{not bound}|\mathbf{x})} = \frac{p(\text{bound}|\mathbf{x})}{1-p(\text{bound}|\mathbf{x})} = \exp\left(-\frac{\Delta G(\mathbf{x})-\mu}{k_B T}\right), \tag{2.1}$$

where k_B is the Boltzmann constant, T is the thermodynamic temperature, and μ is the chemical potential which depends purely on the concentration of the transcription factor.

We denote by $p_{\text{bg}}(\mathbf{x})$ the probability distribution of sequences $\mathbf{x} \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}^W$ in the background set from where the binding sequences were selected. For instance, in ChIP-seq, the background set can be the genomic input, a mock immunoprecipitation (without target protein binding), or sampled from the training sequences by a higher-order Markov model, and in HT-SELEX, it can be the input sequence library prior to the selection cycles.

We denote by $p_{\text{motif}}(\mathbf{x})$ the probability distribution of the dependence of $\Delta G(\mathbf{x})$ on the binding site sequence $\mathbf{x}$. Then we have

$$p_{\text{motif}}(\mathbf{x})/p_{\text{bg}}(\mathbf{x}) \propto \exp(-\Delta G(\mathbf{x})/k_B T). \tag{2.2}$$

The proportionality constant is determined by the normalization. Solving for $p_{\text{motif}}(\mathbf{x})$ and normalising yields

$$p_{\text{motif}}(\mathbf{x}) := \frac{p_{\text{bg}}(\mathbf{x})\exp(-\Delta G(\mathbf{x})/k_B T)}{\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y})\exp(-\Delta G(\mathbf{y})/k_B T)}, \tag{2.3}$$

where the sum in the normalisation constant runs over all possible binding sites $\mathbf{y} \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}^W$. We define the motif score $S(\mathbf{x})$ as

$$S(\mathbf{x}) := \log\frac{p_{\text{motif}}(\mathbf{x})}{p_{\text{bg}}(\mathbf{x})} = -\frac{\Delta G(\mathbf{x})}{k_B T} + \text{const.} \tag{2.4}$$

The motif score $S(\mathbf{x})$ gives us the binding strength of a site $\mathbf{x}$ as quantified by the negative Gibbs energy of binding in units of $k_B T \log 2$. Once knowing $p_{\text{motif}}(\mathbf{x})$, we can compute the motif score $S(\mathbf{x})$.

By combining E.q. (2.1) and (2.61), we can have

$$p(\text{bound}|\mathbf{x}) = \frac{e^{S(\mathbf{x})+\mu'}}{1+e^{S(\mathbf{x})+\mu'}} \approx \frac{e^{S(\mathbf{x})}}{1+e^{S(\mathbf{x})}}, \tag{2.5}$$

where μ' is the constant chemical potential and can be dropped for simplicity.

For the motif discovery task, it is essential to find an approximation to the binding probability $p_{\text{motif}}(\mathbf{x})$, so that it can accurately describe the binding preference of the transcription factors, and thus lead us to the appropriate interpretation about the functions of transcription factor. Similarly important is the approximation of the background probability distribution $p_{\text{bg}}(\mathbf{x})$. Many tools have been developed to solve this task by developing models that balance the simplicity and accuracy. In the following subsections, I will introduce several representative models that approximate the binding probability.

2.1.1 Position weight matrix (PWM)

Given the calculation of binding probability $p(\mathbf{x})$ as

$$p(\mathbf{x}) = p(x_1 ... x_W) = p_1(x_1) \times p_2(x_2|x_1) \times p_3(x_3|x_1x_2) \times ... \times p_W(x_W|x_1x_2...x_W). \quad (2.6)$$

PWM assumes that every position is independent from its neighboring positions so that $p(\mathbf{x})$ can be simplifies as

$$p(\mathbf{x}) = p(x_1 ... x_W) \approx \prod_{i=1}^{W} p_i(x_i). \quad (2.7)$$

Given a set of N sequences, $p_i(x_i)$ can be estimated as the frequency of the base x_i at position i, therefore

$$p_i(x_i) \approx \frac{n_i(x_i)}{N} = f_i(x_i). \quad (2.8)$$

When there are sufficient sequences, E.q. (2.8) can be a good estimate of the probability $p_i(x_i)$. For example, when the motif length W is 6, there are $4^6 (= 4096)$ possible 6-mers. The input sequences should observe all the 6-mers in sufficient amount, in order to give a relatively good estimation of the frequencies. When there are limited sequences, $n_i(x_i)$ can be down to zero and thus cannot reflect $p_i(x_i)$ well. To take that into consideration, it is a common strategy to introduce *pseudo-counts* to balance the information and the noise. In this case, the pseudo-counts are the product of the background frequency $f_{\text{bg}}(x_i)$ and a pseudo-factor (or a hyperparameter) α:

$$p_i(x_i) = \frac{n_i(x_i) + \alpha \times f_{\text{bg}}(x_i)}{N + \alpha}. \quad (2.9)$$

The pseudo-factor α can be a fixed positive number.

When there is sufficient data, $n_i(x_i)$ will dominate over pseudo-counts, and thus $p_i(x_i)$ is approximated as the foreground frequency $f_i(x_i)$. When $n_i(x_i)$ is significantly small, the $p_i(x_i)$ is very close to the background frequency $f_{\text{bg}}(x_i)$.

For a typical PWM, there are $(4-1)\times W(=3W)$ parameters to learn. Despite its model simplicity, a PWM does not learn the nucleotide dependency, which is found to be an important feature for most of the TF-DNA bindings.

2.1.2 Pattern-based motif discovery tool (PEnGmotif)

We introduce PEnGmotif (Pattern-based discovery of enriched genomic or transcriptomic sequence motifs), a tool which learns motifs in PWMs to represent the enriched patterns.

The key idea of PEnGmotif is to find the enriched DNA patterns in the sequences over random expectations from a second-order background model. It first uses an enumerative approach to exhaustively count all the possible non-degenerate W-mers with a fixed length. One big advantage of such an enumerative approach is that it covers the motif space to a large extend efficiently. To get the probability distribution of W-mers observed by chance in the negative set, a background model is needed. The simplest model is built upon the mononucleotide frequencies, similar to E.q. (2.7) and (2.8) (note: here a W-mer from the background set is denoted as $\mathbf{y}$ to distinguish it from the positive set):

$$p_{\text{bg}}(\mathbf{y}) \approx \prod_{i=1}^{W} p_i(y_i) \approx \prod_{i=1}^{W} f_i(y_i). \tag{2.10}$$

However, this model does not account for sequence features such as dinucleotide CG repeats, or poly-A or poly-T sequences, which are commonly present in most non-coding regions of the genome.

Model background distribution with interpolated homogeneous Markov models

To take into account the nucleotide dependencies in the background sequences where no binding events occur, there are a few approaches used in this field. One approach is to use the shuffled k-mers to construct negative sequence set. Another approach is to build the background model using higher-order Markov models. For SELEX-like experiments, there are library sequences which can be used as background sequences. Here I explain the higher-order background models are built upon a kth-order homogeneous (that is, position-nonspecific) Markov model for the pattern with a length of W:

$$p_{\text{bg}}(\mathbf{y}) \approx p(y_1 \ldots y_{k+1}) \prod_{i=k+1}^{W} p(y_i | y_{i-k} \ldots y_{i-1}). \tag{2.11}$$

To reduce the amount of parameters for training, we use an interpolated Markov model (first introduced by [54]) as E.q. (2.9):

$$p(y_{k+1}|y_1...y_k) \approx \frac{n(y_1...y_k) + \alpha_k \times p(y_{k+1}|y_2...y_k)}{n(y_1...y_k) + \alpha_k}. \tag{2.12}$$

with the order-specific pseudo-factor α_k as follows:

$$\alpha_k = \begin{cases} 1, & \text{if} \quad k = 0, \\ \beta \times \gamma^{k-1}, & \text{if} \quad k > 0. \end{cases}$$

with $\beta = 20$ and $\gamma = 3$ as chosen by [35], which improve the performance of using solely the k-mer frequencies and keep robust with increasing motif orders.

We choose the model order k to be 2 for the background models. Because if the order is lower, it does not capture the genomic features such as the CpG islands; and for the higher orders it is prone to over-fitting.

Calculate P-values for W-mers

To check whether a W-mer is enriched in the given sequences over the random expectation, one method is to compute its P-value.

The number of occurrences of W-mer $\mathbf{y}$ should follow a Poisson distribution with expectation value $\mu = L_{\text{tot}} p_{\text{bg}}(\mathbf{y})$, where $L_{\text{tot}} = \sum_{n=1}^{N}(L_n - W + 1)$ is the total number of positions in the input sequences, and $p_{\text{bg}}(\mathbf{y})$ is the probability of k-mer $\mathbf{y}$ according to the kth-order background model as computed previously (E.q. (2.11) and (2.12)).

For $n(\mathbf{y}) \gg 1$ and $n(\mathbf{y}) > \mu$, which is always fulfilled anyway when a motif is significantly over-represented, the P-value can be approximated using Stirling's approximation by:

$$\begin{aligned}
\text{P-value}(\mathbf{y}) &= \sum_{k=n}^{\infty} \frac{\mu^k}{k!} e^{-\mu} \\
&= \frac{\mu^n}{n!} e^{-\mu} \sum_{k=0}^{\infty} \frac{\mu^k}{(n+1)\cdots(n+k)} \\
&\lessapprox \frac{\mu^n}{n!} e^{-\mu} \sum_{k=0}^{\infty} \frac{\mu^k}{(n+1)^k} \\
&\approx \frac{\mu^{n(\mathbf{y})}}{n(\mathbf{y})!} e^{-\mu} \frac{1}{1 - \mu/(n(\mathbf{y})+1)} \\
\log \text{P-value}(\mathbf{y}) &\approx n(\mathbf{y}) \log \frac{\mu}{n(\mathbf{y})} + n(\mathbf{y}) - \mu - \frac{1}{2}\log(2\pi n(\mathbf{y})) - \log\left(1 - \frac{\mu}{n(\mathbf{y})+1}\right).
\end{aligned} \tag{2.13}$$

Calculate Z-values for W-mers

We then compute squared Z-scores for all non-degenerate W-mers. The Z-scores are simply the deviation from expectation divided by the expected standard deviation. The standard deviation of a Poisson distribution is equal to the square root of its mean, therefore:

$$Z(\mathbf{y}) = \frac{n(\mathbf{y}) - L_{\text{tot}} p_{\text{bg}}(\mathbf{y})}{\sqrt{L_{\text{tot}} p_{\text{bg}}(\mathbf{y})}}. \tag{2.14}$$

Z-scores are used later for comparing W-mers to find the optimal ones.

Find local optimal W-mers

To reduce the amount of non-degenerate patterns and select the representative ones around their neighbours (i.e., those that are at most one substitution away), we apply a recursive function which takes a W-mer $\mathbf{y}$ and checks for all its neighbouring W-mers. If it finds a neighbour $\mathbf{y}_{\text{neigh}}$ with a better Z-score, the function is called recursively with $\mathbf{y}_{\text{neigh}}$ as an argument. Otherwise, if no neighbour of $\mathbf{y}$ has better Z-score than $\mathbf{y}$, $\mathbf{y}$ is appended to the list of locally optimal W-mers. By doing this, the number of enriched W-mers are reduced for further computation.

Transform W-mers to degenerate IUPAC patterns

From the local optimal non-degenerate W-mers, we allow some flexibility by replacing the bases $\in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}$ with IUPAC letters $\in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T},\mathrm{S},\mathrm{W},\mathrm{R},\mathrm{Y},\mathrm{M},\mathrm{K},\mathrm{N}\}$ (see Table A.1). Similarly to the previous step, we apply an iterative greedy search for local optimal IUPAC patterns by comparing them with their neighbouring substitutions with regard to their P-values or Z-values.

Convert degenerate IUPAC patterns to PWMs

To derive a position weight matrix (PWM) from an IUPAC pattern $\mathbf{y}$, the simplest way is to count the occurrence of nucleotide a at position j within the motif in the matched sequences, for all $\mathbf{a} \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}$ at all positions.

The probabilities of the PWM are then calculated as:

$$p_{\text{pwm}}(\mathbf{y}) = \prod_{j=1}^{W} p_{ja} = \prod_{j=1}^{W} \frac{n_{ja}}{\sum_{b}^{\mathrm{A,C,G,T}} n_{jb}}. \tag{2.15}$$

A drawback of this method is when the amino acid a is excluded at a certain position j by an IUPAC letter y_j, n_{ja} can be zero and so does p_{ja} equal to zero, which is not allowed in the PWM.

To avoid this from happening, we could use pseudo-counts. But there is a smarter way. It relies on the insight that if we allow any of the four nucleotides at position j, the vast majority of motif matches will still be true positives due to the descriptive power of the other $W-1$ IUPAC letters. Therefore, we count the four nucleotides at motif position j for matches to the pattern $y_{0:j-1}\mathrm{N}y_{j+1:W-1}$ in which we replace the j'th IUPAC letter by an N:

$$p_{ja} = \frac{n(y_{1:j}\,a\,y_{j+2:W})}{n(y_{1:j}\,\mathrm{N}\,y_{j+2:W})}, \tag{2.16}$$

where we denote by $n(\mathbf{y})$ the number of occurrences of W-mer $\mathbf{y}$ in the input set. Note that these PWM probabilities can be computed solely from the W-mer counts in a time $O(W \times D)$ that is independent of the size of the input data set L_{tot} and only depends on the degeneracy $D = \left|\{\mathbf{x} \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}^W : \mathbf{x} \text{ matches } \mathbf{y}\}\right|$ of the motif $\mathbf{y}$, i.e., the number of different W-mers it matches.

After efficiently getting PWMs, we refine the models using expectation maximization (EM) algorithm, which will be explain in section 2.2.5. PWMs with overlaps are merged and extended, and can serve as seeds to be refined to higher-order Markov models using Bayes' rules (explained in the following section 2.2.1).

2.1.3 Higher-order Bayesian Markov model (BaMM)

As mentioned in the introduction section, higher-order Bayesian Markov model (BaMM) adopts the interpolation approach that was first introduced by [54] to control the information flows from variable bases prior to the current base. On top of the interpolated Markov model, BaMM uses a pseudo-factor α to balance between the $(K+1)$-mer counts and pseudo-counts from the lower K-mers as:

$$p_i^{\mathrm{BaMM}}(x_i|x_{i-K:\,i-1}) = \frac{n_i(x_{i-K:\,i}) + \alpha_K \times p_i^{\mathrm{BaMM}}(x_i|x_{i-K+1:\,i-1})}{n_{i-1}(x_{i-K:\,i-1}) + \alpha_K}. \tag{2.17}$$

Same as for the homogeneous background model (E.q. (2.12)), the α_K is chosen as 1 when $K=0$, and $\beta \times \gamma^{K-1}$ when $K>0$. α_K increases when the order K gets larger, which indicates that the influence of prior bases generally decrease with longer distance. For the previous version of BaMMmotif, Matthias et al. [35] tried different β and γ combinations and found that $\beta = 30$ and $\gamma = 3$ lead to relatively good performance and robust to over-fitting.

However, some transcription factors, especially those with multiple DNA-binding domains, can recognize nucleotides that are a few bases away. Therefore, order- and position-specific α_{kj} (with j as the position within the motif with order k) shall capture such features better and thus lead to better motif performance than using uniformly distributed αs. The learning process of the position-specific α_{kj} is described in Section 2.2.6.

2.2 How to train a Bayesian Markov model?

2.2.1 Bayes rules and log likelihood

We use the model likelihood to denote the probability that the observed data (in our case, it is the sequence data $\mathbf{x}$) could have been generated by the motif model (denoted as $\mathbf{m}$). In the process of model optimization, one tries to optimize the logarithm of this probability, or *log likelihood* (*LL*) of a model, with respect to the model parameters:

$$LL \approx \log P(\mathbf{x}|\mathbf{m}) = \sum_i \log P(x_i|\mathbf{m}), \tag{2.18}$$

where $x_i \in \mathbf{x}$ (namely, each sequence x_i in the given sequence set $\mathbf{x}$). The log likelihood is a rough approximation of the binding affinity and it is proportional to the information content that is represented by the motif logo.

Given some prior information about the motif, such as a few binding sites as the seed, we can apply *Bayes theorem* to get an optimal motif model via:

$$P(\mathbf{m}|\mathbf{x}) = \frac{P(\mathbf{x}|\mathbf{m}) \times P(\mathbf{m})}{P(\mathbf{x})}. \tag{2.19}$$

By transforming it to the logarithmic space gives:

$$\log P(\mathbf{m}|\mathbf{x}) \propto \log P(\mathbf{x}|\mathbf{m}) + \log P(\mathbf{m}). \tag{2.20}$$

Here, $P(\mathbf{m})$ is the prior knowledge of the motif model *before* we observe any data and is often initialized by a few enriched DNA sites. $P(\mathbf{x}|\mathbf{m})$ is the model likelihood, same as it in E.q. (2.26). $P(\mathbf{m}|\mathbf{x})$ is the posterior distribution is the probability distribution that is obtained *after* we have observed $\mathbf{x}$. $P(\mathbf{m}|\mathbf{x})$ is the estimated model maximizes the likelihood $P(\mathbf{x}|\mathbf{m})$ using the method of maximum a posteriori (MAP) estimation. It can then be treated as a new prior and applied to E.q. (2.19) and thus iteratively we can optimize the model parameters to which give the optimal likelihood.

2.2.2 Likelihood in weak binding approximation

Given sequences $\mathbf{x}_1, \ldots, \mathbf{x}_N$ of lengths L_n, the task is to discover motifs of length W enriched in them. However, the positions of the potential motifs on the sequences are unknown. To learn the distribution $p_{\text{motif}}(\cdot)$, and therefore also the free binding energy $\Delta G(\mathbf{x})$, from the measured binding sites, we need the likelihood:

$$p(\mathbf{x}_1 \ldots \mathbf{x}_N|\mathbf{m}) = \prod_{n=1}^{N} p(\mathbf{x}_n|\text{bound}, \mathbf{m}) \tag{2.21}$$

of the binding sites given the model parameters $\mathbf{m}$.

We can estimate the probability $p(\mathbf{x}_n|\text{bound}, \mathbf{m})$ of obtaining a training sequence $\mathbf{x}_n$ through its binding to a transcription factor out of a library of possible sequences $\mathbf{x}_1 \ldots \mathbf{x}_N$ described by $p_{\text{bg}}(\mathbf{y})$ by

$$\begin{aligned} p(\mathbf{x}_n|\text{bound}, \mathbf{m}) &= \frac{p(\text{bound}|\mathbf{x}_n, \mathbf{m})\, p_{\text{bg}}(\mathbf{x}_n)}{p_{\text{bg}}(\mathbf{y}) \sum_{\mathbf{y}} p(\text{bound}|\mathbf{y}, \mathbf{m})} \\ &= \frac{p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L_n - W + 1} p_i(\text{bound}|\mathbf{x}_n, \mathbf{m})}{\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y}) \sum_{i=1}^{L_n - W + 1} p_i(\text{bound}|\mathbf{y}, \mathbf{m})} \end{aligned} \tag{2.22}$$

Here, $p_i(\text{bound}|\mathbf{x}_n, \mathbf{m})$ is the probability that $\mathbf{x}_n$ is bound by a factor whose binding site starts at position i in the sequence. Because of steric hindrance two factors cannot bind nearer than approximately W nucleotides from each other. Therefore the probability for the factor to bind at i can depend on the probabilities of binding at other positions.

In a regime of unsaturated binding, we can assume that $p(\text{bound}|\mathbf{x}) \lesssim 0.1$. We can then approximate E.q. (2.5) as $p_i(\text{bound}|\mathbf{x}, \mathbf{m}) \approx \exp\left(S(x_{i:i+W-1}) + \mu\right)$. Inserting this expression into E.q. (2.22) yields

$$\begin{aligned} p(\mathbf{x}|\text{bound}, \mathbf{m}) &= \frac{p_{\text{bg}}(\mathbf{x}) \sum_{i=1}^{L-W+1} p_i(\text{bound}|\mathbf{x}, \mathbf{m})}{\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y}) \sum_{i=1}^{L-W+1} p_i(\text{bound}|\mathbf{y}, \mathbf{m})} \\ &= \frac{p_{\text{bg}}(\mathbf{x}) \sum_{i=1}^{L-W+1} e^{S(x_{i:i+W-1}) + \mu}}{\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y}) \sum_{i=1}^{L-W+1} e^{S(y_{i:i+W-1}) + \mu}} \\ &= \sum_{i=1}^{L-W+1} p_{\text{bg}}(x_{1:i-1}) \frac{p_{\text{bg}}(x_{i:i+W-1}|x_{1:i-1})\, e^{S(x_{i:i+W-1})}}{\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y}) \sum_{i=1}^{L-W+1} e^{S(y_{i:i+W-1})}} p_{\text{bg}}(x_{i+W:L}) \\ &\approx \sum_{i=1}^{L-W+1} p_{\text{bg}}(x_{1:i-1}) \frac{p_{\text{bg}}(x_{i:i+W-1})\, e^{S(x_{i:i+W-1})}}{\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y}) \sum_{i=1}^{L-W+1} e^{S(y_{i:i+W-1})}} p_{\text{bg}}(x_{i+W:L}). \end{aligned} \tag{2.23}$$

The denominator in the sum can be simplified by realising that the sums over nucleotides:

$$\begin{aligned}
&\sum_{\mathbf{y}} p_{\mathrm{bg}}(\mathbf{y}) \sum_{i=1}^{L-W+1} e^{S(y_{i:i+W-1})} \\
&= \sum_{i=1}^{L-W+1} \sum_{y_{1:i-1}} p_{\mathrm{bg}}(y_{1:i-1}) \sum_{y_{i:i+W-1}} p_{\mathrm{bg}}(y_{i:i+W-1}|y_{1:i-1}) e^{S(y_{i:i+W-1})} \cancel{\sum_{y_{i+W:L}} p_{\mathrm{bg}}(y_{i+W:L}|y_{1:i+W-1})} \\
&\approx \sum_{i=1}^{L-W+1} \sum_{y_{i:i+W-1}} p_{\mathrm{bg}}(y_{i:i+W-1}) e^{S(y_{i:i+W-1})} \\
&= (L-W) \sum_{y_{1:W}} p_{\mathrm{bg}}(y_{1:W}) e^{S(y_{1:W})}
\end{aligned} \tag{2.24}$$

Inserting this into the previous equation and using E.q. (2.3) gives:

$$\begin{aligned}
p(\mathbf{x}|\mathrm{bound},\mathbf{m}) &\approx \frac{1}{L-W} \sum_{i=1}^{L-W+1} p_{\mathrm{bg}}(x_{1:i-1})\, p_{\mathrm{motif}}(x_{i:i+W-1})\, p_{\mathrm{bg}}(x_{i+W:L}) \\
&= p_{\mathrm{bg}}(\mathbf{x}) \frac{1}{L-W} \sum_{i=1}^{L-W+1} \frac{p_{\mathrm{motif}}(x_{i:i+W-1})}{p_{\mathrm{bg}}(x_{i:i+W-1})} \\
&= p_{\mathrm{bg}}(\mathbf{x}) \frac{1}{L-W} \sum_{i=1}^{L-W+1} e^{S(x_{i:i+W-1})}.
\end{aligned} \tag{2.25}$$

The sequence sets used for training might not all have been bound directly by the factor of interest. One reason is that they can be bound other co-factors of the factor of our interest, or they are transferred to the immunoprecipitated fraction bound non-specifically to some tube or bead surfaces. To account for that unbound sequences are always present in the training set, we assume that a fraction q of the sequences are specifically bound to the factor.

The likelihood thus can be calculated by:

$$p(\mathbf{X}|\mathbf{m}) = \prod_{n=1}^{N} \left[p_{\mathrm{bg}}(\mathbf{x}_n) \left(1 - q + \frac{q}{L-W} \sum_{i=1}^{L-W+1} \frac{p_{\mathrm{motif}}(x_{n,i:i+W-1})}{p_{\mathrm{bg}}(x_{n,i:i+W-1})} \right) \right]. \tag{2.26}$$

This equation for the likelihood applies to any choice of models for the binding site and background sequences in the regime of unsaturated, weak binding. It is remarkable because it shows that the statistical physics approach to learning a binding energy model that explains the observed binding data leads to the same likelihood as the purely statistical approach using the "zero or one occurrence per sequence" (ZOOPS) model. However, for the "multiple occurrences per sequence" (MOPS) model, there is no straightforward thermodynamic justification yet.

2.2.3 The prior probability distributions

2.2.3.1 The prior on model parameters *m*

As prior probability distribution $p(\mathbf{m}|\mathbf{m}^*,\alpha)$ we choose a Dirichlet (Beta) distribution with pseudo-count parameters αs coming from the lower order,

$$p(\mathbf{m}|\mathbf{m}^*,\alpha) = \prod_{j=0}^{W-1}\prod_{y_{1:K}}^{4} \mathrm{Dir}\left(\mathbf{m}(\cdot|y_{1:K})|\alpha_{Kj}\mathbf{m}_j^*(\cdot|y_{2:K})\right), \tag{2.27}$$

where the Dirichlet distribution is:

$$\begin{aligned}&\mathrm{Dir}\left(\mathbf{m}(\cdot|\mathbf{y})|\alpha_{Kj}\mathbf{m}_j^*(\cdot|\mathbf{y}')\right)\\ &= \frac{\Gamma(\alpha_{Kj})}{\prod_{a=1}^{4}\Gamma(\alpha_{Kj}\mathbf{m}_j^*(a|\mathbf{y}'))}\prod_{a=1}^{4}\mathbf{m}_j(a|\mathbf{y})^{\alpha_{Kj}\mathbf{m}_j^*(a|\mathbf{y}')-1}\,\delta\left(1-\sum_{a=1}^{4}\mathbf{m}_j(a|\mathbf{y})\right),\end{aligned} \tag{2.28}$$

with $\mathbf{y} = y_{1:K}$, $\mathbf{y}' = y_{2:K}$, a as the nucleotide base on position j, and $\Gamma(\alpha_{Kj})$ is a Gamma function defined in E.q. (2.29).

This choice of prior leads to a type of interpolated Markov model.

2.2.3.2 The prior on hyperparameters α_{kj}

We choose as prior on the hyperparameters α_{kj} (for $1 \le k \le K$) an inverse Gamma distribution with parameters 1 and $(\beta\gamma^k)$,

$$p(\alpha_{kj}|\beta,\gamma) = \frac{\beta\,\gamma^k}{\alpha_{kj}^2}e^{-\beta\,\gamma^k/\alpha_{kj}}, \tag{2.29}$$

where $\beta \approx 5$ and $\gamma = 3$ corresponds roughly to Matthias' choice $\alpha_{kj} = \beta\,\gamma^k = 20\times 3^{k-1}$ that worked for all of the data sets in the previous paper [35]. By this definition of the inverse Gamma distribution, the reciprocal α_{kj}^{-1} is distributed according to a Gamma distribution with parameters 1 and $\beta\gamma^k$, which is an exponential with mean $(\beta\gamma^k)^{-1}$. The mean and variance of the prior on α_{kj} are infinite, but the mode is $\beta\gamma^k/2$. Hence this prior very softly pushes the α_{kj} towards $\beta\gamma^k/2$ and barely restrains them in assuming large positive values.

2.2.3.3 The positional prior $p(z_n)$

We choose a flat positional preference prior,

$$\begin{aligned} p(z_n = 0) &= 1 - q \quad \text{(signifies "no motif present")} \\ p(z_n = i) &= \frac{q}{L_n - W + 1}, \text{for} 1 \leq i \leq L_n - W + 1. \end{aligned} \tag{2.30}$$

The hyperparameter q specifies the probability for a sequence to contain a motif. In a thermodynamic interpretation, $\log q$ corresponds to a global shift of ΔG in units of $k_B T$ and $\log p(z_n = i)$ corresponds to a position-dependent shift in binding energy ΔG. Hence we assume here no positional dependence of binding energy. Alternatively, the positional preference profile can be learned from the data, as we will show later (Section 2.2.6).

2.2.4 The posterior probability distribution

Given that the posterior is the product of the likelihood and the prior, normalized with a normalization constant, according to Bayes' rules (E.q. (2.19)), we can get the posterior probability distribution as:

$$\begin{aligned} p(\mathbf{m}, \alpha | \mathbf{X}, q, \mathbf{m}^*) &\propto p(\mathbf{X} | \mathbf{m}, \alpha, q)\, p(\mathbf{m} | \mathbf{m}^* \alpha)\, p(\alpha) \\ &\propto \left(\prod_{n=1}^{N} p(\mathbf{x}_n | \mathbf{m}, \alpha, q) \right) p(\mathbf{m} | \mathbf{m}^* \alpha)\, p(\alpha) \\ &\propto \left(\prod_{n=1}^{N} \sum_{i=1}^{L_n - W + 1} p(\mathbf{x}_n | z_n = i, \mathbf{m}, \alpha) p(z_n = i | q) \right) p(\mathbf{m} | \mathbf{m}^* \alpha)\, p(\alpha) \end{aligned} \tag{2.31}$$

with q as the fraction of sequences that are specifically bound with the transcription factor.

2.2.5 Maximum likelihood algorithm

To obtain the maximum likelihood (ML) solutions for the probabilistic model with latent variables, the expectation maximization (EM) algorithm is a general solution for it [56]. The EM algorithm was first introduced by [50] to the motif discovery field and has been widely adopted for motif finding tools, including the popular MEME Suite [57].

The EM algorithm iterates between the estimation step (E-step) and the maximization step (M-step), till the optimal (or convergence) is reached.

Given a roughly estimated model $\mathbf{m}^{\text{old}}$, in the E-step, it calculates the probability of each site based on the current motif model $\mathbf{m}$, and in the M-step, it re-estimates a new motif model $\mathbf{m}^{\text{new}}$ based on the probabilities $p_{\text{motif}}(\mathbf{m})$. It is similar to a gradient descent procedure, which converges to a maximum of the log likelihood of the resulting model (E.q. (2.26)).

The training data set from input sequences are denoted by $\mathbf{x}$. $\mathbf{z}$ represents motif positions on the sequences and α is the pseudo-factor as the latent variables. The model parameters are denoted as $\mathbf{m}$. The process of the EM algorithm can be visualized in the space of parameters as Figure 2.1.

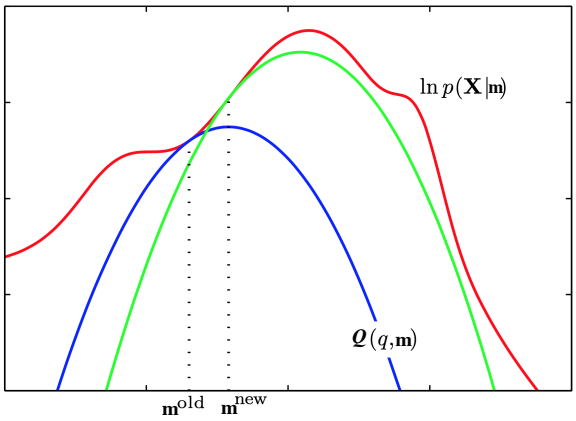


Fig. 2.1 **The EM algorithm in the parameter space.**
Adapted from [58]. See the text below for a full description.

1. **Initialization**: Get an initial model $\mathbf{m}^{\text{old}}$ with parameters and latent variables $\mathbf{z}$,

2. **E-step**: Estimate the posterior distribution of the $p(\mathbf{z}|\mathbf{x},\mathbf{m}^{\text{old}})$, which gives rise to a lower bound $Q(\mathbf{m},\mathbf{m}^{\text{old}})$ whose value equals to the log likelihood at $\mathbf{m}^{\text{old}}$, as the blue curve in Figure 2.1,

3. **M-step**: Get a new motif model $\mathbf{m}^{\text{new}}$ by maximizing the auxiliary function Q:

$$Q(\mathbf{m}|\mathbf{m}^{\text{old}}) = \sum_{\mathbf{z}} P(\mathbf{z}|\mathbf{x},\mathbf{m}^{\text{old}}) \log P(\mathbf{x},\mathbf{z}|\mathbf{m}) \tag{2.32}$$

   to get a new set of model parameters $\mathbf{m}^{\text{new}}$ by

$$\mathbf{m}^{\text{new}} = \arg\max_{\mathbf{m}} Q(\mathbf{m}|\mathbf{m}^{\text{old}}) \tag{2.33}$$

   The subsequent E-step then constructs a bound that is tangential at $\mathbf{m}^{\text{new}}$ as shown by the green curve.

4. **Check point**: Check for convergence of the log likelihood (the red curve).

   If the convergence criterion has not been met, then let

$$\mathbf{m}^{\text{old}} \leftarrow \mathbf{m}^{\text{new}} \tag{2.34}$$

   and return to **step 2**;

   else exit with the model with optimal parameters $\mathbf{m}^{\text{optimal}}$.

To start with the EM for the Bayesian Markov model, I have described the prior distribution of the model $\mathbf{m}$, given the latent (hidden) variables $\mathbf{z}$ for motif positions and α as pseudo-factor (E.q. (2.27)). The likelihood function is described by E.q. (2.26). The posterior probability distribution can thus be calculated using Bayes' theorem (E.q. (2.19)).

Therefore, we can write down this to find the answer:

E-step: the approximation of posterior distribution:

$$r_{ni} = p(z_n = i|\mathbf{x}_n, \mathbf{m}, \alpha) = \frac{p(\mathbf{x}_n|z_n = i, \mathbf{m})\, p(z_n = i)}{\sum_{i'=0}^{L_n - W + 1} p(\mathbf{x}_n|z_n = i', \mathbf{m})\, p(z_n = i')} \tag{2.35}$$

M-step: the auxiliary function Q:

$$\begin{aligned}
Q(\mathbf{m}, \alpha, q|\mathbf{r}, \mathbf{m}^*) &= \sum_{n=1}^{N} \left[\sum_{i=0}^{L_n - W + 1} r_{ni} \log \left(p(\mathbf{x}_n|z_n = i, \mathbf{m})\, p(z_n = i|q) \right) \right] + \log p(\mathbf{m}|\mathbf{m}^*, \alpha) \\
&= \sum_{n=1}^{N} \sum_{i=0}^{L_n - W + 1} r_{ni} \log p(\mathbf{x}_n|z_n = i, \mathbf{m}) \\
&+ \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + (1 - r_{n,0}) \log \frac{q}{L_n - W + 1} \right) \\
&+ \log p(\mathbf{m}|\mathbf{m}^*, \alpha) \\
&+ \log p(\alpha) \\
&+ \log p(q).
\end{aligned} \tag{2.36}$$

In the E-step, we estimate the responsibilities r_{ni} of the binding site x_n occurring at position i on each sequence n, given the priors on model parameters and hyperparameters. In the M-step, we optimize the model parameters $\mathbf{m}$ and hyperparameters α, q and $\mathbf{z}$ to maximize the auxiliary function Q.

2.2.6 Collapsed Gibbs sampling

The model parameter **m** and motif position distribution **z** are coupled in the models above (E.q. 2.31), which makes it difficult to optimize the probability of **z**. A standard practice in conducting Bayesian inference is to integrate out the nuisance parameter. A similar strategy was developed for learning the parameters of *latent Dirichlet allocation* models [59], where it was also called *collapsed Gibbs sampling*. Therefore, here we integrate out analytically the model parameters **m** and only sample **z** and q. As illustrated in Figure 2.2, by integrating out the **m** (as like *collapsing down* the parameter space on to the z-axis), we get the complete parameter space for **z** to sample using Gibbs sampling approach.

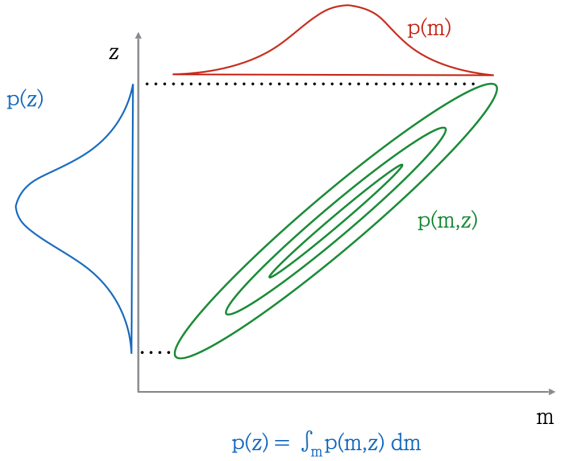


Fig. 2.2 **The collapsed Gibbs sampling algorithm in the parameter space.**

The likelihood of the sequences **X** given the motif positions **z** and the model parameters **m** is proportional to

$$p(\mathbf{X}|\mathbf{z},\mathbf{m}) \propto \prod_{n:z_n>0}^{N} \prod_{j=0}^{W-1} \frac{m_j\left(x_{n,z_n+j}|x_{n,z_n+j-K:\,z_n+j-1}\right)}{m_{\text{bg}}\left(x_{n,z_n+j}|x_{n,z_n+j-K':\,z_n+j-1}\right)}. \tag{2.37}$$

To simplify this expression, we define the k-mer counts (for $1 \leq k \leq K+1$),

$$n_j^{\mathbf{z}}(y_{1:k}) := \sum_{n=1}^{N} I\left(x_{n,z_n+j-k+1:z_n+j} = y_{1:k}\right), \tag{2.38}$$

i.e., the number of times k-mer $y_{1:k}$ has been observed with its rightmost nucleotide at position j of a motif. We obtain for the likelihood:

$$p(\mathbf{X}|\mathbf{z},\mathbf{m}) \propto \prod_{y_{1:K+1}} \prod_{j=0}^{W-1} \left(\frac{m_j(y_{K+1}|y_{1:K})}{m_{\text{bg}}(y_{K+1}|y_{1:K})} \right)^{n_j^{\mathbf{z}}(y_{1:K+1})}. \tag{2.39}$$

We will now integrate out the parameters $\mathbf{m} = (m(a|\mathbf{y}))$ in the likelihood in order to apply Gibbs sampling to draw samples directly from the posterior distribution over $(\mathbf{z}, \alpha, q)$. In the second line we will use the Dirichlet prior on $\mathbf{m}$ from E.q. (2.27) and (2.28):

$$\begin{aligned}
p(\mathbf{X}|\mathbf{z},\alpha,\mathbf{m}^*) &= \int p(\mathbf{X}|\mathbf{z},\mathbf{m})\, p(\mathbf{m}|\mathbf{m}^*,\alpha)\, d\mathbf{m} \\
&\propto \int \prod_{j=0}^{W-1} \prod_{\mathbf{y}} \left(\frac{\Gamma(\alpha_{kj})}{\prod_a \Gamma(\alpha_{kj} m_j^*(a|\mathbf{y}'))} \prod_{a=1}^{4} \left(\frac{m_j(a|\mathbf{y})}{m_{\text{bg}}(a|\mathbf{y})} \right)^{n_j^{\mathbf{z}}(\mathbf{y},a)} m_j(a|\mathbf{y})^{\alpha_{kj} m_j^*(a|\mathbf{y}')-1} \delta\left(1 - \sum_{a=1}^{4} m_j(a|\mathbf{y}) \right) \right) d\mathbf{m} \\
&= \prod_{j=0}^{W-1} \prod_{\mathbf{y}} \frac{\Gamma(\alpha_{kj})}{\prod_a \Gamma(\alpha_{kj} m_j^*(a|\mathbf{y}'))} \frac{1}{\prod_{a=1}^{4} m_{\text{bg}}(a|\mathbf{y})^{n_j^{\mathbf{z}}(\mathbf{y},a)}} \int\limits_{\sum_a^4 m_j(a|\mathbf{y})=1} \prod_{a=1}^{4} m_j(a|\mathbf{y})^{n_j^{\mathbf{z}}(\mathbf{y},a)+\alpha_{kj} m_j^*(a|\mathbf{y}')-1} d^4 m_j(\cdot|\mathbf{y}).
\end{aligned} \tag{2.40}$$

The integrals can be solved by noting that the second integrand is a Dirichlet distribution up to a constant,

$$\begin{aligned}
&\int\limits_{\sum_{a=1}^4 m_j(a|\mathbf{y})=1} \prod_{a=1}^{4} m_j(a|\mathbf{y})^{n_j^{\mathbf{z}}(\mathbf{y},a)+\alpha_{kj} m_j^*(a|\mathbf{y}')-1} d^4 m_j(\cdot|\mathbf{y}) \\
&\quad = \frac{\prod_{a=1}^{4} \Gamma(n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} m_j^*(a|\mathbf{y}'))}{\Gamma(n_{j-1}^{\mathbf{z}}(\mathbf{y}) + \alpha_{kj})} \cancel{\int \text{Dir}(m_j(\cdot|\mathbf{y})|n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} m_j^*(a|\mathbf{y}'))\, d^4 m_j(\cdot|\mathbf{y})} \\
&\quad = \frac{\prod_{a=1}^{4} \Gamma(n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} m_j^*(a|\mathbf{y}'))}{\Gamma(n_{j-1}^{\mathbf{z}}(\mathbf{y}) + \alpha_{kj})}.
\end{aligned} \tag{2.41}$$

Inserting this into the previous equation yields

$$\begin{aligned}
&p(\mathbf{X}|\mathbf{z},\alpha,\mathbf{m}^*) \\
&\propto \prod_{j=0}^{W-1} \prod_{\mathbf{y}} \frac{\Gamma(\alpha_{kj})}{\prod_a \Gamma(\alpha_{kj} m_j^*(a|\mathbf{y}'))} \frac{\prod_{a=1}^{4} \Gamma(n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} m_j^*(a|\mathbf{y}'))}{\Gamma(n_{j-1}^{\mathbf{z}}(\mathbf{y}) + \alpha_{kj})} \cancel{\prod_{a=1}^{4} \frac{1}{m_{\text{bg}}(a|\mathbf{y})^{n_j^{\mathbf{z}}(\mathbf{y},a)}}}.
\end{aligned} \tag{2.42}$$

2.2.6.1 Collapsed Gibbs sampling of z

In Gibbs sampling, we draw each z_n in turn from its conditional posterior probability distribution $p(z_n = i|\mathbf{z}_{-n}, \mathbf{X}, \alpha, q, \mathbf{m}^*)$. Here, $\mathbf{z}_{-n}$ denotes the vector $\mathbf{z}$ with its n'th coordinate removed. By Bayes' theorem, the posterior probability is

$$\begin{aligned} p(z_n = i|\mathbf{z}_{-n}, \mathbf{X}, \alpha, q, \mathbf{m}^*) &= \frac{p(\mathbf{X}|z_n = i, \mathbf{z}_{-n}, \alpha, \mathbf{m}^*)\, p(z_n = i|q)}{p(\mathbf{X}|\mathbf{z}_{-n}, \alpha, \mathbf{m}^*)} \\ &\underset{z_n}{\propto} \frac{p(\mathbf{X}|z_n = i, \mathbf{z}_{-n}, \alpha, \mathbf{m}^*)}{p(\mathbf{X}_{-n}|\mathbf{z}_{-n}, \alpha, \mathbf{m}^*)}\, p(z_n = i|q) \end{aligned} \tag{2.43}$$

The z_n below the $\propto$ indicates that the proportionality constant $p(\mathbf{X}_{-n}|\mathbf{z}_{-n}, \alpha, \mathbf{m}^*)/p(\mathbf{X}|\mathbf{z}_{-n}, \alpha, \mathbf{m}^*)$ does not depend on the value of z_n, but depends on $\mathbf{z}_{-n}, \mathbf{X}, \alpha$, and $\mathbf{m}^*$.

The first factor on the right side of the proportionality can in fact be simplified a lot by noting, first, that $\Gamma(n+1) = n\Gamma(n)$ for any $n \in \mathrm{N}$, and second, that the counts for $\mathbf{z}$ with $z_n = i$ are the same as those for $\mathbf{z}_{-n}$ except for the W k-mers $x_{n,i+j-K:i+j}$ occurring at the motif at position i of the n'th sequence $\mathbf{x}_n$,

$$n_j^{\mathbf{z}}(\mathbf{y}, a) = n_j^{\mathbf{z}_{-n}}(\mathbf{y}, a) + I\left((\mathbf{y}, a) = x_{n,i+j-K:i+j}\right). \tag{2.44}$$

Noting these two points, E.q. (2.43) simplifies to

$$p(z_n = i|\mathbf{z}_{-n}, \mathbf{X}, \alpha, q, \mathbf{m}^*) \propto_{z_n} p(z_n = i) \prod_{j=0}^{W-1} \frac{n_j^{\mathbf{z}_{-n}}(x_{i+j-K:i+j}) + \alpha_{Kj} m^*(x_{i+j}|x_{i+j-K-1:i+j-1})}{(n_{j-1}^{\mathbf{z}_{-n}}(x_{i+j-K:i+j-1}) + \alpha_{Kj})\, m_{\mathrm{bg}}(x_{i+j}|x_{i+j-K:i+j-1})} \tag{2.45}$$

and, with the abbreviation:

$$m_j^{\mathbf{z}_{-n}}(y_{K+1}|\mathbf{y}) := \frac{n_j^{\mathbf{z}_{-n}}(y_{1:K+1}) + \alpha_{Kj} m_j^*(y_{K+1}|y_{2:K})}{n_{j-1}^{\mathbf{z}_{-n}}(y_{1:K}) + \alpha_{Kj}} \tag{2.46}$$

we obtain our sampling equation:

$$p(z_n = i|\mathbf{z}_{-n}, \mathbf{X}, \alpha, q, \mathbf{m}^*) \propto p(z_n = i) \prod_{j=0}^{W-1} \frac{m_j^{\mathbf{z}_{-n}}(x_{i+j}|x_{i+j-K:i+j-1})}{m_{\mathrm{bg}}(x_{i+j}|x_{i+j-K:i+j-1})}. \tag{2.47}$$

We will see in the section on the optimisation of parameters using the EM algorithm (Section 2.2.5) that the conditional probabilities are just the responsibilities defined there, i.e., $r_{ni} = p(z_n = i|\mathbf{z}_{-n}, \mathbf{X}, \alpha, q, \mathbf{m}^*)$ with model parameters $\mathbf{m}$ given by E.q. (2.46). The pseudo-counts from the lower order are simply updated according to $m_j^*(y_{K+1}|y_{2:K}) = m_j^{\mathbf{z}_{-n}}(y_{K+1}|y_{2:K})$,

and hence for $1 \leq k \leq K$:

$$m_j^{\mathbf{z}_{-n}}(y_{k+1}|y_{1:k}) := \frac{n_j^{\mathbf{z}_{-n}}(y_{1:k+1}) + \alpha_{kj}\, m_j^{\mathbf{z}_{-n}}(y_{k+1}|y_{2:k})}{n_{j-1}^{\mathbf{z}_{-n}}(y_{1:k}) + \alpha_{kj}}\,. \tag{2.48}$$

A key feature of the collapsed Gibbs sampling E.q. (2.47) that ensures its efficient exploration of the space of likely motif positions is that, in order to sample the new motif position i in sequence n, the model parameters are computed based on the counts *excluding the previous position i' of the motif in sequence n*. This effectively prevents overtraining and slow mixing, as the old motif position does not "attract" the new position to the same position. In conventional Gibbs sampling, however, z_n is sampled given the model parameters $\mathbf{m}$, which contain information from the previous motif counts of *all* sequences, including sequence n itself. This leads to slower mixing and slower exploration of the parameters space.

2.2.6.2 Sampling of hyperparameter q

Analogous to E.q. (2.43), by Bayes' theorem the posterior probability for q can be written as

$$\begin{aligned} p(q|\mathbf{X},\mathbf{z},\alpha_k,\mathbf{m}^*) &\underset{q}{\propto} p(\mathbf{X}|q,\mathbf{z},\alpha,\mathbf{m}^*)\, p(q|\mathbf{z},\alpha,\mathbf{m}^*) \\ &\underset{q}{\propto} p(q|\mathbf{z},\alpha,\mathbf{m}^*)\,. \end{aligned} \tag{2.49}$$

We apply Bayes' theorem, define $N_0 := |\{n : z_n = 0\}|$ and assume a uniform prior for q, $p(q) = \mathrm{Beta}(q|1,1) = 1$:

$$\begin{aligned} p(q|\mathbf{z},\alpha,\mathbf{m}^*) &\underset{q}{\propto} p(\mathbf{z}|q,\alpha)\, p(q) \\ p(q|\mathbf{z}) &\underset{q}{\propto} q^{N-N_0}(1-q)^{N_0} \end{aligned} \tag{2.50}$$

which has the functional form of a Beta (or Dirichlet) distribution, and therefore

$$p(q|\mathbf{z}) = \mathrm{Beta}(q|N-N_0+1, N_0+1)\,. \tag{2.51}$$

To sample from a Beta distribution, we draw two random numbers, $Q \sim \mathrm{Gamma}(N-N_0+1,1)$ and $P \sim \mathrm{Gamma}(N_0+1,1)$, in which case $q = Q/(Q+P)$ will be distributed according to a Beta distribution $q \sim \mathrm{Beta}(N-N_0+1, N_0+1)$.

2.2.6.3 Sampling α by Gibbs with Metropolis-Hastings

Analogous to E.q. (2.43), by Bayes' theorem the conditional probability of α given $\mathbf{z}$ can be written

$$\begin{aligned} p(\alpha_k|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1}) &\underset{\alpha_k}{\propto} p(\mathbf{X}|\alpha,\mathbf{z},\mathbf{m}^{k-1})\, p(\alpha|q,\mathbf{z},\mathbf{m}^{k-1}) \\ &\underset{\alpha_k}{\propto} p(\mathbf{X}|\alpha,\mathbf{z},\mathbf{m}^{k-1})\, p(\alpha)\,. \end{aligned} \tag{2.52}$$

Inserting (2.29) and (2.42) yields for the conditional probability

$$\begin{aligned} p(\alpha_k|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1}) &= \sum_{j=0}^{W-1} \left(\prod_{\mathbf{y}} \frac{\beta\,\gamma^k}{\alpha_{kj}^2} e^{-\frac{\beta\gamma^k}{\alpha_{kj}}} \frac{\Gamma(\alpha_{kj})}{\prod_a \Gamma(\alpha_{kj} m_j^{k-1}(a|\mathbf{y}'))} \frac{\prod_{a=1}^4 \Gamma(n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} m_j^{k-1}(a|\mathbf{y}'))}{\Gamma(n_{j-1}^{\mathbf{z}}(\mathbf{y}) + \alpha_{kj})} \right) \\ &= \prod_{j=0}^{W-1} p(\alpha_{kj}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1})\,, \end{aligned} \tag{2.53}$$

which factorizes over the α_{kj}. We could therefore use Gibbs sampling to draw each new value of α_{kj} from its probability distribution independent of the others .

But for an efficient optimisation we need to reparameterise α_{kj} as

$$\alpha_{kj} = e^{a_{kj}} \tag{2.54}$$

and sample a_{kj} instead of α_{kj}, because otherwise it would take too long to explore the entire probability distribution by small steps in α_{kj}. If we went in steps of 0.5, for example, it would take almost 20000 directed steps to move from $\alpha_{kj} = 1$ to 10000. With steps of size 0.5, it would take only $2log20000 = 18.4$ directed steps to reach 10000. The probability density also needs to be transformed with the variable:

$$p(a_{kj}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1}) = \left|\frac{d\,\alpha_{kj}}{d\,a_{kj}}\right| p(\alpha_{kj}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1}) \tag{2.55}$$

$$= \alpha_{kj}\, p(\alpha_{kj}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1}) \tag{2.56}$$

The log conditional probability for a_{kl} is

$$\log p(a_{kl}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1}) = \text{const.} - \log\alpha_{kj} - \beta\,\gamma^k/\alpha_{kj} + 4^k \log\Gamma(\alpha_{kj}) \tag{2.57}$$

$$+ \sum_{\mathbf{y}=y_{1:k}} \left(\sum_{a=1}^{4} \left[\log\Gamma(n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} m_j^{k-1}(a|\mathbf{y}')) - \log\Gamma(\alpha_{kj} m_j^{k-1}(a|\mathbf{y}')) \right] - \log\Gamma(n_{j-1}^{\mathbf{z}}(\mathbf{y}) + \alpha_{kj}) \right)$$

We can sample from this distribution using the Metropolis-Hastings algorithm. We draw a new $a_{kl}^{\text{try}} \sim \mathcal{N}(a_{kl}, 1)$ and accept this trial sample with a probability

$$\frac{p(a_{kl}^{\text{try}}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1})}{p(a_{kl}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1})} \text{ if } p(a_{kl}^{\text{try}}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1}) < p(a_{kl}|\mathbf{X},\mathbf{z},q,\mathbf{m}^{k-1})$$
$$1 \text{ if otherwise .} \tag{2.58}$$

Because it is fast to sample a_{kl} in this way, we draw 10 or times in a row and only take record the last accepted sample of a_{kl}. This 10-fold repetition ensures that we can explore almost the entire range of relevant values of a_{kl} within these 10 steps.

2.2.7 Obtaining a motif model

At the start of the sampling, the a_{kj} will move in the direction of the medians of their probability distribution in relatively directed steps until the changes to the a_{kj} become non-directional and begin to fluctuate. We can then fix the a_{kj} to the average of the last 20 or so samples and perform a few (e.g. 5) iterations of the EM algorithm (described in section 2.2.5) to find the optimum model parameters $\mathbf{m}_j^K(a|\mathbf{y})$ given the fixed a_{kj}.

2.2.8 Learning positional preferences of transcription factors

2.2.8.1 Thermodynamic treatment of positional preference

We proceed analogously to Section 2.1 but introduce a positional preference as an additive term ΔG_i in the binding energy. The probability of a factor to bind a binding site consisting of W nucleotides between i and $i+W-1$ in a sequence $\mathbf{x} = x_{1:L}$ then becomes

$$p_i(\text{bound}|\mathbf{x}) = \left(1+\exp\left(\frac{\Delta G(x_{i:i+W-1})+\Delta G_i-\mu}{k_BT}\right)\right)^{-1}. \tag{2.59}$$

We define $p_{\text{motif}}(x_{0:W-1})$ as in E.q. (2.3) and we further define a positional distribution

$$p(z=i|\mathbf{x},\text{bound}) = \frac{\exp(-\Delta G_i/k_BT)}{\sum_{i'=1}^{L}\exp(-\Delta G_{i'}/k_BT)}. \tag{2.60}$$

We abbreviate the denominator as const. gives

$$-\frac{\Delta G_i}{k_BT} + \text{const.} = \log p(z=i|\mathbf{x},\text{bound}) =: s_i. \tag{2.61}$$

Once we know $p_{\text{motif}}(\cdot)$ and $p(z{=}i|\mathbf{x},\text{bound})$, we can compute $S(x_{i:i+W-1})$ and s_i and the relative binding strength $(\Delta G(x_{i:i+W-1})+\Delta G_i)/k_BT$ for any potential binding site position i in any sequence $\mathbf{x}=(x_1\ldots x_L)$.

If we again assume to be in a regime of unsaturated binding, $p(\text{bound}|\mathbf{x}) \lesssim 0.1$ we can approximate the probability $p(\mathbf{x}_n|\text{bound}, p^k_{\text{motif}})$ for pulling out a sequence $\mathbf{x}_n$ from an underlying distribution of possible sequences $p_{\text{bg}}(\mathbf{x})$ as

$$\begin{aligned}
p(\mathbf{x}_n|\text{bound}, p^k_{\text{motif}}) &\propto p(\text{bound}|\mathbf{x}_n, p^k_{\text{motif}})\, p_{\text{bg}}(\mathbf{x}_n) \\
&= p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} p_i(\text{bound}|\mathbf{x}_n, p^k_{\text{motif}}) \\
&= p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} \left(1+\exp\left(\frac{\Delta G(x_{i:i+W-1})+\Delta G_i-\mu}{k_BT}\right)\right)^{-1} \\
&\approx p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} \exp\left(-\frac{\Delta G(x_{i:i+W-1})+\Delta G_i-\mu}{k_BT}\right) \\
&\propto p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} \exp\left(S(x_{i:i+W-1})+s_i\right). \qquad (2.62)
\end{aligned}$$

To find the model parameters $\mathbf{m}$ consisting of $\mathbf{s}=(s_1,\ldots,s_{L-W+1})$, we need to optimise the log likelihood function of these parameters:

$$LL(\mathbf{m}) = \sum_{n=1}^{N} \log p(\mathbf{x}_n|\text{bound}, p^k_{\text{motif}}, \mathbf{s}) \qquad (2.63)$$

2.2.8.2 Flat Bayesian prior on positional preference

Let us define parameters π with $\pi_i = p(z{=}i|z_i{\neq}0) = e^{s_i}$ the probability of a motif to start at position i of a sequence. The M-step will then be given again by E.q. (2.35) but this time using the positional preferences π_i instead of the flat positional distribution. We will use a flat prior distribution

$$p(\pi|\beta) = \text{Dir}(\pi|\beta\mathbb{1}), \qquad (2.64)$$

and we will choose a value around $\beta = 2\ldots10$.

The auxiliary function becomes

$$
\begin{aligned}
&Q(p^k_{\text{motif}}, \alpha, q|\mathbf{r}, p^{k-1}_{\text{motif}}) \\
&= \sum_{n=1}^{N} \left[\sum_{i=0}^{L_n-W+1} r_{ni} \log \Big(p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}})\, p(z_n = i|q) \Big) \right] + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \alpha) + \log p(\pi|\beta) \\
&= \sum_{n=1}^{N} \sum_{i=0}^{L_n-W+1} r_{ni} \log p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \alpha) \\
&\quad + \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + \sum_{i=1}^{L_n-W+1} r_{ni} \log(q\pi_i) \right) + \log \text{Dir}(\pi|\beta\mathbb{1}) \\
&= \sum_{n=1}^{N} \sum_{i=0}^{L_n-W+1} r_{ni} \log p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \alpha) \qquad (2.65) \\
&\quad + \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + (1-r_{n,0}) \log q + \sum_{i=1}^{L_n-W+1} r_{ni} \log \pi_i \right) + \sum_{i=1}^{L_n-W+1} (\beta - 1) \log \pi_i .
\end{aligned}
$$

We use the method or Lagrange multipliers again to find the optimum of $Q(p^k_{\text{motif}}, \alpha, q|\mathbf{r}, p^{k-1}_{\text{motif}})$ under the constraint $\sum_{i=1}^{L-W+1} \pi_i = 1$:

$$
\frac{\partial}{\partial \pi_i} \left(Q(p^k_{\text{motif}}, \alpha, q|\mathbf{r}, p^{k-1}_{\text{motif}}) - \lambda \left(\sum_{i=1}^{L-W+1} \pi_i - 1 \right) \right) = \sum_{n=1}^{N} \frac{r_{ni}}{\pi_i} + \frac{\beta - 1}{\pi_i} - \lambda = 0 \qquad (2.66)
$$

Solving for π_i, normalising the distribution and defining $N_i := \sum_{n=1}^{N} r_{ni}$ yields

$$
\pi_i = \frac{N_i + \beta - 1}{N + (L - W + 1)(\beta - 1)} . \qquad (2.67)
$$

2.2.8.3 Prior penalising jumps in the positional preference profile

For many applications it might be more appropriate to limit the complexity of the positional preference profile by imposing a smoothness on the $p(z = i)$. For example: (i) transcription factor binding sites will be more frequent near the center of ChIP-seq peaks than farther away; (ii) transcription factors bind more strongly to the outer parts of probes on protein binding microarrays than to the parts near the glass slide; (iii) transcription factors in HT-SELEX experiments might prefer the center of probes over the ends. In the following we assume that all training and test sequences have the same length L.

Because the smoothness prior couples neighbouring positional probabilities with each other, there is no closed-form solution for the parameters anymore. We have to use a gradient-based optimisation such as conjugate gradients to minimise Q with respect to the positional

parameters. We therefore parameterise the positional distribution in such a way that the normalisation condition $\sum_i \pi_i = 1$ and the limits $0 \leq \pi_i \leq 1$ automatically hold true during the numerical optimisation,

$$p(z_n = i | z_n \neq 0) = \frac{e^{s_i}}{\sum_{i'=1}^{L-W+1} e^{s_{i'}}}. \tag{2.68}$$

We impose a smoothness prior on the π_i, that encourages the point-wise estimated first derivative to stay small,

$$p(\pi|\beta) = \prod_{i=2}^{L-W+1} \mathcal{N}\left(s_i - s_{i-1} | 0, \beta^{-1}\right), \tag{2.69}$$

with precision (= inverse variance) β.

With this prior, the auxiliary function becomes

$$\begin{aligned} Q(p^k_{\text{motif}}, \alpha, q, \pi | \mathbf{r}, p^{k-1}_{\text{motif}}) = & \sum_{n=1}^{N} \sum_{i=0}^{L-W+1} r_{ni} \log p(\mathbf{x}_n | z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}} | p^{k-1}_{\text{motif}}, \alpha) \\ & + \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + (1-r_{n,0}) \log q + \sum_{i=1}^{L-W+1} r_{ni} \left(s_i - \log \left(\sum_{i'} e^{s_{i'}} \right) \right) \right) \\ & - \frac{\beta}{2} \sum_{i=2}^{L-W+1} (s_i - s_{i-1})^2 + \frac{L-W}{2} \log \beta + \text{const.} \end{aligned} \tag{2.70}$$

The partial derivatives of $Q(p^k_{\text{motif}}, \alpha, q, \pi | \mathbf{r}, p^{k-1}_{\text{motif}})$ are

$$\begin{aligned} \frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \alpha, q, \pi | \mathbf{r}, p^{k-1}_{\text{motif}}) = & \sum_{n=1}^{N} r_{ni} - \sum_{n=1}^{N} \sum_{i'=1}^{L-W+1} r_{ni'} \frac{e^{s_i}}{\sum_{i''} e^{s_{i''}}} \\ & - \beta (s_i - s_{i-1}) I(2 \leq i \leq L-W+1) \\ & + \beta (s_{i+1} - s_i) I(1 \leq i \leq L-W) \end{aligned} \tag{2.71}$$

and

$$\frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \alpha, q, \pi | \mathbf{r}, p^{k-1}_{\text{motif}}) = N_i - (N - N_0)\, p(z = i | z \neq 0) - (\beta \mathbf{A} s)_i$$

with the abbreviations $N_0 := \sum_{n=1}^{N} r_{n,0}$ and

$$\mathbf{A} := \begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ -1 & 2 & -1 & 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & -1 & 2 & -1 & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & -1 & 2 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & 2 & -1 & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 & -1 & 2 & -1 \\ 0 & \dots & \dots & \dots & 0 & 0 & -1 & 1 \end{pmatrix} . \tag{2.72}$$

The partial derivative will adjust s_i such that $p(z=i|z\neq 0) = e^{s_i}/\sum_{i'} e^{s_{i'}}$ equals $N_i/(N-N_0)$ plus a smoothness correction $\mathbf{A}s$ that will pull s_i up or down in order to minimise the estimator of the second derivative of the profile at position i. We run a few iterations of conjugate gradients (e.g. 5 to 10) during each EM step to learn the positional preferences.

Learning the optimal smoothness parameter β from the data. We can regard Q also as a function of β,

$$Q(p^k_{\text{motif}}, \alpha, q, \pi, \beta|\mathbf{r}, p^{k-1}_{\text{motif}}) = -\frac{\beta}{2}\sum_{i=2}^{L-W+1}(\pi_i - \pi_{i-1})^2 + \frac{L-W}{2}\log\beta + \text{const}_\beta\,, \tag{2.73}$$

and optimise is with respect to β:

$$0 = \frac{\partial}{\partial\beta}Q(p^k_{\text{motif}}, \alpha, q, \pi, \beta|\mathbf{r}, p^{k-1}_{\text{motif}}) = -\frac{1}{2}\sum_{i=2}^{L-W+1}(s_i - s_{i-1})^2 + \frac{L-W}{2\beta} \tag{2.74}$$

and therefore

$$\beta = \left(\frac{1}{L-W}\sum_{i=2}^{L-W+1}(s_i - s_{i-1})^2\right)^{-1} \tag{2.75}$$

Instead of optimising β, we can again interpret Q as the likelihood of an ensemble of fractional motif instances with weights r_{ni} and compute the expectation value of β. If we assume a uniform prior on β, $p(\beta) = \text{const}$, the posterior distribution of β is proportional to the likelihood. We note that the functional form of $Q(\beta)$ is that of a Gamma distribution, $Q(\beta) = \log \text{Ga}(\beta|a,b) + \text{const} = (a-1)\log\beta - b\beta + \text{const}$, with $a-1 = (L-W)/2$ and $b = (1/2)\sum_i (s_i - s_{i-1})^2$. Since the expectation value of a Gamma distribution is a/b, we can

conclude for β

$$\mathbb{E}[\beta] = \left(\frac{1}{L-W+2} \sum_{i=2}^{L-W+1} (s_i - s_{i-1})^2 \right)^{-1} . \tag{2.76}$$

We can then update β by its expectation value instead of the mode of $Q(\beta)$. Alternatively, we could sample β from the Gamma distribution $\mathrm{Ga}(\beta|(L-W+2)/2, (1/2)\sum_i (s_i - s_{i-1})^2)$.

2.2.8.4 Prior penalising kinks in the positional preference profile

For various applications such as PBMs and HT-SELEX, we might be interested in more smooth positional preferences. In these cases, it might be better to use a smoothness prior on the π_i that encourages the point wise estimated *third* derivative to stay small,

$$p(\pi|\beta) = \prod_{i=2}^{L-W} \mathcal{N}\left(s_i - \frac{s_{i-1}+s_{i+1}}{2} \middle| 0, \beta^{-1} \right) , \tag{2.77}$$

with precision (= inverse variance) β. With this prior, the auxiliary function becomes

$$\begin{aligned} Q(p^k_{\text{motif}}, \alpha, q, \pi|\mathbf{r}, p^{k-1}_{\text{motif}}) = & \sum_{n=1}^{N} \sum_{i=0}^{L-W+1} r_{ni} \log p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \alpha) \\ & + \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + (1-r_{n,0}) \log q + \sum_{i=1}^{L-W+1} r_{ni} \left(s_i - \log \left(\sum_{i'} e^{s_{i'}} \right) \right) \right) \\ & - \frac{\beta}{2} \sum_{i=2}^{L-W} \left(s_i - \frac{s_{i-1}+s_{i+1}}{2} \right)^2 + \frac{L-W-1}{2} \log \beta + \text{const.} \end{aligned} \tag{2.78}$$

The partial derivatives of $Q(p^k_{\text{motif}}, \alpha, q, \pi|\mathbf{r}, p^{k-1}_{\text{motif}})$ are

$$\begin{aligned} \frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \alpha, q, \pi|\mathbf{r}, p^{k-1}_{\text{motif}}) = & \sum_{n=1}^{N} r_{ni} - \sum_{n=1}^{N} \sum_{i'=1}^{L-W+1} r_{ni'} \frac{e^{s_i}}{\sum_{i''} e^{s_{i''}}} \\ & + \frac{\beta}{2} \left(s_{i-1} - \frac{s_{i-2}+s_i}{2} \right) I(3 \le i \le L-W+1) \\ & - \beta \left(s_i - \frac{s_{i-1}+s_{i+1}}{2} \right) I(2 \le i \le L-W) \\ & + \frac{\beta}{2} \left(s_{i+1} - \frac{s_i+s_{i+2}}{2} \right) I(1 \le i \le L-W-1) \end{aligned} \tag{2.79}$$

and

$$\frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \alpha, q, \pi | \mathbf{r}, p^{k-1}_{\text{motif}}) = N_i - (N - N_0)\, p(z{=}i | z{\neq}0) - \frac{\beta}{4}(\mathbf{B}s)_i$$

with the abbreviations $N_0 := \sum_{n=1}^{N} r_{n,0}$ and

$$\mathbf{B} := \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ -2 & 5 & -4 & 1 & 0 & \ddots & \ddots & \ddots & \vdots \\ 1 & -4 & 6 & -4 & 1 & \ddots & \ddots & \ddots & \vdots \\ 0 & 1 & -4 & 6 & -4 & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 1 & -4 & 6 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 1 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 6 & -4 & 1 \\ \vdots & \ddots & \ddots & \ddots & \ddots & 1 & -4 & 5 & -2 \\ 0 & \dots & \dots & \dots & \dots & 0 & 1 & -2 & 1 \end{pmatrix}. \tag{2.80}$$

The partial derivative will adjust s_i such that $p(z{=}i|z{\neq}0) = e^{s_i} / \sum_{i'} e^{s_{i'}}$ equals $N_i/(N{-}N_0)$ plus a smoothness correction $\mathbf{B}s$ that will pull s_i up or down in order to minimise the estimator of the third derivative of the profile at position i.

Learning the optimal smoothness parameter β from the data. Analogously to the previous smoothness prior, we can learn β from the data using the update

$$\beta = \left(\frac{1}{L{-}W{-}1} \sum_{i=2}^{L-W} \left(s_i - \frac{s_{i-1} + s_{i+1}}{2} \right)^2 \right)^{-1} \tag{2.81}$$

or

$$\beta = \left(\frac{1}{L{-}W{+}1} \sum_{i=2}^{L-W} \left(s_i - \frac{s_{i-1} + s_{i+1}}{2} \right)^2 \right)^{-1}. \tag{2.82}$$

2.3 Training and testing data

in vivo data sets

ENCODE database

We evaluated the performance of the selected algorithms on human ChIP-seq data sets from the ENCODE portal [60] until March 2012. In total, there are 435 data sets for 93 distinct transcription factors. The top 5000 peak regions sorted by their signal value are selected for each data set when peaks are more than 5000, and all peaks are chosen if the peaks are fewer than 5000. Positive sequences are extracted ± 104 bp around the peak summits. Background sequences are sampled by the trimer frequencies from positive sequences, with the same lengths as positive sequences and 10 times the amount of positive sequences. 8 data sets are excluded from all the results because diChIPMunk fails to learn models within 3 hours.

GTRD database

For the GTRD database, we obtained 405 *in vivo* data sets for 405 non-redundant human transcription factors from Yevshin et al. [61]. The top 5000 peak regions are selected after sorting by q-values. Positive sequences are extracted ± 100 bp around the peak summits. Background sequences are sampled in the same way as described previously.

MITOMI data sets

MITOMI is a microfluidics-based approach for *de novo* discovery and quantitative biophysical characterization of DNA target sequences [62]. We downloaded the MITOMI data for 28 *Saccharomyces cerevisiae* transcription factors under the accession GPL10817. The 3 bp and 15 bp long adapters on both ends are truncated. We then downloaded yeast GTRD data sets that are available for 8 transcription factors [61] and use them for training the motif models.

in vitro data sets

HT-SELEX data sets

For HT-SELEX data, we downloaded 164 data sets with 200 bp-long oligomers from Zhu et al. [18], which are deposited in the European Nucleotide Archive (ENA) under accession PRJEB22684. Each data set represents one non-redundant human transcription factor. For each data set, we selected the top 5000 sequences from the 4th cycle without any sorting. Background sequences are sampled in the same way as described previously.

2.4 Assessing motif models and benchmark

As numerous motif discovery tools have become available, guiding the users to choose the proper models for their research becomes important. Because of the incomplete understanding of the regulatory mechanism and the lack of ground truth, it is challenging to determine the correctness of tools (see review [63]). In this thesis, we address this challenge in two aspects: (I) developing a novel motif assessment score, the Average Recall (AvRec), for better describing the accuracy of the motif models; (II) providing a benchmark scheme of data sets from different technique platforms for assessing further tools. This part is described in detail in the manuscript and published paper.

To avoid unnecessary re-writing and self-plagiarism, part of the methods and results are included in the published papers attached here, and contributions of this author are claimed for each publication.

Chapter 3

Result and Discussion

3.1 BaMMmotif2 algorithm

3.1.1 Overview

For *de novo* discovering regulatory motifs from nucleotide sequences with high accuracy, I have implemented BaMMmotif2, a tool using higher-order interpolated Markov models to learn the dependencies of nucleotides with variable lengths for TF binding. I have optimized the pseudo-factor that determines how much lower-order information flows to the higher-order. I have introduced a masking strategy to optimize distinct motifs, if existing in the data. I have trained the model to learn the positional preferences of TFs from the data.

Apart from developing the motif discovery approach, I have completed benchmark tests using both *in vivo* (e.g., ChIP-seq) and *in vitro* (e.g., HT-SELEX) data, to validate different motif finders. I have developed a better validation score to replace the Receiver operating characteristic (ROC) curve and p-values when estimating how well a motif model performs on experimental data, given the most relevant regime of true positives and false positives. I have established a validation scheme to examine the model robustness regardless of the cell conditions and experimental platforms.

It has shown that our approach outperforms other PWM-based and higher-order model-based tools on both *in vivo* and *in vitro* data. I have also introduced a cross-platform scheme for validating the models by training models on *in vivo* data and testing them on *in vitro* data and vice versa, to capture the motif features that are conserved across various experimental conditions. Although TFs are reported to bind to different motifs under different cell states, I have shown with a cross-cell-line validation that BaMMs are robust to learn the TF-DNA specificity regardless of the cell types, compared to other tools.

Most of the results are included in the manuscript (Section 3.1.3) and the publication (Section 3.2.2). I describe the remaining results in the first part of the following:

3.1.2 Hyperparameter optimization

I tried to improve the model by different approaches: optimization of hyper-parameter α using Gibbs sampling (Section 3.1.2.1), learning TF positional preference via positional priors (Section 3.1.2.2, and masking sequences for learning distinct motifs (Section 3.1.2.3). I also applied higher-order BaMMs to predict weak binding events (Section 3.1.2.4).

3.1.2.1 Gibbs sampling of pseudo-factor α

In the BaMM model profile (E.q. (2.17)), there is a hyperparameter α, which tunes how much pseudo-counts from the lower-order shall be accounted for the conditional probabilities of the higher-order. It lies at the core of the BaMM model since it lowers the model complexity by applying pseudo-counts to adopt the variant information from the lower-orders, instead of learning all the parameters for all the orders. However, the choices of αs for each order at each position of the motif can be further improved, since the nucleotide correlations do not only occur within the adjacent positions but also positions a few bases away. The optimized position-specific αs could help to optimize the motif length. Thus, according to E.q. (2.58), I first implemented the Metropolis-Hastings algorithm for sampling the αs.

The core implementation is illustrated as the following code:

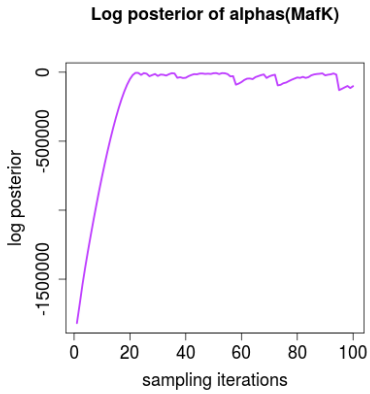
```
void GibbsSampling::GibbsMH_sample_alphas( size_t iter ){
    // sampling alphas in exponential space with MH algorithm
    std::uniform_real_distribution<float> uniform_dist( 0.0f, 1.0f );
    for( size_t k = 0; k < K+1; k++ ){ // for all the orders
        for( size_t j = 0; j < W; j++ ){ // for all motif positions
            // convert Alpha to log space as 'a'
            float a_prev = logf(Alpha[k][j]);
            float lprob_a_prev = calc_logCondProb_a(iter,a_prev,k,j);
            // draw a new 'a' from the distribution of N(a, 1)
            std::normal_distribution<float> norm_dist(a_prev,
                                                1.0f / (float)(k+1));
            float a_new = norm_dist( Global::rngx );
            float lprob_a_new = calc_logCondProb_a( iter,a_new,k,j);
            float accept_ratio;
            float uni_random;
            if( lprob_a_new < lprob_a_prev ){
                // calculate the acceptance ratio
                accept_ratio = expf(lprob_a_new - lprob_a_prev);
```

```
                    // draw a random number uniformly between 0 and 1
                    uni_random = uniform_dist( Global::rngx );
                    // accept the trial sample if the ratio is not
                    // smaller than a random number between (0,1)
                    if( accept_ratio >= uni_random ){
                        Alpha[k][j] = expf(a_new);
                    }
                } else {
                    // accept the trial sample
                    Alpha[k][j] = expf(a_new);
                }
            }
        }
}
```

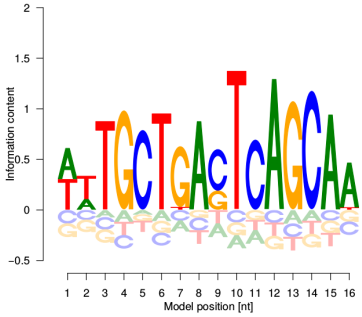
Since it is a numerical sampling procedure, we can approximate the optimal alphas by taking the average alpha values from the last 10 steps or so after sampling for 100 times (i.e., `iter=100`). Then I obtained a motif model by performing 5 iterations of the EM algorithm (Section 2.2.5) with the average alphas to find the optimum model parameters $\mathbf{m}_j^K(a|\mathbf{y})$.

For each iteration, the log posterior is updated after drawing the αs. After a few dozens of iteration, the log posterior begins to fluctuate (e.g., Figure 3.1A). As for the example of MafK motif, the optimized alphas are very close if not identical for lower-orders in the core region, and become larger on the borders (Figure 3.1C).

(A)



(B)



(C)

```
k = 0
0.889 1.04 1.01 1.04 1.03 1.07 1.04 1.04 1.04 1.02 1.08 1.02 1.06 1.03 1.06 1.05
k = 1
20 1.1 1.1 1.11 1.13 1.19 1.19 1.16 1.17 1.17 1.14 1.23 1.18 1.35 1.22 1.19
k = 2
60 60 1.64 1.47 1.16 1.35 1.43 1.47 1.4 1.38 1.62 1.39 1.5 1.58 1.87 1.64
k = 3
180 180 180 102 34.5 4.37 8.45 4.08 3.06 1.14 3.09 2.98 0.883 4.37 4.11 3.39
```

Fig. 3.1 **Alpha optimization using gradient descent.**
Panel (**A**) shows the posteriors in log scale versus the iterations after sampling αs in each iteration, training on a MafK dataset from ChIP-seq. Panel (**B**) shows the motif logo of MafK. Panel (**C**) shows the optimized alphas for the order 0, 1, 2 and 3 over all the motif positions.

Given that the sampled alphas look reasonable on the real dataset for MafK (Figure 3.1C), I carried out benchmark tests on 552 *in vivo* datasets from the GTRD database. The input are 5000 sequences that are either $\pm$ 100 bp or $\pm$ 500 bp around the summit in each dataset. I compared the model performance between optimizations using EM and Gibbs sampling by applying 5-fold cross-validations. I found that there is no major difference between using EM and Gibbs sampling with optimized alphas, when the input sequences are 200 bp long (Figure 3.2A). However, when the sequences are extended to 1000 bp long, the median motif scores are improved by 5.3% (Figure 3.2B and 3.2C). Given that the lengths of input sequences for motif search are usually no longer than a few hundred base pairs, the sampling of alphas does not gain us much. Also, Gibbs sampling has the drawback that it is not as efficient as EM. Thus, I kept EM as the default optimizer for motif refinement procedure.

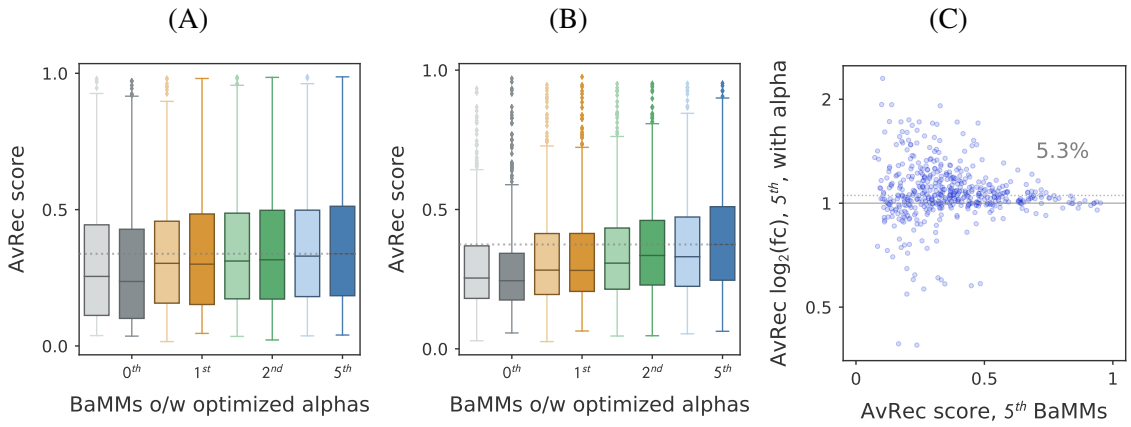


Fig. 3.2 **Performance of BaMMs with alpha learning on *in vivo* data.**
BaMMs with zeroth- (grey), first- (orange), second- (green) and fifth-orders (blue) are trained and tested on 552 GTRD datasets using 5-fold validations. Panel (**A**) shows the benchmark of BaMMs with and without alpha optimization, trained and tested on GTRD datasets with sequences of length 200 bp around the summits, as shown in box plots, with boxes indicating 25%/75% quantiles, whiskers 95%/5% quantiles. Panel (**B**) shows the same benchmark as (**A**) but with sequences of length 500 bp. Panel (**C**) shows the same benchmark as (**B**) in a scatter plot. Each dot represents one dataset. The median fold change increase of AvRec scores is by 5.3%.

3.1.2.2 Optimization of positional prior z

For de-coupling the motif parameters $\mathbf{m}^K$ and positional prior $\mathbf{z}$, I first implemented collapsed Gibbs sampling approach for integrating out the model parameters $\mathbf{m}^K$, according to E.q. 2.47. In each iteration, a sequence-specific positional prior z_n is sampled from a cumulative distribution of the rest $\mathbf{z}_{-n}$. If z_n is larger than 0, then the counts of k-mers within the pattern W (W as the motif length) at z position on the n-th sequence are subtracted from the total counts. The motif profile is then updated without counting the n-th sequence, and a new distribution of positional priors is calculated based on the new motif profile.

I have implemented this algorithm and compared the motif performance of the models optimized with sampled positional priors $\mathbf{z}$ and those without. However, there is no significant difference between these two approaches (Figure 3.3).

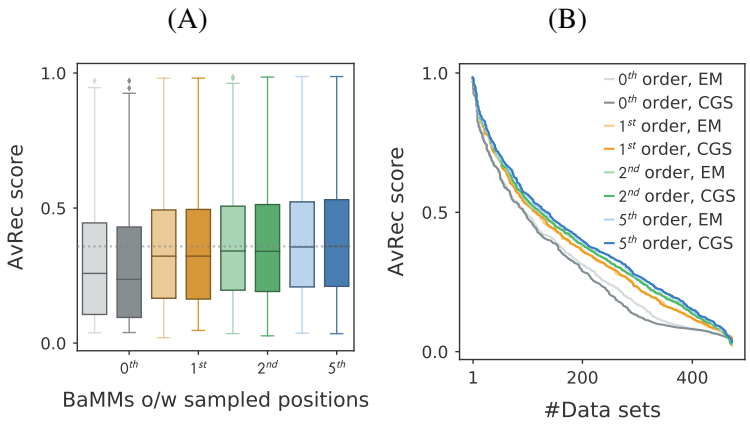


Fig. 3.3 **Performance of BaMMs with positional prior optimization by collapsed Gibbs sampling on in vivo data.**
BaMMs with zeroth- (grey), first- (orange), second- (green) and fifth-orders (blue) are trained and tested on 473 GTRD datasets with input sequences of length 200 bp using 5-fold validations. Panel (**A**) shows the benchmark of BaMMs with (darker colors) and without (lighter colors) sampling positional prior **z** in box plots, with boxes indicating 25%/75% quantiles, whiskers 95%/5% quantiles. Panel (**B**) shows the same results as (**A**) but in cumulative plots.

Since sampling of the positional priors with collapsed Gibbs sampling does not improve the model predictive power, and sampling approach is not as efficient as deterministic approximations, I, therefore, tried to optimize the positional priors using a smooth kernel function. It also reflects the biological properties of the transcription binding preferences. The theoretical part is included in the manuscript (Section 3.1.3).

I tested the performance of optimized positional prior in a simulation. For the simulated data, there are 5000 sequences with a length of 205 bp generated from a second-order Markov model. There are three motifs implanted randomly in these sequences: motif 1 with 30% occurrence at positions 50 $\pm$ 10 bp, motif 2 with 30 % the occurrence at positions 100 $\pm$ 10 bp, and motif 3 with 50 % occurrence at positions 150 $\pm$ 10 bp (Figure 3.4A). Without optimizing the positional priors, the initial motif 1 was refined to a mixture of motif 2 and 3 (Figure 3.4B). In contrast, with optimizing the positional priors, the initial motif 1 was refined to its local optimum (Figure 3.4C), and the distribution of positional priors is consistent with the motif distribution over the sequences (Figure 3.4D).

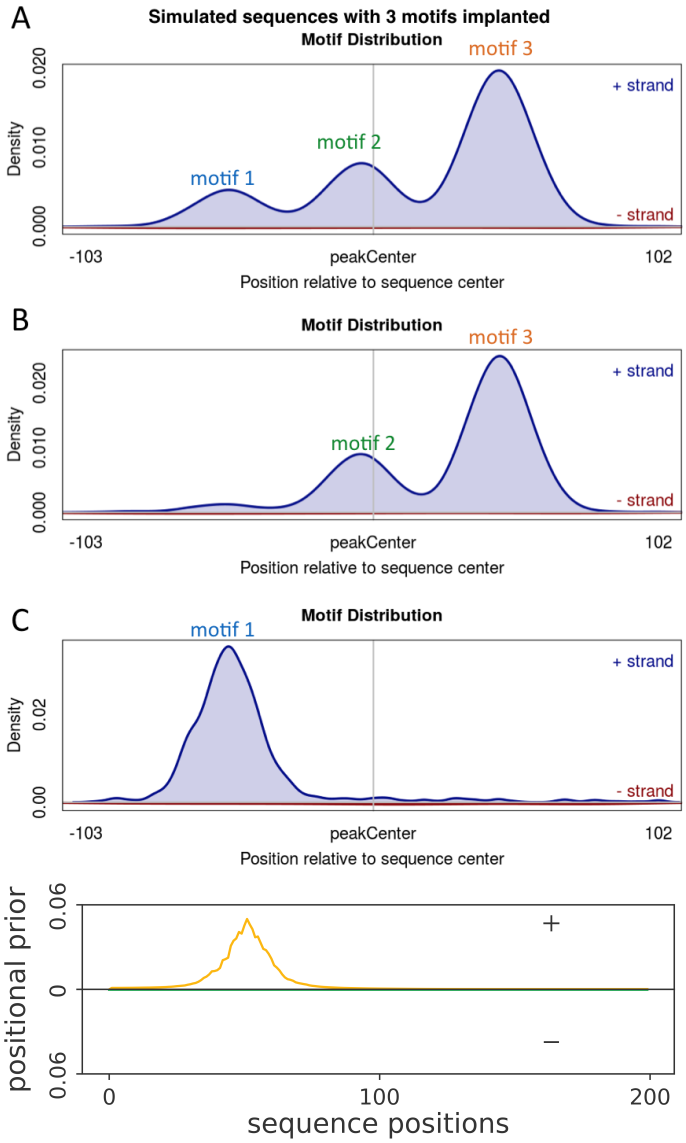


Fig. 3.4 **Positional prior optimization on simulated data.**
Panel (**A**) shows the overall distribution of the three motifs. Panel (**B**) shows the motif distribution after scanning with the model seeded by motif 1 and optimized by EM without positional prior. The EM algorithm tends to converge to a global optimum. Panel (**C**) shows the motif distribution after scanning with the model seeded by motif 1 and optimized by EM with positional prior. It converges to the optimum of motif 1 distribution. Panel (**D**) shows the distribution of optimized positional prior for panel (**C**).

With promising results from the simulations, I did benchmark tests using the ENCODE ChIP-seq datasets. Although the motif positional preferences are learned properly on the real

dataset (e.g., Figure 3.5C), the median motif performance is however, not improved (Figure 3.5A and 3.5B).

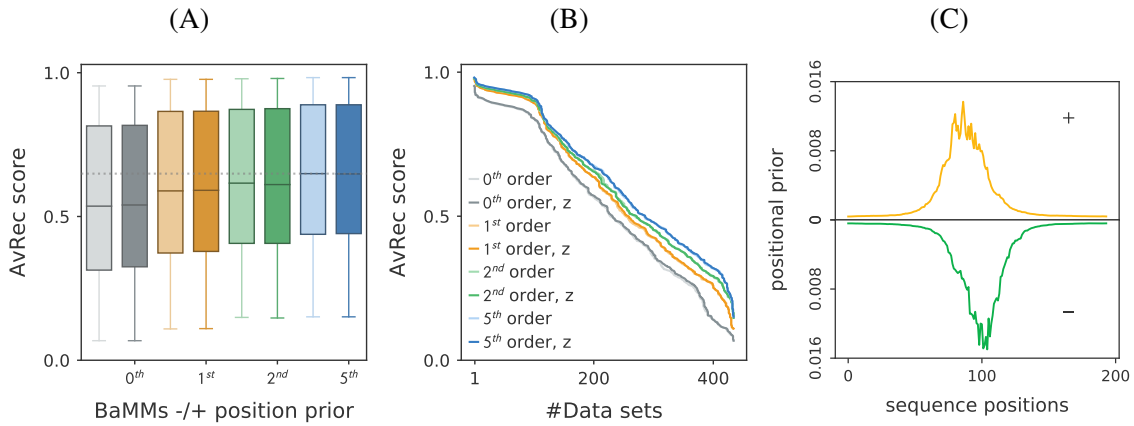


Fig. 3.5 **Optimization of positional prior on *in vivo* data.**
BaMMs with zeroth- (grey), first- (orange), second- (green) and fifth-orders (blue) are trained and tested on 435 ENCODE datasets using 5-fold validations. Panel (**A**) shows the comparison in box plots for different orders with- (dark colors) and without (light colors) optimization of positional prior in the motif training. Panel (**B**) shows the cumulative number of datasets with AvRec score ranging from 0 to 1. Panel (**C**) shows the optimized positional prior on ChIP-seq dataset for GABPα motif (from the GTRD database).

3.1.2.3 Masking input sequences

Multiple motifs can occur in the same data sets, especially when they are co-binding events of TF and co-factors, or if the experiments are explicitly designed for studying the TF cooperativity (Figure 3.6). To distinguish the motifs and prevent EM from running into a global optimum with only the primary or stronger motifs, I introduced a masking step prior to the EM algorithm.

Given the sequences and initial motif models, I first applied one round of E-step (E.q. (2.35)), and calculated the responsibilities of motif occurring at all the positions on all the sequences. I then ranked the motif occurrences with regard to the responsibilities and chose the top 5% for further optimizing the motifs.

In the benchmark on 427 ChIP-seq datasets, the masking step reduces the average performance of fifth-order BaMMs by 4.4% and improves the speed by 10-fold (Figure 3.7B). This result implies that optimizing higher-order models with EM on the full parameter space might lead to the global optimum with multiple motifs. However, limiting the searching space to the most relevant positions helps sustain the motif model's local optimum.

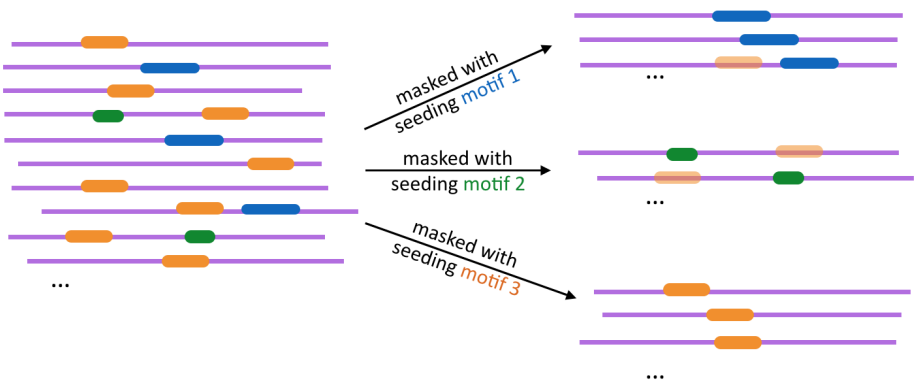


Fig. 3.6 **Scheme of the masking step.**
Mask the input sequences with each seeding motif and optimize motifs based on the top few percent of motif occurrences.

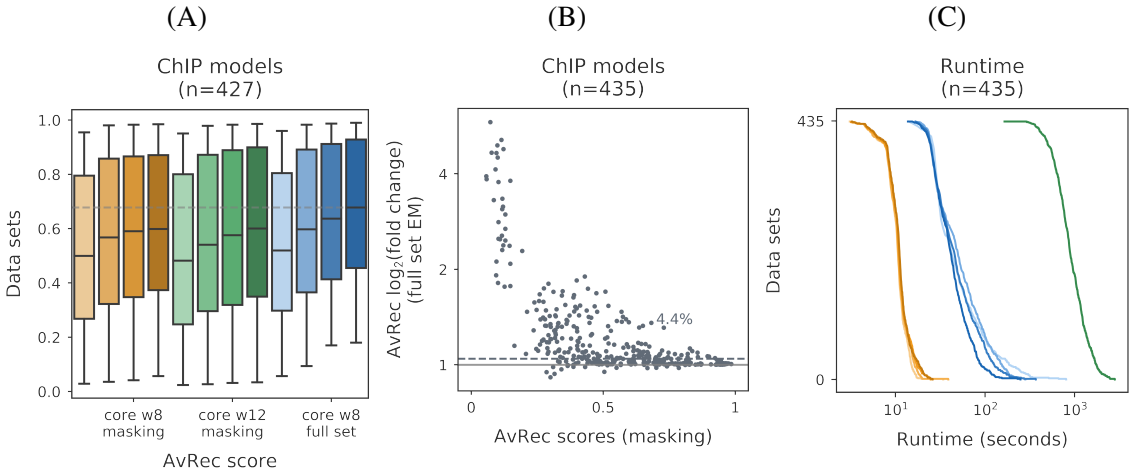


Fig. 3.7 **Performance of using EM on the full versus masked sequences on *in vivo* data.** (**A**) Using the full set of sequences for EM optimization (blue) improves the performance of higher-order models while extending the core regions for searching the enriched patterns (green) does not contribute to motif discovery, in comparison to that with 8 bp for seeding and masking 95% sequences for EM optimization (yellow). All box-plot whiskers show 95th/5th percentile. Each cluster contains models with different orders (zeroth-, first-, second- and fifth-order). (**B**) Fifth-order BaMMs with full set of sequences for optimization have a 4.4% AvRec fold increase compared to those with only 5% sequences for optimization. (**C**) Using a masking step improves the speed by 10-fold, in comparison to using the full set for learning motif model.

3.1.2.4 Prediction on weak binding sites

Predicted binding affinities by BaMMs correlate with measured affinities by MITOMI

Fordyce et al. [62] developed a microfluidics-based approach for measuring the relative binding affinities between 28 yeast TFs and their binding sites quantitatively. In their experiment design, they generated a library of 1457 oligonucleotides of length 70 bp to cover all possible 8-mers. The TFs and oligos were labelled with different fluorescent dyes and incubated in isolated unit cells of the device. After unbound molecules were washed out, the fluorescent intensities were detected. The ratio between the two fluorescent signals should be linearly proportional to the fractional occupancy of protein and thus reflects the relative TF-DNA binding affinity. The binding affinities can also be predicted by a predictive model, as described in E.q. 2.5. Thus, we could validate the model performance by comparing the correlations between the predicted binding affinities by different tools and the measured binding affinities by MITOMIv2 experiments.

Here is one example: for the yeast transcription factor Reb1, I compared the correlations of 8-mer binding affinity measured by MITOMIv2 experiment and predicted by either a 5^{th}-order BaMM learned from a ChIP-seq dataset from GTRD database (Figure 3.8A) or a PWM from the JASPAR database (Figure 3.8B).

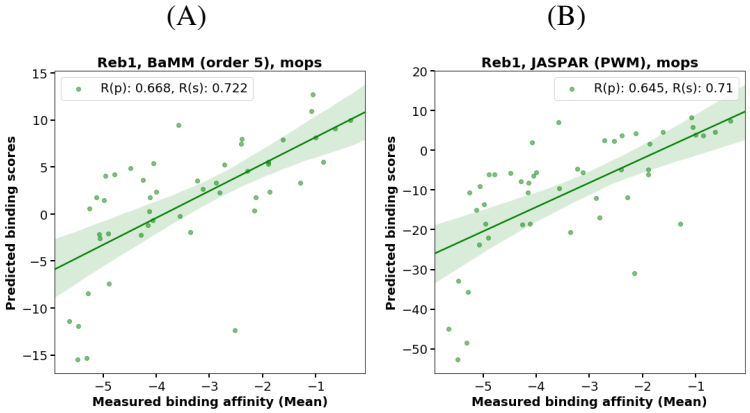


Fig. 3.8 **Correlations between the measured and predicted binding affinity.**
The 8-mer log-odds scores predicted by either a 5^{th}-order BaMM learned from a ChIP-seq dataset from GTRD (**A**) or a PWM from the JASPAR database (**B**), compared to the intensity ratio measured by the MITOMITv2 experiment. Both Pearson and Spearman correlation coefficients are shown on the upper right of each panel.

Furthermore, I validated BaMMs of 1st- and 5th-order and PWMs from the JASPAR database using MITOMIv2 datasets and GTRD datasets. There were eight GTRD datasets for the same TFs as those measured in MITOMIv2, and seven out of the eight showed that fifth-order BaMMs had a better prediction on the binding affinity, compared to PWMs from JASPAR (Table 3.1). There was one exception for which the reported motif consensus was not enriched in the GTRD dataset.

Table 3.1 **Pearson correlations between predicted and measured yeast motifs.**

Motif	Consensus	JASPAR PWM	1^{st}-order BaMM	5^{th}-order BaMM
BAS1	TGACTC	0.365	0.439	0.444
CBF1	RTCACGTG	0.487	0.553	0.523
REB1	CCGGGTAA	0.645	0.656	0.668
SKO1	TTACGTAA	0.544	0.578	0.592
DAL80	cGATAAG	0.356	0.384	0.389
GAT1	GATAAG	0.215	0.373	0.389
GCN4	TGASTCA	0.639	0.645	0.664
PHO4	CACGTG	0.5	0.108*	0.054*

* Here the motif learned from GTRD has a different consensus that is reported in literature.

As for further validations, SMiLE-seq [64] can be a good candidate, which is also a microfluidics-based technique and measures human TF-TF-DNA bindings on a large scale with more flexible lengths.

Higher-order BaMM models predict more human CTCF sites

As good evidence for the higher-order BaMMs being more accurate than PWMs at predicting the weak binding sites, I revisited the CTCF models learned by a fifth-order BaMM from different cell lines in the ENCODE database. The CTCF factor is an essential player in forming the topologically associating domains (TADs) of chromatins and thus bring closer the enhancers and promoters for the open regions to start the transcription. To gain a deeper understanding of its functions, it is crucial to train a more accurate model.

Siebert et al. [35] showed that a CTCF motif model optimized by BaMMmotif with order 5 and length 67bp had a better AUPRC (area-under-the-precision-recall-curve) than a PWM model. I re-checked this and trained fifth-order BaMM models on ChIP-seq sequences

from four cell lines (i.e., GM12878, K562, Hela and Mcf). I scanned both the whole human genome hg19 and the reversed genome (as a negative set) with the optimized BaMMs and a PWM model from the JASPAR database (ID: MA0139.1) for comparison. The log-odds scores were calculated for all the positions on the genome and the negative sequences and the distributions of the log-odds scores were plotted, as shown in Figure 3.9. The distributions with scores predicted by BaMMs from the GM12878 and K562 cell lines showed a clear hump which indicates the CTCF binding sites identified with high accuracy (Figure 3.9A and 3.9B).

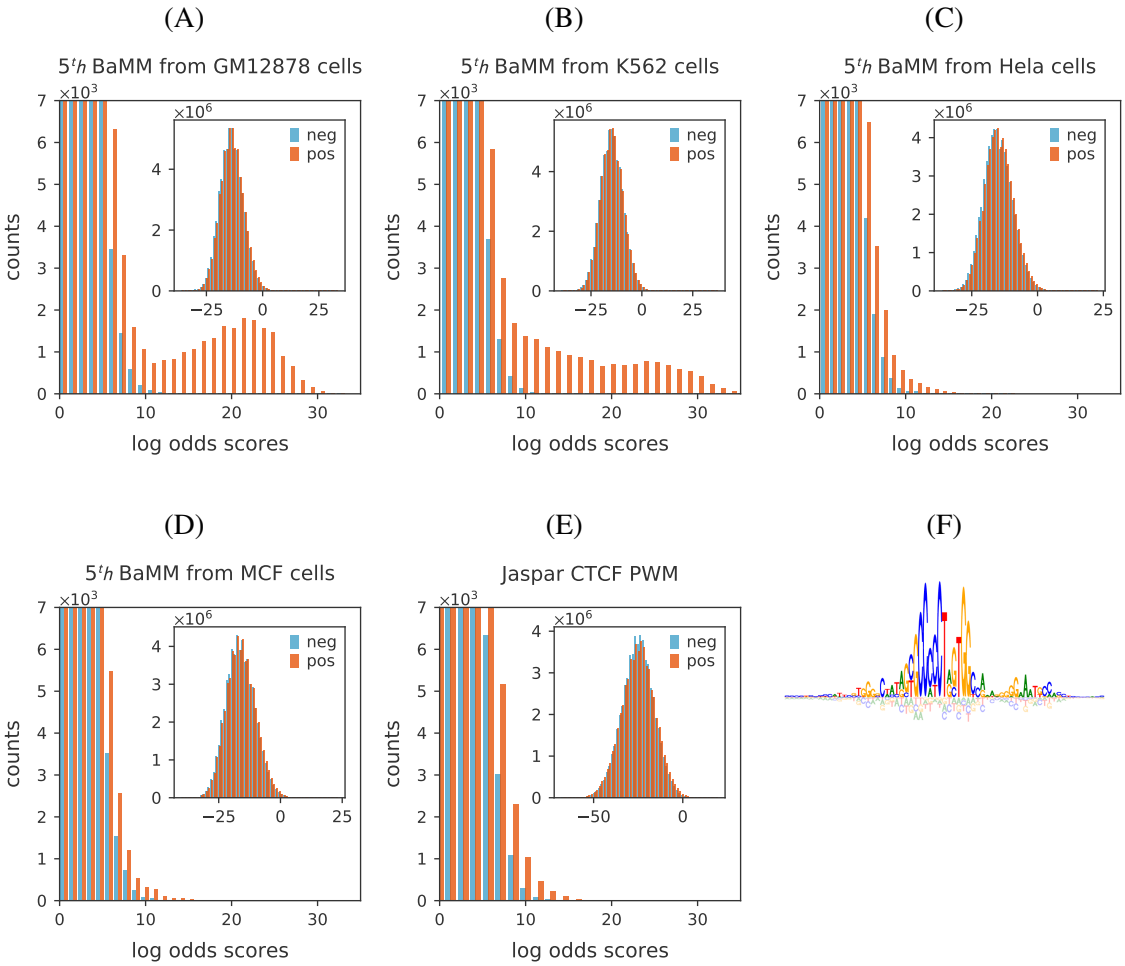


Fig. 3.9 **More CTCF binding sites are predicted by fifth-order BaMMs than PWM.** Panel (**A-E**) show the distribution of CTCF binding sites over motif scores on the whole human genome (hg19) on reversed negative sequences with fifth-order models learned from different cell lines and a PWM model. Panel (**F**) shows the motif logo of CTCF. Note that each distribution is sampled as 1% of all the positions due to the limit of data size.

The total number of CTCF binding sites predicted by a fifth-order BaMM from the GM12878 cell line is 1.5 million, with log-odds scores larger than 15. This prediction is 10-fold larger than the current estimated number of CTCF binding sites [65]. These predicted sites may consist of the weak binding sites which are recognized by the CTCF factors in some cell lines. Further investigation of these weak binding CTCF sites may illustrate the function characteristics of CTCF factors in more depth.

The major results are included in the following manuscript and paper:

3.1.3 Article: Bayesian Markov models improve the prediction of binding motifs beyond first-order without overfitting

Wanwan Ge, Markus Meier, Christian Roth and Johannes Söding*

The manuscript is prepared for submission.

Code availability

The source code is available for command-line versions of PEnGmotif and BaMMmotif2 and supported on Linux and Mac OS X:

PEnGmotif

PEnGmotif repository: github.com/soedinglab/PEnG-motif. For this study, I used parameters `-optimization_score MUTUAL_INFO -w 8 -threads 4`. The output is in MEME-like format. The motifs are sorted by their AvRec scores, and the best one was taken for the benchmark.

BaMMmotif2

BaMMmotif2 repository: github.com/soedinglab/BaMMmotif2. For this study, I seeded with the PWMs discovered by PEnGmotif and used parameters `-EM -k [k] -advanceEM -extend 2 2` for further optimization. `[k]` was chosen as 0, 1, 2 or 5 for each benchmark test. The output format is defined as BaMM format with extensions like `.ihbcp` and `.hbcp`.

Author contributions

Johannes Söding (JS) designed the algorithm. Wanwan Ge (WG) implemented the BaMM approach. Markus Meier (MM), Christian Roth (CR) and JS developed the PEnG approach. WG implemented the statistical approach and conducted all the benchmarks. WG and JS wrote the paper.

Bayesian Markov models improve the prediction of binding motifs beyond first order

Wanwan Ge, Markus Meier, Christian Roth, and Johannes Söding **

Quantitative and Computational Biology, Max Planck Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany

Received 2020-11-02; Revised 2020-12-15; Accepted 2021-03-22

ABSTRACT

Transcription factors (TFs) regulate gene expression by binding to specific DNA motifs. Accurate models for predicting binding affinities are crucial for quantitatively understanding transcriptional regulation. Motifs are commonly described by position weight matrices, which assume that each position contributes independently to the binding energy. Models that can learn dependencies between positions, for instance, induced by DNA structure preferences, have yielded markedly improved predictions for most TFs on *in vivo* data. However, they are more prone to overfit the data and to learn patterns merely correlated with rather than directly involved in TF binding. We present an improved, faster version of our Bayesian Markov model software, BaMMmotif2. We tested it with state-of-the-art motif discovery tools on a large collection of ChIP-seq and HT-SELEX datasets. BaMMmotif2 models of fifth-order achieved a median false-discovery-rate-averaged recall 13.6% and 12.2% higher than the next best tool on 427 ChIP-seq datasets and 164 HT-SELEX datasets, respectively, while being 8 to 1000 times faster. BaMMmotif2 models showed no signs of overtraining in cross-cell line and cross-platform tests, with similar improvements on the next-best tool. These results demonstrate that dependencies beyond first order clearly improve binding models for most TFs.

INTRODUCTION

Gene expression is regulated through the binding of transcription factors (TFs) to specific recognition motifs within promoter and enhancer DNA sequences. These binding motifs typically contain 6 to 12 only partially conserved bases (1, 2, 3). Learning quantitative models from experimental data that allow us to accurately predict the binding affinities of TFs to any given sequence is important for quantitatively predicting transcription rates from regulatory sequences.

The task of *de novo* motif discovery is to infer from experimental data a statistical or thermodynamic model that can then predict the binding affinity of a TF of interest for any sequence up to a constant (see Suppl. Methods subsection 1.2). Motif models can be inferred from numerous types of experiments (4). Common *in vivo* techniques are ChIP-seq (5) and bacterial-one-hybrid (6), while most modern *in vitro* approaches are SELEX-based (7, 8, 9). These measurements result in sets of hundreds to millions of bound sequences from which the binding motif model is deduced based on the statistical enrichment of binding sites compared to a background set of unbound sequences or a background model for random sequences.

The dominant model for describing the binding affinity of transcription factors to DNA target sequences has been the position weight matrix (PWM). This model assumes that the binding energy can be decomposed into a sum of contributions from each of the nucleotides in the binding site. By Boltzmann's law, this is equivalent to assuming statistical independence between nucleotides at different positions of the binding site. The PWM

*To whom correspondence should be addressed. Email: soeding@mpibpc.mpg.de

model has been enormously successful, because for the vast majority of transcription factors it achieves quite high accuracy for predicting the binding affinity of high-affinity binding sites with only $3W$ parameters for a binding site of W nucleotides. However, modeling the nucleotide inter-dependency often yields better motif predictions than PWMs (10, 11, 12). One reason is that the stacked, neighboring bases largely determine the physical properties of DNA, such as their equilibrium bending angle, minor groove width, propeller twist, or helical twist. The information on the geometric orientation of the bases propagates within the DNA for several positions before fading out, creating a dependence of the DNA physical properties on nucleotide pairs, triplets and longer k-mers. Since TFs recognize their target sites not only using hydrogen bonds but also using their structural fit, TF binding motifs show preferences depending on k-mer words (13), particularly in the flanking regions outside the hydrogen bonding core region (14). Furthermore, alternative binding modes of TFs (15, 16) can lead to poor performance of PWMs.

During the past decade, it has become increasingly evident that weak binding sites in enhancers and promoters play an important role in determining transcriptional activity (17, 18, 19, 20, 21), and PWMs have limitations to describe the affinities for weak binding sites accurately. Therefore, various more refined models have been developed that depart from the simplifying assumption of independence of motif positions (22, 23, 24). Prime among them are inhomogeneous Markov models of order k, in which the probability to observe a certain nucleotide at position i depends on the previous k nucleotides at $i-k$ to $i-1$. A zeroth-order Markov model is therefore equivalent to a PWM. Dinucleotide weight matrices (DWMs) are equivalent to first-order models, in which the probability of a nucleotide depends on its direct predecessor, and they have shown improved accuracy over PWMs (25, 26, 27).

For Markov models of higher order k, the large number of $W\times(4^{k+1}-1)$ parameters can lead to overfitting on the training data and hence bad predictive performance. To address this limitation, our group had proposed a special type of Markov model, the Bayesian Markov model (BaMM) (28), in which the probability for a nucleotide at position i of the motif, for example the last nucleotide in `ACTCG`, is estimated by adding to the actual counts of `ACTCG` pseudo counts based on how often the shorter $(k-1)$-mer `CTCG` has been observed in the binding sites. The probability for `CTCG` in turn is estimated by adding its counts to pseudo counts based on how often the word of length $k-1$, `TCG`, has been observed, and so forth. This procedure can be derived formally in a Bayesian framework with Dirichlet priors. Our software BaMMmotif indeed improved on previous PWM-based methods for *de novo* motif discovery and binding site prediction on *in vivo* data (28).

Here we present BaMMmotif2, an open-source software written entirely from scratch in C++. It contains a novel algorithm for its seed finding stage, which gives it greatly improved speed and slightly improved sensitivity in comparison to BaMMmotif. We improved the robustness of the BaMM-based motif refinement stage using sequence masking. BaMMmotif2 can also learn positional preference profiles for binding site locations from the training data.

Higher-order models have the ability to learn several low-order motifs overlaid on top of each other (29). It was therefore surmised that at least a part of the improvements of higher-order models on cross-validation benchmarks using ChIP-seq sequences could stem from learning not only the main binding motif of the ChIPped factor but also, overlaid, the binding motifs of cooperating factors whose binding sites tended to co-occur with it (18). This would of course defeat the purpose of learning the binding affinity of the ChIPped factor. In a different cell type, for instance, in which different co-binding factors are expressed, such a mixed motif might perform badly. It has also been suggested that more complex models could learn complex, nonspecific sequence biases characteristic of the measurement technique, which would allow them to be distinguished from the background sequences. These platform-dependent biases could result from the library preparation, amplification, and ligation biases (30).

We therefore designed a set of benchmark experiments with a focus on detecting such overfitting (Fig. 1): (I) 5-fold cross-validation on ChIP-seq and HT-SELEX data; (II) cross-cell-line validation on ChIP-seq data for the same TFs; (III) model training on ChIP-seq data and testing on HT-SELEX data for the same TFs and (IV) vice versa. Scheme (I) examines how the models generalize to unseen data, especially when data is limited.

Our results demonstrate that BaMMmotif2 does not show signs of overfitting but rather learns the binding affinity of only the factor of interest, and that BaMMmotif2 is the most sensitive and fastest tool among the ones tested here. Furthermore, BaMMmotif2 keeps improving the performance with increasing model orders and scales better with larger datasets.

MATERIALS AND METHODS

The BaMMmotif2 algorithm

BaMMmotif2 consists of a seeding stage and a motif refinement stage. The purpose of the seeding stage is to exhaustively identify motifs enriched in the input sequences in comparison to a second-order Markov background model trained also on the input sequences. Each of the motifs below a P-value cut-off is refined by the BaMM-based refinement stage.

The fast seeding stage. This method is described in detail in Supplementary Section 1.1. Briefly, we first count the number of occurrences of each non-degenerate W-mer word in $\{A,C,G,T\}^W$ ($W=8$ in this study) in the input sequences. From here on, we only inspect the count array and not the sequences anymore, making the runtime of the seeding stage almost independent of the input set size. By default, reverse complements are mapped to the alphabetically lower of the two W-mers.

For each W-mer, an enrichment z-value is calculated, which is the number of standard deviations with which the observed W-mer count surpasses its expected count. The expected count is calculated using a second-order homogeneous Markov model as a background sequence model, trained on the input sequences. Following the idea of (31), we determine all locally optimal w-mers. These are the W-mers with a better enrichment z-value than any of its direct neighbors one substitution away. We use each of the locally optimal w-mers to initialize a search for locally optimal W-mer patterns in the 10-letter IUPAC alphabet {A,C,G,T,S,W,R,Y,M,K,N}, where the last six letters stand for C or G, A or T, A or G, C or T, A or C, and G or T, respectively. For each such locally optimal IUPAC pattern, a PWM is derived from all matches in the input sequences to the degenerate pattern. The PWMs are then refined by applying the expectation-maximization (EM) algorithm in the multiple-occurrences-per-sequence (MOPS) model. We merge PWMs together that overlap by at least $W-2$ highly similar matrix columns. Finally, the PWMs are reranked by their AvRec scores (explained in the next section) and written into an output file in MEME format (32), which is passed to the refinement stage.

The refinement stage. The refinement stage is initialized with the motif occurrences found by the PWMs passed to it from the seeding stage. The length of the motif is extended by 2 bp on both ends by default to ensure that we do not miss information in the flanking regions. Each seed model is refined into an inhomogeneous Bayesian Markov model (BaMM) of order K using the EM algorithm (Supplementary Section 1.5). Each such refinement is independent of the refinements of the other seed motifs. Motifs can overlap with motifs already discovered in a previous refinement stage. A BaMM is an interpolated Markov model in which the conditional probability of base $x_i \in \{A,C,G,T\}$ at position i is calculated by combining the counts $n_i(x_{i-k:\,i})$ of k-mer $x_{i-k:\,i}$ with pseudo counts estimated from lower-order probabilities $p_i^{\mathrm{BaMM}}(x_i|x_{i-k+1:\,i-1})$:

$$p_i^{\mathrm{BaMM}}(x_i|x_{i-k:\,i-1}) = \frac{n_i(x_{i-k:\,i})+\alpha_k\, p_i^{\mathrm{BaMM}}(x_i|x_{i-k+1:\,i-1})}{n_{i-1}(x_{i-k:\,i-1})+\alpha_k}.$$

Here, the hyper-parameter α_k determines how much weight to give to the lower-order. The probabilities of order $k-1$ are again obtained by adding to the k-mer count the pseudo counts from order $k-2$, and so on down to order 0. In this way, when the number of occurrences observed for $(k+1)$-mer $x_{i-k:i}$ is much smaller than the number of pseudo counts $\alpha_k \times p_i^{\mathrm{BaMM}}(x_i|x_{i-k+1:\,i-1})$, the higher order falls back to the lower order: $p_i^{\mathrm{BaMM}}(x_i|x_{i-k:\,i-1}) \approx p_i^{\mathrm{BaMM}}(x_i|x_{i-k+1:\,i-1})$. In this way, BaMMs adapt the order that is learned in a data- and motif position-specific fashion to the amount of data (k-mer counts) available. We assume that the correlation between nearby bases declines with their distance. This is reflected in the pseudo-parameters α_k increasing with order k. For BaMMmotif2, we kept the same setting as in BaMMmotif, $\alpha_k = 7\times 3^k$.

The motif model is optimized with the EM algorithm by maximizing the likelihood of the input sequences assuming zero or one motif occurrence per sequence (the ZOOPS model). It models the bound sequence using a Kth-order inhomogeneous BaMM $p^K_{\mathrm{motif}}(\mathbf{x})$ (where $\mathbf{x}=x_{1:W}$ is the binding site), and models the other unbound sequence regions using a K'th-order homogeneous BaMM $p^{K'}_{\mathrm{bg}}(\mathbf{x})$ (K' is 2 by default). This background sequence model is trained by default on the input sequences. Potential binding site sequences $\mathbf{x}$ are ranked by their score $S(\mathbf{x}) = \log(p^K_{\mathrm{motif}}(\mathbf{x})/p^{K'}_{\mathrm{bg}}(\mathbf{x}))$. In the weak binding limit, this score is proportional to the Gibbs free energy ΔG of binding (Supplementary Section 1.5).

The ZOOPS model is used for its computational convenience. Since actually more than one protein can bind to a sequence, the many-motif-occurrences-per-sequence model would be more appropriate. If the protein can bind in more than one conformation and thereby with more than one distinct motif, ideally all distinct motifs should be modeled and learned at once, using dynamic programming to sum over all possible binding configurations (33).

Learning positional binding preferences. BaMMmotif2 can learn the positional binding preferences for enriched motifs with respect to the center of the input sequences. By aligning the sequences around some anchor feature, such as a transcriptional start site, a 3' splice site, or a binding site of some other transcription factor, the distance preference between enriched motifs and the reference feature can be learned. We parameterize the positional probability distribution with one parameter per position and ensure smoothness by adding L_2 penalties for the differences between successive sequence positions (Supplementary Section 1.5).

Masking sequences during the motif refinement stage. Sequences from *in vivo* experiments such as ChIP-seq commonly contain several distinct motifs from other TFs that together co-regulate their target genes. This can create two types of problems during the refinement stage. First, instead of refining the motif from the seeding stage, the model in some cases tends to learn two or even more motifs in the same higher-order model, as this often improves the likelihood on the training data. Second, if the seed motif is less enriched or less informative than other motifs in the positive sequence set, the model can switch from the seed motif to these other motifs. In this way, the weaker motif is not discovered at all. To avoid these two problems, we introduced a masking step in the EM optimization. We score all possible motif start positions in the input sequences using the PWM passed from the seeding stage to the refinement stage. We mask out all but the top X% of positions ($X = 5$ in this study) and ignore these positions in the EM iterations of the refinement stage.

Motif assessment using average recall (AvRec)

To assess the performance of a classifier such as a motif model, one often plots the true positive predictions (TP) versus the false positive predictions (FP) over all score thresholds. Normalizing FPs and TPs to a maximum of 1 by plotting the true positive rate TPR = TP/Positives versus the false positive rate FPR = FP/Negatives yields the receiver operating curve (ROC). The often-used area under the ROC curve (AUC) is not a good quality measure for a motif model because in many applications the fraction of positive sequences (those carrying the motif) is much smaller than the number of negative sequences. When scanning the human genome for CTCF binding motifs in windows of 100 bp, for example, the ratio is about 1:30. At this ratio, a false discovery rate FDR = TP/(TP+FP) below 50% requires a ratio FPR/TPR < 1/30. So 29/30 = 97% of the ROC plot, the part with FDR > 50%, would be irrelevant. A predictor could have an AUC of 95% and never reach an FDR below 50%.

We therefore previously developed the Average Recall (AvRec) score (34), which averages the recall (the same as true positive rate and sensitivity) over a range of TP:FP ratios from 1:1, corresponding to FDR = 0.5, to 1:100, corresponding to FDR = 1/101 (Fig. 2A). The AvRec score therefore considers the range of FDR most relevant in practice and has the additional benefit that a different positive-to-negative ratio than 1 simply results in a vertical shift of the AvRec curve on the logarithmic y axis.

To calculate the AvRec score, we first simulated 10-fold more negative than positive sequences using a second-order Markov background model learned on the positive set. We computed the motif scores $S(x_{i:i+W-1}) = \log_2\left(p^K_{\text{motif}}(x_{i:i+W-1}) / p^{K'}_{\text{bg}}(x_{i:i+W-1})\right)$ for all possible binding positions i (excluding the masked positions) and took the best score for each sequence. All sequences are sorted by descending score. The false positive count FP is the cumulative number of sequences from the negative set above the score cut-off, and TP is the cumulative number of positive sequences above the score cut-off.

Benchmark design

The performance of BaMMmotif2 was evaluated together with five state-of-the-art motif discovery tools, MEME (32) as the most cited tool, CisFinder (35) for its speed and ability to run on large datasets, ChIPMunk (36) and diChIPMunk (27), which are used for generating the PWMs and dinucleotide PWMs in the HOCOMOCO database (37), and InMoDe (24), which can learn inhomogeneous Markov models of order 2 and beyond.

The processing of the ChIP-seq and HT-SELEX data is described in detail in Supplemental Material II. The motif discovery tools were run on the input sequence sets with default parameters, and four CPU cores were used for tools that could be parallelized (CisFinder, MEME, and BaMMmotif2). For tools that learn multiple motif models from one dataset, the motif models ranked top by the tools were benchmarked.

To assess the model performance over the given sequences, we first performed the benchmark on both human ChIP-seq (38) and HT-SELEX datasets (39) using 5-fold cross-validation (Fig. 1 I). The cross-cell-line validation was applied to ChIP-seq data (Fig. 1 II) and the cross-platform validations were applied to both ChIP-seq data HT-SELEX data (Fig. 1 III and IV). A more detailed description including tool settings can be found in Supplementary Section II.

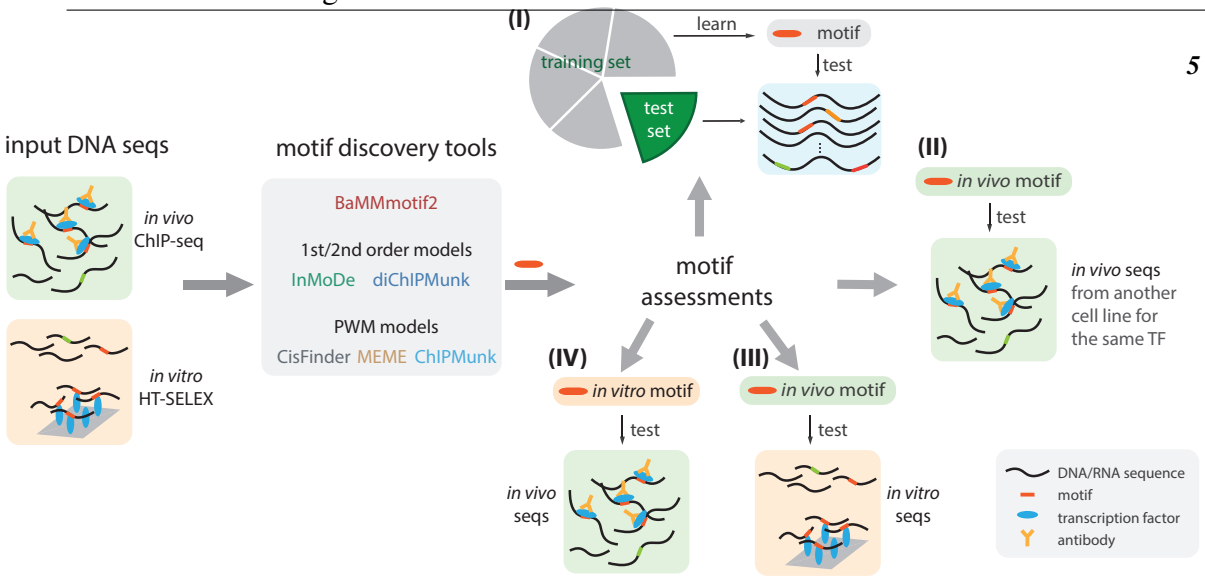


Figure 1. Benchmark pipeline for *de novo* motif discovery. Five state-of-the-art motif discovery tools and BaMMmotif2 learned motif models on *in vivo* and *in vitro* transcription factor binding datasets. The learned models were then assessed (I) by 5-fold cross-validation on the same type of data, (II) by cross-cell-line validation, and (III, IV) by cross-platform validations.

RESULTS

Model performance on *in vivo* and *in vitro* data

We learned *de novo* motifs with each of the six tools on 427 ChIP-seq datasets for 93 transcription factors from the ENCODE project (38) and evaluated their performance using 5-fold cross-validation (Fig. 1I).

As an example, we compare in Fig. 2A and 2B the AvRec plot of a fifth-order BaMM with a second-order InMoDe model for the Elf2 motif, trained and tested on 5000 sequences of length 208 bp via 5-fold cross-validation. At a positives-to-negatives ratio of 1:1 (bold blue line) and a TP:FP-ratio of 10:1 (see y axis, corresponding to an FDR of $1/11$), the BaMMmotif2 model achieves a recall of 0.81 and the InMoDe model achieves 0.69. At a positives-to-negatives ratio of 1:10 and a TP:FP-ratio of 10:1 (broken blue line), or, equivalently, at a positives-to-negatives ratio of 1:1 and a TP:FP-ratio of 100:1, the models achieve recalls of 0.12 and 0.13, respectively. When comparing AvRec scores between fifth-order BaMMs with second-order models from InMoDe across all 427 ChIP-seq datasets, BaMMs attain higher AvRec scores for 415 (97%) of the datasets, and the median AvRec of BaMMs is 13.6% higher than the one of InMoDe models (Fig. 2C). This improvement is universal across TF domain families (40) (Fig. 2C).

Overall, the PWM-based tools, CisFinder, MEME, and ChIPMunk, are outperformed by the tools using higher-order models. BaMMmotif2 with first-order models performs on par with InMoDe and better than the first-order tools such as diChIPMunk. Fifth-order BaMMs achieve even better AvRec scores, as seen in the box plots and AvRec cumulative distributions of Fig. 2D and 2E, and in one-on-one comparisons in Fig. S2A. We also compared BaMMmotif2 with our previous tool BaMMmotif (28). BaMMmotif2 is 10 times faster while being slightly more sensitive (Fig. S3).

Tools that learn higher-order Markov models can learn several motifs in one model, profiting from signals that are merely correlated with the real binding sites (29, 41). To find out whether BaMMs are affected or not, we introduced a masking step in the initial iteration of the EM algorithm (see Materials and Methods). We restrict the model refinement with the higher order BaMM to the 5 % potential motif positions with the highest scores scanned by the seeding PWM. In this way, we avoid overfitting and also speed up the refinement by a factor of 10. However, this robustness is paid by a loss in motif model performance (Fig. S4 and S5). The performance decrease could be caused in part by the limitation of being unable to select better sites during the refinement that were too different from the seeding motif, and in part because sometimes the BaMMs would otherwise have learned more than one distinct motif in a single model. To be on the conservative and robust side, we adopted the masking step in BaMMmotif2 for all

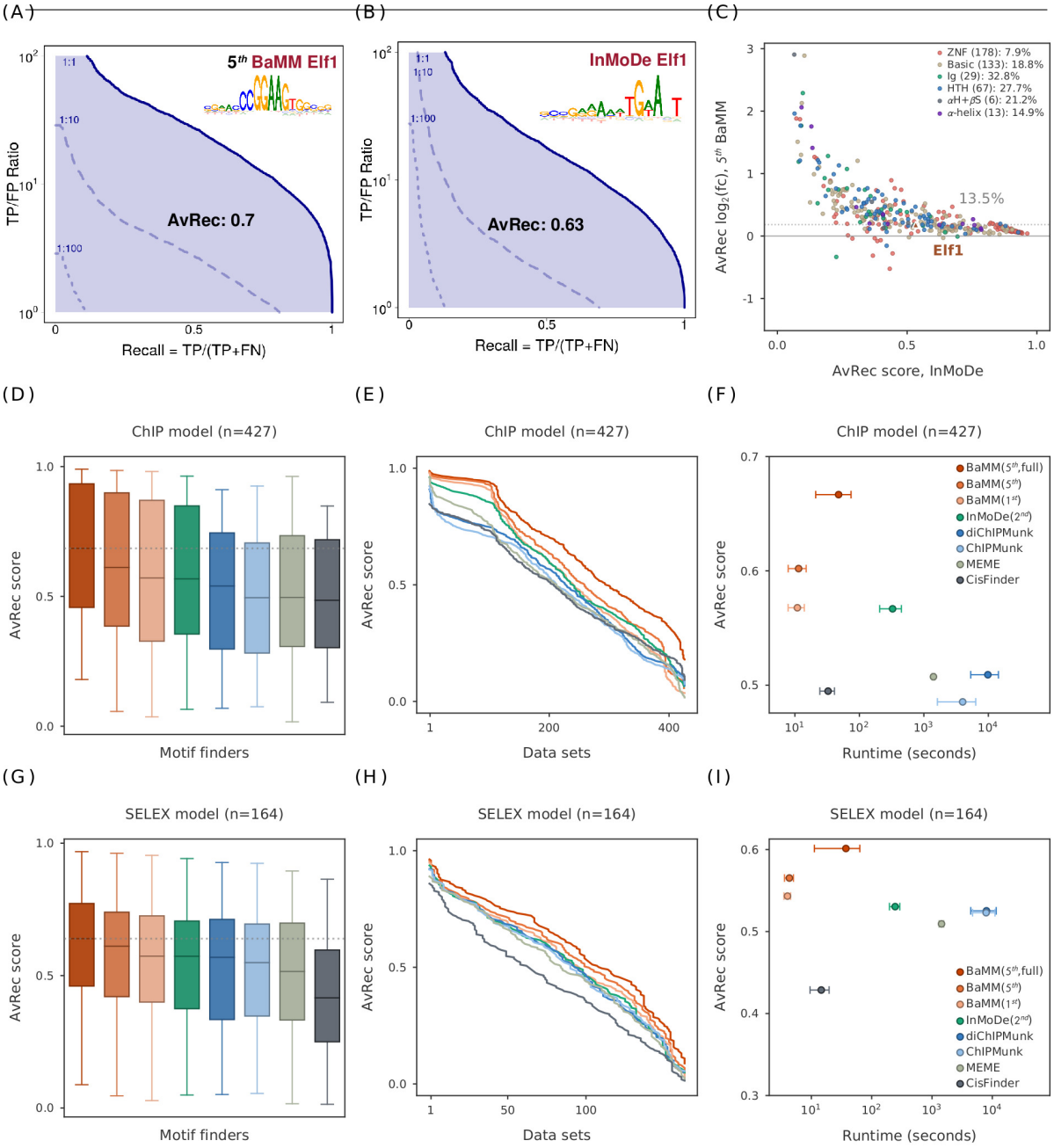


Figure 2. Performance of *de novo* motif discovery tools on *in vivo* and *in vitro* datasets. (**A**) AvRec analysis for fifth-order BaMM on the Elf1 ENCODE dataset. The AvRec is the recall averaged in log space over TP-to-FP ratios between 10^0 and 10^2. This ratio range corresponds to a precision between $1/(1+1)$ and $100/(1+100)=0.99$. Bold line: 1:1 ratio of positives to negatives. At 1:10 ratio (dashed) and 1:100 (dotted), the curves are shifted down by a factor of 10 and 100, respectively. Inset: motif logo of Elf1. (**B**) Same as (**A**) for the InMoDe model of Elf1. (**C**) $\log_2$ of AvRec fold change between fifth-order BaMMmotif2 and InMoDe models versus the AvRec of InMoDe. Each dot represents one dataset. Elf1 is highlighted in a brown triangle. Dot colors represent different TF superfamilies defined by (40). ZNF: Zinc-finger DNA-binding domains, Basic: Basic domains, Ig: Immunoglobulin fold, HTH: Helix-turn-helix domains, αH+βS: alpha-helices exposed by beta-structures, αH: Other all-alpha-helical DNA-binding domains. The median AvRec fold change and the number of motifs are shown in the legend. The overall median $\log_2$ fold change is 13.5%. (**D**) AvRec distributions as box plot, with boxes indicating 25%/75% quantiles and whiskers 95%/5% quantiles. Color code: see the legend in (**F**). (**E**) Cumulative distribution of AvRec scores on the 427 datasets. (**F**) Average runtime per dataset on four cores versus the median AvRec score. InMoDe and (di)ChIPMunk are not parallelized and ran on a single core. Whiskers: ±1 standard deviation. BaMM (5^{th}, full): no masking step. (**G-I**) Analogous to (**D-F**) but for 164 HT-SELEX datasets from the Taipale lab (39).

our benchmarks in this study, unless explicitly stated otherwise.

Next, we assessed the performances of selected tools on 164 *in vitro* HT-SELEX datasets for 164 TFs (39). Each dataset contains long oligomers of 200 bp. We also sampled 10-fold background sequences using the trimer frequencies from the same input set for estimating true negatives.

For the *in vitro* benchmark we observed overall similar trends as on the ChIP-seq data (Fig. 2G-I). CisFinder tends to learn longer motifs than the other tools, which probably helped it on the ChIP-seq data but hurt its performance on the HT-SELEX data. The BaMMs learned without masking (BaMM 5^{th}, full; red) gained only 5% on the masked version (BaMM 5^{th}; orange), whereas the gain had been 12% on the ChIP-seq data. This comparison shows that, on the ChIP-seq data, the fifth-order BaMMs trained without masking indeed tend to learn also motifs of co-occuring TFs that help to distinguish positive from negative sequences. If the goal is to learn the pure binding affinity of the ChIPped TF, masking should therefore be turned on for *in vivo* data.

Assessing consistency of motif models across cell lines

ChIP-seq measurements have cell-type-specific biases associated with difference in chromatin accessibility, in particular of enhancers and promoters, and differences in TF concentrations (42). A motif model that predicts only the binding affinity of the ChIPped TF should also perform well in predicting binding sites of the factor in other cell lines, whereas a motif model that has learned also motifs of co-occurring TFs and other sequence features with no direct effect on the binding affinity of the main TF should generalize badly to other cell lines in which different TFs will often co-occur with the ChIPped TF.

We therefore conducted a cross-cell line benchmark on *in vivo* data. We assessed the performance of models learned on ChIP-seq data from one cell line and tested on ChIP-seq data of the same TF from another cell line. We found 119 pairs of ChIP-seq datasets in the ENCODE database in which the same TF had been ChIPped in two different cell lines. We trained the model on one dataset and tested it on the other, and vice versa, resulting in 238 AvRec scores (Fig. 3).

Remarkably, the AvRec scores are around 0.2 lower for all tools than the AvRec scores in Fig. 2D obtained when training and testing in the same cell lines, with the PWM-based tools going from AvRec 0.5 to as low as 0.3. This quite dramatic decrease indicates that all models, even the simple PWMs, do not perform well for predicting bound sequences in another cell line. Remarkably, except for InMoDe, the predictive power of the higher-order models does not suffer more than that of the PWMs. This indicates that the higher-order models (except InMoDe) do not tend to overfit to sequence features that are specific to one cell line, such as co-occurring TFs. It is surprising that the fifth-order BaMMs trained without masking maintain or even improve their edge on the other models, despite our expectation that they would be the most prone to overfit on cell type-specific features.

In vitro models predict *in vivo* binding and vice versa

Each measurement for detecting TF-DNA interactions has its own biases. ChIP-seq has biases from sequence-dependent PCR amplification, cell-type-specific sonication bias, and chromatin structure(43, 44, 45), while HT-SELEX has biased nucleotide compositions and depleted palindromes as a result of the library preparation, as well as sequence carry-over bias in selection cycles (46, 47). These biases can give optimistic results even in the cross-cell-line benchmark because the model can be overtrained on genomic features that are identical or similar in both cell lines.

To assess how much models base their predictions on technical biases that would improve their performance when tested on the same platform but decrease their performance when tested on a different platform, we performed two cross-platform benchmarks.

First, for each of the tools, we trained a motif model on each of the 140 ChIP-seq datasets for which an HT-SELEX dataset for the same TF, but not necessarily from the same cell line, was available. We discovered that several datasets were of too low quality to give reliable models, and some HT-SELEX datasets showed signs of having had the identity of the TF switched. We therefore selected the 92 ChIP-seq datasets for which at least one of 8 tools achieved an AvRec score of $\geq$ 0.1. The first- and fifth-order BaMMs achieve better accuracies than the PWM-based models (Fig. 4A and 4B).

Second, for each of the tools we trained a motif model on each of the 82 HT-SELEX datasets for which a ChIP-seq dataset with the same TF was available. We selected the HT-SELEX datasets for which at least one of the 8 tools achieved an AvRec score of $\geq$ 0.1. Again, BaMMs achieved the best AvRec scores. However, we observed no major improvements from first to fifth order (Fig. 4C and 4D). This time, the improvements over PWM-based models are minor. ChIPMunk and diChIPMunk fared badly because they only predict one motif per dataset, while other tools

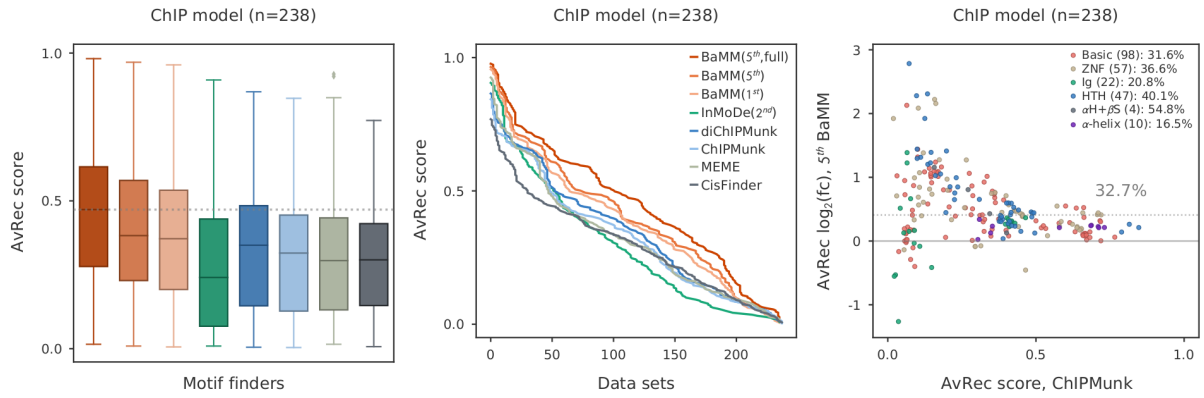


Figure 3. Cross-cell-line validation. 119 pairs of ENCODE datasets were used in this benchmark in which the same TF had been ChIPped in different cell lines. (**A**) AvRec distributions for 2×119 models that were tested on a ChIP-seq dataset from a different cell line than they were trained on. (**B**) Cumulative distributions of AvRec scores. (**C**) Log_2 fold change in AvRec between fifth-order BaMMs and ChIPMunk for each of the 238 datasets. The median improvement is 32.7%. Same legend as Fig 2C.

generate several motif candidates and the best one is chosen for comparison.

BaMMs learned similar information content in the first-order on ChIP-seq and on HT-SELEX data while showing no tendency to learn systematic biases of these platforms (Fig. S7). This demonstrates how the information in the first-order can help to improve cross-platform predictions.

Extended flanking regions increase motif prediction accuracy

Various studies have shown that the flanking regions outside of the core binding sites affect TF binding, by affecting DNA shape preferences or by harboring binding sites of co-cooperatively binding TFs at variable spacings (14, 48, 49, 50). Therefore, we investigated the impact of extending the core motifs, by adding two or four nucleotides on each side in the seeding motifs and refining the extended motifs with BaMMmotif2.

We find that for BaMMs trained on ChIP-seq datasets, extending the models by 2×2 or 2×4 positions indeed improves the motif model performance across all orders, and more so with increasing order (Fig. 5A). The improvement from no added positions to 2×4 bp added is by 3% for zeroth order BaMMs (PWMs) and by 11% for fifth-order BaMMs (Fig. 5B and Fig. S8A). This indicates that flanking regions carry information mostly in the higher orders and not much in preferences for specific nucleotides.

It is not clear, however, if these improvements are due to DNA shape preferences that are reflected by preferences for certain di- and tri-nucleotides or by other sequence features of the genomic sequences such as motifs of co-occurring TFs. We therefore repeated the same analysis on HT-SELEX data. We restricted ourselves to long oligonucleotides of 200 bp because short oligonucleotides of 20 bp to 40 bp might not reflect well enough the physical properties of genomic DNA.

The results on the HT-SELEX data are very similar to those on ChIP-seq data (Fig. 5B and 5D). Again, PWMs gain much less AvRec score through 2×4bp extensions than fifth-order BaMMs (1.3% versus 8%, shown in Fig. S8B and Fig. 5D). This result confirms that the features picked up by the higher orders are not chiefly ones that are specific to genomic sequences but are also learned on *in vitro*-selected sequences and are therefore likely to be associated with DNA structural preferences.

Learning positional binding preferences

Motifs often have certain positional preferences with regard to other motifs or genomic landmarks such as transcription start sites. Therefore, we introduced the possibility to learn the probability distribution of motif positions from the input data (Fig. S1A). Learning the positional distribution of motifs around ChIP-seq peak positions did not improve the median motif performances (Fig. S1B and S1C), probably because the information content of the positional distribution is very low when the the distribution is not much narrower than the window size. (The information content can be calculated as the difference between the entropies of the two positional distributions.) The positional preference is likely to have a positive impact when positioning effects are stronger, such as for

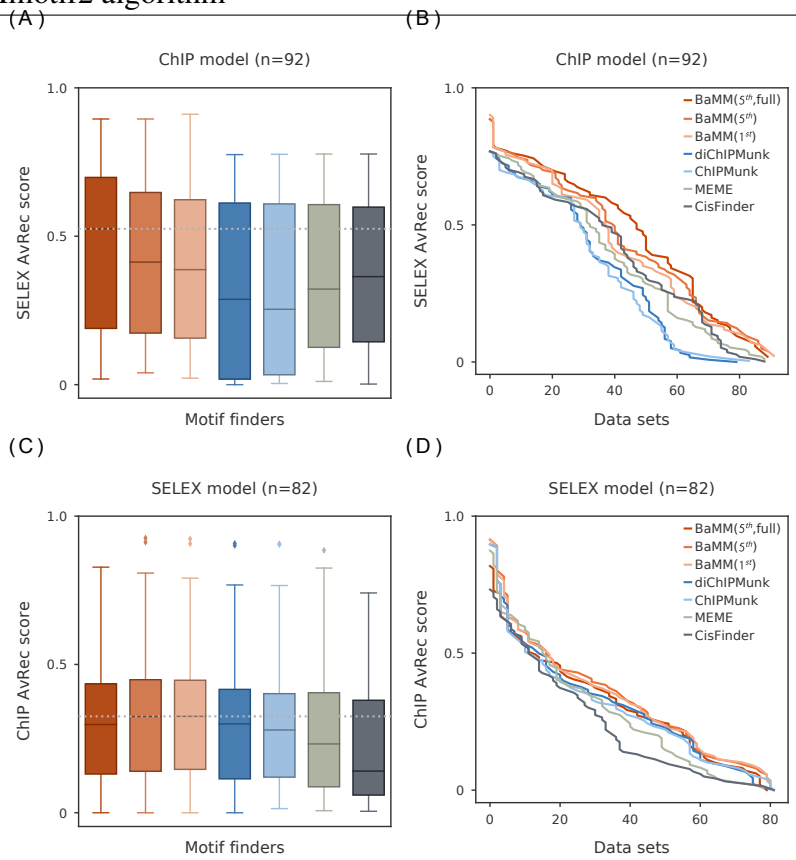


Figure 4. Cross-platform validation. (**A,B**) AvRec distributions and cumulative distributions for 92 models trained on ChIP-seq datasets and tested on HT-SELEX datasets for the same TFs using different tools. (**C,D**) Same as (A,B) for 82 motif models but trained on HT-SELEX datasets and tested on ChIP-seq datasets.

splicing motifs around splice sites, core promoter motifs around transcription start sites, or TF binding sites of cooperatively binding TFs.

DISCUSSION

We presented BaMMmotif2, a fast and accurate *de novo* motif discovery algorithm for large-scale transcriptomic data. BaMMmotif2 builds on our earlier theory of Bayesian Markov models (BaMMs) implemented in BaMMmotif. BaMMs employ pseudocounts from model order $k-1$ to stabilize the estimation of the conditional probabilities for order k, for all orders k from 1 to the maximum order (five in this study). In this way, they can learn higher orders if a sufficient number of k-mer counts was observed to estimate them but otherwise fall back to a lower order that can still be estimated safely.

BaMMmotif2 was written from scratch in C++ using explicit AVX2 vectorization and multi-core parallelization. We developed a novel, fast seeding method to find enriched patterns that scales almost independently of the input set size. We also added a masking step to force the refinement stage to only refine the seed motifs and prevent it from learning in addition other predictive features such as co-occurring motifs of other TFs or experimental sequence biases. We also developed a Bayesian approach to learn position binding preferences from the input data.

By their sheer number, ChIP-seq datasets are the dominant source of information for TF binding affinities. Therefore, most benchmark comparisons of *de novo* motif discovery tools have been performed exclusively or predominantly on ChIP-seq data. However, for assessing the quality of models more complex and informative than PWMs, such as higher-order Markov models and mixture models, ChIP-seq data are problematic for several reasons. First, they often have complex sequence biases (42), which higher-order models can learn to distinguish

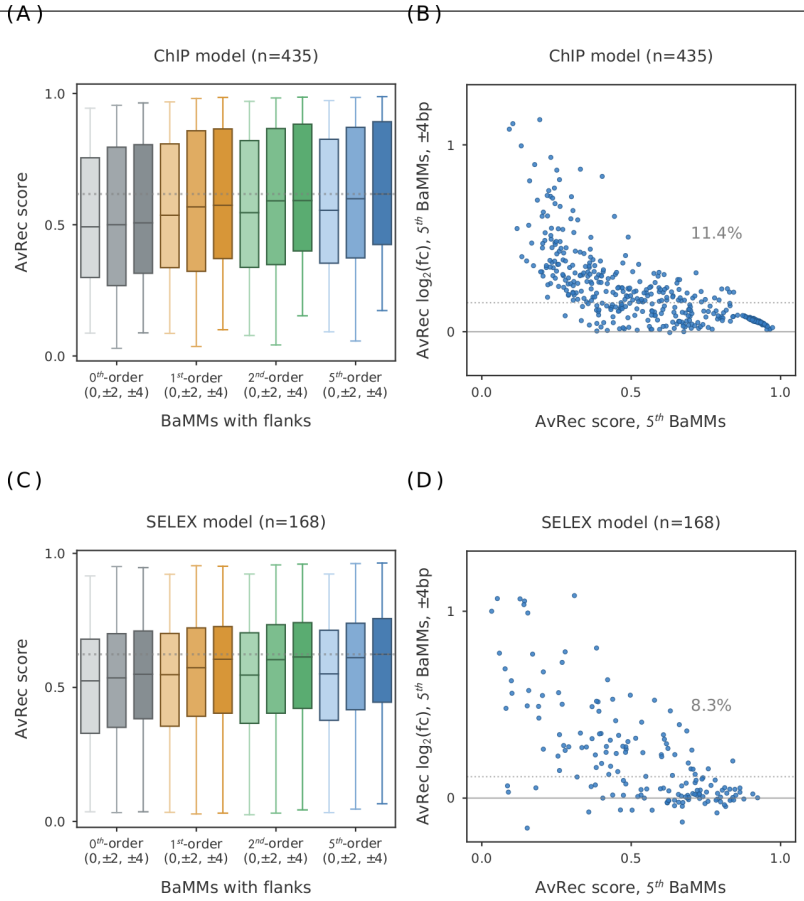


Figure 5. Extending the core motif by flanking positions improves motif performance. AvRec of BaMMs with different numbers of flanking positions added to the core motif, tested by 5-fold cross-validation. (**A**) AvRec distribution on 435 ChIP-seq datasets for models of order 0, 1, 2, and 5, each for three sizes of flanking regions: 0 bp, ± 2 bp, and ± 4 bp. (**B**) Log_2 of fold change between fifth-order BaMMmotif2 models with ± 4 bp flanking positions and no added flanking positions. The median AvRec increase is 11.4 %. (**C,D**) Same as (A,B) for 168 HT-SELEX datasets.

from negative sequences generated with random background models. To alleviate this problem, second order background models should be used, but even this might be insufficient to eliminate learning generic sequence biases of the ChIPped versus random sequences. Second, sequences in ChIP-seq peaks usually contain in addition to the motif of the ChIPped TF the binding motifs of co-binding factors (41). Complex models can improve their predictive performance by scoring sequences highly that contain any of these co-occuring motifs. This is possible even within a short motif length by learning the motifs superposed with each other, with the higher orders preventing mixing and blurring of motifs (29). Although improving the apparent model performance, such models do not describe faithfully the binding affinity of the ChIPped factor.

Our goal was to compare PWM-based motif discovery tools with tools employing more complex models: dinucleotide weight matrices, parsimonious context trees, and BaMMs. We therefore set up a cross-cell line benchmark to assess how well the motif models learned in one cell line can predict binding in another cell line. Furthermore, we conducted a cross-platform benchmark, in which we trained the models on ChIP-seq data and tested them on HT-SELEX data, and vice versa. The results show that among the tested tools, those with more complex models still tend to perform better in these benchmarks, albeit with smaller improvements over the PWM-based tools. The improvements from higher orders were particularly marked for the BaMMs. So, most of the information in higher orders seems to be transferable between cell lines and measurement platforms.

Even though we did not see clear signs of overfitting in our BaMMs, we introduced sequence masking as a precaution against overfitting to other motifs and technology- or cell line-dependent sequence biases. We use the seed PWM to mask out all but the top-scoring 5% of positions, and we train the higher-order BaMM only on the remaining 5%. We thereby ensure that only sequence regions that actually carry the seed motif can be learned by the BaMM. The performance drop between training fifth-order BaMMs with and without masking was 8% on HT-SELEX data and 12% on ChIP-seq data (Fig. 2D,G, Fig. S4). This indicates that if higher-order BaMMs profit from learning co-occurring motifs at all, the effect on their performance is quite limited.

Still, if the goal is to learn binding affinities and not just predict motifs from *in vivo* sequence data, we recommend to run BaMMmotif2 with the masking, because BaMMs can learn several similar motifs in one single model, such as bipartite motifs with a variable-length spacer or motifs of mono- and dimeric binding modes of a transcription factor. The masking option controls how closely the refined motif has to stay to the seed motif. For instance, masking helps to learn the correct partially related motifs for FoxA2 factor, when training 5th-order BaMMs on a ChIP-seq data (Suppl. Fig. S11C). Whether these similar motifs are learned in a single model or are split into two models can vary from case to case. If users want to learn motifs separately, it is therefore recommended to use masking and to experiment with even stricter masking than the default 95%.

On *in vitro* data, masking is not necessary and in order to make use of the 5% improvement we recommend to run BaMMmotif2 without masking. However, even with masking the fifth-order BaMMs still perform competitively with the state-of-the-art tools while being significantly faster.

Transcription factors combine base- with DNA shape readout (13). Instead of studying the TF-DNA binding using only the sequence features, some models utilize DNA shape features predicted from the sequence to enhance motif models (51, 52, 53). The shape descriptors these tools use, like minor groove width, helical tilt and bent, or propeller tilt, are predicted from five-mer tables computed using molecular dynamics calculations. Given enough data, it is therefore evident that higher-order models such as BaMMs can learn these DNA structural preferences implicitly, yet are not limited to the pre-defined shape descriptors.

In recent years, deep learning approaches have become popular for learning motif models with very good predictive performance (51, 54, 55). Such models usually take advantage of contextual information such as co-occurring motifs, which increases their predictive power but serves a different purpose than the models we discuss here: learning a model for the sequence dependence of the binding affinity of a factor. In addition, BaMMs have the advantage or being conceptionally simple and interpretable in terms of k-mer dependent energy terms.

In conclusion, we have shown that higher order models for binding motifs improved binding site predictions on a large collection of ChIP-seq and HT-SELEX datasets, both in cross-validated setting and when training and testing on different experimental platforms and cell lines. Importantly, clear improvements in predictive performance are even seen beyond first order models: BaMMs of fifth order show a solidly improved performance across the bench over the tested state of the art tools, while being significantly faster.

AVAILABILITY

Data

ENCODE database. We evaluated the performance of selected algorithms on human ChIP-seq datasets from the ENCODE portal (38) until March 2020. In total, there are 435 datasets for 93 distinct transcription factors. The top 5000 peak regions sorted by their signal value are selected for each dataset when peaks are more than 5000, and all peaks are chosen if there are fewer than 5000 peaks. Positive sequences are extracted ±104 bp around the peak summits. Background sequences are sampled by the trimer frequencies from positive sequences, with the same lengths as positive sequences and 10 times the amount of positive sequences. 8 datasets are excluded from all the results because diChIPMunk fails to learn models within 3 hours.

HT-SELEX datasets. For HT-SELEX data, we downloaded 164 datasets with 200 bp-long oligomers from Zhu et al. (39), which are deposited in the European Nucleotide Archive (ENA) under the accession PRJEB22684. Each dataset represents one non-redundant transcription factor. For each dataset, we selected 5000 sequences from each selection round without any sorting.

The HT-SELEX data contain reads from at least four selection cycles, and the measured binding affinity iteratively increases with the cycles. Thus, we chose the sequences from the fourth selection rounds with detected high affinities for motif training and testing in the main paper. Since ChIPMunk and diChIPMunk took longer than 2 hours to run on the full datasets,

we selected 5000 sequences out of the millions of reads as training and test sequences. To examine the power of BaMMs in learning the weak binding sites, we also used sequences from the second and third selection rounds. Background sequences are sampled in the same way as described previously.

Software and parameters

The new version of BaMMmotif2 software is implemented in C++ and Python3. The code is licensed under GPLv3 and freely accessible without registration at github.com/soedinglab/PEnG-motif, and github.com/soedinglab/BaMMmotif2, and supported on Linux and MacOS. They are also integrated into our webserver (34).

Results and analysis scripts

The analysis scripts are available in Jupyter Notebook format at github.com/soedinglab/bamm-benchmark.

FUNDING

This work was supported by the DFG SPP1935 grant CR 117/6–1 and the International Max Planck Research School for Genome Science (IMPRS-GS). Funding for open access charge: institutional.

ACKNOWLEDGEMENTS

We thank Matthias Siebert and Anja Kiesel for help with designing the code structure of BaMMmotif2, the members of the Söding lab, especially Eli Levy Karin, Saikat Banerjee, Milot Mirdita and Ruoshi Zhang for discussions, and the genomics research community for sharing their data.

Author Contribution: W.G. developed the BaMMmotif2 software. M.M., C.R. and J.S. developed the seeding stage software (PEnG). W.G. implemented the statistical approach and conducted all the benchmarks. W.G. and J.S. wrote the manuscript. J.S. supervised the research.

Conflict of interest statement. None declared.

REFERENCES

1. Serfling, E., Jasin, M., and Schaffner, W. (1985) Enhancers and eukaryotic gene transcription. *Trends Genet.,* **1**, 224–230.
2. Argos, P. (1988) A sequence motif in many polymerases. *Nucleic Acids Res.,* **16**(21), 9909–9916.
3. Mitchell, P. J. and Tjian, R. (1989) Transcriptional regulation in mammalian cells by sequence-specific DNA binding proteins. *Science,* **245**(4916), 371–378.
4. Jolma, A. and Taipale, J. (2011) Methods for analysis of transcription factor DNA-binding specificity in vitro. In *A Handbook of Transcription Factors* pp. 155–173 Springer.
5. Robertson, G., Hirst, M., Bainbridge, M., Bilenky, M., Zhao, Y., Zeng, T., Euskirchen, G., Bernier, B., Varhol, R., Delaney, A., et al. (2007) Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nat. Methods,* **4**(8), 651–657.
6. Meng, X., Brodsky, M. H., and Wolfe, S. A. (2005) A bacterial one-hybrid system for determining the DNA-binding specificity of transcription factors. *Nat. Biotechnol.,* **23**(8), 988–994.
7. Jolma, A., Yan, J., Whitington, T., Toivonen, J., Nitta, K. R., Rastas, P., Morgunova, E., Enge, M., Taipale, M., Wei, G., et al. (2013) DNA-binding specificities of human transcription factors. *Cell,* **152**(1-2), 327–339.
8. Riley, T. R., Slattery, M., Abe, N., Rastogi, C., Liu, D., Mann, R. S., and Bussemaker, H. J. (2014) SELEX-seq: a method for characterizing the complete repertoire of binding site preferences for transcription factor complexes. In *Hox Genes* pp. 255–278 Springer.
9. Isakova, A., Groux, R., Imbeault, M., Rainer, P., Alpern, D., Dainese, R., Ambrosini, G., Trono, D., Bucher, P., and Deplancke, B. (2017) SMiLE-seq identifies binding motifs of single and dimeric transcription factors. *Nat. Methods,* **14**(3), 316–322.
10. Man, T.-K. and Stormo, G. D. (2001) Non-independence of Mnt repressor–operator interaction determined by a new quantitative multiple fluorescence relative affinity (QuMFRA) assay. *Nucleic Acids Res.,* **29**(12), 2471–2478.
11. Bulyk, M. L., Johnson, P. L., and Church, G. M. (2002) Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucleic Acids Res.,* **30**(5), 1255–1261.
12. Benos, P. V., Lapedes, A. S., and Stormo, G. D. (2002) Probabilistic code for DNA recognition by proteins of the EGR family. *J. Mol. Biol.,* **323**(4), 701–727.
13. Rohs, R., West, S. M., Sosinsky, A., Liu, P., Mann, R. S., and Honig, B. (2009) The role of DNA shape in protein–DNA recognition. *Nature,* **461**(7268), 1248–1253.
14. Gordân, R., Shen, N., Dror, I., Zhou, T., Horton, J., Rohs, R., and Bulyk, M. L. (2013) Genomic regions flanking E-box binding sites influence DNA binding specificity of bHLH transcription factors through DNA shape. *Cell Rep.,* **3**(4), 1093–1104.
15. Fordyce, P. M., Pincus, D., Kimmig, P., Nelson, C. S., El-Samad, H., Walter, P., and DeRisi, J. L. (2012) Basic leucine zipper transcription factor Hac1 binds DNA in two distinct modes as revealed by microfluidic analyses. *Proc. Natl. Acad. Sci. U.S.A.,* **109**(45), E3084–E3093.
16. Zuo, Z. and Stormo, G. D. (2014) High-resolution specificity from DNA sequencing highlights alternative modes of Lac repressor binding. *Genetics,* **198**(3), 1329–1343.
17. Halazonetis, T. D., Georgopoulos, K., Greenberg, M. E., and Leder, P. (1988) c-Jun dimerizes with itself and with c-Fos, forming complexes of different DNA binding affinities. *Cell,* **55**(5), 917–924.
18. Slattery, M., Riley, T., Liu, P., Abe, N., Gomez-Alcala, P., Dror, I., Zhou, T., Rohs, R., Honig, B., Bussemaker, H. J., et al. (2011) Cofactor binding evokes latent differences in DNA binding specificity between Hox proteins. *Cell,* **147**(6), 1270–1282.
19. Crocker, J., Noon, E. P.-B., and Stern, D. L. (2016) The soft touch: low-affinity transcription factor binding sites in development and evolution. In *Curr. Top. Dev. Biol.* Vol. 117, pp. 455–469 Elsevier.
20. Kribelbauer, J. F., Rastogi, C., Bussemaker, H. J., and Mann, R. S. (2019) Low-Affinity Binding Sites and the Transcription Factor Specificity Paradox in Eukaryotes. *Annu. Rev. Cell Dev. Biol.,* **35**, 357–379.
21. Jiang, J. and Levine, M. (1993) Binding affinities and cooperative interactions with bHLH activators delimit threshold

responses to the dorsal gradient morphogen. *Cell,* **72**(5), 741–752.
22. Rastogi, C., Rube, H. T., Kribelbauer, J. F., Crocker, J., Loker, R. E., Martini, G. D., Laptenko, O., Freed-Pastor, W. A., Prives, C., Stern, D. L., et al. (2018) Accurate and sensitive quantification of protein-DNA binding affinity. *Proc. Natl. Acad. Sci. U.S.A.,* **115**(16), E3692–E3701.
23. Mathelier, A. and Wasserman, W. W. (2013) The next generation of transcription factor binding site prediction. *PLoS Comput. Biol.,* **9**(9), e1003214.
24. Eggeling, R., Grosse, I., and Grau, J. (2017) InMoDe: tools for learning and visualizing intra-motif dependencies of DNA binding sites. *Bioinformatics,* **33**(4), 580–582.
25. Gershenzon, N. I., Stormo, G. D., and Ioshikhes, I. P. (2005) Computational technique for improvement of the position-weight matrices for the DNA/protein binding sites. *Nucleic Acids Res.,* **33**(7), 2290–2301.
26. Siddharthan, R. (2010) Dinucleotide weight matrices for predicting transcription factor binding sites: generalizing the position weight matrix. *PloS One,* **5**(3), e9722.
27. Kulakovskiy, I., Levitsky, V., Oshchepkov, D., Bryzgalov, L., Vorontsov, I., and Makeev, V. (2013) From binding motifs in ChIP-Seq data to improved models of transcription factor binding sites. *J. Bioinform. Comput. Biol.,* **11**(01), 1340004.
28. Siebert, M. and Söding, J. (2016) Bayesian Markov models consistently outperform PWMs at predicting motifs in nucleotide sequences. *Nucleic Acids Res.,* **44**(13), 6055–6069.
29. Eggeling, R. (2018) Disentangling transcription factor binding site complexity. *Nucleic Acids Res.,* **46**(20), e121.
30. Orenstein, Y. and Shamir, R. (2014) A comparative analysis of transcription factor binding models learned from PBM, HT-SELEX and ChIP data. *Nucleic Acids Res.,* **42**(8), e63.
31. Nitta, K. R., Jolma, A., Yin, Y., Morgunova, E., Kivioja, T., Akhtar, J., Hens, K., Toivonen, J., Deplancke, B., Furlong, E. E., et al. (2015) Conservation of transcription factor binding specificities across 600 million years of bilateria evolution. *elife,* **4**, e04837.
32. Bailey, T. L., Boden, M., Buske, F. A., Frith, M., Grant, C. E., Clementi, L., Ren, J., Li, W. W., and Noble, W. S. (2009) MEME SUITE: tools for motif discovery and searching. *Nucleic Acids Res.,* **37**(suppl_2), W202–W208.
33. Sohrabi-Jahromi, S. and Söding, J. (2021) Thermodynamic modeling reveals widespread multivalent binding by RNA-binding proteins. *bioRxiv, doi: https://doi.org/10.1101/2021.01.30.428941,*.
34. Kiesel, A., Roth, C., Ge, W., Wess, M., Meier, M., and Söding, J. (2018) The BaMM web server for de-novo motif discovery and regulatory sequence analysis. *Nucleic Acids Res.,* **46**(W1), W215–W220.
35. Sharov, A. A. and Ko, M. S. (2009) Exhaustive search for over-represented DNA sequence motifs with CisFinder. *DNA Res.,* **16**(5), 261–273.
36. Kulakovskiy, I. V., Boeva, V., Favorov, A. V., and Makeev, V. J. (2010) Deep and wide digging for binding motifs in ChIP-Seq data. *Bioinformatics,* **26**(20), 2622–2623.
37. Kulakovskiy, I. V., Vorontsov, I. E., Yevshin, I. S., Sharipov, R. N., Fedorova, A. D., Rumynskiy, E. I., Medvedeva, Y. A., Magana-Mora, A., Bajic, V. B., Papatsenko, D. A., et al. (2018) HOCOMOCO: towards a complete collection of transcription factor binding models for human and mouse via large-scale ChIP-Seq analysis. *Nucleic Acids Res.,* **46**(D1), D252–D259.
38. ENCODE Project Consortium and others (2012) An integrated encyclopedia of DNA elements in the human genome. *Nature,* **489**(7414), 57–74.
39. Zhu, F., Farnung, L., Kaasinen, E., Sahu, B., Yin, Y., Wei, B., Dodonova, S. O., Nitta, K. R., Morgunova, E., Taipale, M., et al. (2018) The interaction landscape between transcription factors and the nucleosome. *Nature,* **562**(7725), 76–81.
40. Wingender, E., Schoeps, T., Haubrock, M., and Dönitz, J. (2015) TFClass: a classification of human transcription factors and their rodent orthologs. *Nucleic acids research,* **43**(D1), D97–D102.
41. Hunt, R. W. and Wasserman, W. W. (2014) Non-targeted transcription factors motifs are a systemic component of ChIP-seq datasets. *Genome Biol.,* **15**(7), 412.
42. Chen, Y., Negre, N., Li, Q., Mieczkowska, J. O., Slattery, M., Liu, T., Zhang, Y., Kim, T.-K., He, H. H., Zieba, J., et al. (2012) Systematic evaluation of factors influencing ChIP-seq fidelity. *Nat. Methods,* **9**(6), 609–614.
43. Aird, D., Ross, M. G., Chen, W.-S., Danielsson, M., Fennell, T., Russ, C., Jaffe, D. B., Nusbaum, C., and Gnirke, A. (2011) Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries. *Genome Biol.,* **12**(2), R18.
44. Diaz, A., Park, K., Lim, D. A., and Song, J. S. (2012) Normalization, bias correction, and peak calling for ChIP-seq. *Stat. Appl. Genet. Mol. Biol,* **11**(3).
45. Teytelman, L., Özaydın, B., Zill, O., Lefrançois, P., Snyder, M., Rine, J., and Eisen, M. B. (2009) Impact of chromatin structures on DNA processing for genomic analyses. *PloS One,* **4**(8), e6700.
46. Zhao, Y., Granas, D., and Stormo, G. D. (2009) Inferring binding energies from selected binding sites. *PLoS Comput. Biol.,* **5**(12), e1000590.
47. Jolma, A., Kivioja, T., Toivonen, J., Cheng, L., Wei, G., Enge, M., Taipale, M., Vaquerizas, J. M., Yan, J., Sillanpää, M. J., et al. (2010) Multiplexed massively parallel SELEX for characterization of human transcription factor binding specificities. *Genome Res.,* **20**(6), 861–873.
48. Levo, M., Zalckvar, E., Sharon, E., Machado, A. C. D., Kalma, Y., Lotam-Pompan, M., Weinberger, A., Yakhini, Z., Rohs, R., and Segal, E. (2015) Unraveling determinants of transcription factor binding outside the core binding site. *Genome Res.,* **25**(7), 1018–1029.
49. Schöne, S., Jurk, M., Helabad, M. B., Dror, I., Lebars, I., Kieffer, B., Imhof, P., Rohs, R., Vingron, M., Thomas-Chollier, M., et al. (2016) Sequences flanking the core-binding site modulate glucocorticoid receptor structure and activity. *Nat. Commun.,* **7**(1), 12621.
50. Yella, V. R., Bhimsaria, D., Ghoshdastidar, D., Rodríguez-Martínez, J. A., Ansari, A. Z., and Bansal, M. (2018) Flexibility and structure of flanking DNA impact transcription factor affinity for its core motif. *Nucleic Acids Res.,* **46**(22), 11883–11897.
51. Mathelier, A., Xin, B., Chiu, T.-P., Yang, L., Rohs, R., and Wasserman, W. W. (2016) DNA shape features improve transcription factor binding site predictions in vivo. *Cell Syst.,* **3**(3), 278–286.
52. Peng, P.-C. and Sinha, S. (2016) Quantitative modeling of gene expression using DNA shape features of binding sites. *Nucleic Acids Res.,* **44**(13), e120–e120.
53. Samee, M. A. H., Bruneau, B. G., and Pollard, K. S. (2019) A de novo shape motif discovery algorithm reveals preferences of transcription factors for DNA shape beyond sequence motifs. *Cell Syst.,* **8**(1), 27–42.
54. Alipanahi, B., Delong, A., Weirauch, M. T., and Frey, B. J. (2015) Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nat. Biotechnol.,* **33**(8), 831–838.
55. Kelley, D. R., Snoek, J., and Rinn, J. L. (2016) Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Res.,* **26**(7), 990–999.

Supplementary materials: Bayesian Markov models improve the prediction of binding motifs beyond first order

Wanwan Ge, Markus Meier, Christian Roth, Johannes Söding*

Quantitative and Computational Biology, Max Planck Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany

* For correspondence: soeding@mpibpc.mpg.de

DOI: https://doi.org/10.1101/2020.07.12.197053

Part I

Supplemental Figures

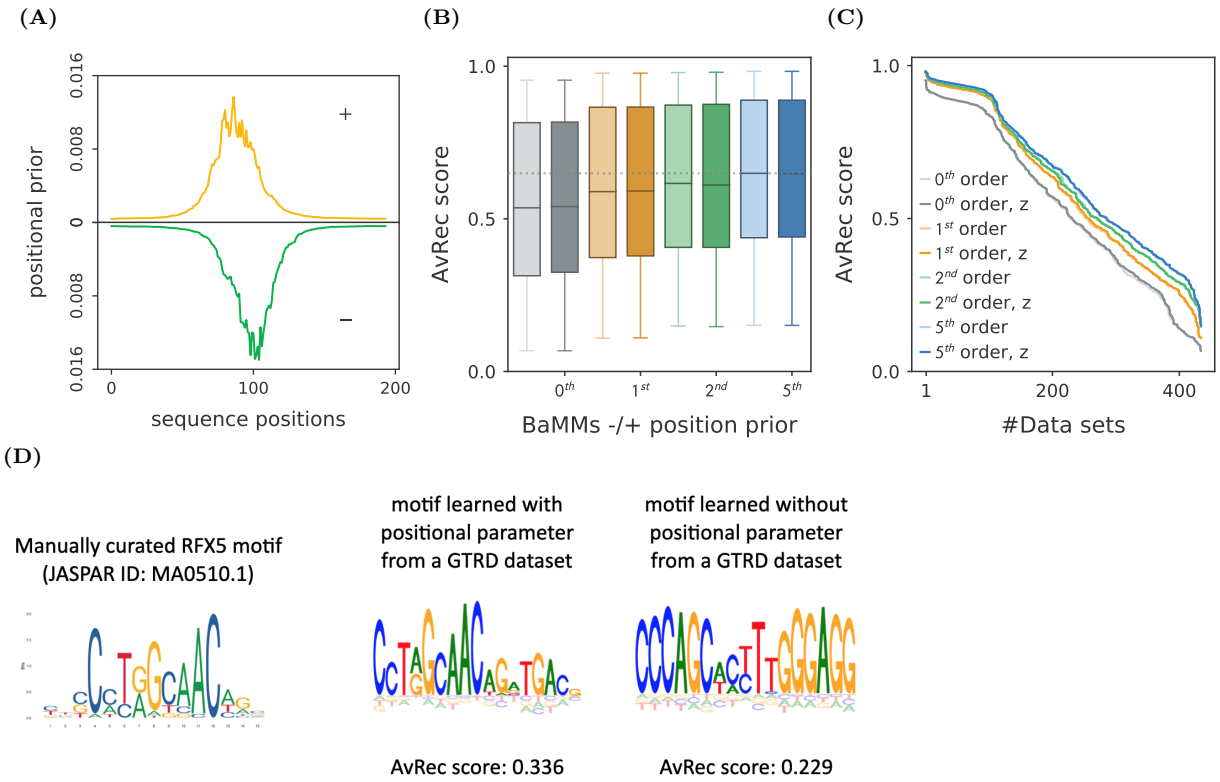


Figure S 1. Optimization of positional prior on *in vivo* data. BaMMs with zeroth- (grey), first- (orange), second- (green) and fifth-orders (blue) are trained and tested on 435 ENCODE datasets using 5-fold cross-validation. Panel (**A**) shows the distribution of optimized positional priors over the positions on both sequence strands that are center around ChIP-seq summits for GABPα motif. Panel (**B**) shows the AvRec distributions as box plot, with boxes indicating 25%/75% quantiles and whiskers 5%/95% quantiles. The colors are for different orders with- (dark colors) and without (light colors) optimization of positional prior in the motif training. Panel (**C**) shows the cumulative distributions of AvRec scores on 435 datasets. There is no major difference before and after the positional prior optimization. Panel (**D**) shows the motif for RFX5 factor learned from a GTRD dataset [1] with (middle) and without (right) the optimized positional parameter, compared to the reference motif reported in the JASPAR database [2] (left).

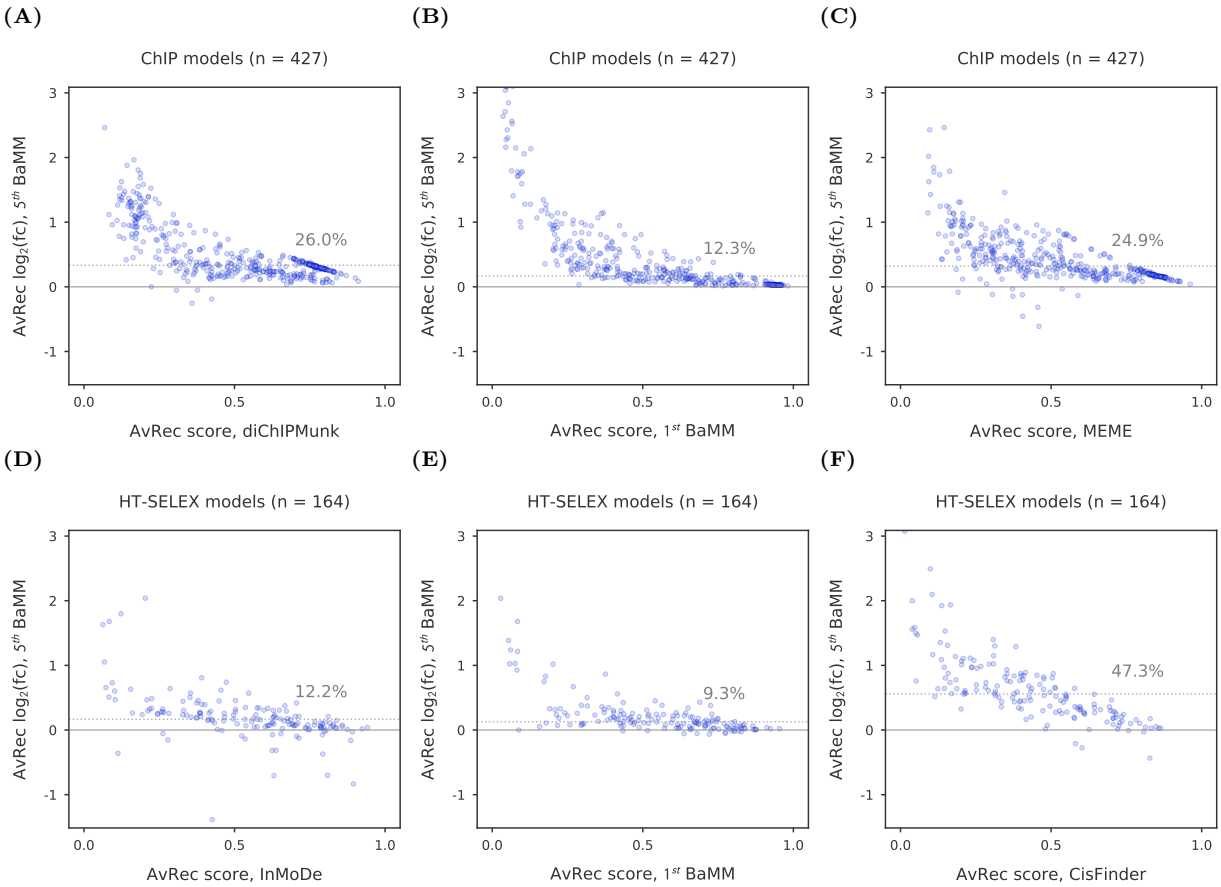


Figure S 2. Performance comparison of motif discovery tools on *in vivo* and *in vitro* data. $\log_2$ of fold change in AvRec between fifth-order BaMMmotif2 and diChIPMunk models versus AvRec of diChIPMunk models (**A**), first-order BaMMmotif2 (**B**) and MEME PWMs (**C**) on 427 ChIP-seq datasets. Each dot represents the test on one dataset from either ChIP-seq or HT-SELEX. The grey dashed lines indicate the median $\log_2$ fold change is 26%, 12.4% and 24.9% respectively. (**D**-**F**) Similar comparisons as (**A**-**C**) but on 164 HT-SELEX datasets.

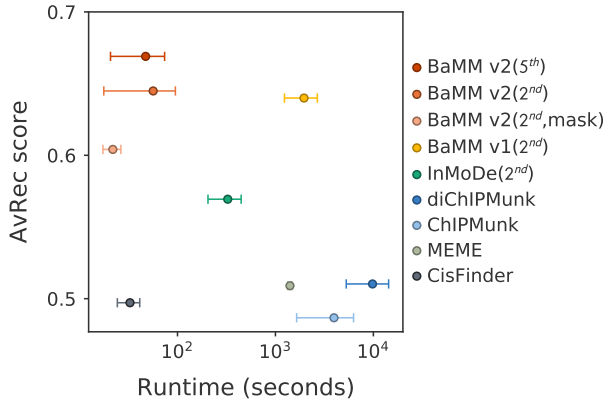


Figure S 3. Benchmark on *in vivo* data. Average runtime per dataset on a server with 4 cores versus the median AvRec score of several *de novo* motif discovery tools, including the previous version of BaMMmotif, validated on 419 datasets with 5-fold cross-validation, with MEME, CisFinder, BaMMmotif and BaMMmotif2 running on 4 CPU cores. Whiskers indicate the standard deviation of AvRec score. 2^{nd}-order models trained using BaMMmotif and BaMMmotif2 have similar average AvRec scores, yet BaMMmotif2 is $>\tilde{1}0$ times faster than BaMMmotif.

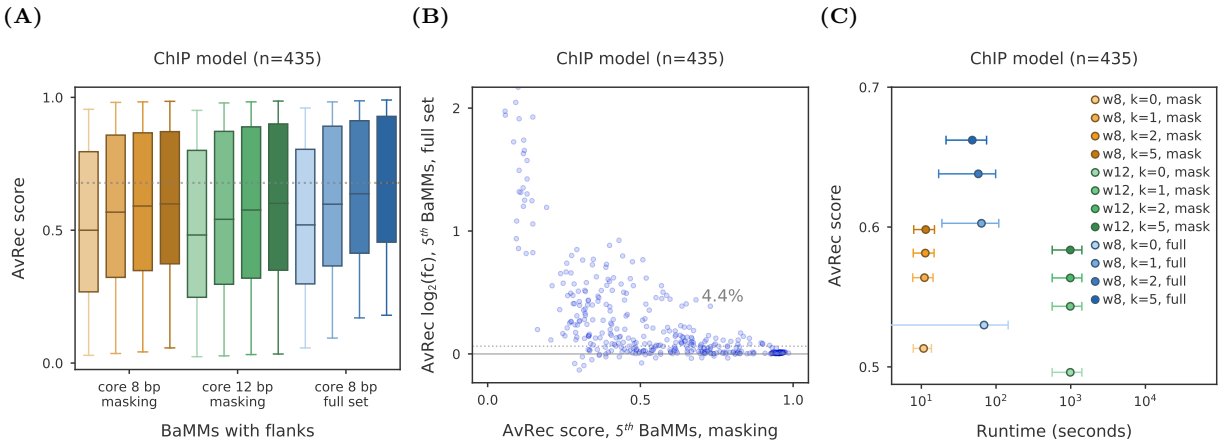


Figure S 4. EM optimization using the full set compared to masking 95% sequences on ChIP-seq datasets. (**A**) Using the full set of sequences for the EM optimization (blue) improves the performance of higher-order models, while extending the core regions for searching the enriched patterns (green) does not contribute to motif discovery, in comparison to that with 8 bp for seeding and masking 95% sequences for the EM optimization (yellow). All box-plot whiskers show 95th/5th percentile. Each cluster contains models with different orders (zeroth-, first-, second- and fifth-order). (**B**) Fifth-order BaMMs optimized on the full sequences set have a 4.4% AvRec fold increase compared to those trained only 5% sequences. (**C**) Using a masking step improves the speed by 10-fold, in comparison to using the full set for learning motif model.

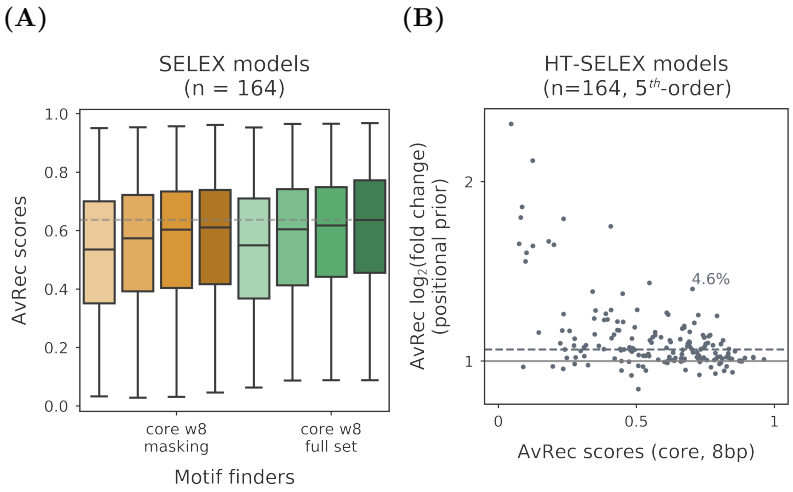


Figure S 5. EM optimization using the full set compared to masking 95% sequences from HT-SELEX datasets. (**A**) Using the full set for motif refinement (green) improves the performance of higher-order models over that using only 5% sequences (yellow). All box-plot whiskers show 95th/5th percentile. Each cluster contains models with different orders (zeroth-, first-, second- and fifth-order). (**B**) Fifth-order BaMMs with full set of sequences for optimization has a 4.6% AvRec fold increase compared to it with only 5% sequences for optimization.

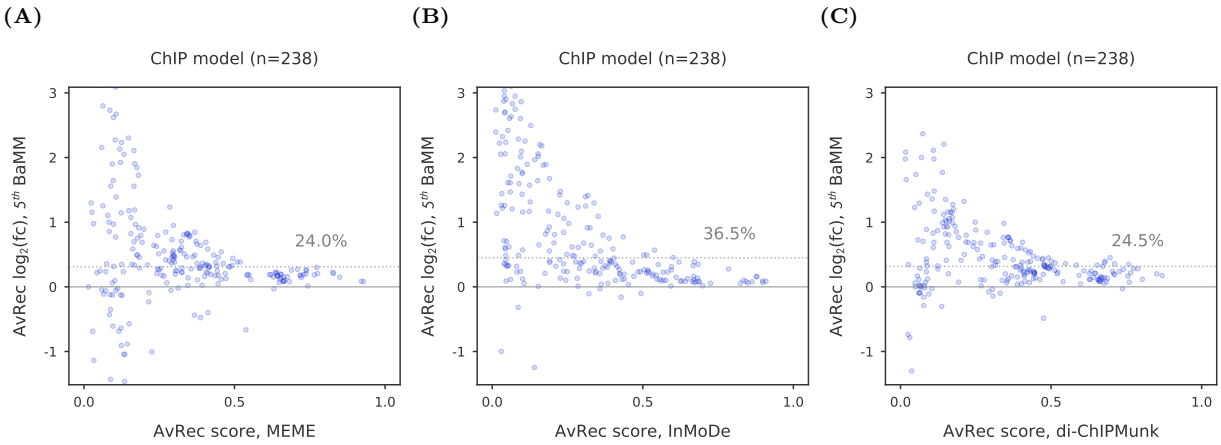


Figure S 6. Cross-cell-line validation. $\log_2$ of fold change in AvRec between fifth-order BaMMmotif2 and PWMs from MEME (**A**), second-order models from InMoDe (**B**), and first-order models from diChIPMunk (**C**), when comparing to AvRec scores of the latter models in 238 paired ENCODE datasets. Each dot represents one test. The range of AvRec scores is chosen from 0.5 to 8 and the outliers are not shown in these plots.

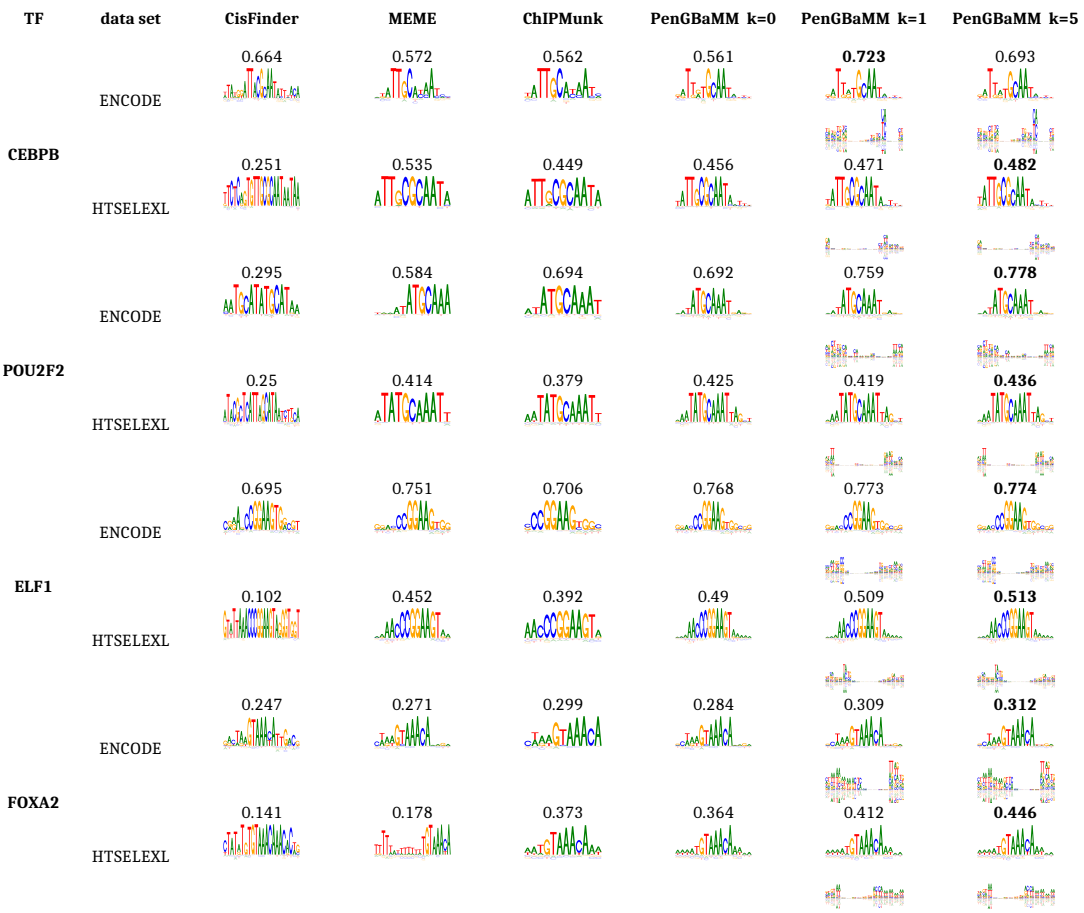


Figure S 7. Sequence logos and AvRec scores of motifs models from cross-platform validation. Motif models are trained by different models for four transcription factors: CEBPB, POU2F2, ELF1, and FOXA2. For each transcription factor, the first row shows models learned on ENCODE data by applying different tools. The number above each logo represents the AvRec score when testing the model on the corresponding HT-SELEX data. The second row shows the models learned on HT-SELEX data and AvRec scores when testing models on ENCODE data. For BaMM models, both zeroth- and first-order logos are plotted.

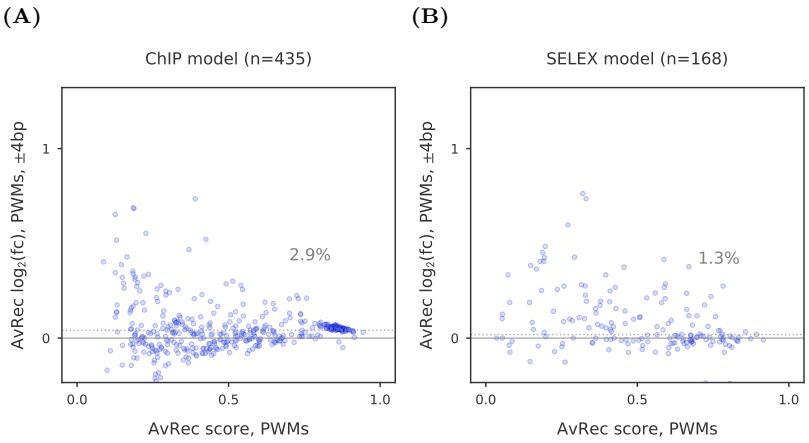


Figure S 8. Impact of extending core motif regions on PWMs. (**A**) Log_2 of fold change between PWM models with ± 4 bp flanking positions and no added flanking positions, using 435 datasets. Median AvRec change is 2.9 %. (**B**) Same as (**A**) but on 168 HT-SELEX datasets.

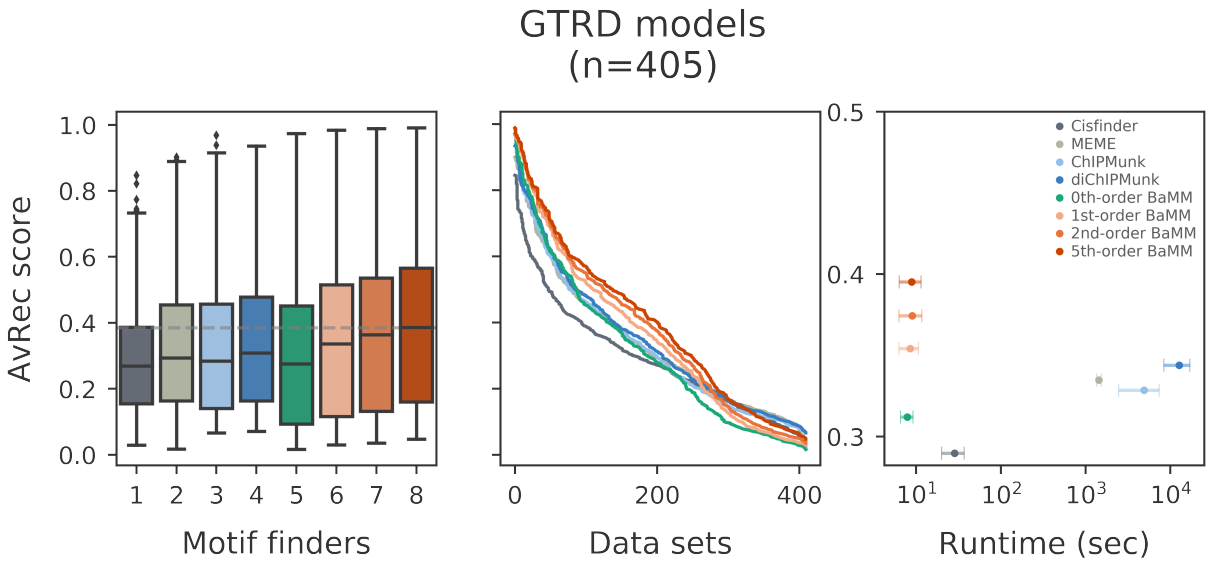


Figure S 9. Quantitative performance on *in vivo* GTRD datasets. The selected tools are applied to 405 GTRD datasets [1] and their AvRec were calculated by 5-fold cross-validation, similar to Figure 2. (**A**) AvRec distributions as box plot. All box-plot whiskers show 95th/5th percentile. (**B**) The cumulative of AvRec scores on 405 datasets. (**C**) Average runtime per dataset on a server with 4 cores versus the median AvRec score. Whiskers: standard deviation.

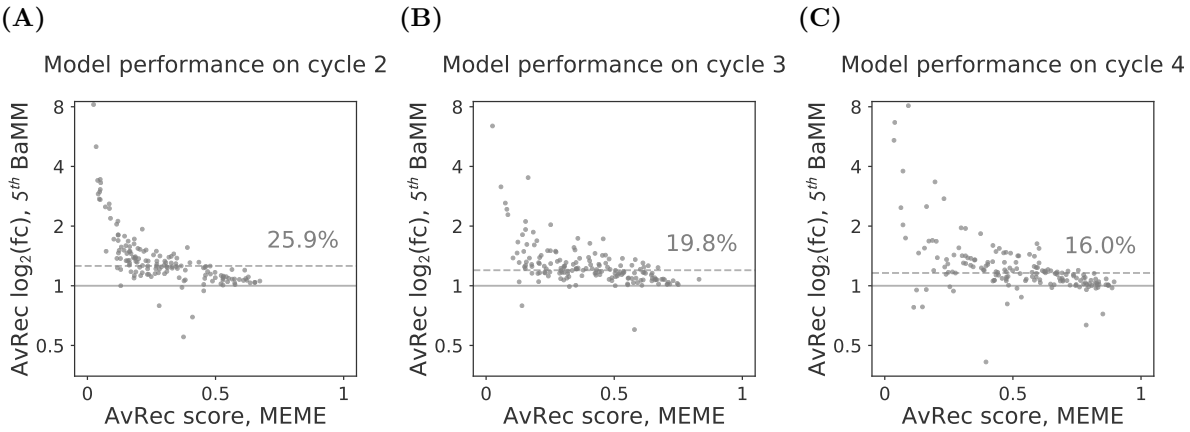
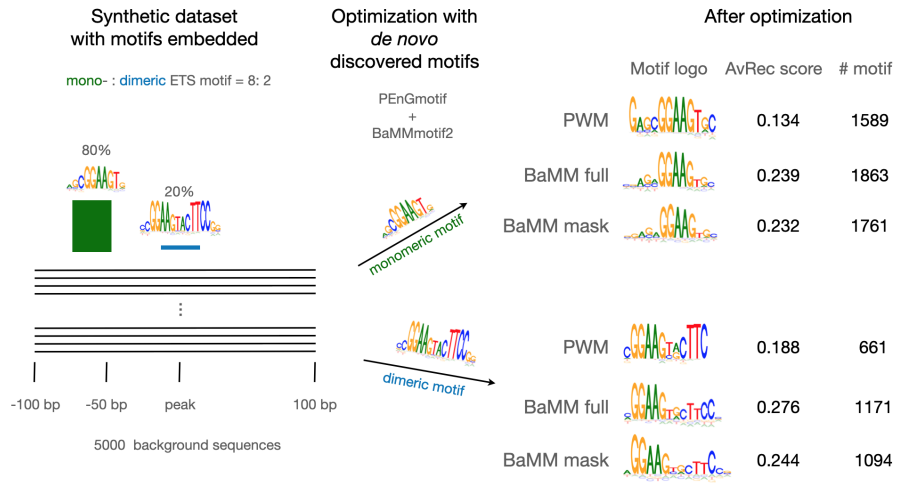
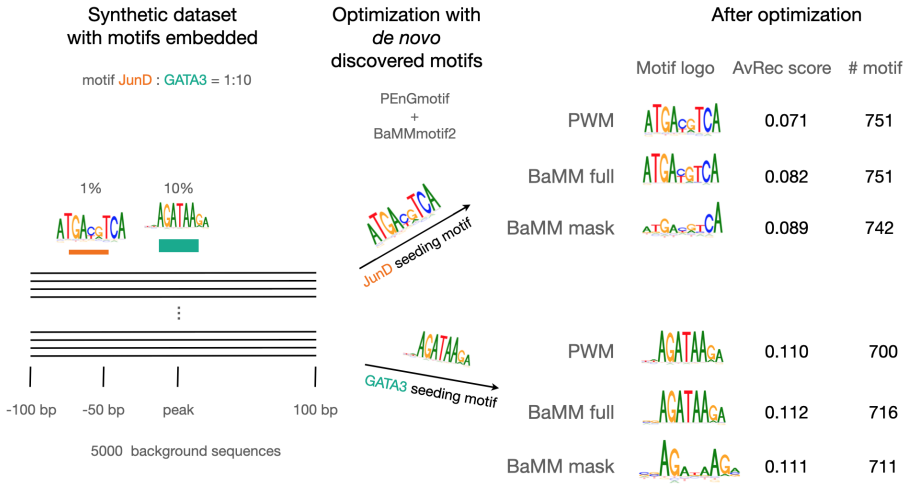


Figure S 10. Performance comparison of BaMMmotif versus MEME on weak binding prediction. $\log_2$ fold change between fifth-order BaMMmotif2 models and MEME models versus AvRec of MEME, with AvRec analyzed by 5-fold cross-validation on sequences from the 2nd- (**A**), 3rd- (**B**) or 4th- (**C**) selection cycle of 164 HT-SELEX datasets. The median fold change increases are 25.9%, 19.8% and 16%, respectively (grey dashed lines). Each dot represents one data set.

(A)



(B)



(C)

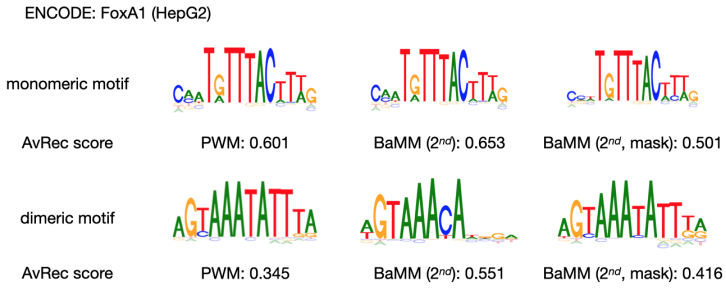


Figure S 11. Show cases for higher-order BaMMs. (A) 0th-order models and higher-order models were trained with and without sequence masking on a set of 5000 synthetic background sequences from a second order null model implanted with monomeric and dimeric ETS motifs in 80% and 20% of the sequences, respectively. With all three settings, the implanted motifs were learned separately as two distinct motifs. (B) 5000 synthetic sequences embedded with GATA3 and JunD motifs with very low occurrences, 10% and 1% respectively. With all three settings, the implanted motifs were learned separately as two distinct motifs. (C) Motif discovery for FoxA1 from a ENCODE dataset (accession: ENCFF648VIL). The *de novo* motif discovery process found two binding modes. But given that the consensus of the dimer motif is palindromic, its fifth-order motif model mixes with the monomer motif when no masking was applied. When masking 99% of the positions, 5th-order BaMM was able to separate these two closely related motifs.

Part II

Supplemental Methods

The supplemental material provides further details of the theoretical basis, the implementation of the BaMMmotif2 package, and the processing procedure of the datasets that are used for this benchmark. It also documents the parameters used for testing the motif discovery tools in the benchmark. It ensures the reproducibility of the results in this paper.

1 *De novo* motif discovery and refinement

1.1 The fast seeding phase: PEnGmotif

We describe PEnGmotif (Pattern-based discovery of enriched genomic or transcriptomic sequence motifs), an efficient method for discovering sequence patterns enriched in a set of nucleotide sequences over random expectation sampled from a second-order background model. The enriched patterns found by PEnGmotif are optimized to PWMs and serve as seeds to initialise the refinement stage by BaMMmotif2.

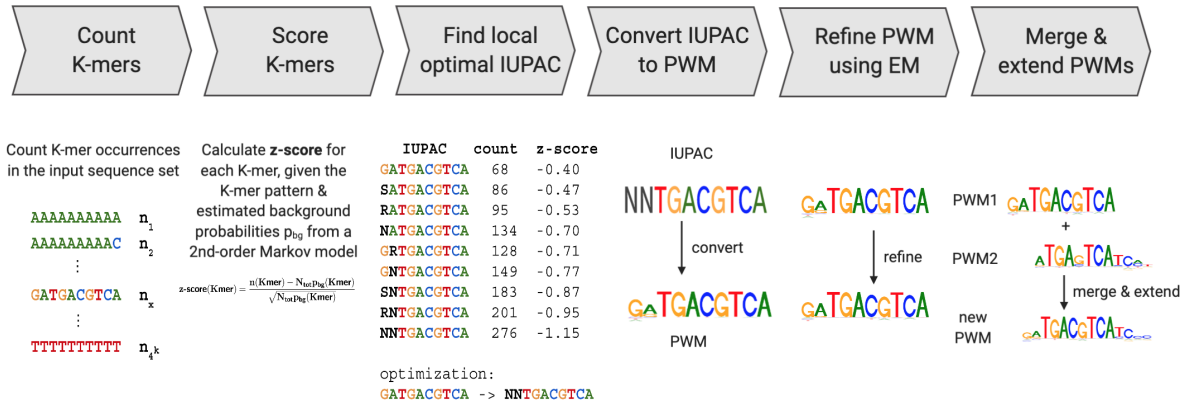


Figure S 12. Workflow of the fast seeding stage. Sequences from high-throughput assays, such as ChIP-seq, SELEX and PBM, are provided as input data. (i) Occurrences of all K-mers of a fixed specified length (default 10) are counted. (ii) An enrichment z-score is calculated for each K-mer based on a Poisson model. (iii) High-scored K-mers are optimized from the nucleotide alphabet (ACGT) to a degenerate IUPAC alphabet with 11 letters (ACGTRYSWSKN). (iv) The locally optimal IUPAC patterns are converted to PWMs. (v) PWMs are refined using the Expectation Maximisation algorithm. (vi) PWMs with similar overlapping regions are merged and extended.

Let K be the length of patterns that will be analysed (e.g. $K = 8$ used in the study). First, the number of occurrences of each of the 4^K non-degenerate seed patterns of length K are counted in a 4^K-dimensional array with $\mathbf{x} \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^K$. $p_{\mathrm{bg}}(\mathbf{x})$ denotes the probability of observing K-mer $\mathbf{x}$ in absence of specific binding. $p_{\mathrm{bg}}(\mathbf{x})$ can be directly counted from large background sequence sets or modelled as a homogeneous Markov model on a background data set or the dataset itself. For example, $p_{\mathrm{bg}}(\mathbf{x})$ is learned from the genomic input, a mock

immunoprecipitation or the input sequence library prior to the selection in HT-SELEX. We model the background probability using a homogeneous Markov model of order K' ($K' = 2$ by default):

$$p_{\mathrm{bg}}(x_{i_0\,:\,i_1}) = \prod_{i=i_0}^{i_1} p_{\mathrm{bg}}(x_i | x_{i-K'\,:\,i-1}). \tag{1}$$

We assume the number of occurrences in absence of specific binding to follow a Poisson distribution: $\mu = L_{\mathrm{tot}} p_{\mathrm{bg}}(\mathbf{y})$, where $L_{\mathrm{tot}} = \sum_{n=1}^{N}(L_n - K + 1)$ is the total number of all counted patterns in the input sequences (N is the total sequence number and L_n is the length of n'th sequence).

z-score We compute Z-scores for all non-degenerate K-mer patterns. The Z-score is the deviation from expectation divided by the standard deviation. As for the Poisson distribution the variance equals the mean, the Z-score is:

$$Z(\mathbf{y}) = \frac{n(\mathbf{y}) - L_{\mathrm{tot}} p_{\mathrm{bg}}(\mathbf{y})}{\sqrt{L_{\mathrm{tot}} p_{\mathrm{bg}}(\mathbf{y})}}\,. \tag{2}$$

The z-score can be used to pre-filter what K-mers should enter the optimization routine.

p-value As we are also interested in highly enriched sequences (x) and (y) are fulfilled and we can use the Stirling approximation to calculate the p-value:

$$\begin{aligned}
\text{p-value}(\mathbf{y}) &= \sum_{k=n}^{\infty} \frac{\mu^k}{k!} e^{-\mu} \\
&= \frac{\mu^n}{n!} e^{-\mu} \sum_{k=0}^{\infty} \frac{\mu^k}{(n+1)\cdots(n+k)} \\
&\lessapprox \frac{\mu^n}{n!} e^{-\mu} \sum_{k=0}^{\infty} \frac{\mu^k}{(n+1)^k} \\
&\approx \frac{\mu^{n(\mathbf{y})}}{n(\mathbf{y})!} e^{-\mu} \frac{1}{1 - \mu/(n(\mathbf{y})+1)} \\
\log \text{p-value}(\mathbf{y}) &\approx n(\mathbf{y}) \log \frac{\mu}{n(\mathbf{y})} + n(\mathbf{y}) - \mu - \frac{1}{2}\log(2\pi n(\mathbf{y})) - \log\left(1 - \frac{\mu}{n(\mathbf{y})+1}\right).
\end{aligned} \tag{3}$$

Mutual information We optimize the mutual information (MI) between two random variables,

$$\mathrm{MI}(q) = -qH(p_{\mathrm{obs}}) - (1-q)H(p_{\mathrm{exp}}) + H(p), \tag{4}$$

with $H(x) := -x\log x - (1-x)\log(1-x)$.

We then find locally optimal non-degenerate patterns with a recursive function which takes a K-mer $\mathbf{y}$ and checks for all its neighbouring K-mers, i.e. those that are at most one substitution away. If it finds a neighbouring $\mathbf{y}_{\mathrm{neigh}}$ with a better mutual information, the

function is called recursively with $\mathbf{y}_{\text{neigh}}$ as an argument. If no neighbour of $\mathbf{y}$ has better mutual information than $\mathbf{y}$, $\mathbf{y}$ is appended to the list of locally optimal K-mers. Similarly, we optimized the high-scored K-mers from the nucleotide alphabet (ACGT) to a degenerate IUPAC alphabet with 11 letters (ACGTRYSWSKN).

The IUPAC patterns can be transformed to PWMs based on the combined occurrences of all non-degenerated K-mers that match the degenerate IUPAC pattern in the input sequences. Alternatively, there is a faster approach based on the insight that if we allow any of the four nucleotides $a \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$ at position j, the vast majority of motif matches will still be true positives due to the descriptive power of the other $K-1$ IUPAC letters. Therefore, we count the four nucleotides at motif position j for matches to the pattern $y_{0:j-1}\mathrm{N}y_{j+1:K-1}$ in which we replaced the jth IUPAC letter by an N:

$$p_{ja} = \frac{n(y_{0:j-1}\, a\, y_{j+1:K-1})}{n(y_{0:j-1}\, \mathrm{N}\, y_{j+1:K-1})}\,, \tag{5}$$

where we have called $n(\mathbf{y})$ the number of occurrences of K-mer $\mathbf{y}$ in the input set. Note that these PWM probabilities can be computed solely from the K-mer counts in a time $O(W \times D)$ that is independent of the size of the input dataset L_{tot}, and only depends on the degeneracy $D = \left|\{\mathbf{x} \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^W : \mathbf{x} \text{ matches } \mathbf{y}\}\right|$ of the motif $\mathbf{y}$, i.e., the number of different K-mers it matches.

We then refine the obtained PWMs by learning a multiple-occurrence-per-sequence model (MOPS) directly on the K-mer counts. The likelihood of a K-mer $\mathbf{x} \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^K$ given a position weight matrix model with probabilities $\mathbf{p} = (p_j(\mathbf{A})$ is

$$\frac{p(\mathbf{x}|\mathbf{p}_{\text{motif}})}{p(\mathbf{x}|\mathbf{p}_{\text{bg}})} = \prod_{j=0}^{K-1} \frac{p_j(x_j)}{p_{\text{bg}}(x_j)}\,. \tag{6}$$

Expectation step: Compute the responsibilities $r(\mathbf{x})$, i.e., the probability that the factor will bind to K-mer $\mathbf{x}$.

$$r(\mathbf{x}) = \frac{p(\mathbf{x}|\mathbf{p}_{\text{motif}})/p(\mathbf{x}|\mathbf{p}_{\text{bg}})}{\sum_{\mathbf{x}' \in \{\mathrm{A,C,G,T}\}^K} n(\mathbf{x}') p(\mathbf{x}'|\mathbf{p}_{\text{motif}})/p(\mathbf{x}'|\mathbf{p}_{\text{bg}})} \tag{7}$$

Maximization step: Update the probabilities of the position weight matrix model.

$$p_j(\mathbf{A}) = \sum_{\mathbf{x} \in \{\mathrm{A,C,G,T}\}^K} I(x_j = a)\, n(\mathbf{x})\, r(\mathbf{x}) \tag{8}$$

By inserting the E-step equation into the M-step, we obtain

$$p_j^{(t)}(\mathbf{A}) \propto \sum_{\mathbf{x} \in \{\mathrm{A,C,G,T}\}^K} I(x_j = a)\, n(\mathbf{x})\, \frac{p(\mathbf{x}|\mathbf{p}_{\text{motif}}^{(t-1)})}{p(\mathbf{x}|\mathbf{p}_{\text{bg}})} \tag{9}$$

and subsequent normalisation for each j over $a \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$ yields the updated motif matrix probabilities.

To model saturation effects at the motifs with high affinities, we can use a saturation function that will limit the weight of the odds ratios to a maximum value A, e.g. $A = 1000$:

$$p_j^{(t)}(\mathbf{A}) \propto \sum_{\mathbf{x}\in\{\mathrm{A,C,G,T}\}^K} I(x_j{=}a)\, n(\mathbf{x}) \left(A^{-1} + \frac{p(\mathbf{x}|\mathbf{p}_{\mathrm{bg}})}{p(\mathbf{x}|\mathbf{p}_{\mathrm{motif}}^{(t-1)})} \right)^{-1} \tag{10}$$

In a thermodynamic interpretation, A is the odds ratio of sites that have an occupancy of 50% at the assumed concentration of the transcription factor in the nucleus.

Merging and extending PWMs. We can reduce the redundancy of the PEnG!motif output and more importantly, generate more specific and sensitive motifs by merging sub-motifs that describe parts of the same underlying biological motif. For that, we first compute a list of pairwise similarity scores between all PWMs $\{p^{(1)}, \ldots, p^{(M)}\}$ with P-values above a user-specified cutoff obtained in the last step. Here, $p_{ja}^{(m)}$ is the probability of observing a nucleotide a at the j'th position of that PWM. The similarity score $S(p^{(m)}, p^{(m')})$ is defined by the maximum similarity score $s(\cdot,\cdot)$ evaluated in the overlapping regions when the two patterns of length l and l' are shifted by $d = -2, -1, \ldots, l' - l + 2$ to each other:

$$S\Big(p^{(m)}, p^{(m')}\Big) = \max_{-2\le d\le l'-l+2} \left\{ s\Big(p^{(m)}_{j_1:j_2}, p^{(m')}_{j_1':j_2'}\Big) \right\} . \tag{11}$$

The indices defining the overlap region in the two PWMs are $j_1 = \max\{0, d\}, j_2 = \min\{l - 1, l' - 1 + d\}$ and $j_1' = \max\{0, -d\}, j_2' = \min\{l' - 1, l - 1 - d\}$. The similarity score between the PWMs in the overlap region is computed using

$$s(p, p') = \frac{1}{2}\left(d(p, p^{(\mathrm{bg})}) + d(p', p^{(\mathrm{bg})})\right) - d(p, p'), \tag{12}$$

The distance $d(p, p')$ between two PWMs p and p' of length l is the sum over the PWM columns of the relative entropies of each with their average distribution $\bar{p} := (p + p')/2$,

$$d(p, p') = \sum_{j=0}^{l-1} \left(H(p||\bar{p}) + H(p'||\bar{p})\right) = \sum_{j=0}^{l-1} \sum_{a\in\mathrm{A,C,G,T}} \left(p_{ja}\log_2 p_{ja} + p'_{ja}\log_2 p'_{ja} - 2\bar{p}_{ja}\log_2 \bar{p}_{ja}\right) . \tag{13}$$

The pair with the highest score will be merged using the positional offset d that yielded the maximum similarity score. The pair of PWMs $(p^{(m)}, p^{(m')})$ has a score above a user-specified threshold ($0.4 \times W$ bits by default) are merged together using the positional offset d that yielded the maximum similarity score. In the overlapping regions, the nucleotide probabilities of merged PWM will be the weighted sum of the nucleotide probabilities of the two merged PWMs, where the weights are the numbers of matches of the associated IUPAC patterns. The new weights of the columns of merged PWM will be the sum of these numbers of matches. In the non-overlapping regions, the probabilities and weights are simply copied over from the one PWM.

1.2 Higher-order inhomogeneous Markov models

BaMMmotif [3] refines the pre-aligned short patterns or position-weight-matrices (PWMs) to higher-order Bayesian Markov models for the enriched motifs.

According to Boltzmann's law, the probability of a genomic site with sequence $\mathbf{x}$ to be bound by the transcription factor divided by the probability of $\mathbf{x}$ not to be bound is

$$\exp\left(-\frac{\Delta G(\mathbf{x})-\mu}{k_B T}\right) = \frac{p(\text{bound}|\mathbf{x})}{p(\text{not bound}|\mathbf{x})} = \frac{p(\text{bound}|\mathbf{x})}{1-p(\text{bound}|\mathbf{x})}, \tag{14}$$

with the chemical potential μ that depends on the factor concentration but not on $\mathbf{x}$. Solving for $p(\text{bound}|\mathbf{x})$ yields the well-known behaviour for saturated binding,

$$p(\text{bound}|\mathbf{x}) = \left(1+\exp\left(\frac{\Delta G(\mathbf{x})-\mu}{k_B T}\right)\right)^{-1}. \tag{15}$$

We parameterise the dependence of $\Delta G(\mathbf{x})$ on the binding site sequence $\mathbf{x}$ by a probability distribution $p_{\text{motif}}(\mathbf{x})$ which is defined by

$$p_{\text{motif}}(\mathbf{x})/p_{\text{bg}}(\mathbf{x}) \propto \exp(\Delta G(\mathbf{x})/k_B T). \tag{16}$$

The proportionality constant is determined by the normalization. Solving for $p_{\text{motif}}(\mathbf{x})$ and normalising yields

$$p_{\text{motif}}(\mathbf{x}) := \frac{p_{\text{bg}}(\mathbf{x})\exp(-\Delta G(\mathbf{x})/k_B T)}{\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y})\exp(-\Delta G(\mathbf{y})/k_B T)}, \tag{17}$$

where the sum in the normalisation constant runs over all possible binding site sequences $\mathbf{y} \in \{\text{A},\text{C},\text{G},\text{T}\}^W$. The motif score

$$S(\mathbf{x}) := \log\frac{p_{\text{motif}}(\mathbf{x})}{p_{\text{bg}}(\mathbf{x})} = -\frac{\Delta G(\mathbf{x})}{k_B T} + \text{const.} \tag{18}$$

gives us, up to the constant $\log\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y})\exp(-\Delta G(\mathbf{y})/k_B T)$, the binding strength of a site $\mathbf{x}$ as quantified by the negative Gibbs energy of binding in units of $k_B T\log 2$. Once we know $p_{\text{motif}}(\cdot)$ we can compute the motif score $S(\mathbf{x})$ which gives us the relative binding strength. If we define $\mu' = \mu/k_B T - \log\sum_{\mathbf{y}} p_{\text{bg}}(\mathbf{y})\exp(-\Delta G(\mathbf{y})/k_B T)$ we see that $S(\mathbf{x})+\mu' = (-\Delta G(\mathbf{x})+\mu)/k_B T$. Hence up to the constant chemical potential μ', $p_{\text{motif}}(\cdot)$ determines the occupancy of any sequence (in the absence of competitive binding through steric hindrance) for any potential binding site sequence $\mathbf{x} = (x_1 \ldots x_W)$,

$$p(\text{bound}|\mathbf{x}) = \frac{e^{S(\mathbf{x})+\mu'}}{1+e^{S(\mathbf{x})+\mu'}}. \tag{19}$$

In the following we drop the prime on μ' for simplicity.

We derive a model for the Gibbs binding energy $\Delta G(\mathbf{x})$ for any potential binding site sequence $\mathbf{x} = x_{1:K} \in \{\text{A},\text{C},\text{G},\text{T}\}^K$ by computing a motif score $S(\mathbf{x})$:

$$S(\mathbf{x}) = -\frac{\Delta G(\mathbf{x})}{k_B T} + \text{const.} := \log\frac{p_{\text{motif}}(\mathbf{x})}{p_{\text{bg}}(\mathbf{x})} = \sum_{j=0}^{K-1}\log\frac{p_j^K(x_j|x_{j-K:j-1})}{p_{\text{bg}}^{K'}(x_j|x_{j-K':j-1})}. \tag{20}$$

where we model the background probability using a homogeneous Markov model of order K':

$$p_{\mathrm{bg}}(x_{i_0\,:\,i_1}) = \prod_{i=i_0}^{i_1} p_{\mathrm{bg}}(x_i|x_{i-K'\,:\,i-1}). \tag{21}$$

We model the motif using an inhomogeneous Markov model of order K:

$$p_{\mathrm{motif}}(x_{0\,:\,K-1}) = \prod_{j=0}^{K-1} p_j(x_j|x_{j-K\,:\,j-1}). \tag{22}$$

We learn the parameters of the inhomogeneous Markov model by maximising the posterior probability. A natural prior is a product of Dirichlet distributions with pseudo-count parameters proportional to the lower-order model probabilities, with proportionality constants α_{kj} for $k = 1, \ldots, K$, whose size determines the strength of the prior. Maximizing the posterior probability yields

$$p_j^k(x_{k+1}|x_{1:k}) = \frac{n_j(x_{1:k+1}|\mathbf{r}) + \alpha_{kj} p_j^{k-1}(x_{k+1}|x_{2:k})}{n_{j-1}(x_{1:k}|\mathbf{r}) + \alpha_{kj}}. \tag{23}$$

1.3 Masking in the motif refinement step

We train BaMMs using the expectation-maximization (EM) algorithm. In the E-step, we (re-) estimate the responsibilities r for a motif to be present at position i of sequence n,

$$r_{ni} := p(z_n = i|\mathbf{x}_n, p^K_{\mathrm{motif}}(\mathbf{x})) = \frac{p(\mathbf{x}_n|z_n = i, p^K_{\mathrm{motif}}(\mathbf{x}))\, p(z_n = i)}{\sum_{i'=0}^{L_n-W+1} p(\mathbf{x}_n|z_n = i', p^K_{\mathrm{motif}}(\mathbf{x}))\, p(z_n = i')} \tag{24}$$

In the M-step, we use the new r_{ni} to update the model parameters $p_{\mathrm{motif}}(\mathbf{x})^K$ for all orders $k = 0, ..., K$. This update equation looks exactly the same as the previous equation for known motifs locations, except that now the counts $n_j(x_{1:k+1})$ are interpreted as fractional counts computed according to

$$n_j(\mathbf{x}, x_{k+1}|\mathbf{r}) := \sum_{n=1}^{N} \sum_{i=1}^{L_n-W+1} r_{ni} I\left(x_{n,\,i+j-k\,:\,i+j} = (\mathbf{x}, x_{k+1})\right). \tag{25}$$

The indicator function I returns 1 if the logical expression is true and 0 otherwise. The parameter updates are done for all orders from 0 to K.

Here we introduce a masking step between the E- and M-step by masking out the first N% of r_{ni} after re-ranking increasingly (N is 90 by default) in the first iteration of the EM. By doing this, we learn the model only on the strong binding sites and thus eliminate the effect of unrelated motifs. We then iterate the EM algorithm until convergence.

1.4 Optimization of order- and position-specific hyperparameters α

In the previous version of BaMMmotif [3], the hyperparameters α_{kj} were empirically chosen. Here in this project, we try to learn the position-specific α_{kj} from the data.

We choose as prior on the hyperparameters α_{kj} (for $1 \leq k \leq K$) an inverse Gamma distribution with parameters 1 and $(\beta\gamma^k)$,

$$p(\alpha_{kj}|\beta,\gamma) = \frac{\beta\,\gamma^k}{\alpha_{kj}^2} e^{-\beta\,\gamma^k/\alpha_{kj}} \tag{26}$$

where $\beta \approx 5$ and $\gamma = 3$ corresponds roughly to the previous choice $\alpha_{kj} = \beta\,\gamma^k = 20 \times 3^{k-1}$ that worked for all of the datasets in the previous study [3].

According to Bayes' theorem, the conditional probability of $\boldsymbol{\alpha}$ given motif positions $\mathbf{z}$ can be written as:

$$\begin{aligned} p(\boldsymbol{\alpha}_k|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) &\underset{\boldsymbol{\alpha}_k}{\propto} p(\mathbf{X}|\boldsymbol{\alpha},\mathbf{z},p^{k-1}_{\text{motif}})\,p(\boldsymbol{\alpha}|\mathbf{z},p^{k-1}_{\text{motif}}) \\ p(\boldsymbol{\alpha}_k|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) &\underset{\boldsymbol{\alpha}_k}{\propto} p(\mathbf{X}|\boldsymbol{\alpha},\mathbf{z},p^{k-1}_{\text{motif}})\,p(\boldsymbol{\alpha}) \end{aligned} \tag{27}$$

where

$$\begin{aligned} &p(\mathbf{X}|\mathbf{z},\boldsymbol{\alpha},p^{k-1}_{\text{motif}}) \\ &\propto \prod_{j=0}^{W-1} \prod_{\mathbf{y}} \frac{\Gamma(\alpha_{kj})}{\prod_a \Gamma(\alpha_{kj} v_j^*(a|\mathbf{y}'))} \frac{\prod_{a=1}^4 \Gamma(n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} v_j^*(a|\mathbf{y}'))}{\Gamma(n_{j-1}^{\mathbf{z}}(\mathbf{y}) + \alpha_{kj})} \prod_{a=1}^4 \cancel{\frac{1}{v_{\text{bg}}(a|\mathbf{y})^{n_j^{\mathbf{z}}(\mathbf{y},a)}}}. \end{aligned} \tag{28}$$

Inserting (26) and (28) yields for the conditional probability

$$\begin{aligned} p(\boldsymbol{\alpha}_k|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) &= \sum_{j=0}^{W-1} \left(\prod_{\mathbf{y}} \frac{\beta\,\gamma^k}{\alpha_{kj}^2} e^{-\frac{\beta\,\gamma^k}{\alpha_{kj}}} \frac{\Gamma(\alpha_{kj})}{\prod_a \Gamma(\alpha_{kj} v_j^*(a|\mathbf{y}'))} \frac{\prod_{a=1}^4 \Gamma(n_j^{\mathbf{z}}(\mathbf{y},a) + \alpha_{kj} v_j^*(a|\mathbf{y}'))}{\Gamma(n_{j-1}^{\mathbf{z}}(\mathbf{y}) + \alpha_{kj})} \right) \\ &= \prod_{j=0}^{W-1} p(\alpha_{kj}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}})\,, \end{aligned} \tag{29}$$

which factorizes over the α_{kj}. We could therefore use Gibbs sampling to draw each new value of α_{kj} from its probability distribution independent of the others.

But for an efficient optimisation we need to reparameterise α_{kj} as

$$\alpha_{kj} = e^{a_{kj}} \tag{30}$$

and sample a_{kj} instead of α_{kj}, because otherwise it would take too long to explore the entire probability distribution by small steps in α_{kj}. If we went in steps of 0.5, for example, it would take almost 20000 directed steps to move from $\alpha_{kj} = 1$ to 10000. With steps of size 0.5, it

would take only $2log20000 = 18.4$ directed steps to reach 10000. The probability density also needs to be transformed with the variable:

$$p(a_{kj}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) = \left|\frac{d\,\alpha_{kj}}{d\,a_{kj}}\right| p(\alpha_{kj}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) \tag{31}$$

$$= \alpha_{kj}\, p(\alpha_{kj}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) \tag{32}$$

The log conditional probability for a_{kl} is

$$\log p(a_{kl}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) = \text{const.} - \log\alpha_{kj} - \beta\,\gamma^k/\alpha_{kj} + 4^k \log\Gamma(\alpha_{kj}) \tag{33}$$

$$+ \sum_{\mathbf{y}=y_{1:k}} \left(\sum_{a=1}^{4} \Big[\log\Gamma(n^{\mathbf{z}}_j(\mathbf{y},a)+\alpha_{kj}p^{k-1}_{\text{motif},j}(a|\mathbf{y}')) - \log\Gamma(\alpha_{kj}v^{k-1}_j(a|\mathbf{y}')) \Big] - \log\Gamma(n^{\mathbf{z}}_{j-1}(\mathbf{y})+\alpha_{kj}) \right)$$

We can sample from this distribution using the Metropolis-Hastings algorithm. We draw a new $a^{\text{try}}_{kl} \sim \mathcal{N}(a_{kl},1)$ and accept this trial sample with a probability

$$\frac{p(a^{\text{try}}_{kl}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}})}{p(a_{kl}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}})} \text{ if } p(a^{\text{try}}_{kl}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}}) < p(a_{kl}|\mathbf{X},\mathbf{z},p^{k-1}_{\text{motif}})$$

$$1 \text{ if otherwise .} \tag{34}$$

Because it is fast to sample a_{kl} in this way, we draw 10 or times in a row and only take record the last accepted sample of a_{kl}. This 10-fold repetition ensures that we can explore almost the entire range of relevant values of a_{kl} within these 10 steps.

At the start of the sampling, the a_{kj} will move in the direction of the medians of their probability distribution in relatively directed steps until the changes to the a_{kj} become non-directional and begin to fluctuate. We can then fix the a_{kj} to the average of the last 20 or so samples and perform a few (e.g. 5) iterations of the EM algorithm (described in section 1.2) to find the optimum model parameters $v^K_j(a|\mathbf{y})$ given the fixed a_{kj}.

1.5 Learning positional preferences of motifs

Thermodynamic treatment of positional preference

We proceed analogously to section 1.2 but introduce a positional preference as an additive term ΔG_i in the binding energy. The probability of a factor to bind a binding site consisting of W nucleotides between i and $i+W-1$ in a sequence $\mathbf{x} = x_{1:L}$ then becomes

$$p(\text{factor bound at position } i|\mathbf{x}) = \left(1+\exp\left(\frac{\Delta G(x_{i:i+W-1}) + \Delta G_i - \mu}{k_B T}\right)\right)^{-1} . \tag{35}$$

We define $p_{\text{motif}}(x_{0:W-1})$ as in eq. (17) and we further define a positional distribution

$$p(z{=}i|\text{factor bound to } \mathbf{x}) = \frac{\exp(-\Delta G_i/k_BT)}{\sum_{i'=1}^{L}\exp(-\Delta G_{i'}/k_BT)} . \tag{36}$$

We abbreviate the denominator as const. gives

$$-\frac{\Delta G_i}{k_B T} + \text{const.} = \log p(z{=}i|\text{factor bound to } \mathbf{x}) =: s_i. \tag{37}$$

Once we know $p_{\text{motif}}(\cdot)$ and $p(z{=}i|\text{factor bound to } \mathbf{x})$, we can compute $S(x_{i:i+W-1})$ and s_i and the relative binding strength $(\Delta G(x_{i:i+W-1}) + \Delta G_i)/k_B T$ for any potential binding site position i in any sequence $\mathbf{x} = (x_1 \dots x_L)$.

If we again assume to be in a regime of unsaturated binding, $p(\text{bound}|\mathbf{x}) \lesssim 0.1$ we can approximate the probability $p(\mathbf{x}_n|\text{bound}, p^k_{\text{motif}})$ for pulling out a sequence $\mathbf{x}_n$ from an underlying distribution of possible sequences $p_{\text{bg}}(\mathbf{x})$ as

$$\begin{aligned}
p(\mathbf{x}_n|\text{bound}, p^k_{\text{motif}}) &\propto p(\text{factor bound}|\mathbf{x}_n, p^k_{\text{motif}})\, p_{\text{bg}}(\mathbf{x}_n) \\
&= p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} p(\text{factor bound at } i|\mathbf{x}_n, p^k_{\text{motif}}) \\
&= p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} \left(1 + \exp\left(\frac{\Delta G(x_{i:i+W-1}) + \Delta G_i - \mu}{k_B T}\right)\right)^{-1} \\
&\approx p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} \exp\left(-\frac{\Delta G(x_{i:i+W-1}) + \Delta G_i - \mu}{k_B T}\right) \\
&\propto p_{\text{bg}}(\mathbf{x}_n) \sum_{i=1}^{L-W+1} \exp\left(S(x_{i:i+W-1}) + s_i\right).
\end{aligned} \tag{38}$$

To find the model parameters $\boldsymbol{\theta}$ consisting of $\mathbf{s} = (s_1, \dots, s_{L-W+1})$ and of p^k_{motif} specifying $p_{\text{motif}}(\cdot)$, we need to optimise the log likelihood function of these parameters:

$$LL(\boldsymbol{\theta}) = \sum_{n=1}^{N} \log p(\mathbf{x}_n|\text{bound}, p^k_{\text{motif}}, \mathbf{s}) \tag{39}$$

Flat Bayesian prior on positional preference

Let us define parameters $\boldsymbol{\pi}$ with $\pi_i = p(z{=}i|z_i{\neq}0) = e^{s_i}$ the probability of a motif to start at position i of a sequence. The M-step will then be given again by equation (24) but this time using the positional preferences π_i instead of the flat positional distribution. We will use a flat prior distribution,

$$p(\boldsymbol{\pi}|\beta) = \text{Dir}(\boldsymbol{\pi}|\beta\mathbb{1}), \tag{40}$$

and we will choose a value around $\beta = 2 \dots 10$.

The auxiliary function becomes

$$
\begin{aligned}
&Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q|\mathbf{r}, p^{k-1}_{\text{motif}}) \\
&= \sum_{n=1}^{N} \left[\sum_{i=0}^{L_n-W+1} r_{ni} \log \left(p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}})\, p(z_n = i|q) \right) \right] + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \boldsymbol{\alpha}) + \log p(\boldsymbol{\pi}|\beta) \\
&= \sum_{n=1}^{N} \sum_{i=0}^{L_n-W+1} r_{ni} \log p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \boldsymbol{\alpha}) \\
&\quad + \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + \sum_{i=1}^{L_n-W+1} r_{ni} \log(q\pi_i) \right) + \log \text{Dir}(\boldsymbol{\pi}|\beta\mathbb{1}) \\
&= \sum_{n=1}^{N} \sum_{i=0}^{L_n-W+1} r_{ni} \log p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \boldsymbol{\alpha}) \qquad (41) \\
&\quad + \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + (1-r_{n,0}) \log q + \sum_{i=1}^{L_n-W+1} r_{ni} \log \pi_i \right) + \sum_{i=1}^{L_n-W+1} (\beta - 1) \log \pi_i .
\end{aligned}
$$

We use the method or Lagrange multipliers again to find the optimum of $Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q|\mathbf{r}, p^{k-1}_{\text{motif}})$ under the constraint $\sum_{i=1}^{L-W+1} \pi_i = 1$:

$$
\frac{\partial}{\partial \pi_i} \left(Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q|\mathbf{r}, p^{k-1}_{\text{motif}}) - \lambda \left(\sum_{i=1}^{L-W+1} \pi_i - 1 \right) \right) = \sum_{n=1}^{N} \frac{r_{ni}}{\pi_i} + \frac{\beta - 1}{\pi_i} - \lambda = 0 \qquad (42)
$$

Solving for π_i, normalising the distribution and defining $N_i := \sum_{n=1}^{N} r_{ni}$ yields

$$
\boxed{\pi_i = \frac{N_i + \beta - 1}{N + (L - W + 1)(\beta - 1)}\,.} \qquad (43)
$$

Prior penalising jumps in the positional preference profile

For many applications it might be more appropriate to limit the complexity of the positional preference profile by imposing a smoothness on the $p(z = i)$. For example, transcription factor binding sites will be more frequent near the center of ChIP-seq peaks than farther away; factors bind more strongly to the outer parts of probes on protein binding microarrays than to the parts near the glass slide; transcription factors in HT-SELEX experiments might prefer the center of probes over the ends. In the following we assume that all training and test sequences have the same length L.

Because the smoothness prior couples neighbouring positional probabilities with each other, there is no closed-form solution for the parameters anymore. We have to use a gradient-based optimisation such as conjugate gradients to minimise Q with respect to the positional parameters. We therefore parameterise the positional distribution in such a way that the normalisation condition $\sum_i \pi_i = 1$ and the limits $0 \leq \pi_i \leq 1$ automatically hold true during the numerical optimisation,

$$
p(z_n = i|z_n \neq 0) = \frac{e^{s_i}}{\sum_{i'=1}^{L-W+1} e^{s_{i'}}}. \qquad (44)
$$

We impose a smoothness prior on the π_i, that encourages the point-wise estimated first derivative to stay small,

$$p(\boldsymbol{\pi}|\beta) = \prod_{i=2}^{L-W+1} \mathcal{N}\left(s_i - s_{i-1}|\, 0, \beta^{-1}\right) , \tag{45}$$

with precision (= inverse variance) β.

With this prior, the auxiliary function becomes

$$\begin{aligned} Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}|\mathbf{r}, p^{k-1}_{\text{motif}}) = &\sum_{n=1}^{N} \sum_{i=0}^{L-W+1} r_{ni} \log p(\mathbf{x}_n|z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}}|p^{k-1}_{\text{motif}}, \boldsymbol{\alpha}) \\ &+ \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + (1-r_{n,0}) \log q + \sum_{i=1}^{L-W+1} r_{ni} \left(s_i - \log \left(\sum_{i'} e^{s_{i'}} \right) \right) \right) \\ &- \frac{\beta}{2} \sum_{i=2}^{L-W+1} (s_i - s_{i-1})^2 + \frac{L-W}{2} \log \beta + \text{const.} \end{aligned} \tag{46}$$

The partial derivatives of $Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}|\mathbf{r}, p^{k-1}_{\text{motif}})$ are

$$\begin{aligned} \frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}|\mathbf{r}, p^{k-1}_{\text{motif}}) = &\sum_{n=1}^{N} r_{ni} - \sum_{n=1}^{N} \sum_{i'=1}^{L-W+1} r_{ni'} \frac{e^{s_i}}{\sum_{i''} e^{s_{i''}}} \\ &- \beta \left(s_i - s_{i-1}\right) I(2 \le i \le L-W+1) \\ &+ \beta \left(s_{i+1} - s_i\right) I(1 \le i \le L-W) \end{aligned} \tag{47}$$

and

$$\boxed{\frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}|\mathbf{r}, p^{k-1}_{\text{motif}}) = N_i - (N - N_0)\, p(z{=}i|z{\neq}0) - (\beta \mathbf{A} \boldsymbol{s})_i}$$

with the abbreviations $N_0 := \sum_{n=1}^{N} r_{n,0}$ and

$$\mathbf{A} := \begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ -1 & 2 & -1 & 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & -1 & 2 & -1 & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & -1 & 2 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & 2 & -1 & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 & -1 & 2 & -1 \\ 0 & \dots & \dots & \dots & 0 & 0 & -1 & 1 \end{pmatrix} . \tag{48}$$

The partial derivative will adjust s_i such that $p(z{=}i|z{\neq}0) = e^{s_i} / \sum_{i'} e^{s_{i'}}$ equals $N_i/(N - N_0)$ plus a smoothness correction $\mathbf{A}\boldsymbol{s}$ that will pull s_i up or down in order to minimise the estimator of the second derivative of the profile at position i. We run a few iterations of conjugate gradients (e.g. 5 to 10) during each EM step to learn the positional preferences.

Learning the optimal smoothness parameter β from the data. We can regard Q also as a function of β,

$$Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}, \beta | \mathbf{r}, p^{k-1}_{\text{motif}}) = -\frac{\beta}{2} \sum_{i=2}^{L-W+1} (\pi_i - \pi_{i-1})^2 + \frac{L-W}{2} \log \beta + \text{const}_\beta \,, \tag{49}$$

and optimise is with respect to β:

$$0 = \frac{\partial}{\partial \beta} Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}, \beta | \mathbf{r}, p^{k-1}_{\text{motif}}) = -\frac{1}{2} \sum_{i=2}^{L-W+1} (s_i - s_{i-1})^2 + \frac{L-W}{2\beta} \tag{50}$$

and therefore

$$\boxed{\beta = \left(\frac{1}{L-W} \sum_{i=2}^{L-W+1} (s_i - s_{i-1})^2 \right)^{-1}} \tag{51}$$

Instead of optimising β, we can again interpret Q as the likelihood of an ensemble of fractional motif instances with weights r_{ni} and compute the expectation value of β. If we assume a uniform prior on β, $p(\beta) = \text{const}$, the posterior distribution of β is proportional to the likelihood. We note that the functional form of $Q(\beta)$ is that of a Gamma distribution, $Q(\beta) = \log \text{Ga}(\beta | a, b) + \text{const} = (a-1) \log \beta - b\beta + \text{const}$, with $a - 1 = (L-W)/2$ and $b = (1/2) \sum_i (s_i - s_{i-1})^2$. Since the expectation value of a Gamma distribution is a/b, we can conclude for β

$$\boxed{\mathbb{E}[\beta] = \left(\frac{1}{L-W+2} \sum_{i=2}^{L-W+1} (s_i - s_{i-1})^2 \right)^{-1} .} \tag{52}$$

We can then update β by its expectation value instead of the mode of $Q(\beta)$. Alternatively, we could sample β from the Gamma distribution $\text{Ga}(\beta | (L-W+2)/2, (1/2) \sum_i (s_i - s_{i-1})^2)$.

Prior penalising kinks in the positional preference profile

For various applications such as PBMs and HT-SELEC, we might be interested in more smooth positional preferences. In these cases, it might be better to use a smoothness prior on the π_i that encourages the point wise estimated *third* derivative to stay small,

$$p(\boldsymbol{\pi} | \beta) = \prod_{i=2}^{L-W} \mathcal{N} \left(s_i - \frac{s_{i-1} + s_{i+1}}{2} \middle| 0, \beta^{-1} \right) , \tag{53}$$

with precision (= inverse variance) β. With this prior, the auxiliary function becomes

$$\begin{aligned} Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi} | \mathbf{r}, p^{k-1}_{\text{motif}}) = & \sum_{n=1}^{N} \sum_{i=0}^{L-W+1} r_{ni} \log p(\mathbf{x}_n | z_n = i, p^k_{\text{motif}}) + \log p(p^k_{\text{motif}} | p^{k-1}_{\text{motif}}, \boldsymbol{\alpha}) \\ & + \sum_{n=1}^{N} \left(r_{n,0} \log(1-q) + (1-r_{n,0}) \log q + \sum_{i=1}^{L-W+1} r_{ni} \left(s_i - \log \left(\sum_{i'} e^{s_{i'}} \right) \right) \right) \\ & - \frac{\beta}{2} \sum_{i=2}^{L-W} \left(s_i - \frac{s_{i-1} + s_{i+1}}{2} \right)^2 + \frac{L-W-1}{2} \log \beta + \text{const.} \end{aligned} \tag{54}$$

The partial derivatives of $Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}|\mathbf{r}, p^{k-1}_{\text{motif}})$ are

$$\begin{aligned}\frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}|\mathbf{r}, p^{k-1}_{\text{motif}}) = &\sum_{n=1}^{N} r_{ni} - \sum_{n=1}^{N} \sum_{i'=1}^{L-W+1} r_{ni'} \frac{e^{s_i}}{\sum_{i''} e^{s_{i''}}} \\ &+ \frac{\beta}{2}\left(s_{i-1} - \frac{s_{i-2}+s_i}{2}\right) I(3 \le i \le L-W+1) \\ &- \beta\left(s_i - \frac{s_{i-1}+s_{i+1}}{2}\right) I(2 \le i \le L-W) \\ &+ \frac{\beta}{2}\left(s_{i+1} - \frac{s_i+s_{i+2}}{2}\right) I(1 \le i \le L-W-1) \end{aligned} \tag{55}$$

and

$$\boxed{\frac{\partial}{\partial s_i} Q(p^k_{\text{motif}}, \boldsymbol{\alpha}, q, \boldsymbol{\pi}|\mathbf{r}, p^{k-1}_{\text{motif}}) = N_i - (N - N_0)\, p(z{=}i|z{\neq}0) - \frac{\beta}{4}(\mathbf{B}\boldsymbol{s})_i}$$

with the abbreviations $N_0 := \sum_{n=1}^{N} r_{n,0}$ and

$$\mathbf{B} := \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ -2 & 5 & -4 & 1 & 0 & \ddots & \ddots & \ddots & \vdots \\ 1 & -4 & 6 & -4 & 1 & \ddots & \ddots & \ddots & \vdots \\ 0 & 1 & -4 & 6 & -4 & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 1 & -4 & 6 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 1 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 6 & -4 & 1 \\ \vdots & \ddots & \ddots & \ddots & \ddots & 1 & -4 & 5 & -2 \\ 0 & \dots & \dots & \dots & \dots & 0 & 1 & -2 & 1 \end{pmatrix}. \tag{56}$$

The partial derivative will adjust s_i such that $p(z{=}i|z{\neq}0) = e^{s_i}/\sum_{i'} e^{s_{i'}}$ equals $N_i/(N-N_0)$ plus a smoothness correction $\mathbf{B}\boldsymbol{s}$ that will pull s_i up or down in order to minimise the estimator of the third derivative of the profile at position i.

Learning the optimal smoothness parameter β from the data. Analogously to the previous smoothness prior, we can learn β from the data using the update

$$\boxed{\beta = \left(\frac{1}{L-W-1} \sum_{i=2}^{L-W} \left(s_i - \frac{s_{i-1}+s_{i+1}}{2}\right)^2\right)^{-1}} \tag{57}$$

or

$$\boxed{\beta = \left(\frac{1}{L-W+1} \sum_{i=2}^{L-W} \left(s_i - \frac{s_{i-1}+s_{i+1}}{2}\right)^2\right)^{-1}}. \tag{58}$$

1.6 Scanning sequences for motif occurrences

To obtain the motif occurrences from the sequences, given a known or learned motif, we developed a motif scanning tool BaMMScan to evaluate the possible motif occurrences on the input sequences. The motif score $s_i(x_{1:K})$ is calculated for each position i on every sequence x for the order K. A background score distribution is created by generating M-fold background sequences from a second-order homogeneous Markov model from input set (M can be 10). We sort the list of $N^+ + N^-$ positive- and negative-set scores jointly in descending order. We denote the cumulative number of scores from the negative set up to rank l in this list by FP_l and then compute the P-value of entry l with score S_l in that list by

$$P\text{-value}(S_l) = \frac{1}{N^-}\left(\mathrm{FP}_l + \frac{S_l^{\text{higher}} - S_l}{S_l^{\text{higher}} - S_l^{\text{lower}} + \epsilon}\right). \tag{59}$$

and the E-values are obtained simply as

$$E\text{-value} = N^+ \times P\text{-value}. \tag{60}$$

The motif occurrences with a P-value smaller than certain cutoff (e.g. $1e^{-4}$) are reported.

1.7 Evaluation criteria using the average recall (AvRec) score

To assess the predictive performance of the motif finders, we first defined an average recall (AvRec) score (details also described in [4]). The AvRec score represents the averaged recall over the range of precision from 0 to 1. The advantage of AvRec score over commonly used p-value is that it covers the most relevant range of False-discovery-rates (FDR) in practical applications and allows the user to intuitively estimate the motif performance in her particular application.

We obtain a p-value for each sequence by

$$p\text{-value}_l = \frac{\mathrm{FP}_l + 0.5}{\mathrm{N}^- + 1} \tag{61}$$

After having a p-value for every motif occurrence (as described in eq.59), we obtain a list of corresponding local FDR values and an estimate of the weight of the null component η_0 by applying fdrtool [5] on the p-value distribution (Figure S 13A). We then calculate FDR and recall for each entry by

$$\mathrm{FDR}_l = \frac{\mathrm{FP}_l}{\mathrm{FP}_l + \mathrm{TP}_l} \tag{62}$$

$$\mathrm{recall}_l = (1 - \mathrm{FDR}_l)\frac{l}{(1-\eta_0)N} \tag{63}$$

The ratio between true positive (TP) and false positive (FP) is calculated by

$$R_{l[\mathrm{TP/FP}]} = \frac{1 - \mathrm{FDR}_l}{\mathrm{FDR}_l} \times M \tag{64}$$

with M as the ratio between negative and positive sequences.

We visualize the characteristics by plotting the TP/FP ratio $R_{\mathrm{TP/FP}}$ on the y-axis against the recall on the x-axis, and define the calculated area-under-the-curve as the AvRec score for motif evaluation (Figure S 13B).

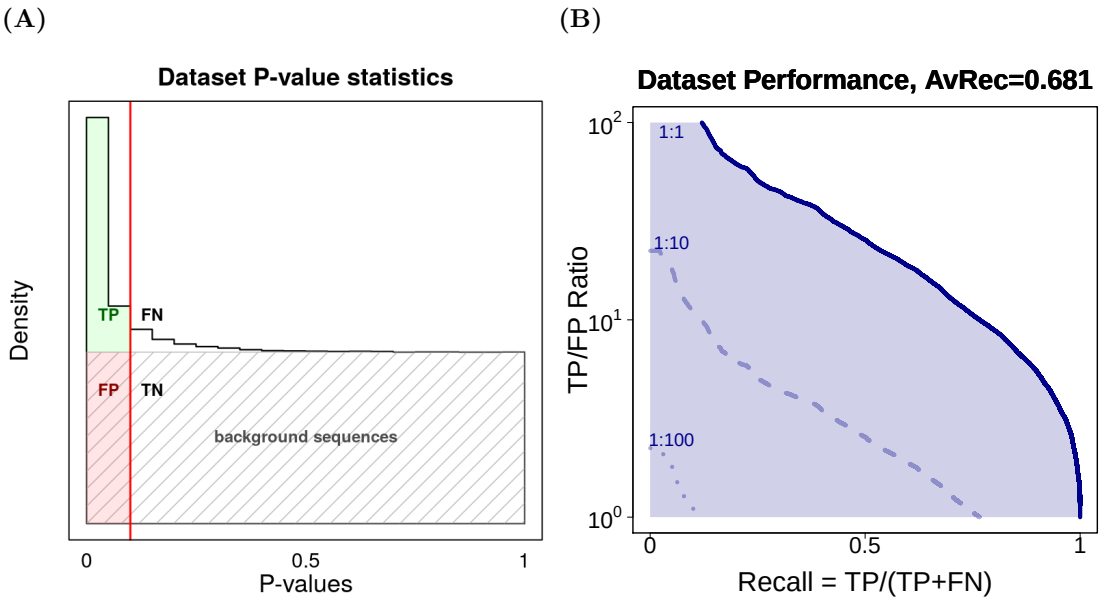


Figure S 13. Schemes of motif assessment on sequences. (**A**) We calculate p-values for the most likely positions on both positive and background sequences, given the motif and a second-order background model learned from the sequences. We plot the density of the p-values and choose a cutoff at 0.1 (the solid red line). The background sequences are mapped in the grey shadow, given the ratio between the background and positive sequences. The true positives (TP, in green), false negatives (FN, in white), false positives (FP, in red) and true negatives (TN) are visualized on the plot. (**B**) For each p-value for positive sequences, we calculate the recall and the ratio between TP and FP, and then plot the recall against the ratio of TP/FP. The solid dark blue line represent for the scenario when the ratio between positive and background sequences is 1:1, the dash lines under it are for the cases when the ratio is 1:10 and 1:100, respectively. An average recall (AvRec) score is calculated as the area under the curve for the 1:1 ratio scenario, and used as a measurement for motif quality on the positive sequence.

2 Datasets used for the benchmark

2.1 ENCODE database

We evaluated the performance of the selected algorithms on human ChIP-seq datasets from the ENCODE portal [6] till March 2012. In total, there are 435 datasets for 93 distinct transcription factors. The top 5000 peak regions, sorted by their signal value, were selected for each dataset. If fewer than 5000 peaks were contained in a dataset, all peaks were chosen. Positive sequences were extracted ±104 bp around the peak summits. Background sequences were sampled by trimer frequencies from positive sequences, with the same length as positive sequences and 10 times the amount of positive sequences. 8 datasets were excluded from all

the results because diChIPMunk failed to learn models within 3 hours.

2.2 HT-SELEX datasets

For HT-SELEX data, we downloaded 164 datasets with 200 bp-long oligomers from Zhu et al. [7], which are deposited in the European Nucleotide Archive (ENA) under accession PRJEB22684. Each dataset represents one non-redundant human transcription factor. For each dataset, we selected the top 5000 sequences from the 4th cycle without any sorting. Background sequences are sampled in the same way as described previously.

2.3 GTRD database

For the GTRD database, we obtained 405 *in vivo* datasets for 405 non-redundant human transcription factors from Yevshin et al. [1]. The top 5000 peak regions are selected after sorting by q-values. Positive sequences are extracted ±100 bp around the peak summits. Background sequences are sampled in the same way as described previously.

2.4 MITOMI datasets

MITOMI is a microfluidics-based approach for *de novo* discovery and quantitative biophysical characterization of DNA target sequences [8]. We downloaded the MITOMI data for 28 *Saccharomyces cerevisiae* transcription factors under the accession GPL10817. The 3 bp and 15 bp long adapters on both ends are truncated. We then downloaded yeast GTRD datasets for 8 transcription factors [1] for the motif discovery.

2.5 Cross-platform datasets

Out of 435 ENCODE datasets for 93 TFs and 164 HT-SELEX datasets for 164 non-redundant TFs, there are 66 TFs which have both *in vivo* and *in vitro* datasets. Out of 66 TFs, most of them have very low AvRec scores when performing the cross-platform validations. We investigated into details and found out that for most of them, the learned motifs were very distinct from the two platforms. This result confirms that TFs can bind to different motifs when experimenting either *in vivo* or *in vitro*. For the left 16 paired tests, they are motifs for 4 TFs, namely CEBPB, POU2F2, ELF1, and FOXA2, which were used in our benchmark.

3 Motif finders used in the benchmark

The source code is available for command-line versions of PEnGmotif and BaMMmotif2 and supported on Linux and MacOS:

3.1 PEnGmotif

PEnGmotif repository: github.com/soedinglab/PEnG-motif. For this study, we used parameters `--optimization_score MUTUAL_INFO -w 8 --threads 4`. The output is in MEME-like format. The motifs are sorted by their AvRec scores, and the best one was taken for the benchmark.

3.2 BaMMmotif2

BaMMmotif2 repository: github.com/soedinglab/BaMMmotif2. For this study, we seeded with the PWMs discovered by PEnGmotif and used parameters `--EM -k [k] --advanceEM --extend 2 2` for further optimization. `[k]` is chosen as 1 and 5 for the benchmark for this study. The output format is defined as BaMM format with extensions like `.ihbcp` and `.hbcp`.

3.3 BaMMmotif

BaMMmotif repository: github.com/soedinglab/BaMMmotif. For this study, we seeded with PWMs by triggering XXmotif internally and used parameters `--reverseComp --XX-localization --XX-localizationRanking --XX-K 2 --maxPValue 0.05 --maxPWMs 3 --extend 2 2` for further optimization. The output format is defined as BaMM format with extensions like `.ihbcp` and `.hbcp`.

3.4 CisFinder

CisFinder was installed from https://lgsun.grc.nia.nih.gov/CisFinder/download.html. We ran patternFind for identifying motifs, patternCluster for clustering motifs, and patternTest for improving motifs. Default parameters were applied. The discovered motifs were converted to MEME-like output format and re-ranked by our motif sorting script, and only the best motif was taken for the benchmarks.

3.5 MEME

MEME version 5.1.1 was installed and applied with parameters `-dna -mod zoops -nmotifs 3 -revcomp -p 4 -V 2`. Maximum 3 motifs were saved in the output, and the best one according to the AvRec score was taken for the benchmarks.

3.6 ChIPMunk

ChIPMunk version v8 was downloaded and applied with parameters `ru.autosome.ChIPMunk 8 12 yes 1.0 100 10 1 4`. The discovered motifs were converted to MEME-like output format.

3.7 diChIPMunk

diChIPMunk was implemented in the same package as ChIPMunk. We ran it with parameters `ru.autosome.di.ChIPMunk 8 12 yes 1.0 200 20 1 4`. The discovered motifs were converted to BaMM-like output format for further comparison.

3.8 InMoDe

InMoDe was downloaded from http://www.jstacs.de/index.php/InMoDe. We applied the module `flexible`, which allows us to customize the learning task. The discovered motifs were converted to BaMM-like output format for further comparison.

References

1. Yevshin, I., Sharipov, R., Kolmykov, S., Kondrakhin, Y., and Kolpakov, F. (2019) GTRD: a database on gene transcription regulation—2019 update. *Nucleic Acids Res.,* **47**(D1), D100–D105.

2. Khan, A., Fornes, O., Stigliani, A., Gheorghe, M., Castro-Mondragon, J. A., van der Lee, R., Bessy, A., Cheneby, J., Kulkarni, S. R., Tan, G., et al. (2018) JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. *Nucleic Acids Res.,* **46**(D1), D260–D266.

3. Siebert, M. and Söding, J. (2016) Bayesian Markov models consistently outperform PWMs at predicting motifs in nucleotide sequences. *Nucleic Acids Res.,* **44**(13), 6055–6069.

4. Kiesel, A., Roth, C., Ge, W., Wess, M., Meier, M., and Söding, J. (2018) The BaMM web server for de-novo motif discovery and regulatory sequence analysis. *Nucleic Acids Res.,* **46**(W1), W215–W220.

5. Strimmer, K. (2008) fdrtool: a versatile R package for estimating local and tail area-based false discovery rates. *Bioinformatics,* **24**(12), 1461–1462.

6. ENCODE Project Consortium and others (2012) An integrated encyclopedia of DNA elements in the human genome. *Nature,* **489**(7414), 57–74.

7. Zhu, F., Farnung, L., Kaasinen, E., Sahu, B., Yin, Y., Wei, B., Dodonova, S. O., Nitta, K. R., Morgunova, E., Taipale, M., et al. (2018) The interaction landscape between transcription factors and the nucleosome. *Nature,* **562**(7725), 76–81.

8. Fordyce, P. M., Gerber, D., Tran, D., Zheng, J., Li, H., DeRisi, J. L., and Quake, S. R. (2010) De novo identification and biophysical characterization of transcription-factor binding sites with microfluidic affinity analysis. *Nat. Biotechnol.,* **28**(9), 970–975.

3.2 BaMM webserver

3.2.1 Overview

We have developed BaMM webserver, a platform for

1. predicting motifs from DNA/RNA sequences
2. finding motif occurrences given a sequence and a motif model
3. searching for similar known motifs in the database, given a novel motif model
4. offering databases with higher-order BaMM models for different organisms

3.2.2 Article: The BaMM webserver for *de novo* motif discovery and regulatory sequence analysis

Anja Kiesel†, Christian Roth†, **Wanwan Ge**, Maximilian Wess, Markus Meier and Johannes Söding*

This article was published online on Nucleic Acids Research, Volume 46, Issue W1, 2 July 2018, Pages W215–W220.

DOI: 10.1093/nar/gky431

URL: bammmotif.soedinglab.org

Author contributions

Johannes Söding (JS) and Anja Kiesel (AK) initialized the idea. AK initially built the framework of the webserver. Wanwan Ge (WG) implemented the algorithm of BaMMmotif2 and realized functionalities such as motif evaluation, motif scanning and prepared the databases for different species. Christian Roth (CR) re-structured the webserver to boost the functionalities. Maximilian Wess (MW) and Markus Meier (MM) helped to set up the webserver. JS, AK, CR and WG jointly wrote the manuscript. JS, CR and WG revised the manuscript for the re-submission.

Published online 28 May 2018 *Nucleic Acids Research, 2018, Vol. 46, Web Server issue* ***W215–W220***
doi: 10.1093/nar/gky431

The BaMM web server for *de-novo* motif discovery and regulatory sequence analysis

Anja Kiesel†, Christian Roth†, Wanwan Ge, Maximilian Wess, Markus Meier and Johannes Söding*

Quantitative and Computational Biology, Max Planck Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany

Received February 14, 2018; Revised May 05, 2018; Editorial Decision May 06, 2018; Accepted May 09, 2018

ABSTRACT

The BaMM web server offers four tools: (i) *de-novo* discovery of enriched motifs in a set of nucleotide sequences, (ii) scanning a set of nucleotide sequences with motifs to find motif occurrences, (iii) searching with an input motif for similar motifs in our BaMM database with motifs for >1000 transcription factors, trained from the GTRD ChIP-seq database and (iv) browsing and keyword searching the motif database. In contrast to most other servers, we represent sequence motifs not by position weight matrices (PWMs) but by Bayesian Markov Models (BaMMs) of order 4, which we showed previously to perform substantially better in ROC analyses than PWMs or first order models. To address the inadequacy of P- and E-values as measures of motif quality, we introduce the AvRec score, the average recall over the TP-to-FP ratio between 1 and 100. The BaMM server is freely accessible without registration at https://bammmotif.mpibpc.mpg.de.

INTRODUCTION

Many methods such as ChIP-seq or high-throughput SELEX (1) produce a set of nucleotide sequences that are preferentially bound by a protein of interest *in vitro* or *in vivo*. From such data, a motif model for the sequence dependence of the binding affinity of the protein to the DNA or RNA can be derived. This model can then be used to predict binding sites and their strengths in other sequences.

Position weight matrices (PWMs) are the standard model to describe binding motifs. In the PWM every motif position contributes additively and independently from other positions to the total binding energy. Even though the approximation of independence of positions works well for many transcription factors, dependencies do occur (2,3), for example due to bendability or shape constraints during binding (4), to multiple binding configurations of the protein (5), or to cooperative interactions between closely binding factors that can modulate each others' binding affinities (6).

PWMs can be generalized to Markov models of order k that account for nucleotide dependencies by conditioning the probability for the four nucleotides at each motif position on the previous k nucleotides. First-order Markov models have been added to the popular motif databases JASPAR and HOCOMOCO (7,8). Models of order 2 and higher have not yet been adopted in the major databases, probably due to the difficulties to robustly train the many parameters of these models on limited data.

We recently developed Bayesian Markov Models (BaMMs) (9), which efficiently prevent overfitting by automatically learning conditional probabilities only up to an order k at which they can still be estimated reliably. The key idea is that the conditional probabilities of order $k-1$ are used as prior probabilities for the conditional probabilities of order k. We have shown that BaMMs of order 4 and 5 systematically outperform PWMs and first-order models in distinguishing bound sequences from negative sequences generated by a second-order Markov model (9).

A very popular web server for regulatory sequence analysis based on PWMs offering a wide choice of tools is the MEME server (10). The RSAT web server (11) provides a general toolbox for the analysis of regulatory sequences including motif-based analyses. Furthermore, other web resources and databases are available for training first-order models (12,13).

The BaMMmotif server brings the improved quality of BaMM motif models within reach of users unfamiliar with command-line tools, in a largely self-explanatory web interface designed for ease of use. The user can discover BaMM models enriched in a set of input sequences, scan sequence sets with BaMM models for motif occurrences, and compare discovered or uploaded motifs with a database of BaMM models learned from ChIP-seq datasets.

*To whom correspondence should be addressed. Tel: +49 551 201 2890; Fax: +49 551 201 2803; Email: soeding@mpibpc.mpg.de
†The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

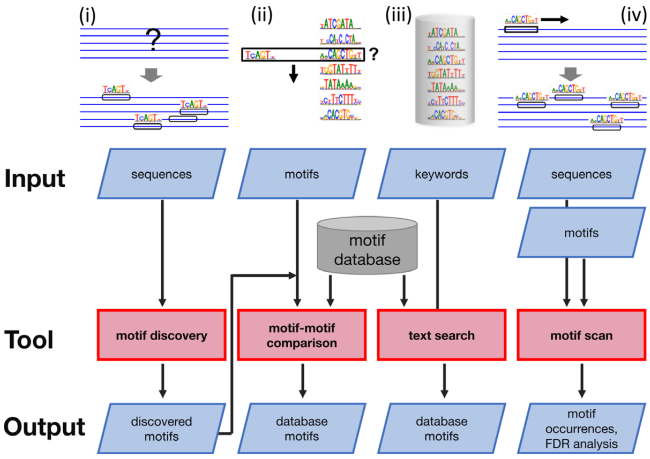


Figure 1. Tools offered by the BaMM server: (i) de-novo discovery of motifs enriched in a nucleotide sequence set. Motifs are represented by higher order BaMMs, which capture correlations between nucleotides. (ii) Searching with an input BaMM or PWM motif for similar motifs in our database of over 1000 fourth-order BaMM motifs. (iii) Browsing and keyword searching in our motif database. (iv) Scanning a set of nucleotide sequences with BaMM or PWM motifs to find motif occurrences.

BAMM TOOLS

In the following we describe the four tools offered by the BaMM server (Figure 1).

De-novo motif discovery using higher-order BaMMs

This tool discovers the motifs enriched in an input set of nucleotide sequences in comparison to the expectation from a background model. For example in sequences obtained from a ChIP-seq or HT-SELEX experiment, the BaMM motif models will approximately describe the sequence dependence of the binding energy of the protein to DNA (see page 2 of supplementary material in (9)). The motif model can be used to scan other sequences for motif occurrences (see next subsection).

Method. The motif discovery proceeds in two stages, seed pattern discovery and motif refinement. For the pattern discovery we developed a fast and sensitive algorithm (PEnGmotif) that will be described in detail elsewhere. Briefly, it finds all locally optimal W-mers (default $W = 8$) over an alphabet of 11 IUPAC letters (A, C, G, T, R = A or G, Y = C or T, W = A or T, S = C or G, M= A or C, K = G or T, N = A, C, G or T), where locally optimal patterns are those for which changing any single one of its letters would result in a decreased enrichment relative to the random expectation from the background model. (Alternatively, the P-value or the mutual information between presence/absence of motifs and input versus background sequence can be optimized.) With each locally optimal pattern, a PWM of length W is initialized and optimized using an expectation maximization (EM) algorithm. PWMs that have very similar overlapping regions are merged and ranked by our new AvRec score (next section).

The seed motifs are then refined using BaMM!motif (9). It learns the parameters of the BaMMs with an EM algorithm that maximizes the log likelihood of the motif model under a zero-or-one-occurrence-per-sequence (ZOOPS) model (14). The BaMM server offers to train motifs of up to fourth order.

By default, BaMM learns a second order Markov model from the input sequences as a background model. The background model is needed first in the motif discovery to model the sequence stretches not modeled by the motif model and second in the motif quality assessment step to generate negative sequences to estimate motif occurrence P-values. A second order model is generally preferable to first or zeroth order as it can better describe sequence biases observed in open versus closed chromatin, ChIPped versus unChIPped sequences etc. (15). A model of order 1 or 0 is recommended for the discovery of very short motifs (e.g. four to five nucleotides) such as to RNA-binding sites, as such short motifs could be learned to some extent even by a second order background model, severely reducing the sensitivity to discover them.

Usage of de-novo motif discovery. After uploading a FASTA file of up to 50 MB with the input sequences, the motif discovery can be started. A drop-down menu offers advanced options in four categories: general settings, seeding stage, model refinement stage and settings for plots and analyses.

In the general settings category the user can choose whether the motif can be present on both strands, set the order of the background model (default 2) and upload an optional sequence set to train the background model on. Settings of the seeding stage include the initial pattern length W, the z-score significance threshold for refining a motif, and the objective function to optimize in the search for locally optimal patterns. For the refinement stage the user can choose the motif model order (default 2) and the number of flanking positions on the left and right of the core model found in the seed stage. Finally, the user can choose to skip motif scanning, motif performance evaluation or motif annotation, and change the significance thresholds for scanning and annotation.

By default up to four best-performing seed patterns are refined to higher-order models. Seed patterns are ranked by their average recall (AvRec) score (see below). Alternatively, the user can choose to select seed patterns manually for refinement after the seeding stage.

The results page (Figure 2A) lists in a summary table the discovered enriched motifs with their IUPAC patterns, the sequence logos of the 0th-order model (forward and reverse complement), the AvRec motif quality score and the fraction of sequences with motifs ('frac. occurrence'), estimated using the fdrtool (16) (explained in subsection 'Dataset AvRec and motif AvRec'). By clicking on the motifs or scrolling down, detailed results for the motifs are shown: 0th-order (forward and reverse complement), first- and second-order sequence logos (Figure 2B); four motif quality assessment plots and a plot of the positional distribution of the motif occurrences relative to the center of the sequences (Figure 2C). (Sequences do not have to be of the same length.) Clicking on the download button in the summary table above saves a zip file containing motif files in BaMM format with the extension ihbcp and all analysis

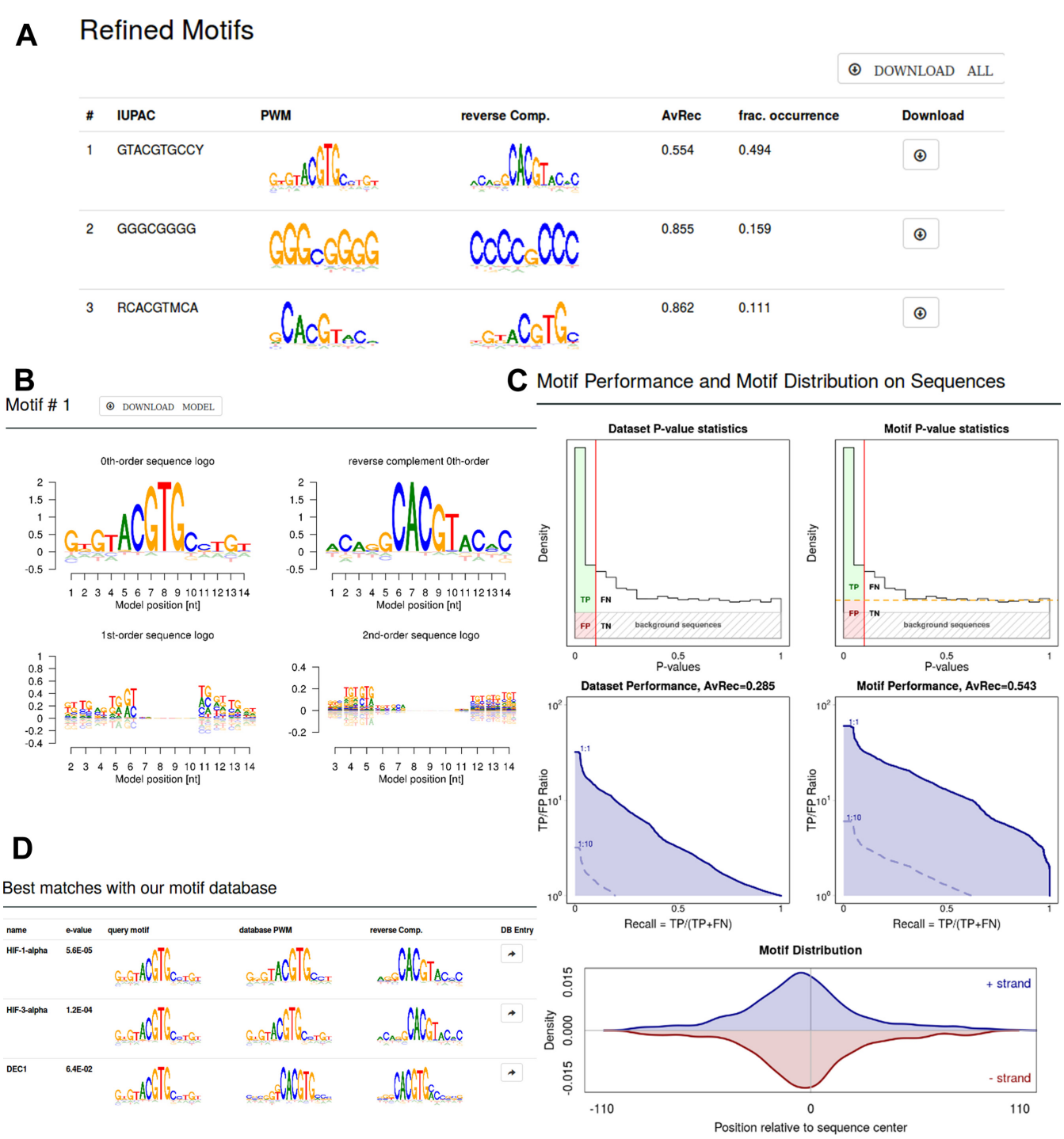


Figure 2. Selected results from a de-novo motif discovery run. (**A**) Summary table of discovered motifs. (**B**) Sequence logos of order 0, 1 and 2 for one discovered motif. (**C**) Motif quality analysis and positional distribution. In the dataset-centered analysis (left) all input sequences are defined as positives. In the motif-centered analysis (right), only input sequences carrying a motif occurrence are positives. Their fraction is estimated using fdrtool (orange broken line on the upper right). The quality of motifs is quantified by average recall (AvRec), the blue area under the TP-to-FP-versus-recall curves. The curves for positive-to-negative ratios in the dataset of 1:1, 1:10 and 1:100 are plotted. Recall = TP/(TP + FN), where TP = true positives, FP = false positives, FN = false negatives. Positional distribution of the motif occurrences relative to the center of the sequences is shown on the bottom. (**D**) List of database motifs similar to discovered motif.

plots for the motif. Last, the database motifs found similar to the discovered motif are listed (see 'motif-motif comparison' below) with links to the database entry ('Best matches with our motif database', Figure 2D). The results page can later be retrieved by giving the job ID on the 'Find my job' page. Results are stored for up to 3 months.

SCAN SEQUENCES FOR MOTIF OCCURRENCES

A set of input sequences can be scanned with a motif or a set of motifs for motif occurrences. The input motifs can be in MEME (version 4 and above) or BaMM format and could have been discovered de-novo by BaMM or they could come from the BaMM database or some other database.

We developed a motif scanning tool that evaluates the log odds score for BaMMs (and PWMs) of any order. A table with the motif occurrences can be downloaded in a zip file, together with the motif analysis on the supplied sequences. The table of motif occurrences contains in each line the sequence length, motif position, binding sites, *P*-value, and E-value of the occurrence. The *P*-values are computed by maximum-likelihood fitting of the high-scoring tail of the log-odds score distribution on sequences generated with the background model with an exponential function, which gave good fits (see PhD thesis at https://edoc.ub.uni-muenchen.de/21504/). Each motif is also evaluated using the dataset and motif-based average recall (AvRec, see below) and the positional distribution of the motif occurrences around the center of the sequences (Figure 2C).

BAMM MOTIF DATABASE

Our database contains 1021 fourth-order BaMMs trained on ChIP-seq datasets of 620 human transcription factors (TFs), 345 mouse TFs, 19 rat TFs, 16 zebrafish TFs and 21 yeast TFs from the GTRD database (17). For each motif, a meta table, details with higher-order sequence logos, positional enrichment around the centers of training sequences, and motif quality assessment plots, evaluated on the ChIP-seq training sequences, are presented. The user can browse the database or perform a text search through the list of names of the transcription factor.

SEARCH WITH QUERY MOTIFS THROUGH THE MOTIF DATABASE

This tool searches for motifs in our BaMM motif database that are similar to the query motifs (in MEME or BaMM format). This motif-motif search is automatically run after de-novo motif discovery using each of discovered motifs as query. The query motifs can also be provided by the user. The output of this tool is shown in Figure 2D.

Motif-motif similarities are computed between the zeroth order contribution of the motifs. The distance between two motifs is the minimum distance for any gapless alignment of their columns that leaves at least four columns aligned. The similarity between aligned motifs M_1 and M_2 is defined as

$$\sum_j \left(-d^{\mathrm{JS}}(M_{1j}, M_{2j}) + d^{\mathrm{JS}}(M_{1j}, M_{\mathrm{bg}}) + d^{\mathrm{JS}}(M_{2j}, M_{\mathrm{bg}})\right).$$

Here, the sum runs over all aligned columns j. $d^{\mathrm{JS}}(M_{1j}, M_{2j})$ is the Jenssen-Shannon divergence between the four nucleotide probabilities of model 1 and of model 2 at aligned column j, and M_{bg} is the zeroth order background distribution in the set on which the query model was learned.

The *E*-values for the motif-motif matches are computed from these similarity scores by fitting the density of scores computed between 100 randomized query motifs and the databases motifs and fitting the high-scoring tail with an exponential distribution (see PhD thesis of Anja Kiesel at https://edoc.ub.uni-muenchen.de/21504/). The randomization of the query motif is achieved by exchanging A with T probabilities of each position with probability 0.5, and analogously for C and G. In addition columns within 2 positions of each other were randomly swapped. This motif randomization keeps the local GC vs. AT content conserved. In our benchmarks, this score performed as well as the best of the TOMTOM scores (Pearson correlation) (18). An example of results of the motif search is shown in Figure 2D.

MOTIF QUALITY ASSESSMENT AND RANKING

P-values do not assess biological relevance of motifs

P-values and *E*-values have a severe drawback for ranking motif models: They can be very significant and yet the motifs have no biological relevance at all. For a fixed x-fold enrichment of motif occurrences on the input set in comparison to the background model, the *P*-value decreases exponentially with the number of sequences in the zero-or-one-occurrence-per-sequence (ZOOPS) model. For that reason, even biologically irrelevant motifs with very slight enrichment factors (e.g. 1.1) can obtain an extremely significant *E*-value if the input set is large enough. Small enrichment factors can occur frequently in practice simply due to an imperfect background model that slightly underestimates the expected frequency of occurrence.

Precision, recall and false discovery rate

To get a more relevant measure of how well the motif model can separate sequences with a motif (positives) from the background sequences (negatives), we first generate for each input sequence one random sequence of the same length sampled with the second-order Markov background model learned from the input sequences. The score for an input or background sequence is the maximum of the log odds scores of the BaMM over all possible motif positions (ZOOPS model). Every sequence with a score above a cut-off is predicted to carry a motif. We rank all sequences by their score and, for each cut-off score, we count the number of correct predictions above that score, called true positives (TP), and the number of incorrect predictions above the cut-off score, called false positives (FP). The precision is the fraction of predictions that are correct, TP/(TP + FP), and the recall (=sensitivity) is the fraction of positive sequences that are actually predicted, TP/(TP + FN). The false discovery rate is FDR = 1 − precision = FP/(TP + FP).

If we did this analysis on the same sequences from which we had trained the model, we could easily overestimate the motif model performance by overtraining. We therefore use

four-fold cross-validation to assess the motif model performance: We split the input and background sequences into four equal-sized parts, retrain the model on three. The results from the four hold-out sets are then combined.

The AUPRC assesses models partly in irrelevant regimes

The area under the recall-precision curve (AUPRC) (see Supplementary Figure S2B) can be interpreted as mean model recall (=sensitivity) averaged over the entire range of precision from 0 to 1. Consider two models: one achieves a maximum precision of 0.99 and the other achieves at any recall a 1% higher precision, with a maximum at 0.9999. Even though the two models have AUPRCs that only differ by 1%, their minimum false discovery rates differ by two orders of magnitude (0.01 and 0.0001), which can make a huge difference in practice.

Consider two application cases. In the first, the expected ratio of sequences with and without true binding sites is ∼1:1, e.g. for a ChIP-seq experiment, and in the second case it is 1:100, e.g. when scanning 10^4 promoter regions in the human genome for motif occurrences, of which 100 are expected to carry the motif. In the first case, an FDR of 0.1, determined at ratio 1:1 between positive and negative (background) sequences, is quite satisfactory to identify sequences with true binding sites. In the second case, an FDR of 0.1 would result in $0.1 \times 10^4 = 1000$ false predictions, which would swamp the expected 100 true binding occurrences. A model with an FDR of 0.001 determined at ratio 1:1 between positive and negative sequences would give us $0.001 \times 10^4 = 10$ false predictions, which would result in an acceptable FDR of 10/110.

So the FDR (estimated for a ratio 1:1 of positives to negatives) that is relevant to assess the quality of motif models depends on the application, more precisely, on the expected ratio of positives to negatives in the sequence data. In contrast, the AUPRC puts much weight on very high FDRs, e.g. the range between 0.9 and 1 has as much weight as the range between 0 and 0.1. Another popular measure, the area under the receiver operator curve (AUROC), can be shown to be even less relevant and difficult to interpret for motif model assessment.

Average recall (AvRec)

We sought a motif quality analysis plot and associated quality measure (i) that covers the range of FDRs most relevant in practical applications and (ii) that allows the user to easily estimate the performance of the motif in her particular application, that is, given the ratio between positive and negative sequences expected for her application.

We replace the precision in the precision-recall plot by $\log_{10}$ of the ratio R = TP/FP between true and false positives, $\log_{10}$TP/FP (Figure 2C, middle). From the ratio R one can immediately obtain the false discovery rate, FDR = $1/(1 + R)$, and vice versa, $R = (1 - \text{FDR})/\text{FDR}$. $R = 100$ corresponds to FDR = 1/101, $R = 1$ corresponds to FDR = 0.5. We define the AvRec quality measure as the average recall computed over a range of $\log_{10} R$-values from 0 to 2, which corresponds to an FDR-range from 1/101 to 0.5. We argue that this range of FDRs is most relevant in practice, as illustrated by the two previous examples.

The new quality measure also satisfies the second requirement. The user can simply pick the curve in the AvRec plot that corresponds to the ratio of positive to negative sequences that she expects in her application. Nicely, the curve at ratio 1:10 is the curve at ratio 1:1 shifted down by one unit ($\log_{10}10$), because R is proportional to the ratio of positive to negative sequences in the dataset: When the number of negative sequences is amplified by 10, the number of false positive predictions will also be increased by a factor of 10. On the web server, we show the curves with ratios of 1:1, 1:10 and 1:100 (if visible on the y-scale).

Dataset AvRec and motif AvRec

We used two definitions of positive and negative sequences. In the *dataset-centered analysis* (Figure 2C, left), the true positive sequences are all sequences from the input set above the cut-off score and the false positive sequences are all background sequences above the cut-off score. The upper left plot in Figure 2C shows the distribution of the motif occurrence P-values computed from their scores. The curve below shows the $\log_{10}$TP/FP values over the recall for this definition of true and false positives.

In the *motif-centered analysis* (Figure 2C, right), we consider only those sequences as true positives that actually contain a motif instance. In order to estimate the number of TPs for a given score cut-off, we first estimate the fraction of input sequences that contain motif instances using the fdrtool (16). This tool assumes that the negative sequences in the positive set are uniformly distributed over all P-values between 0 and 1 and fits a horizontal line giving the fraction of negatives in the input set to the distribution (orange broken line in Figure 2C, top right). The definition of TPs and FPs illustrated in the top right graph of Figure 2C results in the motif-based AvRec analysis plot below.

When the fraction of motifs in the input sequences is near 100%, both approaches yield very similar results. But when this fraction is small, the motif model may still be very accurate. The motif-centered analysis takes account of that, while the dataset-centered analysis severely underestimates the model performance in these cases.

DOCUMENTATION, USABILITY AND SPEED

Each input parameter is briefly explained in a mouse-over text. A detailed documentation is accessible via the 'Documentation' tab on the top of each page. A motif discovery run with 10k (100k) sequences of length 200nt takes around 3.0 (12.5) min. Scanning 100k sequences of length 200nt on both strands for motif matches takes about 6 min per three motifs. A motif-motif search through the largest subcollection of motifs in our database (620 models) takes around 3.5 min per three motifs.

IMPLEMENTATION

The BaMM web server is built on the Django Web framework using Nginx as reverse proxy. Jobs are scheduled via Celery's asynchronous task queuing system, with the help of

Redis as a message broker, and executed on a Linux computer with 28 physical cores using 4 cores per job. MySQL is used as back end database to store results and job parameters. The web front end, back end and the database run in separate Docker containers, enabling easy deployment (Supplementary Figure S1).

CONCLUSION

We hope the BaMM web server will enable many users to exploit the greater descriptive power of BaMMs for motif discovery and regulatory sequence analysis. In the future we will work on extending the database of motifs, especially by training on HT-SELEX datasets.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We thank our beta users for testing and feedback and Fëdor Kolpakov of BioUML (http://gtrd.biouml.org) for support with their GTRD database.

FUNDING

German Federal Ministry of Education and Research (BMBF) within the frameworks of e:Bio [SysCore, project 0316176A]; SPP 1935 (project CR 227/6-1) of the German Research Foundation (DFG); International Max Planck Research School for Genome Science (IMPRS-GS). Funding for open access charge: Institutional.
Conflict of interest statement. None declared.

REFERENCES

1. Lambert,S.A., Jolma,A., Campitelli,L.F., Das,P.K., Yin,Y., Albu,M., Chen,X., Taipale,J., Hughes,T.R. and Weirauch,M.T. (2018) The human transcription factors. *Cell*, **172**, 650–665.
2. Bulyk,M.L., Johnson,P.L. and Church,G.M. (2002) Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucleic Acids Res.*, **30**, 1255–1261.
3. Jolma,A. and Taipale,J. (2011) Methods for analysis of transcription factor DNA-binding specificity in vitro. In *A Handbook of Transcription Factors*, Springer pp. 155–173.
4. Rohs,R., West,S.M., Sosinsky,A., Liu,P., Mann,R.S. and Honig,B. (2009) The role of DNA shape in protein–DNA recognition. *Nature*, **461**, 1248
5. Badis,G., Berger,M.F., Philippakis,A.A., Talukder,S., Gehrke,A.R., Jaeger,S.A., Chan,E.T., Metzler,G., Vedenko,A., Chen,X. *et al.* (2009) Diversity and complexity in DNA recognition by transcription factors. *Science*, **324**, 1720–1723.
6. Jolma,A., Yin,Y., Nitta,K.R., Dave,K., Popov,A., Taipale,M., Enge,M., Kivioja,T., Morgunova,E. and Taipale,J. (2015) DNA-dependent formation of transcription factor pairs alters their binding specificity. *Nature*, **527**, 384.
7. Khan,A., Fornes,O., Stigliani,A., Gheorghe,M., Castro-Mondragon,J.A., van der Lee,R., Bessy,A., Chèneby,J., Kulkarni,S.R., Tan,G. *et al.* (2017) JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. *Nucleic Acids Res.*, **46**, D260–D266.
8. Kulakovskiy,I.V., Vorontsov,I.E., Yevshin,I.S., Sharipov,R.N., Fedorova,A.D., Rumynskiy,E.I., Medvedeva,Y.A., Magana-Mora,A., Bajic,V.B., Papatsenko,D.A. *et al.* (2018) HOCOMOCO: towards a complete collection of transcription factor binding models for human and mouse via large-scale ChIP-Seq analysis. *Nucleic Acids Res.*, **46**, D252–D259.
9. Siebert,M. and Söding,J. (2016) Bayesian Markov models consistently outperform PWMs at predicting motifs in nucleotide sequences. *Nucleic Acids Res.*, **44**, 6055–6069.
10. Bailey,T.L., Boden,M., Buske,F.A., Frith,M., Grant,C.E., Clementi,L., Ren,J., Li,W.W. and Noble,W.S. (2009) MEME SUITE: tools for motif discovery and searching. *Nucleic Acids Res.*, **37**, W202–W208.
11. Medina-Rivera,A., Defrance,M., Sand,O., Herrmann,C., Castro-Mondragon,J.A., Delerce,J., Jaeger,S., Blanchet,C., Vincens,P., Caron,C. *et al.* (2015) RSAT 2015: regulatory sequence analysis tools. *Nucleic Acids Res.*, **43**, W50–W56.
12. Mathelier,A. and Wasserman,W.W. (2013) The next generation of transcription factor binding site prediction. *PLOS Comput. Biol.*, **9**, e1003214
13. Kulakovskiy,I., Levitsky,V., Oshchepkov,D., Bryzgalov,L., Vorontsov,I. and Makeev,V. (2013) From binding motifs in ChIP-Seq data to improved models of transcription factor binding sites. *J. Bioinf. Comput. Biol.*, **11**, 1340004.
14. Bailey,T.L. and Elkan,C. *et al.* (1994) Fitting a mixture model by expectation maximization to discover motifs in bipolymers. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **2**, 28–36.
15. Hartmann,H., Guthöhrlein,E.W., Siebert,M., Luehr,S. and Söding,J. (2013) *P*-value-based regulatory motif discovery using positional weight matrices. *Genome Res.*, **23**, 181–194.
16. Strimmer,K. (2008) fdrtool: a versatile R package for estimating local and tail area-based false discovery rates. *Bioinformatics*, **24**, 1461–1462.
17. Yevshin,I., Sharipov,R., Valeev,T., Kel,A. and Kolpakov,F. (2017) GTRD: a database of transcription factor binding sites identified by ChIP-seq experiments. *Nucleic Acids Res.*, **45**, D61–D67.
18. Gupta,S., Stamatoyannopoulos,J.A., Bailey,T.L. and Noble,W.S. (2007) Quantifying similarity between motifs. *Genome Biol.*, **8**, R24

Published online

Nucleic Acids Research, 0000, Vol. 00, No. 00 ***1–1***
doi:10.1093/nar/gkn000

Supplementary information for: The BaMM webserver for de-novo motif discovery and regulatory sequence analysis

Anja Kiesel,[1] **Christian Roth,**[1] **Wanwan Ge,**[1] **Maximilian Wess,**[1] **Markus Meier,**[1] **and Johannes Söding** [1,*]

[1]Quantitative and Computational Biology, Max Planck Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany *To whom correspondence should be addressed. Email: soeding@mpibpc.mpg.de

SUPPLEMENTARY FIGURE

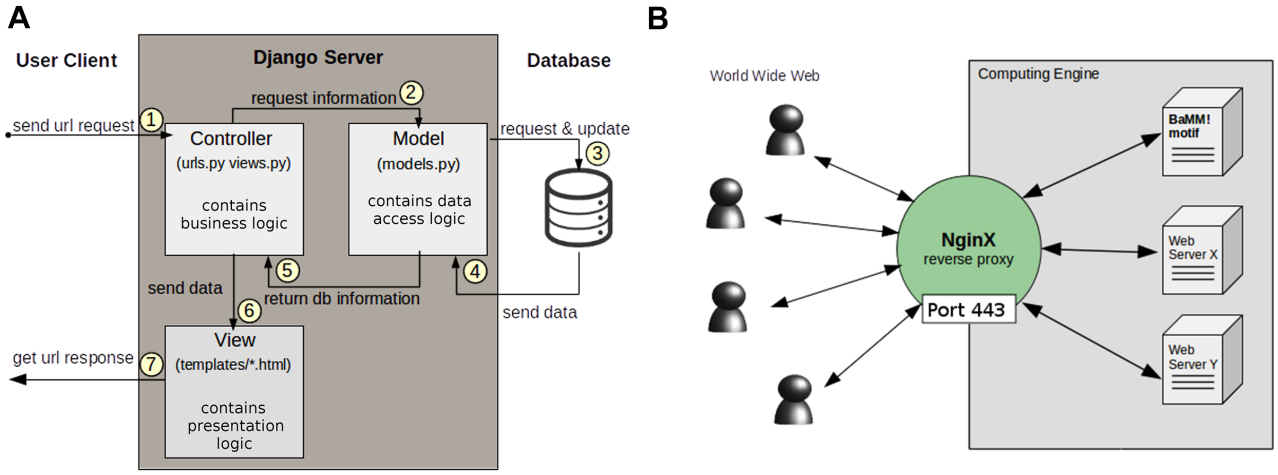


Figure 1. The BaMM server is built on the Django framework. (A) Scheme of Django working environment. (B) The Nginx server controls ports for secure data submission.

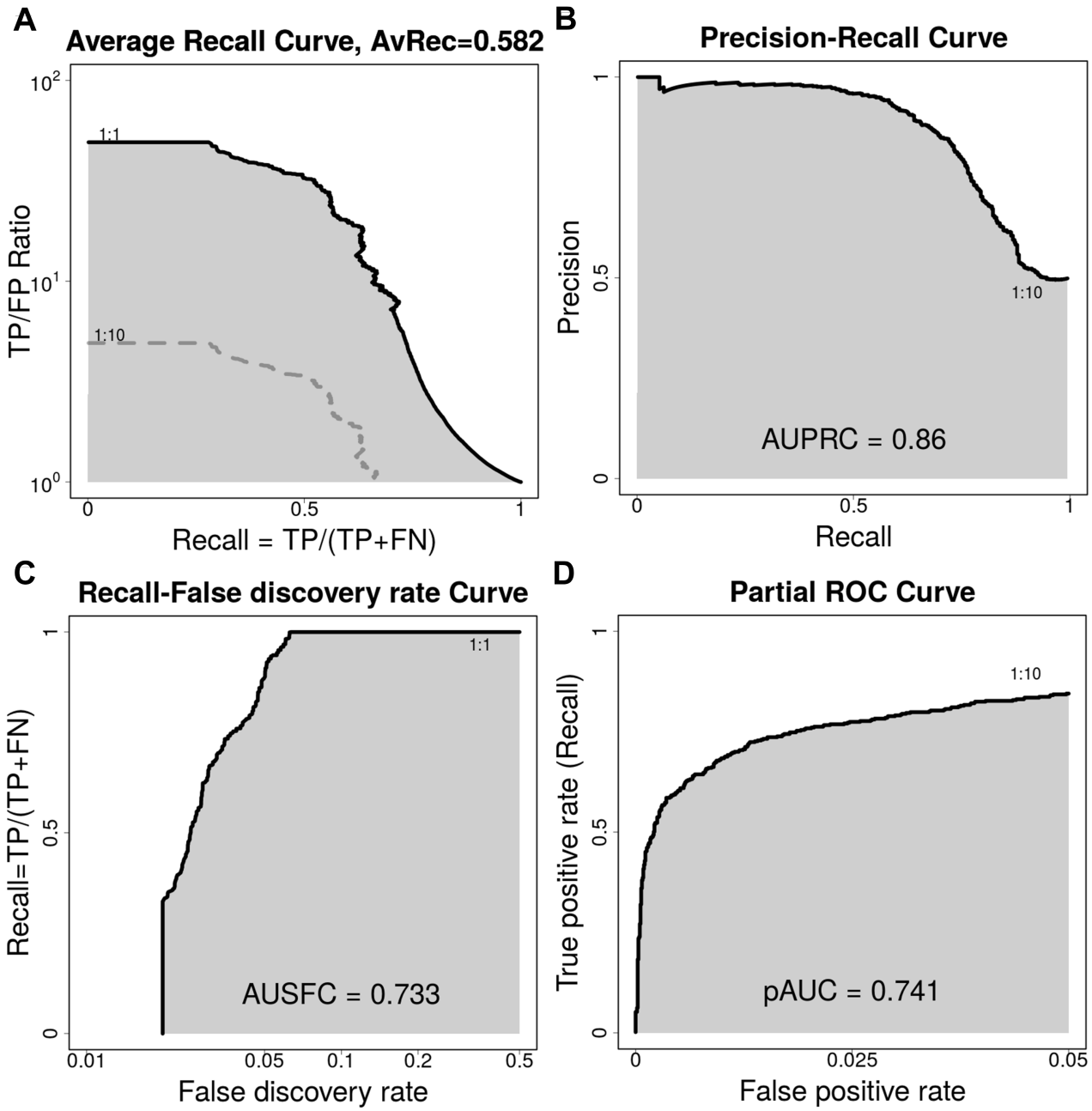


Figure 2. Comparison between different measures of motif model performance on the input dataset. (A) Recall vs. TP-to-FP ratio curve. (B) Precision-Recall curve. (C) Recall-False discovery rate curve. (D) Partial Receiver operating characteristic (ROC) curve. Definitions of axes: recall = true positive rate = TP / (TP + FN); false discovery rate (FDR) = FP / (TP + FP); precision = 1 - FDR = TP / (TP + FP); false positive rate = FP / (FP + TN), where TP = true positives, FP = false positives, FN = false negatives, TN = true negatives.

Chapter 4

Conclusion

An essential step to understanding the gene regulation mechanism is the ability to identify the regulatory factor binding sites. Thanks to the advances of high-throughput sequencing technologies and the development of machine learning tools, it enables a quantitative view to interpret these binding elements by constructing the predictive models. To accurately predict the unseen TF-bound motifs from the DNA/RNA sequences, we have previously developed BaMMmotif to learning the inter-dependencies of the nucleotides by training interpolated inhomogeneous Markov models. Given the outstanding performance of BaMMmotif over PWM-based models on *in vivo* data, I continued the work to improve the model further.

In the scheme of higher-order BaMMs (E.q. (2.17)), the higher-order model profile is built upon the K-mer frequencies and the pseudo-counts from the lower-order profile with hyper-parameter αs. The hyper-parameters αs tune how much low-order information is passed onto the higher-orders. Thus, the model avoids overfitting by automatically adapting the complexity to the available data. The α's were set to fixed values with larger numbers for higher orders, which means BaMMs learn the neighboring nucleotide dependency with decreasing relevance as the distance gets larger. However, to learn more precise models for motifs with varying spaces or non-neighboring nucleotide inter-dependency, it might require to optimize the αs from the datasets. I thus tried to sample position-specific αs for different orders using the Gibbs sampling algorithm. With the optimized αs, it improved the performance of fifth-order BaMMs by 5.3 % median increase in AvRec on 552 GTRD datasets with 500 bp-long sequences (Figure 3.2C).

Most transcription factors have particular positional preferences when recognizing the regulatory elements on the genome. For instance, many TFs show notable preferences for specific regions upstream of the transcription start sites of genes they regulate. Therefore, I introduced a prior to learn this positional preference feature from the data. The optimized positional priors helped to find the weak local motifs on the simulated dataset (Figure 3.4).

However, it did not contribute to improving the overall performance on 435 ChIP-seq datasets, compared to models with uniformly distributed positional priors, although the optimized positional priors corresponded well to the real motif distributions on the ChIP-seq datasets (Figure 3.5).

Some studies suggested that the full EM algorithm might lead to the overestimation of intra-motif dependencies, particularly when there are multiple motifs in the sequences [66]. To figure out whether BaMMs were affected, I selected the 5% best ranked possible binding sites in the first iteration of the EM algorithm for optimizing the models. Applying such a masking step restricted the noise when learning the higher-order model for a motif, and thus avoided overestimation (Section 3.1.2.3). It lowered the overall motif performance of fifth-order BaMMs by 4.4% on ENCODE data (Figure 3.7B) but helped to learn distinct motifs when more than one motif was present. Since it optimized models using fewer training data, it was also fast for the EM algorithm to converge, thus sped up the optimization process (Figure 3.7C).

I was also interested in comparing our approach with others, namely MEME [57] (most commonly used in the field), CisFinder [67] (fast and scales well for large datasets), ChIPMunk [68] (used for generating PWMs in HOCOMOCO database), and higher-order model-based tools including diChIPMunk [69] (used for generating di-PWMs in HOCOMOCO database) and InMoDe [70] (learns intra-motif dependencies). Therefore, I developed a more reliable measure score, the AvRec score, for evaluating the motif quality. It takes into account the most relevant regions for true positives and false positives in real biological applications. The benchmark tests of *de novo* motif discovery tools on large-scale data have only been done in the DeepBind paper [45] to date. Thus, I performed the benchmark tests on both *in vivo* and *in vitro* data and showed that fifth-order BaMMs achieved 13.6% median improvements in AvRec score on 427 ChIP-seq datasets and 12.2% on 164 HT-SELEX datasets in 5-fold cross-validations. Besides, I also carried out cross-cell-line and cross-platform validations for eliminating the biases that were either cell type- or experiment-specific and demonstrated the robustness of BaMMs on 237 cross-cell-line tests and 16 cross-platform tests.

Apart from learning the nucleotide inter-dependency, BaMM is also capable of detecting the weak bindings. Studies show that weak bindings are essential for TFs to rapidly respond to cellular changes to express different proteins with a sufficient amount. Hence I trained the BaMMs learned on ChIP-seq data for eight *Saccharomyces cerevisiae* TFs, and tested on the corresponding MITOMIv2 datasets. Predicted TF-DNA binding affinities with higher-order BaMMs showed better correlations to the measured binding affinities by MITOMIv2 on 7 out of 8 datasets compared to PWMs from the JASPAR database (Table 3.1).

Besides, a fifth-order BaMM learned from the GM cell line predicted more CTCF binding sites on the human genome than the current estimated number of CTCF sites (Figure 3.9). We hypothesize that the excess amount of CTCF binding sites could come from weak binding sites that help to regulate the chromatin structures and mediate the enhancer-promoter interactions under different cellular states.

With the demonstration of the robust and reliable performance of BaMMmotif2, I also helped to develop the BaMM webserver to provide the community with the software and pre-trained models for transfer learning purposes [71]. Different from the MEME suite [57], BaMM webserver provides more sophisticated BaMMs for training models on various types of data (e.g., ChIP-seq, PBM, HT-SELEX) and offers the databases with pre-trained higher-order BaMMs for searching for motif occurrences in the given sequence set, compared with the known motifs, and serving as seeds for further model optimization on the given dataset.

Outlook

Algorithm-related extensions

During the parameter tuning, I have observed that the hyperparameter q, which indicates how many fractions of sequences contain a motif (E.q. (2.30)), is crucial for learning the correct motif when multiple motifs are present in the data. In the current model, I introduced a masking step for preventing it from falling to the global optimum when initialized by a secondary motif that has a local optimum. However, the motif model may benefit from learning a dataset-specific q value.

In this work, although the optimization of positional prior does not improve the motif scores on the ChIP-seq data, the positional prior distribution is learned correctly from the real datasets. The positional prior can be used for optimizing the motif length and replacing the distribution of motif log-odds scores to illustrate the motif distribution on the sequences better.

Currently, the fast-seeding stage of BaMMmotif2 is a standalone tool ("PEnGmotif"). For better user experience, I plan to integrate it to our BaMMmotif2 toolkit, so that BaMMmotif2 can internally seed and then refine to higher-order models.

Weak binding affinities of TF and DNA

Given that weak bindings of TF-DNA are common in eukaryotic cells, especially it may help to maintain the condensates in the phase-separated hubs during the transcription regulation, it would be interesting to look more closely at the weak binding motifs. For example, in the formation of enhancer-promoter loops, binding sites with low affinities allow TF paralog-specific binding [13]. In this study, I have shown the robust performance of BaMMs compared to PWMs on yeast binding-affinity measurements. With the rapid development of techniques and more massive data, I hope our tool can be applied to more applications, thus enhancing the understanding of DNA-TF weak binding affinities. For example, SMiLE-seq [64] can be a good candidate, which is a microfluidics-based technique and measures human TF-TF-DNA bindings on a large scale with more flexible lengths.

Effects of higher-order models on different TF classes

Different mammalian TFs can recognize similar binding sites due to their common evolutionary origin. Thus, the motif prediction could benefit from the classification of TFs according to the structures of their DNA-binding domains (see Table A.3). It would be interesting to

check how much higher-order BaMMs improve the motifs classified by their TF classes and interpret which TF classes trend to bind motifs with longer distance or have more complicated binding modes.

With our tool and ideas described in this thesis, I hope it will help the community to have new and exciting discoveries.

References

[1] Jennifer E Phillips and Victor G Corces. Ctcf: master weaver of the genome. *Cell*, 137(7):1194–1211, 2009.

[2] Charles V Clevenger. Roles and regulation of stat family transcription factors in human breast cancer. *The American journal of pathology*, 165(5):1449–1460, 2004.

[3] Faizeh Al-Quobaili and Mathias Montenarh. Pancreatic duodenal homeobox factor-1and diabetes mellitus type 2. *International journal of molecular medicine*, 21(4):399–404, 2008.

[4] Miguel Angel Maestro, Carina Cardalda, Sylvia F Boj, Reini F Luco, Joan Marc Servitja, and Jorge Ferrer. Distinct roles of hnf1 b, hnf1 α, and hnf4 α in regulating pancreas development, b-cell function and growth. In *Development of the Pancreas and Neonatal Diabetes*, volume 12, pages 33–45. Karger Publishers, 2007.

[5] Mélanie Lambert, Samy Jambon, Sabine Depauw, and Marie-Hélène David-Cordonnier. Targeting transcription factors for cancer treatment. *Molecules*, 23(6):1479, 2018.

[6] Jie Wang, Jiali Zhuang, Sowmya Iyer, XinYing Lin, Troy W Whitfield, Melissa C Greven, Brian G Pierce, Xianjun Dong, Anshul Kundaje, Yong Cheng, et al. Sequence features and chromatin structure around the genomic regions bound by 119 human transcription factors. *Genome research*, 22(9):1798–1812, 2012.

[7] Otto G Berg and Peter H von Hippel. Selection of dna binding sites by regulatory proteins: Statistical-mechanical theory and application to operators and promoters. *Journal of molecular biology*, 193(4):723–743, 1987.

[8] Remo Rohs, Sean M West, Alona Sosinsky, Peng Liu, Richard S Mann, and Barry Honig. The role of dna shape in protein–dna recognition. *Nature*, 461(7268):1248–1253, 2009.

[9] Janice A Fischer, Edward Giniger, Tom Maniatis, and Mark Ptashne. Gal4 activates transcription in drosophila. *Nature*, 332(6167):853, 1988.

[10] Ann Hochschild, John Douhan III, and Mark Ptashne. How λ repressor and λ cro distinguish between or1 and or3. *Cell*, 47(5):807–816, 1986.

[11] Hamed S Najafabadi, Sanie Mnaimneh, Frank W Schmitges, Michael Garton, Kathy N Lam, Ally Yang, Mihai Albu, Matthew T Weirauch, Ernest Radovani, Philip M Kim, et al. C2h2 zinc finger proteins greatly expand the human regulatory lexicon. *Nature biotechnology*, 33(5):555, 2015.

[12] Justin Crocker, Ella Preger-Ben Noon, and David L Stern. The soft touch: Low-affinity transcription factor binding sites in development and evolution. In *Current topics in developmental biology*, volume 117, pages 455–469. Elsevier, 2016.

[13] Judith F Kribelbauer, Chaitanya Rastogi, Harmen J Bussemaker, and Richard S Mann. Low-affinity binding sites and the transcription factor specificity paradox in eukaryotes. *Annual review of cell and developmental biology*, 35:357–379, 2019.

[14] Hirofumi Kurokawa, Hozumi Motohashi, Shinji Sueno, Momoko Kimura, Hiroaki Takagawa, Yousuke Kanno, Masayuki Yamamoto, and Toshiyuki Tanaka. Structural basis of alternative dna recognition by maf transcription factors. *Molecular and cellular biology*, 29(23):6232–6244, 2009.

[15] Polly M Fordyce, David Pincus, Philipp Kimmig, Christopher S Nelson, Hana El-Samad, Peter Walter, and Joseph L DeRisi. Basic leucine zipper transcription factor hac1 binds dna in two distinct modes as revealed by microfluidic analyses. *Proceedings of the National Academy of Sciences*, 109(45):E3084–E3093, 2012.

[16] Matthew Slattery, Tianyin Zhou, Lin Yang, Ana Carolina Dantas Machado, Raluca Gordân, and Remo Rohs. Absence of a simple code: how transcription factors read the genome. *Trends in biochemical sciences*, 39(9):381–399, 2014.

[17] Ruipeng Lu, Eliseos J Mucaki, and Peter K Rogan. Discovery and validation of information theory-based transcription factor and cofactor binding site motifs. *Nucleic acids research*, 45(5):e27–e27, 2017.

[18] Fangjie Zhu, Lucas Farnung, Eevi Kaasinen, Biswajyoti Sahu, Yimeng Yin, Bei Wei, Svetlana O Dodonova, Kazuhiro R Nitta, Ekaterina Morgunova, Minna Taipale, et al. The interaction landscape between transcription factors and the nucleosome. *Nature*, 562(7725):76–81, 2018.

[19] David J Galas and Albert Schmitz. Dnaase footprinting a simple method for the detection of protein-dna binding specificity. *Nucleic acids research*, 5(9):3157–3170, 1978.

[20] Sebastian J Maerkl and Stephen R Quake. A systems approach to measuring the binding energy landscapes of transcription factors. *Science*, 315(5809):233–237, 2007.

[21] Gordon Robertson, Martin Hirst, Matthew Bainbridge, Misha Bilenky, Yongjun Zhao, Thomas Zeng, Ghia Euskirchen, Bridget Bernier, Richard Varhol, Allen Delaney, et al. Genome-wide profiles of stat1 dna association using chromatin immunoprecipitation and massively parallel sequencing. *Nature methods*, 4(8):651–657, 2007.

[22] Yue Zhao, David Granas, and Gary D Stormo. Inferring binding energies from selected binding sites. *PLoS computational biology*, 5(12), 2009.

[23] Ho Sung Rhee and B Franklin Pugh. Comprehensive genome-wide protein-dna interactions detected at single-nucleotide resolution. *Cell*, 147(6):1408–1419, 2011.

[24] Dustin E Schones, Kairong Cui, Suresh Cuddapah, Tae-Young Roh, Artem Barski, Zhibin Wang, Gang Wei, and Keji Zhao. Dynamic regulation of nucleosome positioning in the human genome. *Cell*, 132(5):887–898, 2008.

[25] Arttu Jolma, Yimeng Yin, Kazuhiro R Nitta, Kashyap Dave, Alexander Popov, Minna Taipale, Martin Enge, Teemu Kivioja, Ekaterina Morgunova, and Jussi Taipale. Dna-dependent formation of transcription factor pairs alters their binding specificity. *Nature*, 527(7578):384–388, 2015.

[26] Jason D Buenrostro, Paul G Giresi, Lisa C Zaba, Howard Y Chang, and William J Greenleaf. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, dna-binding proteins and nucleosome position. *Nature methods*, 10(12):1213, 2013.

[27] Gary D Stormo and Yue Zhao. Determining the specificity of protein–dna interactions. *Nature Reviews Genetics*, 11(11):751–760, 2010.

[28] Patrik D'haeseleer. How does dna sequence motif discovery work? *Nature biotechnology*, 24(8):959–961, 2006.

[29] Gary D Stormo, Thomas D Schneider, Larry Gold, and Andrzej Ehrenfeucht. Use of the 'perceptron'algorithm to distinguish translational initiation sites in e. coli. *Nucleic acids research*, 10(9):2997–3011, 1982.

[30] Gary D Stormo. Modeling the specificity of protein-dna interactions. *Quantitative biology*, 1(2):115–130, 2013.

[31] Dana S Fields, Yi-yuan He, Ahmed Y Al-Uzri, and Gary D Stormo. Quantitative specificity of the mnt repressor. *Journal of molecular biology*, 271(2):178–194, 1997.

[32] Martha L Bulyk, Philip LF Johnson, and George M Church. Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucleic acids research*, 30(5):1255–1261, 2002.

[33] Andrija Tomovic and Edward J Oakeley. Position dependencies in transcription factor binding sites. *Bioinformatics*, 23(8):933–941, 2007.

[34] Thomas D Schneider and R Michael Stephens. Sequence logos: a new way to display consensus sequences. *Nucleic acids research*, 18(20):6097–6100, 1990.

[35] Matthias Siebert and Johannes Söding. Bayesian markov models consistently outperform pwms at predicting motifs in nucleotide sequences. *Nucleic acids research*, 44(13):6055–6069, 2016.

[36] Ralf Eggeling, André Gohr, Jens Keilwagen, Michaela Mohr, Stefan Posch, Andrew D Smith, and Ivo Grosse. On the value of intra-motif dependencies of human insulator protein ctcf. *PLoS One*, 9(1):e85629, 2014.

[37] Jens Keilwagen and Jan Grau. Varying levels of complexity in transcription factor binding motifs. *Nucleic acids research*, 43(18):e119–e119, 2015.

[38] Lin Yang, Tianyin Zhou, Iris Dror, Anthony Mathelier, Wyeth W Wasserman, Raluca Gordân, and Remo Rohs. Tfbsshape: a motif database for dna shape features of transcription factor binding sites. *Nucleic acids research*, 42(D1):D148–D155, 2014.

[39] Tianyin Zhou, Ning Shen, Lin Yang, Namiko Abe, John Horton, Richard S Mann, Harmen J Bussemaker, Raluca Gordân, and Remo Rohs. Quantitative modeling of transcription factor binding specificities using dna shape. *Proceedings of the National Academy of Sciences*, 112(15):4654–4659, 2015.

[40] Tsu-Pei Chiu, Federico Comoglio, Tianyin Zhou, Lin Yang, Renato Paro, and Remo Rohs. Dnashaper: an r/bioconductor package for dna shape prediction and feature encoding. *Bioinformatics*, 32(8):1211–1213, 2016.

[41] Anthony Mathelier, Beibei Xin, Tsu-Pei Chiu, Lin Yang, Remo Rohs, and Wyeth W Wasserman. Dna shape features improve transcription factor binding site predictions in vivo. *Cell systems*, 3(3):278–286, 2016.

[42] Pei-Chen Peng and Saurabh Sinha. Quantitative modeling of gene expression using dna shape features of binding sites. *Nucleic acids research*, 44(13):e120–e120, 2016.

[43] Md Abul Hassan Samee, Benoit G Bruneau, and Katherine S Pollard. A de novo shape motif discovery algorithm reveals preferences of transcription factors for dna shape beyond sequence motifs. *Cell systems*, 8(1):27–42, 2019.

[44] Jian Zhou and Olga G Troyanskaya. Predicting effects of noncoding variants with deep learning–based sequence model. *Nature methods*, 12(10):931–934, 2015.

[45] Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 33(8):831, 2015.

[46] David R Kelley, Jasper Snoek, and John L Rinn. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome research*, 26(7):990–999, 2016.

[47] James Zou, Mikael Huss, Abubakar Abid, Pejman Mohammadi, Ali Torkamani, and Amalio Telenti. A primer on deep learning in genomics. *Nature genetics*, 51(1):12–18, 2019.

[48] Gökcen Eraslan, Žiga Avsec, Julien Gagneur, and Fabian J Theis. Deep learning: new computational modelling techniques for genomics. *Nature Reviews Genetics*, 20(7):389–403, 2019.

[49] Ziga Avsec, Roman Kreuzhuber, Johnny Israeli, Nancy Xu, Jun Cheng, Avanti Shrikumar, Abhimanyu Banerjee, Daniel S Kim, Lara Urban, Anshul Kundaje, et al. Kipoi: accelerating the community exchange and reuse of predictive models for genomics. *BioRxiv*, page 375345, 2018.

[50] Charles E Lawrence and Andrew A Reilly. An expectation maximization (em) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins: Structure, Function, and Bioinformatics*, 7(1):41–51, 1990.

[51] Tetsushi Yada, Yasushi Totoki, Masato Ishikawa, Kiyoshi Asai, and Kenta Nakai. Automatic extraction of motifs represented in the hidden markov model from a number of dna sequences. *Bioinformatics (Oxford, England)*, 14(4):317–325, 1998.

[52] Eric P Xing, Michael I Jordan, Richard M Karp, and Stuart J Russell. A hierarchical bayesian markovian model for motifs in biopolymer sequences. In *Advances in Neural Information Processing Systems*, pages 1513–1520, 2003.

[53] Weichun Huang, David M Umbach, Uwe Ohler, and Leping Li. Optimized mixed markov models for motif identification. *BMC bioinformatics*, 7(1):279, 2006.

[54] Steven L Salzberg, Arthur L Delcher, Simon Kasif, and Owen White. Microbial gene identification using interpolated markov models. *Nucleic acids research*, 26(2):544–548, 1998.

[55] Arthur L Delcher, Douglas Harmon, Simon Kasif, Owen White, and Steven L Salzberg. Improved microbial gene identification with glimmer. *Nucleic acids research*, 27(23):4636–4641, 1999.

[56] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[57] Timothy L Bailey, Mikael Boden, Fabian A Buske, Martin Frith, Charles E Grant, Luca Clementi, Jingyuan Ren, Wilfred W Li, and William S Noble. Meme suite: tools for motif discovery and searching. *Nucleic acids research*, 37(suppl_2):W202–W208, 2009.

[58] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[59] Thomas L Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National academy of Sciences*, 101(suppl 1):5228–5235, 2004.

[60] ENCODE Project Consortium et al. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57–74, 2012.

[61] Ivan Yevshin, Ruslan Sharipov, Semyon Kolmykov, Yury Kondrakhin, and Fedor Kolpakov. Gtrd: a database on gene transcription regulation—2019 update. *Nucleic acids research*, 47(D1):D100–D105, 2019.

[62] Polly M Fordyce, Doron Gerber, Danh Tran, Jiashun Zheng, Hao Li, Joseph L DeRisi, and Stephen R Quake. De novo identification and biophysical characterization of transcription-factor binding sites with microfluidic affinity analysis. *Nature biotechnology*, 28(9):970, 2010.

[63] Martin Tompa, Nan Li, Timothy L Bailey, George M Church, Bart De Moor, Eleazar Eskin, Alexander V Favorov, Martin C Frith, Yutao Fu, W James Kent, et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nature biotechnology*, 23(1):137–144, 2005.

[64] Alina Isakova, Romain Groux, Michael Imbeault, Pernille Rainer, Daniel Alpern, Riccardo Dainese, Giovanna Ambrosini, Didier Trono, Philipp Bucher, and Bart Deplancke. Smile-seq identifies binding motifs of single and dimeric transcription factors. *Nature methods*, 14(3):316, 2017.

[65] Ruochi Zhang, Yuchuan Wang, Yang Yang, Yang Zhang, and Jian Ma. Predicting ctcf-mediated chromatin loops using ctcf-mp. *Bioinformatics*, 34(13):i133–i141, 2018.

[66] Ralf Eggeling. Disentangling transcription factor binding site complexity. *Nucleic acids research*, 46(20):e121–e121, 2018.

[67] Alexei A Sharov and Minoru SH Ko. Exhaustive search for over-represented dna sequence motifs with cisfinder. *DNA research*, 16(5):261–273, 2009.

[68] Ivan V Kulakovskiy, VA Boeva, Alexander V Favorov, and Vsevolod J Makeev. Deep and wide digging for binding motifs in chip-seq data. *Bioinformatics*, 26(20):2622–2623, 2010.

[69] Ivan Kulakovskiy, Victor Levitsky, Dmitry Oshchepkov, Leonid Bryzgalov, Ilya Vorontsov, and Vsevolod Makeev. From binding motifs in chip-seq data to improved models of transcription factor binding sites. *Journal of bioinformatics and computational biology*, 11(01):1340004, 2013.

[70] Ralf Eggeling, Ivo Grosse, and Jan Grau. Inmode: tools for learning and visualizing intra-motif dependencies of dna binding sites. *Bioinformatics*, 33(4):580–582, 2017.

[71] Anja Kiesel, Christian Roth, Wanwan Ge, Maximilian Wess, Markus Meier, and Johannes Söding. The bamm web server for de-novo motif discovery and regulatory sequence analysis. *Nucleic acids research*, 46(W1):W215–W220, 2018.

[72] Athel Cornish-Bowden. Nomenclature for incompletely specified bases in nucleic acid sequences: recommendations 1984. *Nucleic acids research*, 13(9):3021, 1985.

[73] Edgar Wingender, Torsten Schoeps, and Jürgen Dönitz. Tfclass: an expandable hierarchical classification of human transcription factors. *Nucleic acids research*, 41(D1):D165–D170, 2013.

[74] Mark M Garner and Arnold Revzin. A gel electrophoresis method for quantifying the binding of proteins to specific dna regions: application to components of the escherichia coli lactose operon regulatory system. *Nucleic acids research*, 9(13):3047–3060, 1981.

[75] Michael Fried and Donald M Crothers. Equilibria and kinetics of lac repressor-operator interactions by polyacrylamide gel electrophoresis. *Nucleic acids research*, 9(23):6505–6525, 1981.

[76] Jerker Porath and Birgit Olin. Immobilized metal affinity adsorption and immobilized metal affinity chromatography of biomaterials. serum protein affinities for gel-immobilized iron and nickel ions. *Biochemistry*, 22(7):1621–1630, 1983.

[77] David Scott Gilmour. *Detection of DNA-protein Interactions by Protein-DNA Crosslinking*. Cornell University, Aug., 1984.

[78] Craig Tuerk and Larry Gold. Systematic evolution of ligands by exponential enrichment: Rna ligands to bacteriophage t4 dna polymerase. *science*, 249(4968):505–510, 1990.

[79] Valerio Orlando. Mapping chromosomal proteins in vivo by formaldehyde-crosslinked-chromatin immunoprecipitation. *Trends in biochemical sciences*, 25(3):99–104, 2000.

[80] Xiangdong Meng, Michael H Brodsky, and Scot A Wolfe. A bacterial one-hybrid system for determining the dna-binding specificity of transcription factors. *Nature biotechnology*, 23(8):988–994, 2005.

[81] Christopher L Warren, Natasha CS Kratochvil, Karl E Hauschild, Shane Foister, Mary L Brezinski, Peter B Dervan, George N Phillips, and Aseem Z Ansari. Defining the sequence-recognition profile of dna-binding molecules. *Proceedings of the National Academy of Sciences*, 103(4):867–872, 2006.

[82] Paul G Giresi, Jonghwan Kim, Ryan M McDaniell, Vishwanath R Iyer, and Jason D Lieb. Faire (formaldehyde-assisted isolation of regulatory elements) isolates active regulatory elements from human chromatin. *Genome research*, 17(6):877–885, 2007.

[83] Melissa J Fullwood, Mei Hui Liu, You Fu Pan, Jun Liu, Han Xu, Yusoff Bin Mohamed, Yuriy L Orlov, Stoyan Velkov, Andrea Ho, Poh Huay Mei, et al. An oestrogen-receptor-α-bound human chromatin interactome. *Nature*, 462(7269):58–64, 2009.

[84] Erez Lieberman-Aiden, Nynke L Van Berkum, Louise Williams, Maxim Imakaev, Tobias Ragoczy, Agnes Telling, Ido Amit, Bryan R Lajoie, Peter J Sabo, Michael O Dorschner, et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *science*, 326(5950):289–293, 2009.

[85] Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377, 2009.

[86] Lingyun Song and Gregory E Crawford. Dnase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harbor Protocols*, 2010(2):pdb–prot5384, 2010.

[87] Razvan Nutiu, Robin C Friedman, Shujun Luo, Irina Khrebtukova, David Silva, Robin Li, Lu Zhang, Gary P Schroth, and Christopher B Burge. Direct measurement of dna affinity landscapes on a high-throughput sequencing instrument. *Nature biotechnology*, 29(7):659, 2011.

[88] Sivakanthan Kasinathan, Guillermo A Orsi, Gabriel E Zentner, Kami Ahmad, and Steven Henikoff. High-resolution mapping of transcription factor binding sites on native chromatin. *Nature methods*, 11(2):203, 2014.

[89] Qiye He, Jeff Johnston, and Julia Zeitlinger. Chip-nexus enables improved detection of in vivo transcription factor binding footprints. *Nature biotechnology*, 33(4):395, 2015.

[90] Holger Hartmann, Eckhart W Guthöhrlein, Matthias Siebert, Sebastian Luehr, and Johannes Söding. P-value-based regulatory motif discovery using positional weight matrices. *Genome research*, 23(1):181–194, 2013.

[91] Ralf Eggeling, Ivo Grosse, and Jan Grau. Inmode: tools for learning and visualizing intra-motif dependencies of dna binding sites. *Bioinformatics*, 33(4):580–582, 2016.

[92] Albin Sandelin, Wynand Alkema, Pär Engström, Wyeth W Wasserman, and Boris Lenhard. Jaspar: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic acids research*, 32(suppl_1):D91–D94, 2004.

[93] Matthew T Weirauch, Ally Yang, Mihai Albu, Atina G Cote, Alejandro Montenegro-Montero, Philipp Drewe, Hamed S Najafabadi, Samuel A Lambert, Ishminder Mann, Kate Cook, et al. Determination and inference of eukaryotic transcription factor sequence specificity. *Cell*, 158(6):1431–1443, 2014.

[94] Ivan V Kulakovskiy, Ilya E Vorontsov, Ivan S Yevshin, Ruslan N Sharipov, Alla D Fedorova, Eugene I Rumynskiy, Yulia A Medvedeva, Arturo Magana-Mora, Vladimir B Bajic, Dmitry A Papatsenko, et al. Hocomoco: towards a complete collection of transcription factor binding models for human and mouse via large-scale chip-seq analysis. *Nucleic acids research*, 46(D1):D252–D259, 2018.

[95] Maxwell A Hume, Luis A Barrera, Stephen S Gisselbrecht, and Martha L Bulyk. Uniprobe, update 2015: new tools and content for the online database of protein-binding microarray data on protein–dna interactions. *Nucleic acids research*, 43(D1):D117–D122, 2015.

[96] Vea Matys, Ellen Fricke, Robert Geffers, Ellen Gößling, Martin Haubrock, Reinhard Hehl, Klaus Hornischer, Dagmar Karas, Alexander E Kel, Olga V Kel-Margoulis, et al. Transfac®: transcriptional regulation, from patterns to profiles. *Nucleic acids research*, 31(1):374–378, 2003.

Appendix A

Supplementary material

A.1 IUPAC letter nomenclature

Table A.1 **The IUPAC letter nomenclature.**

The IUPAC letter nomenclature represents unique alphabet to represent bases in the nucleotide sequence. Each single-letter code encodes for either one single nucleobase or, or more than one nucleobase to allow multiple bases in one particular position (adapted from [72]).

Symbol	Representation	Description	Complement
A	A	Adenine	T
C	C	Cytosine	G
G	G	Guanine	C
T	T	Thymine	A
U	U	Uracil	A
W	A or T	Weak interaction (2 hydrogen bonds)	W
S	C or G	Strong interaction (3 hydrogen bonds)	S
M	A or C	aMino	K
K	G or T	Keto	M
R	A or G	puRine	Y
Y	C or T	pYrimidine	R
B	C, G or T	not A (B comes after A)	V
D	A, G or T	not C (D comes after C)	H
H	A, C or T	not G (H comes after G)	D
V	A, C or G	not T (V comes after T and U)	B
N	A, C, G or T	any Nucleotide (not a gap)	N

A.2 Abbreviations

DNA	deoxyribonucleic acid
RNA	ribonucleic acid
IUPAC	international union of pure and applied chemistry
TAD	topologically associating domains
PIC	pre-initiation complex
TSS	transcription start site
TF	transcription factor
HT-SELEX	high-throughput systematic evolution of ligands by exponential enrichment
ChIP	chromatin immunoprecipitation
CSI	cognate site identifier
PBM	protein binding microarray
B1H	bacterial one-hybrid
EMSA	Electrophoretic mobility shift assays
FAIRE-seq	Formaldehyde-Assisted Isolation of Regulatory Elements sequencing
MITOMI	Mechanically Induced Trapping of Molecular Interactions
ChIA-PET	chromatin interaction analysis by paired-end tag sequencing
HiTS-FLIP	high-throughput sequencing-fluorescent ligand interaction profiling
ATAC-seq	Assay for Transposase-Accessible Chromatin using sequencing
ORGANIC	occupied regions of genomes from affinity-purified naturally isolated chromatin
MNase-seq	micrococcal nuclease sequencing
ChIP-nexus	ChIP with nucleotide resolution through exonuclease, unique barcode and single ligation
CAP-SELEX	consecutive affinity-purification SELEX
NCAP-SELEX	nucleosome consecutive affinity-purification SELEX
SMiLE-seq	selective microfluidics-based ligand enrichment followed by sequencing
PWM	position weight matrix
BaMM	Bayesian Markov model
IMM	interpolated Markov model
iIMM	inhomogenous interpolated Markov model
ZOOPS	zero or one occurrence per sequence
MOPS	more than one occurrence per sequence
EM	expectation maximization
CGS	collapsed Gibbs sampling
PCT	parsimonious context trees
GMLA	gapless multiple local alignment

ReLU	rectified-linear unit
CNN	convolutional neural network
ROC	Receiver operating characteristic
FDR	false-discovery-rate
AvRec	false-discovery-rate-averaged recall

A.3 Transcription factor classes

Transcription factors can be classified based on their DNA-binding domains. Human TFs can be classified into 9 superclasses (Table A.3) [73].

Table A.3 **Human transcription factor superclasses.**

The classification of human transcription factors. (Adapted from [73]*)

TF superclass	Percentage	Examples
Basic domain	11%	bZIP, bHLH, bHSH
Zinc-coordinating domain	52%	C2H2-XF
Helix-turn-helix domain	27%	Fork head factors
Other all-α-helical DNA-binding domain	3%	HMG factors
α-helices exposed by β-structures	1%	MADS box factors
Immunoglobulin fold	4%	p53, T-box
β-hairpin exposed by an α/β-scaffold	1%	SMAD/NF-1 factors
β-sheet binding to DNA	< 1%	TATA-binding factors
β-barrel DNA-binding domain	< 1%	cold-shock factors
Yet undefined DNA-binding domain	1%	leucine-rich repeats-biding factors

(* Also see URL: genexplain.com/tfclass/huTF_classification_Classes.html.)

A.4 Experiments for detecting DNA-protein binding

Table A.4 **Development of DNA-protein interaction experiments.**

1978	DNA footprinting [19]
1981	EMSA [74, 75]
1983	PBM [76]
1984	ChIP [77]
1990	SELEX [78]
2000	ChIP-ChIP [79]
2005	B1H system [80]
2006	CSI array [81]
2007	ChIP-seq [21], FAIRE-seq [82], MITOMI [20]
2008	MNase-seq [24]
2009	ChIA-PET [83], HT-SELEX [22], Hi-C [84], single-cell sequencing [85]
2010	DNase-seq [86]
2011	ChIP-exo [23], HiTS-FLIP [87]
2013	ATAC-seq [26]
2014	ORGANIC [88]
2015	ChIP-nexus [89], CAP-SELEX [25]
2017	SMiLE-seq [64]
2018	NCAP-SELEX [18]

A.5 Selected tools for motif discovery

Table A.5 **List of representative motif discovery tools.**

Tool	**Core Algorithm**	**Feature**	**Ref** (Citation by Jan. 2020)
MEME	EM	PWMs	[57] (4367)
CisFinder	word counts-based PWMs	C++, fast on large-scale data	[67] (122)
ChIPMunk	greedy optimization + bootstrapping	Java, uses coverage profiles as motif positional preferences	[68] (142)
diChIPMunk	GMLA, dinucleotide PWMs	dinucleotide motif discovery	[69] (54)
XXmotif	*p*-value-based PWMs	C++, fast	[90] (60)
DeepBind	deep neural networks	PWMs, handles millions of sequences	[45] (1121)
ShapeMF	shape structure-based, Gibbs sampling	*de novo* discovery using only structure features	[43] (9)
InMoDe	parsimonious context trees (PCTs)	learn intra-motif dependency with higher-order models	[91] (7)

A.6 Motif web servers and databases

There are many web servers and databases developed over the last 30 years for DNA/RNA motifs. This section is not intended to do the comprehensive analysis but list some commonly used collections of motifs.

Table A.6 **List of motif databases and/or web servers.**

Database	Website	Description	Ref.
MEME Suite	meme-suite.org/	motif discovery, scanning, comparison, no database yet	[57]
JASPAR	jaspar.genereg.net/	manually curated, continuous updates, various organisms, various platforms	[92]
BaMM DB	bammmotif.soedinglab.org/database/	motif discovery, scanning, comparison, database with fifth-order BaMMs, various species	[71]
GTRD	gtrd.biouml.org/	uniformly processed ChIP-seq data, mainly for human & mouse	[61]
CisBP	cisbp.ccbr.utoronto.ca/	various organisms, various platforms, incorporated motifs from other databases	[93]
HOCOMOCO	hocomoco.autosome.ru/	human & mouse TFs, curated from various sources, PWMs and di-PWMs, motif scanning	[94]
UniPROBE	thebrain.bwh.harvard.edu/uniprobe/	motifs from PBM	[95]
TRANSFAC	genexplain.com/transfac/	commercial, eukaryotic TFs, regular maintained and curated	[96]

List of figures

List of tables