Learning a Hybrid Architecture for Sequence Regression and Annotation

Yizhe Zhang and **Ricardo Henao** and **Lawrence Carin**
Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA

Jianling Zhong and **Alexander J. Hartemink**
Program in Computational Biology and Bioinformatics, Duke University, Durham, NC 27708, USA

Abstract

When learning a hidden Markov model (HMM), sequential observations can often be complemented by real-valued summary response variables generated from the path of hidden states. Such settings arise in numerous domains, including many applications in biology, like motif discovery and genome annotation. In this paper, we present a flexible framework for jointly modeling both latent sequence features and the functional mapping that relates the summary response variables to the hidden state sequence. The algorithm is compatible with a rich set of mapping functions. Results show that the availability of additional continuous response variables can simultaneously improve the annotation of the sequential observations and yield good prediction performance in both synthetic data and real-world datasets.

Introduction

The hidden Markov model (HMM) is a probabilistic formalism for reasoning about sequences of random variables, which naturally arise in temporal (e.g., in natural language) and spatial settings (e.g., in images and biological sequences). HMMs are especially valued for their efficient inference, broad applicability, and rich extensibility. In the traditional context, an HMM enables one to reason about a sequence of unobserved state variables that evolve under a Markov assumption, based only on a sequence of observations corresponding to the hidden state sequence (Bishop 2006). Here, we extend this model to incorporate additional side information. Specifically, we assume that in addition to the sequential observations, we have access to a corresponding vector of real-valued "response variables" that somehow provide summary information about the hidden state sequence (if this vector is empty, we have a traditional HMM). We are interested in the following dual task: simultaneously learning the *hidden state sequence* from the sequential observations and summary response variables (sequence annotation), as well as the *mapping function* that produces real-valued response variables from the hidden state sequence (sequence regression).

Although applications exist in many domains, this extension to a traditional HMM was originally motivated by a

large collection of biological applications. For example, promoter DNA sequences can be complemented by promoter activity to learn a model that annotates the DNA sequence, and simultaneously predicts the activity level of putative promoters. In this paper, we focus on another question: understanding protein-DNA binding preferences, which is also important in unraveling gene regulation. We are particularly interested in learning how transcription factors (TFs) exhibit sequence binding preferences (motifs), in our case modeled as position weight matrices (PWMs) (Berg and von Hippel 1987). In situations where bound genomic DNA sequences are available, HMMs have been applied to jointly learn binding locations and motifs in an unsupervised manner (Bailey and Elkan 1994; Lawrence et al. 1993). However, newer protein binding microarray (PBM) technology allows us to learn motifs from *in vitro* sequence data (Mukherjee et al. 2004). In a typical PBM, about 40,000 synthetic DNA sequences (probes) are placed on a microarray, and the binding affinity of a TF for each probe is reported in the form of a real-valued fluorescence intensity. Thus, motif inference on PBM data requires jointly modeling DNA sequences and real-valued measurements, which poses new challenges for model design.

Many algorithms have been developed for analyzing PBM data. Approaches based on K-mers (K-grams of DNA sequence) feed occurrences of each K-mer as features into more sophisticated models (Weirauch et al. 2013; Berger et al. 2006). RankMotif++ (Chen, Hughes, and Morris 2007) maximizes the likelihood for binarized preference observations. Other approaches attempt to jointly learn a probabilistic model. For example, BEELM-PBM (Zhao and Stormo 2011) estimates parameters based on a biophysical energy model. One advantage of probabilistic models over K-mer models is that binding preferences can be directly described via PWMs. DeepBind (Alipanahi et al. 2015) uses a deep convolutional neural network to improve accuracy. A systematic survey of several deterministic and probabilistic models was provided by Weirauch et al. (Weirauch et al. 2013). In a PBM scenario, the observed responses reflect the average intensity level over time and probe instances, i.e., a cluster of probes with same DNA sequence. Probabilistic frameworks such as HMMs can represent the probability distribution over all possible binding configurations, and thus may have advantages over deterministic methods.

In this paper, we develop inference methods, which we call RegHMM, that efficiently estimate both the hidden state sequence and all the various parameters of the regression-coupled HMM model. The highlights are:

- A probabilistic generative model that jointly considers sequential observations and multiple real-valued responses.

- Customized link functions, summary functions, and position-wise transition probabilities providing rich and flexible functional mapping.

- Fast and efficient inference with approximate expectation-maximization via Viterbi path integration.

- Our methodology enables prediction at least as accurate as any other approach to date, while learning a much more interpretable model.

Related Work

Many methods have considered the task of mapping sequential observations to categorical or real responses. Discriminative HMMs use Bayesian model comparison by cross-entropy for sequence classification. Previous algorithms (Collins 2002; Ben-Yishai and Burshtein 2004; Sak et al. 2014) investigated discriminative training of HMM or recurrent neural networks (RNN) using maximum mutual information (MMI). (Srivastava et al. 2007) uses a profile HMM to classify biological sequences. However, many of these methods cannot straightforwardly extend to regression problems. Kernel methods (Leslie, Eskin, and Noble 2002) have also been devised to map sequential data to categorical or real responses. However, these kernel methods usually employ predefined kernels and the learned latent features encoding the original sequences are less interpretable than those coming from HMMs. Our approach, in contrast, can be applied for both classification and regression tasks and directly learn the hidden features explaining the sequences.

Previous research has also examined the use of continuous side information to complement sequential observation when learning latent features. Input-output HMMs (Bengio and Frasconi 1996) and TRBMs (Zeiler et al. 2011) consider the task of supervised learning where the observed variables influence both latent and output variables. However, these approaches assume that input observations for each time point are known. GCHMMs (Fan et al. 2015) consider an HMM where transition probabilities are associated with side information. CRBMs (Taylor and Hinton 2009) incorporate side information into the factorization of weights in conditional RBMs. The RUPA method (Noto and Craven 2008) learns an HMM with additional responses via augmented visit aggregation. However, point estimate approximation over expected visits ignores the true distribution of visits. As a result, convergence problems may arise, especially when the distribution of visits has multiple modes. In contrast, our method uses Viterbi path integration to directly approximate the true posterior of visits. In summary, we consider the responses as both extra information for learning a latent representation of sequential observation, and a variable to predict. Considering the dual task simultaneously enforces interpretability, while maintaining predictive power.

Learning a coupled regression HMM model

Learning task

We propose the RegHMM (regression-coupled HMM), which can be understood as a combination of HMM and regression. As shown in Figure 1, suppose we have N sequence observations of length L, we denote $N \times L$ matrices: $\boldsymbol{X} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N)$ to be the sequential observations and $\boldsymbol{Z}$ to be the hidden states. An HMM is defined by parameters $\boldsymbol{\Theta} = \{\boldsymbol{E}, \boldsymbol{A}, \boldsymbol{\pi}\}$, where $\boldsymbol{E}$ is the emission matrix, $\boldsymbol{A}$ is the transition matrix and $\boldsymbol{\pi}$ is the vector of initial probabilities. Further, the hidden variables, $\boldsymbol{Z}$, of the HMM are employed as covariates for M regression tasks based on the assumption that the real-valued responses, $\boldsymbol{Y} = (\boldsymbol{Y}^{(1)}, \ldots, \boldsymbol{Y}^{(M)})$, are generated from summary statistics, $\boldsymbol{V} = (\boldsymbol{V}^{(1)}, \ldots, \boldsymbol{V}^{(M)})$, that encode the information of hidden state paths, where $\boldsymbol{Y}^{(m)} = (y_1^{(m)}, \ldots, y_N^{(m)})$ and $\boldsymbol{V}^{(m)} = (v_1^{(m)}, \ldots, v_N^{(m)})$.

As a generative process, the m-th response for i-th sequential observation, $y_i^{(m)}$, is drawn from hidden states encoded by hidden variables $\boldsymbol{Z} \in \{1 \ldots S\}^{N \times L}$, where S is the total number of discrete states. This generative process of responses can be formally represented as,

$$
\begin{aligned}
y_i^{(m)} &\sim \mathcal{N}(\mu_i^{(m)}, \sigma^{(m)}), \\
\mu_i^{(m)} &= f^{(m)}(v_i^{(m)}; \boldsymbol{\Gamma}^{(m)}), \\
v_i^{(m)} &= g^{(m)}(\boldsymbol{Z}_i),
\end{aligned}
$$

where $v_i^{(m)}$ first summarize all hidden states, $\boldsymbol{Z}_i = (z_{i1}, \ldots, z_{iL})$, via certain summary function, such as counting the visits of certain states. These hidden states summaries, $v_i^{(m)}$, are then transformed through a link function, yielding response means $\mu_i^{(m)}$. Finally, $y_i^{(m)}$ is drawn from a Gaussian distribution with mean $\mu_i^{(m)}$. The standard error $\sigma^{(m)}$, is shared across samples and learned from data. We let $f^{(m)}(\cdot)$ and $g^{(m)}(\cdot)$ to denote the link function and the summary function, respectively. The link function can take either linear form, $f^{(m)}(v) = \alpha + \beta v$ or a generalized hyperbolic tangent form (tanh), $f^{(m)}(v; \alpha, \beta, s, t) = \alpha + \frac{\beta}{1 + \exp[-s(v-t)]}$, where α and β characterize the intercept and slope of $f^{(m)}(\cdot)$, while s and t characterize its shape and center position. Further, Let $\boldsymbol{\Gamma}^{(m)}$ denote all parameters defining the link function, i.e., $\{\alpha, \beta, s, t\}$ for the m-th task.

The learning aim is to jointly estimate regression parameters, $\boldsymbol{\Gamma}^{(m)}$, describing the link functions, as well as the HMM parameters, $\boldsymbol{\Theta}$.

Analyzing PBM data

In this section, we consider an application of RegHMM to PBM data analysis. In particular, we desire to infer the emission matrix, representing the binding preference, from probe sequences $\boldsymbol{X} \in \{A, C, G, T\}^{N \times L}$ and real responses $\boldsymbol{Y} \in \mathbb{R}^N$ where each sequential observation $\boldsymbol{X_i}$, correspond to a single response. Here and throughout the rest of this paper, we only discuss the case of a single response task
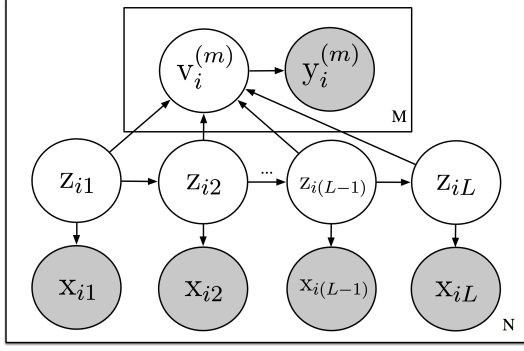
Figure 1: Graphical model representation of RegHMM.

Figure 2: (Top) PBM setup: background state can either transit to itself, or to a series of motif states $a_1 \to a_K$ or $s_1 \to s_K$. (Bottom) Example of a hidden states path.

($M = 1$). Thus, for simplicity the subscript m is omitted, further details on multiple regression tasks are provided in the supplements [1].

In this scenario, we have the background (non-binding) hidden states denoted as B, and the motif hidden states $a_1, \ldots, a_K$ and $s_1, \ldots, s_K$ indicating being bound by an assayed transcription factor, where K denotes the motif length. Motifs can be represented as sense or antisense, $s_1 \to \ldots \to s_K$ or $a_1 \to \ldots \to a_K$, respectively. Hidden variables are enforced to traverse through all K subsequential motif states, once entering a motif start state s_1 or a_1, as shown in Figure 2. Corresponding states, s_k and a_{K-k}, have complementary emissions, i.e., $p(x = A|z = s_k) = p(x = T|z = a_{K-k})$, $p(x = C|z = s_k) = p(x = G|z = a_{K-k})$. The transition probabilities from background states to motif start states, defined as $A_{B \to m} \triangleq A_{B \to a_1} + A_{B \to s_1}$, characterize the *prior* information on the frequency of binding. In a real scenario, this frequency may vary along the probe because the free end of the probe is more susceptible to being bound (Orenstein and Shamir 2014). To model this *positional bias*, we introduce a position-dependent transition, specifically,

$$P(z_{l+1} = \{s_1, a_1, B\}|z_l = B) = \boldsymbol{A}^{(l)}_{B \to \{s_1, a_1, B\}},$$

where $l = 1, \ldots, L$ indicates the position along the sequence. Note that in the PBM setting, the elements of transition matrix A other than $A_{B \to s_1}$, $A_{B \to a_1}$ and $A_{B \to B}$ are deterministic (either 0 or 1). Thus, only $A^{(l)}_{B \to s_1}$, $A^{(l)}_{B \to a_1}$ and $A^{(l)}_{B \to B}$ can vary along the sequences.

To align with the fact that signal intensities reflect the number of TFs bound, we explicitly choose the summary function, $g(\cdot)$, to count the total occurrences of a certain set of states, $\mathbb{S}$, on each sequence, i.e.,

$$v_i = g(\boldsymbol{Z_i}) = \sum_{l=1}^{L} I(z_{il} \in \mathbb{S}),$$

where $I(\cdot)$ is the indicator function and $\mathbb{S} \in 1 \ldots S$ is the collection of states of interest. Three possible settings for $\mathbb{S}$ are examined: counting the motif "entering"
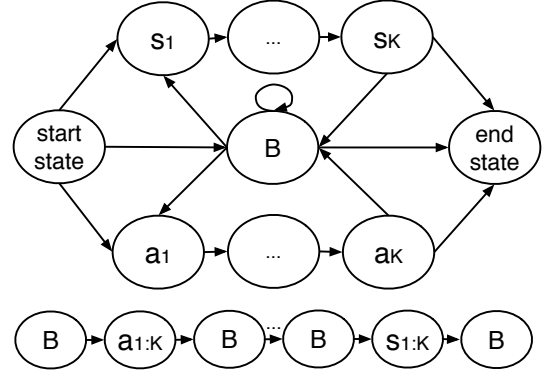
states, $\mathbb{S} = \{s_1, a_1\}$, counting the motif "exiting" states, $\mathbb{S} = \{s_K, a_K\}$ and counting the "total bound" states, $\mathbb{S} = \{s_{1,\ldots,K}, a_{1,\ldots,K}\}$.

The link function itself, $f(\cdot)$, can be either linear or hyperbolic tangent. The motivation for a non-linear link comes from the observation that probes can "saturate" with increasing binding instances, leading to a nonlinear association between number of bound TFs and signal intensity responses (Berger and Bulyk 2009). Empirically, the nonlinear hyperbolic tangent link outperforms a linear one. When a nonlinear transformation via tanh is used, the mapping from hidden space to responses can be understood as sigmoid belief network (SBN) with a single continuous output and constrained weights. Comparisons between different summary and link functions are provided in the supplements.

EM inference for coupled regression HMM model

Suppose we work on PBM data, where our goal is to use expectation maximization (EM) to maximize joint likelihood $p(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{\Theta}, \boldsymbol{\Gamma})$ with respect to HMM parameters $\boldsymbol{\Theta}$ and regression parameters $\boldsymbol{\Gamma}$. A review and comparison with the standard Baum-Welch algorithm, which considers maximizing $p(\boldsymbol{X}|\boldsymbol{\Theta})$ can be found in the supplements.

In *E-step*, we estimate the conditional distribution of hidden state sequence $\boldsymbol{Z}_i$ for i-th sequence, then we calculate the Q function, i.e., the expected value of the log complete likelihood as

$$Q(\boldsymbol{\Theta}, \boldsymbol{\Gamma}) = \mathbb{E}_{\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\Theta}^s, \boldsymbol{\Gamma}^s}[\log p(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}; \boldsymbol{\Theta}, \boldsymbol{\Gamma})] \quad (1)$$
$$= \mathbb{E}_{\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\Theta}^s, \boldsymbol{\Gamma}^s}[\log p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\Theta}) + \log p(\boldsymbol{Y}|\boldsymbol{Z}; \boldsymbol{\Gamma})]$$
$$= Q_1(\boldsymbol{\Theta}) + Q_2(\boldsymbol{\Gamma}), \quad (2)$$

where $\{\boldsymbol{\Theta}^s, \boldsymbol{\Gamma}^s\}$ denote the model parameters estimated by previous step, s. The expectation in (1) can be factorized provided that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are conditionally independent given $\boldsymbol{Z}$. In *M-step*, $Q_1(\boldsymbol{\Theta})$ and $Q_2(\boldsymbol{\Gamma})$ from (2) are then maximized.

We first examine $Q_1(\boldsymbol{\Theta})$, a key observation is that this expectation can be reformulated as the expectation with respect to $P(z_{il}|\boldsymbol{X}_i, \boldsymbol{Y}_i, \boldsymbol{\Theta}^s, \boldsymbol{\Gamma}^s)$ or

---
[1] http://people.duke.edu/~yz196/pdf/AAAI_suppl.zip

$P(z_{il}, z_{i(l+1)}|\boldsymbol{X}_i, \boldsymbol{Y}_i, \boldsymbol{\Theta}^s, \boldsymbol{\Gamma}^s)$. Thereby, HMM parameters $\boldsymbol{\Theta}$ can be updated with procedures similar to Baum-Welch. The only difference lies in the fact that in standard Baum-Welch, $P(z_{il}|\boldsymbol{X}_i, \boldsymbol{\Theta}^s)$ can be fast and accurately computed via the forward-backward algorithm, whereas in our case, the conditional distribution of hidden variables depends on both $\boldsymbol{X}$ and $\boldsymbol{Y}$, preventing us from directly applying forward-backward filtering. As we discussed in the section above, the transition probabilities can vary along the sequence. This can be easily handled via a position-wise update of transition parameters.

Denoting $\boldsymbol{X}_i$ to be the i-th sequence. For i-th input,

$$
\begin{aligned}
p(z_{il}|y_i, \Gamma^s, &\boldsymbol{X}_i, \boldsymbol{\Theta}^s) \\
&\propto p(z_{il}|\boldsymbol{X}_i, \boldsymbol{\Theta}^s)p(y_i|z_{il}, \boldsymbol{X}_i, \Gamma^s, \boldsymbol{\Theta^s}) \\
&\propto p(z_{il}|\boldsymbol{X}_i, \boldsymbol{\Theta}^s)\sum_{v_i} p(y_i|v_i, \Gamma^s)p(v_i|z_{il}, \boldsymbol{X}_i, \boldsymbol{\Theta^s}),
\end{aligned}
$$

where $p(z_{il}|\boldsymbol{X}_i, \boldsymbol{\Theta}^s)$ and $p(y_i|v_i, \Gamma^s)$ can be cheaply obtained. However, the conditional distributions over hidden summaries, i.e., $p(v_i|z_{il}, \boldsymbol{X}_i, \boldsymbol{\Theta}^s)$, are not straightforward. A naive approach would utilize a Gaussian distribution to approximate the true posterior, where mean and variances of the Gaussian distribution can be estimated analytically using propagation tricks similar to forward-backward calculations, as detailed in the supplements. Unfortunately, this approach only works when the real valued distribution $p(v_i|z_{il}, \boldsymbol{X}_i, \boldsymbol{\Theta}^s)$ is unimodal, which is generally not the case.

Instead, we appeal to a Viterbi path integration approximation. The approximated distribution is achieved by integrating over a truncated hidden state sequence space, denoted here as $\mathbb{P}_{\text{vit}}$, where the number of included Viterbi paths is dynamically determined to cover 99% of the total probability mass, thus

$$
p(v_i|z_{il}, \boldsymbol{X}_i, \boldsymbol{\Theta}^s) \propto \mathbb{E}_{\boldsymbol{Z}_i \in \mathbb{P}_{\text{vit}}}p(v_i|\boldsymbol{Z}_i)p(\boldsymbol{Z}_i|z_{il}, \boldsymbol{X}_i, \boldsymbol{\Theta}^s),
$$

where the $\boldsymbol{Z}_i$ are the Viterbi paths conditioned on $\boldsymbol{X}_i$ and z_{il}, and $p(v_i|\boldsymbol{Z}_i)$ is an indicator function $I(v_i = g(\boldsymbol{Z}_i))$, provided that summaries v_i are deterministic given a hidden state sequence Z_i. The calculation of $p(\boldsymbol{Z}_i|z_{il} = z', \boldsymbol{X}_i, \boldsymbol{\Theta}^s)$ is performed for each position l and each possible state z' that z_{il} can take. For each $\{l, z'\}$ pair, the computational complexity is $\mathcal{O}(S^2 L)$, thus a naive approach would cost $\mathcal{O}(S^3 L^2)$. To reduce computational cost we adopted a dynamic programming approach described in Algorithm 1, in which no additional cost is required. Briefly, this procedure first propagates through the sequence to calculate a forward and a backward tensor, while keeping track of the top paths, then it computes the conditional probabilities for each $\{l, z'\}$ pair altogether. In Algorithm 1, the symbol $\otimes$ represents the Kronecker product and we use D to denote the number of top paths.

In the worst case, Viterbi path integration may require a large number of top Viterbi paths, which could be computationally prohibitive. However, in our experiments on PBM data, around 100 Viterbi paths usually cover 99% probability mass for most probe sequences.

Algorithm 1 Top D Conditional Viterbi Path Integration.

1: *(For each (l, z') combination, estimate the distribution of summary variable v_i conditioned on $z_{il} = z'$)*
2: Initialize forward tensor at position 0, $F(0, 1: S, 1: D)$
3: **for** $l = 1 \text{to} L$ **do**
4:     **for** $z' = 1 \text{to} S$ **do**
5:         $F(l, z', 1: D) = \text{MaxK}(F(l - 1, 1: S, 1: D) + A(1: S, z')) + E(z', x_{il})$
6:     **end for**
7:     *(where MaxK($\cdot$) is a routine to find top D elements in a matrix via partial sorting)*
8: **end for**
9: Initialize backward tensor at pos L, $B(L, 1: S, 1: D)$
10: **for** $l = L - 1 \text{to} 1$ **do**
11:     **for** $z' = 1 \text{to} S$ **do**
12:         $B(l, z', 1: D) = \text{MaxK}(B(l + 1, 1: S, 1: D) + A(z', 1: S)\} + E(z', X_l)$
13:     **end for**
14: **end for**
15: **for** $l = 1 \text{to} L$ **do**
16:     **for** $z' = 1 \text{to} S$ **do**
17:         $C(l, z', 1: D) = \text{MaxK}(F(l, z', 1: D) \otimes B(l, z', 1: D) - E(z', x_{il}))$
18:         *(where $C(l, z', k)$ represents the probability of top kth conditional Viterbi path)*
19:         Calculate summary variable $V(l, z', 1: D)$
20:     **end for**
21: **end for**
22: **if** Any $\sum_{k=1}^{D} C(l, z', k) <$ threshold **then**
23:     Dynamically increase D, redo the forward-backward procedure
24: **else**
25:     Renormalize C
26: **end if**
27: Return C,V

So far we have considered maximizing the $Q_1(\boldsymbol{\Theta})$ with Viterbi path integration. We can rewrite $Q_2(\boldsymbol{\Gamma})$ as

$$
Q_2(\boldsymbol{\Gamma}) = \mathbb{E}_{\boldsymbol{V}|\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\Theta}^s, \Gamma^s} \log p(\boldsymbol{Y}|\boldsymbol{V}, \boldsymbol{\Gamma}, \sigma). \tag{3}
$$

Maximizing $Q_2(\boldsymbol{\Gamma})$ can be considered as a least square regression task with random variable covariates, $\boldsymbol{V}$. If the link function between $\boldsymbol{Y}$ and $\boldsymbol{V}$ is chosen to be linear, an analytic solution based on *expected summary sufficient statistics* (ESSS), $\mathbb{E}_{p(v_i|\boldsymbol{X}_i, y_i, \boldsymbol{\Theta}^s, \Gamma^s)}v_i$ and $\mathbb{E}_{p(v_i|\boldsymbol{X}_i, y_i, \boldsymbol{\Theta}^s, \Gamma^s)}v_i^2$ can be computed. For a tanh link function, coordinate gradient descent (CGD) can be utilized to update α and β analytically using ESSS. The shape and position parameters, $\{s, t\}$, can be updated using standard trust-region algorithm (Sorensen 1982). The expectation in (3) is with respect to $p(v_i|\boldsymbol{X}_i, y_i, \boldsymbol{\Theta}^s, \Gamma^s) \propto p(y_i|v_i, \boldsymbol{\Gamma}^s)p(v_i|\boldsymbol{X}_i, \boldsymbol{\Theta}^s)$, which can be calculated using same Viterbi path integration approach described before.

Full details of EM inference are provided in the supplements. MCMC approaches such as Gibbs sampling with embedded Metropolis-Hastings can be applied to this model. However, directly sampling hidden states may be negatively affected due to local modes, rendering mixing arbitrarily

slow. Inherent forward-backward calculation in EM may alleviate this problem. Backpropagation may not be appropriate for two reasons. First, in our model the hidden variables generate both $\boldsymbol{X}$ and $\boldsymbol{Y}$, where backpropagation approaches are not straightforward. Second, we wish to estimate a distribution over hidden state sequences rather than a point estimate. Variational inference methods such as mean field or variational auto-encoders (Kingma and Welling 2013; Fabius and van Amersfoort 2014) might be possible, but at the cost of accuracy and implementation complexity. Benefitting from caching and dynamic programing, we achieve time complexity of $\mathcal{O}(NS^2L)$ multiplied by the number of iterations, which is essentially the same as in Baum-Welch.

Experiments

Simulation studies

We conducted a simulation study in a PBM scenario, where we are interested in recovering the correct emission matrix and predicting for new observations. As a baseline, we considered a two-stage approach, where $\boldsymbol{\Theta}$ is first estimated with Baum-Welch algorithm, followed by a regression task to estimate $\boldsymbol{\Gamma}$. In our synthetic PBM data, 20,000 sequences (sequence length $L = 20$) and corresponding signals were generated from the generative model parameterized by $\boldsymbol{\Theta}$ and $\boldsymbol{\Gamma}$. 10,000 sequence instances were used for training, while the remaining 10,000 instances were used for evaluation. A motif of length 5 with both sense and antisense representation was used. To make the task more challenging, the transition probability from background state to motif states was set to be relatively small, so that motif states were not enriched. We examined 6 different setups for combinations of summary functions (all states, entering states, exiting states) and link functions (linear, tanh). For clarity, we only show the results of one setting (all state + linear), as the other 5 settings gave similar results. Detailed results are provided in the supplements.

Table 1 shows that our method outperformed two-stage baseline model in three aspects. First, the coupled model could recover parameters $\boldsymbol{\Theta}$ and $\boldsymbol{\Gamma}$ close to the truth, whereas the two-stage model diverged significantly. The PWM model implied by emission matrix is shown in Figure 3, RegHMM could capture the emission and transition probability well even under the case where sequential observations were less informative. Note that we were only interested in background to motif transition $A_{B \to M}^{(l)}$. Second, mean squared prediction error for held-out responses (prediction MSE) from test sequences was lower. Finally, as a generative model, the likelihood of unseens (test log-likelihood) was higher. These results indicate that our model can bridge the information between sequential observations and real responses, and leverages all the information to parse the sequences correctly. Having a good understanding about data, the algorithm can then predict reasonably well. To see how the additional real responses reinforce the learning on HMM side, intuitively, these extra responses function via both re-weighting contribution of different sequences and highlighting individual positions in each sequence according to consistency between predicted responses and observed responses. Allowing transitions to vary along the sequence yield slight improvement of performances (see supplements).

	Oracle	RegHMM	Two-stage
α	5	5.00	4.90
β	4	4.10	20.61
$\|\hat{\boldsymbol{E}} - \boldsymbol{E}\|_\infty$	0	0.19	1.16
$\|\hat{\boldsymbol{A}}_{B \to M} - \boldsymbol{A}_{B \to M}\|_\infty$	0	4.5×10^{-5}	3.6×10^{-4}
Pred. MSE	0.962	0.969	1.355
Test LL	-29.13	-29.14	-29.81

Table 1: Comparison between RegHMM and baseline (two-stage) model on synthetic data.

We also simulated a case where the observed sequences cannot be distinguished from sequences generated by a null model, i.e., where only background state transmits to itself. Remarkably, RegHMM recovered the correct motif pattern even when HMM structure was seemingly missing, whereas two-stage model failed (see supplements).

Real data analysis

We evaluated our algorithm on DREAM5, a publicly available dataset of PBM experiments for methods competition (Weirauch et al. 2013). The DREAM5 dataset consists of experimental data for 66 transcription factors under two array designs. The overall task is to use one array type ($\sim 41,000$ probe sequences) for training and predict the real signal intensity responses on the other type. Apart from the prediction task, the inferred motif patterns are also of interest.

We compared RegHMM with BEEML (Zhao and Stormo 2011), Seed and Wobble (Berger et al. 2006), RankMOTIF++ (Chen, Hughes, and Morris 2007), FeatureREDUCE (Weirauch et al. 2013) and DeepBind (Alipanahi et al. 2015). As described in Weirauch et al., all sequences were tailored to 35 bases to retain only unique sequences ($L = 35$). Also, original signal intensities were first transformed using a logarithmic function (Weirauch et al. 2013). For RegHMM, we considered both linear and tanh link functions and two summary functions: counting all motif states and count motif entering states. We also compared between models with and without position-wise transitions. The motif length K for all scenarios was set to 6. The initial value for the emission matrix was established from the most frequent 6-mer.

Following Weirauch et al., the comparison was conducted based on two metrics. The Pearson correlation coefficient between predicted responses and observed responses, to assess the predictive ability of each model for unseen queries. Apart from this, we also compared the area under the receiver operating characteristic (AUROC), as the experimental data can be subject to noise and outliers. To align with the comparison done by Weirauch et al., the positives were defined as the probes that are 4 standard deviations above the mean, with an enforced minimum of 50.

As shown in Figure 4, RegHMM achieved comparable average Pearson correlation and AUROC compared with other methods over 66 experiments. We emphasize that our
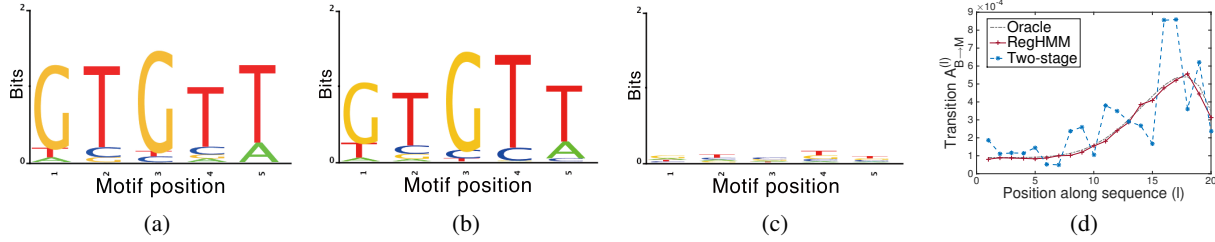
Figure 3: Inferred HMM parameters from simulated data. a): Ground truth. b-c): Motif patterns learned by RegHMM and two-stage model, respectively. d): Inferred transition probabilities along the probe, from background state to motif state $A_{B \to M}^{(l)}$, where $A_{B \to M}^{(l)} \triangleq p(z_l = \text{motif} | z_{l-1} = \text{background})$.
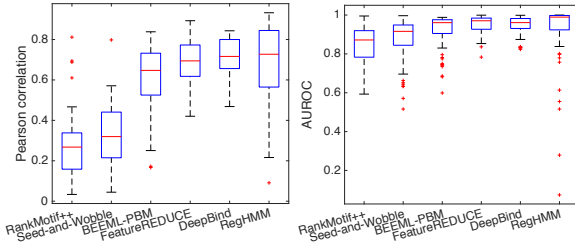

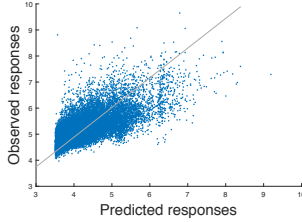Figure 4: DREAM5 probe level comparison based on correlation (left panel) and AUROC (right panel).

Figure 5: Scatter plot of predicted responses and observed response of Zscan10. Pearson correlation $\rho = 0.82$.

model can directly infer a motif pattern, whereas K-mer based models parse sequences into features that do not explicitly explain the data. Thus, our model is predicting "reasonably" by attempting to simultaneously interpret the data. Interestingly, for several transcription factors (such as TF-60, TF-66) reported to be hard to predict by other methods, our model achieved good prediction (figure 5). Unexpectedly, the link functions learned from some transcription factors are monotonically decreasing, indicating the number of motif instances is negatively correlated with signal intensity responses, which may explain why other models assuming only positive associations fail. Possible explanations would be that the motif found captures sequence bias of PBM experiments to some extent, or the assayed transcription factor inclined to avoid binding to certain motif pattern.

Another observation was that if we train two models separately for each one of the two distinct array designs, even though the inferred regression parameters showed slightly discrepancy, the learned HMM parameters corresponded

reasonably well, suggesting a good consistency between arrays. Moreover, we found the transition probability from background to motif states (which we defined as positional bias in previous sections) decreased with positional index (see supplements), which is consistent with our knowledge that free ends of the probe favor binding.

We compared different link functions and summary functions. As shown in the supplements, using a tanh link function marginally outperformed a linear one. The choice of summary function was more TF-specific. Nevertheless, for most TFs, a summary function considering all motif states led to better performance.

Discussion

Motivated by a TF motif discovery challenge, we examined the problem of learning a regression-coupled HMM. We proposed RegHMM, a probabilistic generative model learning hidden labeling of observed sequences by leveraging real-valued responses via a flexible set of link and summary functions. An approximate EM algorithm using Viterbi path integration is employed for fast and accurate inference. The experiments on simulated data showed that, by incorporating additional responses, RegHMM can accurately label the sequences and predict on future observations in situations where inference from sequential observation alone fails. The application to a real-world dataset showed that RegHMM can explicitly characterize protein-binding preferences, while maintaining predictive power. The results suggested potential biological insights for further validation.

Our algorithm was designed for complementing sequential data with real-valued summaries. However, using a softmax or Poisson link, our model could be naturally extended to handle categorical or count responses. Applications may include motif finding from *in vivo* data where multiple tracks of count responses are available. In addition, task-specific summary functions can be engineered to reflect domain knowledge. For example, indicator functions of motif orderings may associate with certain signal responses.

Acknowledgments

References

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. *Nature Biotechnology* 33(8):831–838.

Bailey, T. L., and Elkan, C. 1994. Fitting a mixture model by expectation maximization to discover motifs in bipolymers. Technical report, University of California at San Diego.

Ben-Yishai, A., and Burshtein, D. 2004. A Discriminative Training Algorithm for Hidden Markov Models. *IEEE Transactions on Speech and Audio Processing* 12(3):204–217.

Bengio, Y., and Frasconi, P. 1996. Input-output HMMs for sequence processing. *Neural Networks*.

Berg, O. G., and von Hippel, P. H. 1987. Selection of DNA binding sites by regulatory proteins. Statistical-mechanical theory and application to operators and promoters. *Journal of Molecular Biology* 193(4):723–750.

Berger, M. F., and Bulyk, M. L. 2009. Universal protein-binding microarrays for the comprehensive characterization of the DNA-binding specificities of transcription factors. *Nature Protocols* 4(3):393–411.

Berger, M. F.; Philippakis, A. A.; Qureshi, A. M.; He, F. S.; Estep, P. W.; and Bulyk, M. L. 2006. Compact, universal DNA microarrays to comprehensively determine transcription-factor binding site specificities. *Nature Biotechnology* 24(11):1429–1435.

Bishop, C. M. 2006. *Pattern Recognition and Machine Learning*. Springer Verlag.

Chen, X.; Hughes, T. R.; and Morris, Q. 2007. RankMotif++: a motif-search algorithm that accounts for relative ranks of K-mers in binding transcription factors. *Bioinformatics* 23(13):i72–9.

Collins, M. 2002. Discriminative training methods for hidden Markov models. In *the ACL-02 conference*, 1–8. Morristown, NJ, USA: Association for Computational Linguistics.

Fabius, O., and van Amersfoort, J. R. 2014. Variational Recurrent Auto-Encoders. *arXiv.org*.

Fan, K.; Eisenberg, M.; Walsh, A.; Aiello, A.; and Heller, K. A. 2015. Hierarchical Graph-Coupled HMMs for Heterogeneous Personalized Health Data. *KDD* 239–248.

Kingma, D. P., and Welling, M. 2013. Auto-Encoding Variational Bayes. *arXiv.org*.

Lawrence, C.; Altschul, S.; Boguski, M.; Liu, J.; Neuwald, A.; and Wootton, J. 1993. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science* 262(5131):208–214.

Leslie, C. S.; Eskin, E.; and Noble, W. S. 2002. The Spectrum Kernel: A String Kernel for SVM Protein Classification. *Pacific Symposium on Biocomputing* 566–575.

Mukherjee, S.; Berger, M. F.; Jona, G.; Wang, X. S.; Muzzey, D.; Snyder, M.; Young, R. A.; and Bulyk, M. L. 2004. Rapid analysis of the DNA-binding specificities of transcription factors with DNA microarrays. *Nature genetics* 36(12):1331–1339.

Noto, K., and Craven, M. 2008. Learning Hidden Markov Models for Regression using Path Aggregation. *UAI* 2008:444–451.

Orenstein, Y., and Shamir, R. 2014. A comparative analysis of transcription factor binding models learned from PBM, HT-SELEX and ChIP data. *Nucleic Acids Research* 42(8):e63–e63.

Sak, H.; Vinyals, O.; Heigold, G.; Senior, A. W.; McDermott, E.; Monga, R.; and Mao, M. Z. 2014. Sequence discriminative distributed training of long short-term memory recurrent neural networks. *INTERSPEECH* 1209–1213.

Sorensen, D. C. 1982. Newton's method with a model trust region modification. *SIAM Journal on Numerical Analysis* 19(2):409–426.

Srivastava, P. K.; Desai, D. K.; Nandi, S.; and Lynn, A. M. 2007. HMM-ModE – Improved classification using profile hidden Markov models by optimising the discrimination threshold and modifying emission probabilities with negative training sequences. *BMC Bioinformatics* 8(1):104.

Taylor, G. W., and Hinton, G. E. 2009. Factored conditional restricted Boltzmann machines for modeling motion style. In *ICML*.

Weirauch, M. T.; Cote, A.; Norel, R.; Annala, M.; Zhao, Y.; Riley, T. R.; Saez-Rodriguez, J.; Cokelaer, T.; Vedenko, A.; Talukder, S.; DREAM5 Consortium; Bussemaker, H. J.; Morris, Q. D.; Bulyk, M. L.; Stolovitzky, G.; and Hughes, T. R. 2013. Evaluation of methods for modeling transcription factor sequence specificity. *Nature Biotechnology* 31(2):126–134.

Zeiler, M. D.; Taylor, G. W.; Sigal, L.; Matthews, I.; and Fergus, R. 2011. Facial Expression Transfer with Input-Output Temporal Restricted Boltzmann Machines. In *NIPS*, 1629–1637.

Zhao, Y., and Stormo, G. D. 2011. Quantitative analysis demonstrates most transcription factors require only simple models of specificity. *Nature Biotechnology* 29(6):480–483.