# A Spectral Algorithm for Learning Hidden Markov Models

**Daniel Hsu**
UC San Diego
djhsu@cs.ucsd.edu

**Sham M. Kakade**
Toyota Technological Institute at Chicago
sham@tti-c.org

**Tong Zhang**
Rutgers University
tzhang@stat.rutgers.edu

## Abstract

Hidden Markov Models (HMMs) are one of the most fundamental and widely used statistical tools for modeling discrete time series. In general, learning HMMs from data is computationally hard (under cryptographic assumptions), and practitioners typically resort to search heuristics which suffer from the usual local optima issues. We prove that under a natural separation condition (bounds on the smallest singular value of the HMM parameters), there is an efficient and provably correct algorithm for learning HMMs. The sample complexity of the algorithm does not explicitly depend on the number of distinct (discrete) observations—it implicitly depends on this quantity through spectral properties of the underlying HMM. This makes the algorithm particularly applicable to settings with a large number of observations, such as those in natural language processing where the space of observation is sometimes the words in a language. The algorithm is also simple: it employs only a singular value decomposition and matrix multiplications.

## 1 Introduction

Hidden Markov Models (HMMs) [BE67] are the workhorse statistical model for discrete time series, with widely diverse applications including automatic speech recognition, natural language processing (NLP), and genomic sequence modeling. In this model, a discrete hidden state evolves according to some Markovian dynamics, and observations at particular time depend only on the hidden state at that time. The learning problem is to estimate the model only with observation samples from the underlying distribution. Thus far, the predominant learning algorithms have been local search heuristics, such as the Baum-Welch / EM algorithm [BPSW70, DLR77].

It is not surprising that practical algorithms have resorted to heuristics, as the general learning problem has been shown to be hard under cryptographic assumptions [Ter02]. Fortunately, the hardness results are for HMMs that seem divorced from those that we are likely to encounter in practical applications.

The situation is in many ways analogous to learning mixture distributions with samples from the underlying distribution. There, the general problem is believed to be hard. However, much recent progress has been made when certain separation assumptions are made with respect to the component mixture distributions (*e.g.* [Das99, DS07, VW02, CR08, BV08]). Roughly speaking, these separation assumptions imply that with high probability, given a point sampled from the distribution, we can recover which component distribution generated this point. In fact, there is prevalent sentiment that we are often only interested in clustering the data when such a separation condition holds. Much of the theoretical work here has been on how small this separation need be in order to permit an efficient algorithm to recover the model.

We present a simple and efficient algorithm for learning HMMs under a certain natural separation condition. We provide two results for learning. The first is that we can approximate the joint distribution over observation sequences of length $t$ (here, the quality of approximation is measured by total variation distance). As $t$ increases, the approximation quality degrades polynomially. Our second result is on approximating the *conditional* distribution over a future observation, conditioned on some history of observations. We show that this error is asymptotically bounded—*i.e.* for any $t$, conditioned on the observations prior to time $t$, the error in predicting the $t$-th outcome is controlled. Our algorithm can be thought of as 'improperly' learning an HMM in that we do not explicitly recover the transition and observation models. However, our model does maintain a hidden state representation which is closely (in fact, linearly) related to the HMM's, and can be used for interpreting the hidden state.

The separation condition we require is a spectral condition on both the observation matrix and the transition matrix. Roughly speaking, we require that the observation distributions arising from distinct hidden states be distinct (which we formalize by singular value conditions on the observation matrix). This requirement can be thought of as being weaker than the separation condition for clustering in that the observation distributions can overlap quite a bit—given one observation, we do not necessarily have the information to determine which hidden state it was generated from (unlike in the clustering literature). We also have a spectral condition on the correlation between adjacent observations. We believe both of these conditions to be quite reasonable in many practical applications. Furthermore, given our analysis, extensions to our algorithm which relax these assumptions should

be possible.

The algorithm we present has both polynomial sample and computational complexity. Computationally, the algorithm is quite simple—at its core is a singular value decomposition (SVD) of a correlation matrix between past and future observations. This SVD can be viewed as a Canonical Correlation Analysis (CCA) [Hot35] between past and future observations. The sample complexity results we present do not explicitly depend on the number of distinct observations; rather, they implicitly depend on this number through spectral properties of the HMM. This makes the algorithm particularly applicable to settings with a large number of observations, such as those in NLP where the space of observations is sometimes the words in a language.

### 1.1 Related Work

There are two ideas closely related to this work. The first comes from the subspace identification literature in control theory [Lju87, OM96, Kat05]. The second idea is that, rather than explicitly modeling the hidden states, we can represent the probabilities of sequences of observations as products of matrix observation operators, an idea which dates back to the literature on multiplicity automata [Sch61, CP71, Fli74].

The subspace identification methods, used in control theory, use spectral approaches to discover the relationship between hidden states and the observations. In this literature, the relationship is discovered for linear dynamical systems such as Kalman filters. The basic idea is that the relationship between observations and hidden states can often be discovered by spectral/SVD methods correlating the past and future observations (in particular, such methods often do a CCA between the past and future observations). However, algorithms presented in the literature cannot be directly used to learn HMMs because they assume additive noise models with noise distributions independent of the underlying states, and such models are not suitable for HMMs (an exception is [ARJ03]). In our setting, we use this idea of performing a CCA between past and future observations to uncover information about the observation process (this is done through an SVD on a correlation matrix between past and future observations). The state-independent additive noise condition is avoided through the second idea.

The second idea is that we can represent the probability of sequences as products of matrix operators, as in the literature on multiplicity automata [Sch61, CP71, Fli74] (see [EDKM05] for discussion of this relationship). This idea was re-used in both the Observable Operator Model of [Jae00] and the Predictive State Representations of [LSS01], both of which are closely related and both of which can model HMMs. In fact, the former work by [Jae00] provides a non-iterative algorithm for learning HMMs, with an asymptotic analysis. However, this algorithm assumed knowing a set of 'characteristic events', which is a rather strong assumption that effectively reveals some relationship between the hidden states and observations. In our algorithm, this problem is avoided through the first idea.

Some of the techniques in the work in [EDKM07] for tracking belief states in an HMM are used here. As discussed earlier, we provide a result showing how the model's conditional distributions over observations (conditioned on a history) do not asymptotically diverge. This result was proven in [EDKM07] when an approximate model is *already known*. Roughly speaking, the reason this error does not diverge is that the previous observations are always revealing information about the next observation; so with some appropriate contraction property, we would not expect our errors to diverge. Our work borrows from this contraction analysis.

Among recent efforts in various communities [ARJ03, VWM07, ZJ07, CC08], the only previous efficient algorithm shown to PAC-learn HMMs in a setting similar to ours is due to [MR06]. Their algorithm for HMMs is a specialization of a more general method for learning phylogenetic trees from leaf observations. While both this algorithm and ours rely on the same rank condition and compute similar statistics, they differ in two significant regards. First, [MR06] were not concerned with large observation spaces, and thus their algorithm assumes the state and observation spaces to have the same dimension. In addition, [MR06] take the more ambitious approach of learning the observation and transition matrices explicitly, which unfortunately results in a less stable and less sample-efficient algorithm that injects noise to artificially spread apart the eigenspectrum of a probability matrix. Our algorithm avoids recovering the observation and transition matrix explicitly[1], and instead uses subspace identification to learn an alternative representation.

## 2 Preliminaries

### 2.1 Hidden Markov Models

The HMM defines a probability distribution over sequences of hidden states $(h_t)$ and observations $(x_t)$. We write the set of hidden states as $[m] = \{1, \ldots, m\}$ and set of observations as $[n] = \{1, \ldots, n\}$, where $m \leq n$.

Let $T \in \mathbb{R}^{m \times m}$ be the state transition probability matrix with $T_{ij} = \Pr[h_{t+1} = i | h_t = j]$, $O \in \mathbb{R}^{n \times m}$ be the observation probability matrix with $O_{ij} = \Pr[x_t = i | h_t = j]$, and $\vec{\pi} \in \mathbb{R}^m$ be the initial state distribution with $\vec{\pi}_i = \Pr[h_1 = i]$. The conditional independence properties that an HMM satisfies are: 1) conditioned on the previous hidden state, the current hidden state is sampled independently of all other events in the history; and 2) conditioned on the current hidden state, the current observation is sampled independently from all other events in the history. These conditional independence properties of the HMM imply that $T$ and $O$ fully characterize the probability distribution of any sequence of states and observations.

A useful way of computing the probability of sequences is in terms of 'observation operators', an idea which dates back to the literature on multiplicity automata (see [Sch61, CP71, Fli74]). The following lemma is straightforward to verify (see [Jae00, EDKM07]).

**Lemma 1.** *For $x = 1, \ldots, n$, define*

$$A_x = T \operatorname{diag}(O_{x,1}, \ldots, O_{x,m}).$$

[1] In Appendix A, we discuss the key step in [MR06], and also show how to use their technique in conjunction with our algorithm to recover the HMM observation and transition matrices. Our algorithm does not rely on this extra step—we believe it to be generally unstable—but it can be taken if desired.

*For any $t$:*

$$\Pr[x_1, \ldots, x_t] \;=\; \vec{1}_m^\top A_{x_t} \ldots A_{x_1} \vec{\pi}.$$

Our algorithm learns a representation that is based on this observable operator view of HMMs.

### 2.2 Notation

As already used in Lemma 1, the vector $\vec{1}_m$ is the all-ones vector in $\mathbb{R}^m$. We denote by $x_{1:t}$ the sequence $(x_1, \ldots, x_t)$, and by $x_{t:1}$ its reverse $(x_t, \ldots, x_1)$. When we use a sequence as a subscript, we mean the product of quantities indexed by the sequence elements. So for example, the probability calculation in Lemma 1 can be written $\vec{1}_m^\top A_{x_{t:1}} \vec{\pi}$. We will use $\vec{h}_t$ to denote a probability vector (a distribution over hidden states), with the arrow distinguishing it from the random hidden state variable $h_t$. Additional notation used in the theorem statements and proofs is listed in Table 1.

### 2.3 Assumptions

We assume the HMM obeys the following condition.

**Condition 1** (HMM Rank Condition)**.** $\vec{\pi} > 0$ *element-wise, and $O$ and $T$ are rank $m$.*

The conditions on $\vec{\pi}$ and $T$ are satisfied if, say, the Markov chain specified by $T$ is ergodic and $\vec{\pi}$ is its stationary distribution. The condition on $O$ rules out the problematic case in which some state $i$ has an output distribution equal to a convex combination (mixture) of some other states' output distributions. Such a case could cause a learner to confuse state $i$ with a mixture of these other states. As mentioned before, the general task of learning HMMs (even the specific goal of simply accurately modeling the distribution probabilities [Ter02]) is hard under cryptographic assumptions; the rank condition is a natural way to exclude the malicious instances created by the hardness reduction.

The rank condition of $O$ can be relaxed through a simple modification of our algorithm that looks at multiple observation symbols simultaneously to form the probability estimation tables. For example, if two hidden states have identical observation probability in $O$ but different transition probability in $T$, then they may be differentiated by using two consecutive observations. Although our analysis can be applied in this case with minimal modifications, for clarity, we only state our results for an algorithm that estimates probability tables with rows and columns corresponding to single observations.

### 2.4 Learning Model

Our learning model is similar to those of [KMR$^+$94, MR06] for PAC-learning discrete probability distributions. We assume we can sample observation sequences from an HMM. In particular, we assume each sequence is generated starting from the same initial state distribution (*e.g.* the stationary distribution of the Markov chain specified by $T$). This setting is valid for practical applications including speech recognition, natural language processing, and DNA sequence modeling, where multiple independent sequences are available.

For simplicity, this paper only analyzes an algorithm that uses the initial few observations of each sequence, and ignores the rest. We do this to avoid using concentration bounds with complicated mixing conditions for Markov chains in our sample complexity calculation, as these conditions are not essential to the main ideas we present. In practice, however, one should use the full sequences to form the probability estimation tables required by our algorithm. In such scenarios, a single long sequence is sufficient for learning, and the effective sample size can be simply discounted by the mixing rate of the underlying Markov chain.

Our goal is to derive accurate estimators for the cumulative (joint) distribution $\Pr[x_{1:t}]$ and the conditional distribution $\Pr[x_t|x_{1:t-1}]$ for any sequence length $t$. For the conditional distribution, we obtain an approximation that does not depend on $t$, while for the joint distribution, the approximation quality degrades gracefully with $t$.

## 3 Observable Representations of Hidden Markov Models

A typical strategy for learning HMMs is to estimate the observation and transition probabilities for each hidden state (say, by maximizing the likelihood of a sample). However, since the hidden states are not directly observed by the learner, one often resorts to heuristics (*e.g.* EM) that alternate between imputing the hidden states and selecting parameters $\widehat{O}$ and $\widehat{T}$ that maximize the likelihood of the sample and current state estimates. Such heuristics can suffer from local optima issues and require careful initialization (*e.g.* an accurate guess of the hidden states) to avoid failure.

However, under Condition 1, HMMs admit an efficiently learnable parameterization that depends only on *observable quantities*. Because such quantities can be estimated from data, learning this representation avoids any guesswork about the hidden states and thus allows for algorithms with strong guarantees of success.

This parameterization is natural in the context of Observable Operator Models [Jae00], but here we emphasize its connection to subspace identification.

### 3.1 Definition

Our HMM representation is defined in terms of the following vector and matrix quantities:

$$\begin{aligned}
[P_1]_i &= \Pr[x_1 = i] \\
[P_{2,1}]_{ij} &= \Pr[x_2 = i, x_1 = j] \\
[P_{3,x,1}]_{ij} &= \Pr[x_3 = i, x_2 = x, x_1 = j] \quad \forall x \in [n],
\end{aligned}$$

where $P_1 \in \mathbb{R}^n$ is a vector, and $P_{2,1} \in \mathbb{R}^{n \times n}$ and the $P_{3,x,1} \in \mathbb{R}^{n \times n}$ are matrices. These are the marginal probabilities of observation singletons, pairs, and triples.

The representation further depends on a matrix $U \in \mathbb{R}^{n \times m}$ that obeys the following condition.

**Condition 2** (Invertibility Condition)**.** $U^\top O$ *is invertible.*

In other words, $U$ defines an $m$-dimensional subspace that preserves the state dynamics—this will become evident in the next few lemmas.

A natural choice for $U$ is given by the 'thin' SVD of $P_{2,1}$, as the next lemma exhibits.

**Lemma 2.** *Assume $\vec{\pi} > 0$ and that $O$ and $T$ have column rank $m$. Then* $\mathrm{rank}(P_{2,1}) = m$. *Moreover, if $U$ is the matrix of left singular vectors of $P_{2,1}$ corresponding to non-zero singular values, then* $\mathrm{range}(U) = \mathrm{range}(O)$, *so $U \in \mathbb{R}^{n \times m}$ obeys Condition 2.*

*Proof.* Using the conditional independence properties of the HMM, entries of the matrix $P_{2,1}$ can be factored as

$$\begin{aligned}[P_{2,1}]_{ij} &= \sum_{k=1}^{m}\sum_{\ell=1}^{m} \Pr[x_2 = i, x_1 = j, h_2 = k, h_1 = \ell] \\ &= \sum_{k=1}^{m}\sum_{\ell=1}^{m} O_{ik}\, T_{k\ell}\, \vec{\pi}_\ell\, [O^\top]_{\ell j}\end{aligned}$$

so $P_{2,1} = OT\,\mathrm{diag}(\vec{\pi})O^\top$ and thus $\mathrm{range}(P_{2,1}) \subseteq \mathrm{range}(O)$. The assumptions on $O$, $T$, and $\vec{\pi}$ imply that $T\,\mathrm{diag}(\vec{\pi})O^\top$ has linearly independent rows and that $P_{2,1}$ has $m$ non-zero singular values. Therefore

$$O = P_{2,1}(T\,\mathrm{diag}(\vec{\pi})O^\top)^+$$

(where $X^+$ denotes the Moore-Penrose pseudo-inverse of a matrix $X$), which in turn implies $\mathrm{range}(O) \subseteq \mathrm{range}(P_{2,1})$. Thus $\mathrm{rank}(P_{2,1}) = \mathrm{rank}(O) = m$, and also $\mathrm{range}(U) = \mathrm{range}(P_{2,1}) = \mathrm{range}(O)$. ∎

Our algorithm is motivated by Lemma 2 in that we compute the SVD of an empirical estimate of $P_{2,1}$ to discover a $U$ that satisfies Condition 2. We also note that this choice for $U$ can be thought of as a surrogate for the observation matrix $O$ (see Remark 5).

Now given such a matrix $U$, we can finally define the observable representation:

$$\begin{aligned}\vec{b}_1 &= U^\top P_1 \\ \vec{b}_\infty &= \left(P_{2,1}^\top U\right)^+ P_1 \\ B_x &= \left(U^\top P_{3,x,1}\right)\left(U^\top P_{2,1}\right)^+ \quad \forall x \in [n]\,.\end{aligned}$$

### 3.2 Basic Properties

The following lemma shows that the observable representation $\{\vec{b}_\infty, \vec{b}_1, B_1, \ldots, B_n\}$ is sufficient to compute the probabilities of any sequence of observations.

**Lemma 3** (Observable HMM Representation)**.** *Assume the HMM obeys Condition 1 and that $U \in \mathbb{R}^{n \times m}$ obeys Condition 2. Then:*

1. $\vec{b}_1 = (U^\top O)\vec{\pi}$.
2. $\vec{b}_\infty^\top = \vec{1}_m^\top (U^\top O)^{-1}$.
3. $B_x = (U^\top O) A_x (U^\top O)^{-1} \;\; \forall x \in [n]$.
4. $\Pr[x_{1:t}] = \vec{b}_\infty^\top B_{x_{t:1}} \vec{b}_1 \;\; \forall t \in \mathbb{N}, x_1, \ldots, x_t \in [n]$.

In addition to joint probabilities, we can compute conditional probabilities using the observable representation. We do so through (normalized) conditional 'internal states' that depend on a history of observations. We should emphasize that these states are *not* in fact probability distributions over hidden states (though the following lemma shows that they are linearly related). As per Lemma 3, the initial state is

$$\vec{b}_1 = (U^\top O)\vec{\pi}.$$

Generally, for any $t \geq 1$, given observations $x_{1:t-1}$ with $\Pr[x_{1:t-1}] > 0$, we define the internal state as:

$$\vec{b}_t = \vec{b}_t(x_{1:t-1}) = \frac{B_{x_{t-1:1}} \vec{b}_1}{\vec{b}_\infty^\top B_{x_{t-1:1}} \vec{b}_1}.$$

The case $t = 1$ is consistent with the general definition of $\vec{b}_t$ because the denominator is $\vec{b}_\infty^\top \vec{b}_1 = \vec{1}_m^\top (U^\top O)^{-1}(U^\top O)\vec{\pi} = \vec{1}_m^\top \vec{\pi} = 1$. The following result shows how these internal states can be used to compute conditional probabilities $\Pr[x_t = i | x_{1:t-1}]$.

**Lemma 4** (Conditional Internal States)**.** *Assume the conditions in Lemma 3. Then, for any time $t$:*

1. *(Recursive update of states) If $\Pr[x_{1:t}] > 0$, then*

$$\vec{b}_{t+1} = \frac{B_{x_t} \vec{b}_t}{\vec{b}_\infty^\top B_{x_t} \vec{b}_t},$$

2. *(Relation to hidden states)*

$$\vec{b}_t = (U^\top O)\, \vec{h}_t(x_{1:t-1})$$

*where $[\vec{h}_t(x_{1:t-1})]_i = \Pr[h_t = i | x_{1:t-1}]$ is the conditional probability of the hidden state at time $t$ given the observations $x_{1:t-1}$,*

3. *(Conditional observation probabilities)*

$$\Pr[x_t | x_{1:t-1}] = \vec{b}_\infty^\top B_{x_t} \vec{b}_t.$$

**Remark 5.** *If $U$ is the matrix of left singular vectors of $P_{2,1}$ corresponding to non-zero singular values, then $U$ acts much like the observation probability matrix $O$ in the following sense:*

*Let $\vec{b}_t = \vec{b}_t(x_{1:t-1})$ and $\vec{h}_t = \vec{h}_t(x_{1:t-1})$. Then*

$$\Pr[x_t = i | x_{1:t-1}] = [U\vec{b}_t]_i = [O\vec{h}_t]_i.$$

*To see this, note that $UU^\top$ is the projection operator to* $\mathrm{range}(U)$. *Since* $\mathrm{range}(U) = \mathrm{range}(O)$ *(Lemma 2), we have $UU^\top O = O$, so $U\vec{b}_t = U(U^\top O)\vec{h}_t = O\vec{h}_t$.*

### 3.3 Proofs

*Proof of Lemma 3.* The first claim is immediate from the fact $P_1 = O\vec{\pi}$. For the second claim, we write $P_1$ in the following unusual (but easily verified) form:

$$\begin{aligned}P_1^\top &= \vec{1}_m^\top T\,\mathrm{diag}(\vec{\pi})O^\top \\ &= \vec{1}_m^\top (U^\top O)^{-1}(U^\top O) T\,\mathrm{diag}(\vec{\pi})O^\top \\ &= \vec{1}_m^\top (U^\top O)^{-1} U^\top P_{2,1}.\end{aligned}$$

The matrix $U^\top P_{2,1}$ has linearly independent rows (by the assumptions on $\vec{\pi}, O, T$, and the condition on $U$), so

$$\begin{aligned}\vec{b}_\infty^\top &= P_1^\top (U^\top P_{2,1})^+ \\ &= \vec{1}_m^\top (U^\top O)^{-1} (U^\top P_{2,1}) (U^\top P_{2,1})^+ \\ &= \vec{1}_m^\top (U^\top O)^{-1}.\end{aligned}$$

To prove the third claim, we first express $P_{3,x,1}$ in terms of $A_x$:

$$\begin{aligned} P_{3,x,1} &= OA_xT\,\mathrm{diag}(\vec{\pi})O^\top \\ &= OA_x(U^\top O)^{-1}(U^\top O)T\,\mathrm{diag}(\vec{\pi})O^\top \\ &= OA_x(U^\top O)^{-1}U^\top P_{2,1}. \end{aligned}$$

Again, using the fact that $U^\top P_{2,1}$ has full row rank,

$$\begin{aligned} B_x &= \left(U^\top P_{3,x,1}\right)\left(U^\top P_{2,1}\right)^+ \\ &= (U^\top O)A_x(U^\top O)^{-1}\left(U^\top P_{2,1}\right)\left(U^\top P_{2,1}\right)^+ \\ &= (U^\top O)A_x(U^\top O)^{-1}. \end{aligned}$$

The probability calculation in the fourth claim is now readily seen as a telescoping product that reduces to the product in Lemma 1. ∎

*Proof of Lemma 4.* The first claim is a simple induction. The second and third claims are also proved by induction as follows. The base case is clear from Lemma 3 since $\vec{h}_1 = \vec{\pi}$ and $\vec{b}_1 = (U^\top O)\vec{\pi}$, and also $\vec{b}_\infty^\top B_{x_1}\vec{b}_1 = \vec{1}_m^\top A_{x_1}\vec{\pi} = \Pr[x_1]$. For the inductive step,

$$\begin{aligned} \vec{b}_{t+1} &= \frac{B_{x_t}\vec{b}_t}{\vec{b}_\infty^\top B_{x_t}\vec{b}_t} = \frac{B_{x_t}(U^\top O)\vec{h}_t}{\Pr[x_t|x_{1:t-1}]} = \frac{(U^\top O)A_{x_t}\vec{h}_t}{\Pr[x_t|x_{1:t-1}]} \\ &= (U^\top O)\,\frac{\Pr[h_{t+1}=\cdot, x_t|x_{1:t-1}]}{\Pr[x_t|x_{1:t-1}]} \\ &= (U^\top O)\,\frac{\Pr[h_{t+1}=\cdot|x_{1:t}]\Pr[x_t|x_{1:t-1}]}{\Pr[x_t|x_{1:t-1}]} \\ &= (U^\top O)\,\vec{h}_{t+1}(x_{1:t}) \end{aligned}$$

(the first three equalities follow from the first claim, the inductive hypothesis, and Lemma 3), and

$$\vec{b}_\infty^\top B_{x_{t+1}}\vec{b}_{t+1} = \vec{1}_m^\top A_{x_{t+1}}\vec{h}_{t+1} = \Pr[x_{t+1}|x_{1:t}]$$

(again, using Lemma 3). ∎

# 4 Spectral Learning of Hidden Markov Models

## 4.1 Algorithm

The representation in the previous section suggests the algorithm LEARNHMM$(m, N)$ detailed in Figure 1, which simply uses random samples to estimate the model parameters. Note that in practice, knowing $m$ is not essential because the method presented here tolerates models that are not exactly HMMs, and the parameter $m$ may be tuned using cross-validation. As we discussed earlier, the requirement for independent samples is only for the convenience of our sample complexity analysis.

The model returned by LEARNHMM$(m, N)$ can be used as follows:

- To predict the probability of a sequence:

$$\widehat{\Pr}[x_1, \ldots, x_t] = \widehat{b}_\infty^\top \widehat{B}_{x_t} \ldots \widehat{B}_{x_1}\widehat{b}_1.$$

Algorithm LEARNHMM$(m, N)$:
Inputs: $m$ - number of states, $N$ - sample size
Returns: HMM model parameterized by $\{\widehat{b}_1, \widehat{b}_\infty, \widehat{B}_x\ \forall x \in [n]\}$

1. Independently sample $N$ observation triples $(x_1, x_2, x_3)$ from the HMM to form empirical estimates $\widehat{P}_1, \widehat{P}_{2,1}, \widehat{P}_{3,x,1}\ \forall x \in [n]$ of $P_1, P_{2,1}, P_{3,x,1}$ $\forall x \in [n]$.
2. Compute the SVD of $\widehat{P}_{2,1}$, and let $\widehat{U}$ be the matrix of left singular vectors corresponding to the $m$ largest singular values.
3. Compute model parameters:
   (a) $\widehat{b}_1 = \widehat{U}^\top \widehat{P}_1$,
   (b) $\widehat{b}_\infty = (\widehat{P}_{2,1}^\top \widehat{U})^+ P_1$,
   (c) $\widehat{B}_x = \widehat{U}^\top \widehat{P}_{3,x,1}(\widehat{U}^\top \widehat{P}_{2,1})^+\ \forall x \in [n]$.

Figure 1: HMM learning algorithm.

- Given an observation $x_t$, the 'internal state' update is:

$$\widehat{b}_{t+1} = \frac{\widehat{B}_{x_t}\widehat{b}_t}{\widehat{b}_\infty^\top \widehat{B}_{x_t}\widehat{b}_t}.$$

- To predict the conditional probability of $x_t$ given $x_{1:t-1}$:

$$\widehat{\Pr}[x_t|x_{1:t-1}] = \frac{\widehat{b}_\infty^\top \widehat{B}_{x_t}\widehat{b}_t}{\sum_x \widehat{b}_\infty^\top \widehat{B}_x \widehat{b}_t}.$$

Aside from the random sampling, the running time of the learning algorithm is dominated by the SVD computation of an $n \times n$ matrix. The time required for computing joint probability calculations is $O(tm^2)$ for length $t$ sequences—same as if one used the ordinary HMM parameters ($O$ and $T$). For conditional probabilities, we require some extra work (proportional to $n$) to compute the normalization factor. However, our analysis shows that this normalization factor is always close to 1 (see Lemma 13), so it can be safely omitted in many applications.

## 4.2 Main Results

We now present our main results. The first result is a guarantee on the accuracy of our joint probability estimates for observation sequences. The second result concerns the accuracy of conditional probability estimates — a much more delicate quantity to bound due to conditioning on unlikely events. We also remark that if the probability distribution is only approximately modeled as an HMM, then our results degrade gracefully based on this approximation quality.

### 4.2.1 Joint Probability Accuracy

Let $\sigma_m(M)$ denote the $m$th largest singular value of a matrix $M$. Our sample complexity bound will depend polynomially on $1/\sigma_m(P_{2,1})$ and $1/\sigma_m(O)$.

Also, define

$$\epsilon(k) \;=\; \min\left\{\sum_{j\in S}\Pr[x_2=j] : S\subseteq[n], |S|=n-k\right\}, \tag{1}$$

and let

$$n_0(\epsilon) \;=\; \min\{k : \epsilon(k)\le \epsilon/4\}.$$

In other words, $n_0(\epsilon)$ is the minimum number of observations that account for about $1-\epsilon/4$ of the total probability mass. Clearly $n_0(\epsilon)\le n$, but it can often be much smaller in real applications. For example, in many practical applications, the frequencies of observation symbols observe a power law (called Zipf's law) of the form $f(k)\propto 1/k^s$, where $f(k)$ is the frequency of the $k$-th most frequently observed symbol. If $s>1$, then $\epsilon(k)=O(k^{1-s})$, and $n_0(\epsilon)=O(\epsilon^{1/(1-s)})$ becomes independent of the number of observations $n$. This means that for such problems, our analysis below leads to a sample complexity bound for the cumulative distribution $\Pr[x_{1:t}]$ that can be independent of $n$. This is useful in domains with large $n$ such as natural language processing.

**Theorem 6.** *There exists a constant $C>0$ such that the following holds. Pick any $0<\epsilon,\eta<1$ and $t\ge 1$. Assume the HMM obeys Condition 1, and*

$$N \ge C\cdot\frac{t^2}{\epsilon^2}\cdot\left(\frac{m\cdot\log(1/\eta)}{\sigma_m(O)^2\sigma_m(P_{2,1})^4}+\frac{m\cdot n_0(\epsilon)\cdot\log(1/\eta)}{\sigma_m(O)^2\sigma_m(P_{2,1})^2}\right).$$

*With probability at least $1-\eta$, the model returned by the algorithm* LEARNHMM$(m,N)$ *satisfies*

$$\sum_{x_1,\ldots,x_t}|\Pr[x_1,\ldots,x_t]-\widehat{\Pr}[x_1,\ldots,x_t]|\le\epsilon.$$

The main challenge in proving Theorem 6 is understanding how the estimation errors accumulate in the algorithm's probability calculation. This would have been less problematic if we had estimates of the usual HMM parameters $T$ and $O$; the fully observable representation forces us to deal with more cumbersome matrix and vector products.

### 4.2.2 Conditional Probability Accuracy

In this section, we analyze the accuracy of our conditional predictions $\widehat{\Pr}[x_t|x_1,\ldots,x_{t-1}]$. Intuitively, we might hope that these predictive distributions do not become arbitrarily bad over time (as $t\to\infty$). The reason is that while estimation errors propagate into long-term probability predictions (as evident in Theorem 6), the history of observations constantly provides feedback about the underlying hidden state, and this information is incorporated using Bayes' rule (implicitly via our internal state updates).

This intuition was confirmed by [EDKM07], who showed that if one has an approximate model of $T$ and $O$ for the HMM, then under certain conditions, the conditional prediction does not diverge. This condition is the positivity of the 'value of observation' $\gamma$, defined as

$$\gamma \;=\; \inf_{\vec{v}:\|\vec{v}\|_1=1}\|O\vec{v}\|_1.$$

Note that $\gamma\ge\sigma_m(O)/\sqrt{n}$, so it is guaranteed to be positive by Condition 1. However, $\gamma$ can be much larger than what this crude lower bound suggests.

To interpret this quantity $\gamma$, consider any two distributions over hidden states $\vec{h},\widehat{h}\in\mathbb{R}^m$. Then $\|O(\vec{h}-\widehat{h})\|_1\ge\gamma\|\vec{h}-\widehat{h}\|_1$. Regarding $\vec{h}$ as the true hidden state distribution and $\widehat{h}$ as the estimated hidden state distribution, this inequality gives a lower bound on the error of the estimated observation distributions under $O$. In other words, the observation process, on average, reveal errors in our hidden state estimation. [EDKM07] uses this as a contraction property to show how prediction errors (due to using an approximate model) do not diverge. In our setting, this is more difficult as we do not explicitly estimate $O$ nor do we explicitly maintain distributions over hidden states.

We also need the following assumption, which we discuss further following the theorem statement.

**Condition 3** (Stochasticity Condition)**.** *For all observations $x$ and all states $i$ and $j$, $[A_x]_{ij}\ge\alpha>0$.*

**Theorem 7.** *There exists a constant $C>0$ such that the following holds. Pick any $0<\epsilon,\eta<1$. Assume the HMM obeys Conditions 1 and 3, and*

$$N\ge C\cdot\left[\frac{m\cdot n_0(\epsilon)}{\epsilon^2\sigma_m(O)^2\sigma_m(P_{2,1})^2}+\frac{m}{\sigma_m(O)^2\sigma_m(P_{2,1})^4}\cdot\left(\frac{m}{\epsilon^2\alpha^2}+\frac{(\log(2/\alpha))^4}{\epsilon^4\alpha^4\gamma^4}+\frac{(\log(2/\alpha))^4}{\alpha^{10}\gamma^4}\right)\right]\cdot\log\frac{1}{\eta}.$$

*With probability at least $1-\eta$, then the model returned by* LEARNHMM$(m,N)$ *satisfies, for any time $t$,*

$$KL(\Pr[x_t|x_1,\ldots,x_{t-1}]\;||\;\widehat{\Pr}[x_t|x_1,\ldots,x_{t-1}])$$
$$=\mathbb{E}_{x_{1:t}}\left[\ln\frac{\Pr[x_t|x_{1:t-1}]}{\widehat{\Pr}[x_t|x_{1:t-1}]}\right]\le\epsilon.$$

To justify our choice of error measure, note that the problem of bounding the errors of conditional probabilities is complicated by the issue of that, over the long run, we may have to condition on a very low probability event. Thus we need to control the relative accuracy of our predictions. This makes the KL-divergence a natural choice for the error measure. Unfortunately, because our HMM conditions are more naturally interpreted in terms of spectral and normed quantities, we end up switching back and forth between KL and $L_1$ errors via Pinsker-style inequalities (as in [EDKM07]). It is not clear to us if a significantly better guarantee could be obtained with a pure $L_1$ error analysis (nor is it clear how to do such an analysis).

The analysis in [EDKM07] (which assumed that approximations to $T$ and $O$ were provided) dealt with this problem of dividing by zero (during a Bayes' rule update) by explicitly modifying the approximate model so that it *never* assigns the probability of any event to be zero (since if this event occurred, then the conditional probability is no longer defined). In our setting, Condition 3 ensures that true model never assigns the probability of any event to be zero. We can relax this condition somewhat (so that we need not quantify over all observations), though we do not discuss this here.

We should also remark that while our sample complexity bound is significantly larger than in Theorem 6, we are also bounding the more stringent KL-error measure on conditional distributions.

| | |
|---|---|
| $m, n$ | Number of states and observations |
| $n_0(\epsilon)$ | Number of significant observations |
| $O, T, A_x$ | HMM parameters |
| $P_1, P_{2,1}, P_{3,x,1}$ | Marginal probabilities |
| $\widehat{P}_1, \widehat{P}_{2,1}, \widehat{P}_{3,x,1}$ | Empirical marginal probabilities |
| $\epsilon_1, \epsilon_{2,1}, \epsilon_{3,x,1}$ | Sampling errors [Section 5.1] |
| $\widehat{U}$ | Matrix of $m$ left singular vectors of $\widehat{P}_{2,1}$ |
| $\widetilde{b}_\infty, \widetilde{B}_x, \widetilde{b}_1$ | True observable parameters using $\widehat{U}$ [Section 5.1] |
| $\widehat{b}_\infty, \widehat{B}_x, \widehat{b}_1$ | Estimated observable parameters using $\widehat{U}$ |
| $\delta_\infty, \Delta_x, \delta_1$ | Parameter errors [Section 5.1] |
| $\Delta$ | $\sum_x \Delta_x$ [Section 5.1] |
| $\sigma_m(M)$ | $m$-th largest singular value of matrix $M$ |
| $\vec{b}_t, \widehat{b}_t$ | True and estimated states [Section 5.3] |
| $\vec{h}_t,\ \widehat{h}_t,\ \widehat{g}_t$ | $(\widehat{U}^\top O)^{-1}\vec{b}_t,\ (\widehat{U}^\top O)^{-1}\widehat{b}_t,\ \widehat{h}_t/(\vec{1}_m^\top \widehat{h}_t)$ [Section 5.3] |
| $\widehat{A}_x$ | $(\widehat{U}^\top O)^{-1}\widehat{B}_x(\widehat{U}^\top O)$ [Section 5.3] |
| $\gamma,\ \alpha$ | $\inf\{\|Ov\|_1 : \|v\|_1 = 1\},\ \min\{[A_x]_{i,j}\}$ |

Table 1: Summary of notation.

#### 4.2.3 Learning Distributions $\epsilon$-close to HMMs

Our $L_1$ error guarantee for predicting joint probabilities still holds if the sample used to estimate $\widehat{P}_1, \widehat{P}_{2,1}, \widehat{P}_{3,x,1}$ come from a probability distribution $\Pr[\cdot]$ that is merely close to an HMM. Specifically, all we need is that there exists some $t_{\text{max}} \geq 3$ and some $m$ state HMM with distribution $\Pr^{\text{HMM}}[\cdot]$ such that:

1. $\Pr^{\text{HMM}}$ satisfies Condition 1 (HMM Rank Condition),
2. $\forall t \leq t_{\text{max}}, \sum_{x_{1:t}} |\Pr[x_{1:t}] - \Pr^{\text{HMM}}[x_{1:t}]| \leq \epsilon^{\text{HMM}}(t)$,
3. $\epsilon^{\text{HMM}}(2) \ll \frac{1}{2}\sigma_m(P_{2,1}^{\text{HMM}})$.

The resulting error of our learned model $\widehat{\Pr}$ is

$$\begin{aligned}&\sum_{x_{1:t}} |\Pr[x_{1:t}] - \widehat{\Pr}[x_{1:t}]| \\ &\quad\leq\ \epsilon^{\text{HMM}}(t) + \sum_{x_{1:t}} |\Pr^{\text{HMM}}[x_{1:t}] - \widehat{\Pr}[x_{1:t}]|\end{aligned}$$

for all $t \leq t_{\text{max}}$. The second term is now bounded as in Theorem 6, with spectral parameters corresponding to $\Pr^{\text{HMM}}$.

## 5 Proof ideas

We outline the main ideas for proving Theorems 6 and 7. Full proofs can be found in a technical report available from arXiv (`http://arxiv.org/abs/0811.4413`).

Throughout this section, we assume the HMM obeys Condition 1. Table 1 summarizes the notation that will be used throughout the analysis in this section.

### 5.1 Estimation Errors

Define the following sampling error quantities:

$$\begin{aligned}\epsilon_1 &= \|\widehat{P}_1 - P_1\|_2 \\ \epsilon_{2,1} &= \|\widehat{P}_{2,1} - P_{2,1}\|_2 \\ \epsilon_{3,x,1} &= \|\widehat{P}_{3,x,1} - P_{3,x,1}\|_2\end{aligned}$$

The following lemma bounds these errors with high probability as a function of the number of observation samples used to form the estimates.

**Lemma 8.** *If the algorithm independently samples $N$ observation triples from the HMM, then with probability at least $1 - \eta$:*

$$\begin{aligned}\epsilon_1 &\leq \sqrt{\frac{1}{N}\ln\frac{3}{\eta}} + \sqrt{\frac{1}{N}} \\ \epsilon_{2,1} &\leq \sqrt{\frac{1}{N}\ln\frac{3}{\eta}} + \sqrt{\frac{1}{N}} \\ \sum_x \epsilon_{3,x,1} &\leq \min_k \left(\sqrt{\frac{k}{N}\ln\frac{3}{\eta}} + \sqrt{\frac{k}{N}} + 2\epsilon(k)\right) \\ &\quad + \sqrt{\frac{1}{N}\ln\frac{3}{\eta}} + \sqrt{\frac{1}{N}}\end{aligned}$$

*where $\epsilon(k)$ is defined in* (1).

The rest of the analysis estimates how the sampling errors affect the accuracies of the model parameters (which in turn affect the prediction quality).

Let $U \in \mathbb{R}^{n\times m}$ be matrix of left singular vectors of $P_{2,1}$. The first lemma implies that if $\widehat{P}_{2,1}$ is sufficiently close to $P_{2,1}$, *i.e.* $\epsilon_{2,1}$ is small enough, then the difference between projecting to $\text{range}(\widehat{U})$ and to $\text{range}(U)$ is small. In particular, $\widehat{U}^\top O$ will be invertible and be nearly as well-conditioned as $U^\top O$.

**Lemma 9.** *Suppose $\epsilon_{2,1} \leq \varepsilon \cdot \sigma_m(P_{2,1})$ for some $\varepsilon < 1/2$. Let $\varepsilon_0 = \epsilon_{2,1}^2/((1-\varepsilon)\sigma_m(P_{2,1}))^2$. Then:*

1. $\varepsilon_0 < 1$,
2. $\sigma_m(\widehat{U}^\top \widehat{P}_{2,1}) \geq (1-\varepsilon)\sigma_m(P_{2,1})$,
3. $\sigma_m(\widehat{U}^\top P_{2,1}) \geq \sqrt{1-\varepsilon_0}\sigma_m(P_{2,1})$,
4. $\sigma_m(\widehat{U}^\top O) \geq \sqrt{1-\varepsilon_0}\sigma_m(O)$.

Now we will argue that the estimated parameters $\widehat{b}_\infty, \widehat{B}_x, \widehat{b}_1$ are close to the following true parameters from the observable representation when $\widehat{U}$ is used for $U$:

$$\begin{aligned}\widetilde{b}_\infty &= (P_{2,1}^\top \widehat{U})^+ P_1 \ =\ (\widehat{U}^\top O)^{-\top}\vec{1}_m, \\ \widetilde{B}_x &= (\widehat{U}^\top P_{3,x,1})(\widehat{U}^\top P_{2,1})^+ \\ &= (\widehat{U}^\top O)A_x(\widehat{U}^\top O)^{-1} \quad \forall x \in [n] \\ \widetilde{b}_1 &= \widehat{U}^\top P_1.\end{aligned}$$

By Lemma 3, as long as $\widehat{U}^\top O$ is invertible, these parameters $\widetilde{b}_\infty, \widetilde{B}_x, \widetilde{b}_1$ constitute a valid observable representation for the HMM.

Define the following errors of the estimated parameters:

$$\begin{aligned}
\delta_\infty &= \left\|(\widehat{U}^\top O)^\top(\widehat{b}_\infty - \widetilde{b}_\infty)\right\|_\infty \\
&= \left\|(\widehat{U}^\top O)^\top \widehat{b}_\infty - \vec{1}_m\right\|_\infty, \\
\Delta_x &= \left\|(\widehat{U}^\top O)^{-1}\left(\widehat{B}_x - \widetilde{B}_x\right)(\widehat{U}^\top O)\right\|_1 \\
&= \left\|(\widehat{U}^\top O)^{-1}\widehat{B}_x(\widehat{U}^\top O) - A_x\right\|_1, \\
\Delta &= \sum_x \Delta_x \\
\delta_1 &= \left\|(\widehat{U}^\top O)^{-1}(\widehat{b}_1 - \widetilde{b}_1)\right\|_1 = \left\|(\widehat{U}^\top O)^{-1}\widehat{b}_1 - \vec{\pi}\right\|_1.
\end{aligned}$$

We can relate these to the sampling errors as follows.

**Lemma 10.** *Assume $\epsilon_{2,1} \le \sigma_m(P_{2,1})/3$. Then:*

$$\begin{aligned}
\delta_\infty &\le 4 \cdot \left(\frac{\epsilon_{2,1}}{\sigma_m(P_{2,1})^2} + \frac{\epsilon_1}{3\sigma_m(P_{2,1})}\right), \\
\Delta_x &\le \frac{8}{\sqrt{3}} \cdot \frac{\sqrt{m}}{\sigma_m(O)} \cdot \\
&\quad \left(\Pr[x_2 = x] \cdot \frac{\epsilon_{2,1}}{\sigma_m(P_{2,1})^2} + \frac{\epsilon_{3,x,1}}{3\sigma_m(P_{2,1})}\right), \\
\Delta &\le \frac{8}{\sqrt{3}} \cdot \frac{\sqrt{m}}{\sigma_m(O)} \cdot \left(\frac{\epsilon_{2,1}}{\sigma_m(P_{2,1})^2} + \frac{\sum_x \epsilon_{3,x,1}}{3\sigma_m(P_{2,1})}\right), \\
\delta_1 &\le \frac{2}{\sqrt{3}} \cdot \frac{\sqrt{m}}{\sigma_m(O)} \cdot \epsilon_1.
\end{aligned}$$

### 5.2 Proof of Theorem 6

We need to quantify how estimation errors propagate in the probability calculation. Because the joint probability of a length $t$ sequence is computed by multiplying together $t$ matrices, there is a danger of magnifying the estimation errors exponentially. Fortunately, this is not the case: the following lemma (readily proved by induction) shows that these errors accumulate roughly additively.

**Lemma 11.** *Assume $\widehat{U}^\top O$ is invertible. For any time $t$:*

$$\begin{aligned}
&\sum_{x_{1:t}} \left\|(\widehat{U}^\top O)^{-1}\left(\widehat{B}_{x_{t:1}}\widehat{b}_1 - \widetilde{B}_{x_{t:1}}\widetilde{b}_1\right)\right\|_1 \\
&\quad \le (1+\Delta)^t\delta_1 + (1+\Delta)^t - 1.
\end{aligned}$$

All that remains is to bound the effect of errors in $\widehat{b}_\infty$. Theorem 6 will follow from the following lemma combined with the sampling error bounds of Lemma 8.

**Lemma 12.** *Assume $\epsilon_{2,1} \le \sigma_m(P_{2,1})/3$. Then for any $t$,*

$$\begin{aligned}
&\sum_{x_{1:t}} \left|\Pr[x_{1:t}] - \widehat{\Pr}[x_{1:t}]\right| \\
&\quad \le \delta_\infty + (1+\delta_\infty)\left((1+\Delta)^t\delta_1 + (1+\Delta)^t - 1\right).
\end{aligned}$$

### 5.3 Proof of Theorem 7

In this subsection, we assume the HMM obeys Condition 3 (in addition to Condition 1).

We introduce the following notation. Let the unnormalized estimated conditional hidden state distributions be

$$\widehat{h}_t = (\widehat{U}^\top O)^{-1}\widehat{b}_t,$$

and its normalized version,

$$\widehat{g}_t = \widehat{h}_t/(\vec{1}_m^\top \widehat{h}_t).$$

Also, let

$$\widehat{A}_x = (\widehat{U}^\top O)^{-1}\widehat{B}_x(\widehat{U}^\top O).$$

This notation lets us succinctly compare the updates made by our estimated model to the updates of the true model. Our algorithm never explicitly computes these hidden state distributions $\widehat{g}_t$ (as it would require knowledge of the unobserved $O$). However, under certain conditions (namely Conditions 1 and 3 and some estimation accuracy requirements), these distributions are well-defined and thus we use them for sake of analysis.

The following lemma shows that if the estimated parameters are accurate, then the state updates behave much like the true hidden state updates.

**Lemma 13.** *For any probability vector $\vec{w} \in \mathbb{R}^m$ and any observation $x$,*

$$\left|\sum_x \widehat{b}_\infty^\top(\widehat{U}^\top O)\widehat{A}_x\vec{w} - 1\right| \le \delta_\infty + \delta_\infty\Delta + \Delta \quad \textit{and}$$

$$\frac{[\widehat{A}_x\vec{w}]_i}{\widehat{b}_\infty^\top(\widehat{U}^\top O)\widehat{A}_x\vec{w}} \ge \frac{[A_x\vec{w}]_i - \Delta_x}{\vec{1}_m^\top A_x\vec{w} + \delta_\infty + \delta_\infty\Delta_x + \Delta_x}$$

*for all $i = 1, \ldots, m$. Moreover, for any non-zero vector $\vec{w} \in \mathbb{R}^m$,*

$$\frac{\vec{1}_m^\top \widehat{A}_x\vec{w}}{\widehat{b}_\infty^\top(\widehat{U}^\top O)\widehat{A}_x\vec{w}} \le \frac{1}{1-\delta_\infty}.$$

A consequence of Lemma 13 is that if the estimated parameters are sufficiently accurate, then the state updates never allow predictions of very small hidden state probabilities.

**Corollary 14.** *Assume $\delta_\infty \le 1/2$, $\max_x \Delta_x \le \alpha/3$, $\delta_1 \le \alpha/8$, and $\max_x \delta_\infty + \delta_\infty\Delta_x + \Delta_x \le 1/3$. Then $[\widehat{g}_t]_i \ge \alpha/2$ for all $t$ and $i$.*

Lemma 13 and Corollary 14 can now be used to prove the contraction property of the KL-divergence between the true hidden states and the estimated hidden states. The analysis shares ideas from [EDKM07], though the added difficulty is due to the fact that the state maintained by our algorithm is not a probability distribution.

**Lemma 15.** *Let $\varepsilon_0 = \max_x 2\Delta_x/\alpha + (\delta_\infty + \delta_\infty\Delta_x + \Delta_x)/\alpha + 2\delta_\infty$. Assume $\delta_\infty \le 1/2$, $\max_x \Delta_x \le \alpha/3$, and $\max_x \delta_\infty + \delta_\infty\Delta_x + \Delta_x \le 1/3$. For all $t$, if $\widehat{g}_t \in \mathbb{R}^m$ is a probability vector, then*

$$KL(\vec{h}_{t+1}||\widehat{g}_{t+1}) \le KL(\vec{h}_t||\widehat{g}_t) - \frac{\gamma^2}{2\left(\ln\frac{2}{\alpha}\right)^2}KL(\vec{h}_t||\widehat{g}_t)^2 + \varepsilon_0.$$

Finally, the recurrence from Lemma 15 easily gives the following lemma.

**Lemma 16.** *Let* $\varepsilon_0 = \max_x 2\Delta_x/\alpha + (\delta_\infty + \delta_\infty\Delta_x + \Delta_x)/\alpha + 2\delta_\infty$ *and* $\varepsilon_1 = \max_x(\delta_\infty + \sqrt{m}\delta_\infty\Delta_x + \sqrt{m}\Delta_x)/\alpha$. *Assume* $\delta_\infty \leq 1/2$, $\max_x \Delta_x \leq \alpha/3$, *and* $\max_x \delta_\infty + \delta_\infty\Delta_x + \Delta_x \leq 1/3$. *Also assume*

$$\delta_1 \leq \sqrt{\frac{\varepsilon_0}{8\gamma^2}} \leq \frac{\alpha}{8} \leq \frac{1}{2}, \quad \varepsilon_0 \leq \frac{\alpha^4\gamma^2}{128\left(\ln\frac{2}{\alpha}\right)^2}, \text{ and } \quad \varepsilon_1 < \frac{1}{2}.$$

*Then for all* $t$,

$$KL(\vec{h}_t||\widehat{g}_t) \leq \sqrt{\frac{2\left(\ln\frac{2}{\alpha}\right)^2\varepsilon_0}{\gamma^2}} \quad \text{and}$$

$$KL(\Pr[x_t|x_{1:t-1}] \,||\, \widehat{\Pr}[x_t|x_{1:t-1}]) \leq \sqrt{\frac{2\left(\ln\frac{2}{\alpha}\right)^2\varepsilon_0}{\gamma^2}} + \delta_\infty + \delta_\infty\Delta + \Delta + 2\varepsilon_1.$$

Theorem 7 follows by combining the previous lemma and the sampling error bounds of Lemma 8.

**Acknowledgments**

The authors would like to thank John Langford and Ruslan Salakhutdinov for earlier discussions on using bottleneck methods to learn nonlinear dynamic systems. The linearization of the bottleneck idea was the basis of this paper. We also thank Yishay Mansour for pointing out hardness results for learning HMMs. This work was completed while the first author was an intern at TTI-C in 2008.

## References

[ARJ03] S. Andersson, T. Ryden, and R. Johansson. Linear optimal prediction and innovations representations of hidden markov models. *Stochastic Processes and their Applications*, 108:131–149, 2003.

[BE67] Leonard E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bull. Amer. Math. Soc.*, 73(3):360–363, 1967.

[BPSW70] Leonard E. Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[BV08] S. Charles Brubaker and Santosh Vempala. Isotropic PCA and affine-invariant clustering. In *FOCS*, 2008.

[CC08] G. Cybenko and V. Crespi. Learning hidden markov models using non-negative matrix factorization. Technical report, 2008. arXiv:0809.4086.

[CP71] J.W Carlyle and A. Paz. Realization by stochastic finite automaton. *J. Comput. Syst. Sci.*, 5:26–40, 1971.

[CR08] Kamalika Chaudhuri and Satish Rao. Learning mixtures of product distributions using correlations and independence. In *COLT*, 2008.

[Das99] Sanjoy Dasgupta. Learning mixutres of Gaussians. In *FOCS*, 1999.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[DS07] Sanjoy Dasgupta and Leonard Schulman. A probabilistic analysis of EM for mixtures of separated, spherical Gaussians. *JMLR*, 8(Feb):203–226, 2007.

[EDKM05] Eyal Even-Dar, Sham M. Kakade, and Yishay Mansour. Planning in POMDPs using multiplicity automata. In *UAI*, 2005.

[EDKM07] Eyal Even-Dar, Sham M. Kakade, and Yishay Mansour. The value of observation for monitoring dynamic systems. In *IJCAI*, 2007.

[Fli74] M. Fliess. Matrices deHankel. *J. Math. Pures Appl.*, 53:197–222, 1974.

[Hot35] H. Hotelling. The most predictable criterion. *Journal of Educational Psychology*, 1935.

[Jae00] Herbert Jaeger. Observable operator models for discrete stochastic time series. *Neural Comput.*, 12(6), 2000.

[Kat05] T. Katayama. *Subspace Methods for System Identification*. Springer, 2005.

[KMR+94] M. Kearns, Y. Mansour, D. Ron, R. Rubinfeld, R. Schapire, and L. Sellie. On the learnability of discrete distributions. In *STOC*, pages 273–282, 1994.

[Lju87] L. Ljung. *System Identification: Theory for the User*. NJ: Prentice-Hall Englewood Cliffs, 1987.

[LSS01] Michael Littman, Richard Sutton, and Satinder Singh. Predictive representations of state. In *Advances in Neural Information Processing Systems 14 (NIPS)*, pages 1555–1561, 2001.

[MR06] E. Mossel and S. Roch. Learning nonsingular phylogenies and hidden Markov models. *Annals of Applied Probability*, 16(2):583–614, 2006.

[OM96] P. V. Overschee and B. De Moor. *Subspace Identification of Linear Systems*. Kluwer Academic Publishers, 1996.

[Sch61] M.P. Schützenberger. On the definition of a family of automata. *Inf. Control*, 4:245–270, 1961.

[Ter02] Sebastiaan Terwijn. On the learnability of hidden Markov models. In *International Colloquium on Grammatical Inference*, 2002.

[VW02] Santosh Vempala and Grant Wang. A spectral algorithm for learning mixtures of distributions. In *FOCS*, 2002.

[VWM07] B. Vanluyten, J. Willems, and B. De Moor. A new approach for the identification of hidden markov models. In *Conference on Decision and Control*, 2007.

[ZJ07] MingJie Zhao and Herbert Jaeger. The error controlling algorithm for learning OOMs. Technical Report 6, International University Bremen, 2007.

# A Recovering the Observation and Transition Matrices

We sketch how to use the technique of [MR06] to recover the observation and transition matrices explicitly. This is an extra step that can be used in conjunction with our algorithm.

Define the $n \times n$ matrix $[P_{3,1}]_{i,j} = \Pr[x_3 = i, x_1 = j]$. Let $O_x = \operatorname{diag}(O_{x,1}, \ldots, O_{x,m})$, so $A_x = TO_x$. Since $P_{3,x,1} = OA_xT \operatorname{diag}(\vec{\pi})O^\top$, we have $P_{3,1} = \sum_x P_{3,x,1} = OTT \operatorname{diag}(\vec{\pi})O^\top$. Therefore

$$\begin{aligned} U^\top P_{3,x,1} &= U^\top OTO_xT \operatorname{diag}(\vec{\pi})O^\top \\ &= (U^\top OT)O_x(U^\top OT)^{-1} \\ &\quad (U^\top OT)T \operatorname{diag}(\vec{\pi})O^\top \\ &= (U^\top OT)O_x(U^\top OT)^{-1}(U^\top P_{3,1}). \end{aligned}$$

The matrix $U^\top P_{3,1}$ has full row rank, so it follows that

$$(U^\top P_{3,1})(U^\top P_{3,1})^+ = I,$$

and thus

$$(U^\top P_{3,x,1})(U^\top P_{3,1})^+ \;=\; (U^\top OT)\, O_x\, (U^\top OT)^{-1}.$$

Since $O_x$ is diagonal, the eigenvalues of $(U^\top P_{3,x,1})(U^\top P_{3,1})^+$ are exactly the observation probabilities $O_{r,1}, \ldots, O_{r,m}$.

Define i.i.d. random variables $g_x \sim N(0,1)$ for each $x$. It is shown in [MR06] that the eigenvalues of

$$\begin{aligned} &\sum_x g_x (U^\top P_{3,x,1})(U^\top P_{3,1})^+ \\ &\quad = \; (U^\top OT) \left( \sum_x g_x O_x \right) (U^\top OT)^{-1}. \end{aligned}$$

will be separated with high probability (though the separation is roughly on the same order as the failure probability; this is the main source of instability with this method). Therefore an eigen-decomposition will recover the columns of $(U^\top OT)$ up to a diagonal scaling matrix $S$, *i.e.* $U^\top OTS$. Then for each $x$, we can diagonalize $(U^\top P_{3,x,1})(U^\top P_{3,1})^+$:

$$(U^\top OTS)^{-1} \, (U^\top P_{3,x,1})(U^\top P_{3,1})^+ \, (U^\top OTS) \;=\; O_x.$$

Now we can form $O$ from the diagonals of $O_x$. Since $O$ has full column rank, $O^+O = I_m$, so it is now easy to also recover $\vec{\pi}$ and $T$ from $P_1$ and $P_{2,1}$:

$$O^+ P_1 \;=\; O^+ O\vec{\pi} \;=\; \vec{\pi}$$

and

$$\begin{aligned} &O^+ P_{2,1} (O^+)^\top \operatorname{diag}(\vec{\pi})^{-1} \\ &\quad = \; O^+ (OT \operatorname{diag}(\vec{\pi}) O^\top)(O^+)^\top \operatorname{diag}(\vec{\pi})^{-1} \\ &\quad = \; T. \end{aligned}$$

Note that because [MR06] do not allow more observations than states, they do not need to work in a lower dimensional subspace such as $\operatorname{range}(U)$. Thus, they perform an eigen-decomposition of the matrix

$$\sum_x g_x P_{3,x,1} P_{3,1}^{-1} = (OT) \left( \sum_x g_x O_x \right) (OT)^{-1},$$

and then use the eigenvectors to form the matrix $OT$. Thus they rely on the stability of the eigenvectors, which depends heavily on the spacing of the eigenvalues.