On learning parametric-output HMMs

Aryeh Kontorovich KARYEH@CS.BGU.AC.IL
Department of Computer Science, Ben-Gurion University, Beer Sheva 84105, Israel.

Boaz Nadler BOAZ.NADLER@WEIZMANN.AC.IL
Department of Computer Science and Applied Math, Weizmann Institute of Science, Rehovot, 76100, Israel.

Roi Weiss ROIWEI@CS.BGU.AC.IL
Department of Computer Science, Ben-Gurion University, Beer Sheva 84105, Israel.

Abstract

We present a novel approach to learning an HMM whose outputs are distributed according to a parametric family. This is done by *decoupling* the learning task into two steps: first estimating the output parameters, and then estimating the hidden state transition probabilities. The first step is accomplished by fitting a mixture model to the output stationary distribution. Given the parameters of this mixture model, the second step is formulated as the solution of an easily solvable convex quadratic program. We provide an error analysis for the estimated transition probabilities and show they are robust to small perturbations in the estimates of the mixture parameters. Finally, we support our analysis with some encouraging empirical results.

1. Introduction

Hidden Markov Models (HMM) are a standard tool in the modeling and analysis of time series with a wide variety of applications. When the number of hidden states is known, the standard method for estimating the HMM parameters from observed data is the Baum-Welch algorithm (Baum et al., 1970). The latter is known to suffer from two serious drawbacks: it tends to converge (i) very slowly and (ii) only to a local maximum. Indeed, the problem of recovering the parameters of a general HMM is provably hard, in several distinct senses (Abe & Warmuth, 1992; Lyngsø & Pedersen, 2001; Terwijn, 2002).

In this paper we consider learning parametric-output HMMs with a finite and known number of hidden states, where the output from each hidden state follows a parametric distribution from a given family. A notable example is a Gaussian HMM, where from each state x, the output is a (possibly multivariate) Gaussian, $\mathcal{N}(\mu_x, \Sigma_x)$, typically with unknown μ_x, Σ_x.

Main results. We propose a novel approach to learning parametric output HMMs, based on the following two insights: (i) in an ergodic HMM, the stationary output is a mixture of distributions from the parametric family, and (ii) given the output parameters, or their approximate values, one can efficiently recover the corresponding transition probabilities up to small additive errors.

Combining these two insights leads to our proposed *decoupling* approach to learning parametric HMMs. Rather than attempting, as in the Baum-Welch algorithm, to jointly estimate both the transition probabilities and the output density parameters, we instead learn each of them separately. First, given one or several long observed sequences, the HMM output parameters are estimated by a general purpose parametric mixture learner, such as the Expectation-Maximization (EM) algorithm. Next, once these parameters are approximately known, we learn the hidden state transition probabilities by solving a computationally efficient convex quadratic program (QP).

The key idea behind our approach is to treat the underlying hidden process as if it were sampled independently from the Markov chain's stationary distribution, and operate only on the empirical distribution of singletons and consecutive pairs. Thus we avoid computing the exact likelihood, which depends on the full sequence, and obtain considerable gains in computational efficiency. Under standard assumptions on

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28. Copyright 2013 by the author(s).

the Markov chain and on its output probabilities, we prove in Theorem 1 that given the exact output probabilities, our estimator for the hidden state transition matrix is asymptotically consistent. Additionally, this estimator is robust to small perturbations in the output probabilities (Theorems 2-6).

Beyond its practical prospects, our proposed approach also sheds light on the theoretical difficulty of the full HMM learning problem: It shows that for parametric-output HMMs the *key difficulty is fitting a mixture model*, since once its parameters have been accurately estimated, learning the transition matrix can be cast as a convex program. While learning a general mixture is considered a hard problem, recently much progress has been made under various separation conditions on the mixture components, see e.g. Moitra & Valiant (2010); Belkin & Sinha (2010) and references therein.

Related work. The problem of estimating HMM parameters from observations has been actively studied since the 1970's, see Cappé et al. (2005); Rabiner (1990); Roweis & Ghahramani (1999). While computing the maximum-likelihood estimator for an HMM is in general computationally intractable, under mild conditions, such an estimator is asymptotically consistent and normally distributed, see Bickel et al. (1998); Chang (1996); Douc & Matias (2001).

In recent years, there has been a renewed interest in learning HMMs, in particular under assumptions that render the learning problem tractable (Faragó & Lugosi, 1989; Hsu et al., 2009; Mossel & Roch, 2006; Siddiqi et al., 2010; Song et al., 2010; Bailly, 2011; Anandkumar et al., 2012; Balle et al., 2012).

Additionally, Lakshminarayanan & Raich (2010); Cybenko & Crespi (2011) recently suggested Non-negative Matrix Factorization (NNMF) approaches for learning HMMs. These are related to our approach, since with known output probabilities, NNMF reduces to a convex program similar to the one considered here. Hence, our stability and consistency analysis may be relevant to NNMF-based approaches as well.

Paper outline. In Section 2 we present our problem setup. The HMM learning algorithm appears in Section 3, and its statistical analysis in Section 4. Section 5 contains some simulation results. Technical details are given in the full version (Kontorovich et al., 2013).

2. Problem Setup

Notation. When $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ take values in a discrete set we abbreviate $P(x)$ for $\Pr(X = x)$ and $P(y \,|\, x)$ for $\Pr(Y = y \,|\, X = x)$. When $Y \in \mathcal{Y}$ is continuous-valued, we denote by $P(y \,|\, x)$ the probability density function of Y given X.

For $x, w \in \mathbb{R}^n$, $\operatorname{diag}(x)$ is the $n \times n$ diagonal matrix with entries x_i on its diagonal, x/w is the vector with entries x_i/w_i, and $\|x\|_w^2 = \sum_i w_i x_i^2$ is a w-weighted ℓ_2 norm (for $w_i > 0$). The shorthand $x \lesssim y$ means $x \leq (1+o(1))y$. Similarly, $x \lesssim_P y$ means $x \leq (1+o_P(1))y$. Finally, for $n \in \mathbb{N}$, we write $[n] = \{1, 2, \ldots, n\}$.

Hidden Markov Model. We consider a discrete-time, discrete-space HMMs with n hidden states. The HMM output alphabet, denoted $\mathcal{Y}$, may be either discrete or continuous. A parametric-output HMM is characterized by a tuple $(A, \mathcal{F}_n^\theta, P_0)$ where A is an $n \times n$ column stochastic matrix, P_0 is the distribution of the initial state and $\mathcal{F}_n^\theta = (f_{\theta_1}, \ldots, f_{\theta_n})$ is an ordered tuple of *parametrized* probability density functions. In the sequel we sometimes write f_i instead of f_{θ_i}.

To generate the output sequence of the HMM, first an unobserved Markov sequence of hidden states $x = (x_t)_{t=0}^{T-1}$ is generated with the following distribution.

$$P(x) \quad = \quad P_0(x_0) \prod_{t=1}^{T-1} A_{x_t, x_{t-1}},$$

where $A_{ij} = P(X_t = i \,|\, X_{t-1} = j)$ are the transition probabilities. Then, each hidden state X_t independently emits an observation $Y_t \in \mathcal{Y}$ according to the distribution $P(y_t \,|\, x_t) \equiv f_{x_t}(y_t)$. Hence the output sequence $y = (y_t)_{t=0}^{T-1}$ has the conditional probability

$$P(y \,|\, x) = \prod_{t=0}^{T-1} P(y_t \,|\, x_t) = \prod_{t=0}^{T-1} f_{x_t}(y_t).$$

The HMM Learning Problem. Given one or several HMM output sequences $(Y_t)_{t=0}^{T-1}$, the HMM learning problem is to estimate both the transition matrix A and the parameters of the output distributions $\mathcal{F}_n^\theta$.

3. Learning Parametric-Output HMMs

The standard approach to learning the parameters of an HMM is to maximize the likelihood

$$\sum_{x \in [n]^T} P_0(x_0) P(y_0 \,|\, x_0) \prod_{t=1}^{T-1} A_{x_t, x_{t-1}} P(y_t \,|\, x_t).$$

As discussed in the Introduction, this problem is in general computationally hard. In practice, neglecting the small effect of the initial distribution $P_0(x_0)$ on the likelihood, A and $\mathcal{F}_n^\theta$ are usually estimated via the Baum-Welch algorithm, which is computationally slow and only guaranteed to converge to a local maximum.

3.1. A Decoupling Approach

In what follows we show that when the output distributions are parametric, we can *decouple* the HMM learning task into two steps: first learning the output parameters $\theta_1, \ldots, \theta_n$, followed by learning the transition probabilities of the HMM. Under some structural assumptions on the HMM, this decoupling implies that the difficulty of learning a parametric-output HMM can be reduced to that of learning a parametric mixture model. Indeed, given (an approximation to) $\mathcal{F}_n^\theta$'s parameters, we propose an efficient, single-pass, statistically-consistent algorithm for estimating the transition matrix A.

As an example, consider learning a Gaussian HMM with univariate outputs. While the Baum-Welch approach jointly estimates $n^2 + 2n$ parameters (the matrix A and the parameters μ_i, σ_i^2), our decoupling approach first fits a mixture model with only $3n$ parameters $(\pi_i, \mu_i, \sigma_i^2)$, and then solves a convex problem for the remaining n^2 entries of the matrix A. While both problems are in general computationally hard, ours has a significantly lower dimensionality for large n.

Assumptions. To recover the matrix A and the output parameters θ_j we make the following assumptions:

(1a) The Markov chain has a unique stationary distribution $\boldsymbol{\pi}$ over the n hidden states. Moreover, each hidden state is recurrent with a frequency bounded away from zero: $\min_k \pi_k \geq a_0$ for some constant $a_0 > 0$.

(1b) The $n \times n$ transition matrix A is geometrically ergodic[1]: there exists parameters $G < \infty$ and $\psi \in [0, 1)$ such that from any initial distribution P_0

$$\left\| A^t P_0 - \boldsymbol{\pi} \right\|_1 \leq 2G\psi^t, \qquad \forall t \in \mathbb{N}. \tag{1}$$

(1c) The output parameters of the n states are all distinct: $\theta_i \neq \theta_j$ for $i \neq j$ and the parametric family is identifiable. In addition we assume "observability" which will be cast as a full rank condition on specially defined matrices.

Remarks: Assumption (1a) rules out transient states, whose presence makes it generally impossible to estimate all entries in A from one or a few long observed sequences. Assumption (1b) implies mixing and is used later on to bound the error and the number of samples needed to learn the matrix A. Assumption (1c) is crucial to our approach, which uses the distribution of only single and pairs of consecutive observations. If two states i, j had same output parameters, it would be impossible to distinguish between them based on single outputs.

3.2. Learning the output parameters.

Assumptions (1a,1b) imply that the Markov chain over the hidden states is mixing, and so after only a few time steps, the distribution of X_t is nearly stationary. Assuming for simplicity that already X_0 is sampled from the stationary distribution, or alternatively neglecting the first few outputs, this implies that each observable Y_t is a random realization from the following *parametric mixture model*,

$$Y \sim \sum_{i=1}^{n} \pi_i f_{\theta_i}(y). \tag{2}$$

Hence, given an output sequence $(Y_t)_{t=0}^{T-1}$ the output parameters θ_i and the stationary distribution π_i can be estimated by fitting the mixture model (2) to the observations, typically via the EM algorithm.

Like its more sophisticated cousin Baum-Welch, the mixture-learning EM algorithm also suffers from local maxima. Indeed, from a theoretical viewpoint, learning such a mixture model (i.e. the parameters of $\mathcal{F}_n^\theta$) is a non-trivial task considered in general to be computationally hard. Nonetheless, under various separation assumptions, efficient algorithms with rigorous guarantees have been recently proposed (see e.g. Belkin & Sinha (2010)).[2] Note that while these algorithms have polynomial complexity in sample size and output dimension, they are still exponential in the number of mixture components (i.e., in the number of hidden states of the HMM). Hence, these methods do not imply polynomial learnability of parametric-output HMMs.

In what follows we assume that using some mixture-learning procedure, the output parameters θ_j have been estimated with a relatively small error (say $|\hat{\theta}_j - \theta_j| = O(1/\sqrt{T})$). Furthermore, to allow for cases where θ_j were estimated from separate observed sequences of perhaps other HMMs with same output parameters but potentially different stationary distributions, we do not assume that π_i have been estimated.

3.3. Learning the transition matrix A

Next, we describe how to recover the matrix A given either exact or approximate knowledge of the HMM output probabilities. For clarity and completeness, we

[1] Any finite-state ergodic Markov chain is geometrically ergodic.

[2] Note that the techniques for learning mixtures assume iid data. However, if these are algorithmically stable — as such methods typically are — the iid assumption can be replaced by strong mixing (Mohri & Rostamizadeh, 2010).

first give an estimation procedure for the stationary distribution π.

Discrete observations. As a warm-up to the case of continuous outputs, we first consider HMMs with a discrete observation space of size $|\mathcal{Y}| = m$. In this case we can replace $\mathcal{F}_n^\theta$ by an $m \times n$ column-stochastic matrix B where $B_{ki} \equiv P(k \,|\, i)$ is the probability of observing an output k given that the Markov chain hidden state is i. In what follows, we assume that the size of the output set is larger or equal to the number of hidden states, $m \geq n$, and that the $m \times n$ matrix B has full rank n. The latter is the discrete analogue of assumption (1c) mentioned above.

First note that since the matrix A has a stationary distribution $\boldsymbol{\pi}$, the process Y_t also has a stationary distribution $\boldsymbol{\rho}$, which by analogy to Eq. (2), is

$$\boldsymbol{\rho} = B\boldsymbol{\pi}. \tag{3}$$

Similarly, the pair (Y_t, Y_{t+1}) has a unique stationary distribution $\boldsymbol{\sigma}$, given by

$$\sigma_{k,k'} \;=\; \sum_{\ell,\ell' \in [n]} \pi_\ell A_{\ell',\ell} B_{k,\ell} B_{k',\ell'}. \tag{4}$$

As we shall see below, with a known matrix B, knowledge of $\boldsymbol{\rho}$ and $\boldsymbol{\sigma}$ suffices to estimate $\boldsymbol{\pi}$ and A. Although $\boldsymbol{\rho}$ and $\boldsymbol{\sigma}$ are themselves unknown, they are easily estimated from a single pass on the data $(y_t)_{t=0}^{T-1}$:

$$\begin{aligned} \hat{\rho}_k &= \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{1}_{\{y_t = k\}}, \\ \hat{\sigma}_{k,k'} &= \frac{1}{T-1}\sum_{t=1}^{T-1} \mathbb{1}_{\{y_{t-1}=k\}} \mathbb{1}_{\{y_t = k'\}}. \end{aligned} \tag{5}$$

Estimating the stationary distribution $\boldsymbol{\pi}$. The key idea in our approach is to replace the exact, but complicated and non-convex likelihood function by a "pseudo-likelihood", which treats the hidden state sequence (X_t) as if they were iid draws from the unknown stationary distribution $\boldsymbol{\pi}$. The pseudo-likelihood has the advantage of having an easily computed global maximum, which, as we show in in Section 4, yields an asymptotically consistent estimator. Approximating the (X_t) as iid draws from $\boldsymbol{\pi}$ means that the (Y_t) are treated as iid draws from $\boldsymbol{\rho} = B\boldsymbol{\pi}$. Thus, given a sequence $(Y_t)_{t=0}^{T-1}$ the pseudo-likelihood for a vector $\boldsymbol{\pi}$ is

$$\mathcal{L}(y_0, \ldots, y_{T-1} \,|\, \boldsymbol{\pi}) = \prod_{i=0}^{T-1} (B\boldsymbol{\pi})_{y_i} = \prod_{k=1}^{m} (B\boldsymbol{\pi})_k^{n_k}$$

where $n_k = \sum_{i=0}^{T-1} \mathbb{1}_{\{y_t = k\}} = T\hat{\rho}_k$. Its maximizer is

$$\hat{\boldsymbol{\pi}}^{\mathrm{ML}} = \operatorname*{arg\,min}_{x_i \geq 0,\, \|x\|_1 = 1} \; -\sum_{k=1}^{m} \hat{\rho}_k \log(Bx)_k. \tag{6}$$

Since $-\log(x)$ is convex, and both $(Bx)_k$ and the constraints are linear in the unknown variables x_j, (6) is a *convex* program, easily solved via standard optimization methods (Nesterov & Nemirovskii, 1994).

However, to facilitate the analysis and to increase the computational efficiency, we consider the asymptotic behavior of the pseudo-likelihood in (6), for T sufficiently large so that $\hat{\boldsymbol{\rho}}$ is close to $\boldsymbol{\rho}$. First, we write

$$(Bx)_k = \hat{\rho}_k \left(1 + \frac{(Bx)_k - \hat{\rho}_k}{\hat{\rho}_k}\right).$$

Next, assuming that $T \gg 1$ is sufficiently large to ensure $|(Bx)_k - \hat{\rho}_k| \ll \hat{\rho}_k$, we take a second order Taylor expansion of $\log(Bx)_k$ in (6). This gives

$$\begin{aligned} &-\sum_{k=1}^{n} \hat{\rho}_k \log \hat{\rho}_k - \sum_{k=1}^{n} ((Bx)_k - \hat{\rho}_k) + \\ &+ \sum_{k=1}^{n} \hat{\rho}_k \left(\frac{(Bx)_k - \hat{\rho}_k}{\hat{\rho}_k}\right)^2 + O\left(\frac{\|Bx - \hat{\boldsymbol{\rho}}\|_\infty^3}{\min_j \rho_j^2}\right). \end{aligned}$$

The first term is independent of x, whereas the second term vanishes. Thus, we may approximate (6) by the *quadratic program*

$$\operatorname*{argmin}_{x_i \geq 0,\, \|x\|_1 = 1} \|\hat{\boldsymbol{\rho}} - Bx\|_{(1/\hat{\boldsymbol{\rho}})}^2 \tag{7}$$

where $\|x\|_w^2 = \sum_k w_k x_k^2$ is a weighted ℓ_2 norm w.r.t. the weight vector w. Eq. (7) is also a convex problem, easily solved via standard optimization techniques. However, let us temporarily ignore the non-negativity constraints $x_i \geq 0$ and add a Lagrange multiplier for the equality constraint $\sum x_i = 1$:

$$\min \frac{1}{2}\sum_{k=1}^{m} \frac{1}{\hat{\rho}_k}\Big(\hat{\rho}_k - \sum_{j=1}^{n} B_{kj} x_j\Big)^2 - \lambda\Big(\sum_j x_j - 1\Big). \tag{8}$$

Differentiating with respect to x_i yields

$$Wx = (1+\lambda)\mathbf{1}, \tag{9}$$

where $W = B^\intercal \mathrm{diag}(1/\hat{\boldsymbol{\rho}})B$. Enforcing the normalization constraint is equivalent to solving for $x^* = W^{-1}\mathbf{1}$ and normalizing $\hat{\boldsymbol{\pi}} = x^* / \|x^*\|_1$. Note that if all entries of x^* are positive, $\hat{\boldsymbol{\pi}}$ is the solution of the optimization problem in (7), and we need not invoke a QP solver. Assumptions (1a,1b) that π_k is bounded away from zero and that the chain is mixing imply that for sufficiently large T, all entries of $\hat{\boldsymbol{\pi}}$ will be positive with high probability, see Section 4.

Estimating the transition matrix A. To estimate A, we consider pairs (Y_t, Y_{t+1}) of consecutive observations. By definition we have that for a single pair,

$$P(Y_t = k, Y_{t+1} = k') = \sum_{i,j} B_{k'i} B_{kj} A_{ij} P(X_t = j).$$

As above, we treat the $T-1$ consecutive pairs (Y_t, Y_{t+1}) as independent of each other, with the hidden state X_t sampled from the stationary distribution $\boldsymbol{\pi}$. When the output probability matrix B and the stationary distribution $\boldsymbol{\pi}$ are both known, the pseudo-likelihood is given by

$$\mathcal{L}(y \,|\, A) = \prod_{(k,k')} \Big(\sum_{ij} B_{k'i} B_{kj} A_{ij} \pi_j \Big)^{n_{kk'}},$$

where $n_{kk'} = \sum_{t=1}^{T-1} \mathbb{1}_{\{y_{t-1}=k\}} \mathbb{1}_{\{y_t=k'\}} = (T-1)\hat{\sigma}_{kk'}$. The resulting estimator is

$$\operatorname*{argmin}_{A_{ij}\geq 0, \sum_i A_{ij}=1, A\boldsymbol{\pi}=\boldsymbol{\pi}} -\sum \hat{\sigma}_{kk'} \log \Big(\sum_{ij} C_{ij}^{kk'} A_{ij} \Big) \tag{10}$$

where $C_{ij}^{kk'} = \pi_j B_{kj} B_{k'i}$. In practice, since $\boldsymbol{\pi}$ is not known, we use $\hat{C}_{ij}^{kk'} = \hat{\pi}_j B_{kj} B_{k'i}$, with $\hat{\boldsymbol{\pi}}$ instead of $\boldsymbol{\pi}$. Again, (10) is a convex program in A and may be solved by standard constrained convex optimization methods. To obtain a more computationally efficient formulation, let us assume that $\min_{k,k'} \sigma_{k,k'} \geq a_2 > 0$ (this assumption is removed later, see sec. 4), and that $\min_{k,k'} T\hat{\sigma}_{kk'} \gg 1$, so that $|(\hat{C}A)_{kk'} - \hat{\sigma}_{kk'}| \ll \hat{\sigma}_{kk'}$, where $(\hat{C}A)_{kk'} = \sum_{ij} \hat{C}_{ij}^{kk'} A_{ij}$. Then, as above, the approximate minimization problem is

$$\operatorname*{argmin}_{A_{ij}\geq 0, \sum_i A_{ij}=1, A\hat{\boldsymbol{\pi}}=\hat{\boldsymbol{\pi}}} \left\| \hat{\boldsymbol{\sigma}} - \hat{C}A \right\|_{1/\hat{\boldsymbol{\sigma}}}^2. \tag{11}$$

In contrast to the estimation of $\boldsymbol{\pi}$, where we could ignore the non-negativity constraints, here the constraints $A_{ij} \geq 0$ are essential, since for realistic HMMs, some entries in A might be strictly zero. Note that if $\hat{\boldsymbol{\pi}} = \boldsymbol{\pi}$ and $\hat{\boldsymbol{\sigma}} = \boldsymbol{\sigma}$, the true matrix A satisfies $\boldsymbol{\sigma} = CA$ and is the minimizer of (10).

Finally, note that (11) with the standard ℓ_2 norm and an unknown matrix B is precisely the NNMF functional proposed by Lakshminarayanan & Raich (2010).

In summary, given one or more output sequences $(y_t)_{t=0}^{T-1}$ and an estimate of B, we first make a single pass over the data and construct the estimators $\hat{\boldsymbol{\rho}}$ and $\hat{\boldsymbol{\sigma}}$, with complexity $O(T)$. Then, the stationary distribution $\boldsymbol{\pi}$ is estimated via (9), and its transition matrix A via (11). To estimate A, we first compute the matrix product $\hat{C}^{\intercal}\hat{C}$, with $O(n^4m^2)$ operations. The resulting QP has size n^2, and is thus solvable (den Hertog, 1994) in time $O(n^6)$ — which is dominated by $O(n^4m^2)$ since $m \geq n$ by assumption. Hence, the overall time complexity of estimating A is $O(T + n^4m^2)$.

Extension to continuous observations. We now extend the above results to the case of continuous outputs distributed according to a known parametric family. Recall that in this case, each hidden state $i \in [n]$ has an associated output probability density $f_{\theta_i}(y)$. As with discrete observations, we assume that an approximation $(\hat{\theta}_1, \ldots, \hat{\theta}_n)$ to f_i's parameters is given and use it to construct estimates of $\boldsymbol{\pi}$ and A.

To this end, we seek analogues of (3) and (4), which relate the observable quantities to the latent ones. This will enable us to construct the appropriate empirical estimates and the corresponding quadratic programs, whose solutions will be our estimators $\hat{\boldsymbol{\pi}}$ and $\hat{A}$. To handle infinite output alphabets, we map each observation y to an n-dimensional vector $\varphi(y) = (f_{\theta_1}(y), \ldots, f_{\theta_n}(y))$, whose entries are the likelihood of y from each of the underlying hidden states. As shown below, this allows us to reduce the problem to a discrete "observation" space which can be solved by the methods introduced in the previous subsection.

Estimating the stationary distribution $\boldsymbol{\pi}$. To obtain an analogue of (3), we define the following vector $\boldsymbol{\xi} = \mathbf{E}[\varphi(Y)] \in \mathbb{R}^n$, and matrix $K \in \mathbb{R}^{n\times n}$, which play the roles of $\boldsymbol{\rho}$ and B for discrete output alphabets,

$$\xi_k \quad \equiv \quad \mathbf{E}[\varphi_k(Y)] = \sum_{j=1}^{n} \pi_j \int_{\mathcal{Y}} f_k(y) P(y \,|\, j) dy.$$

$$K_{ij} \equiv \mathbf{E}[\varphi_i(Y) \,|\, X = j] = \int_{\mathcal{Y}} f_i(y) P(y \,|\, j) dy. \tag{12}$$

With these definitions we have, as in Eq. (3),

$$\boldsymbol{\xi} = K\boldsymbol{\pi}. \tag{13}$$

Thus, given an observed sequence $(y_t)_{t=0}^{T-1}$ we construct the empirical estimate

$$\hat{\xi}_k = \frac{1}{T} \sum_{t=0}^{T-1} f_k(y_t), \tag{14}$$

and consequently solve the QP

$$\hat{\boldsymbol{\pi}} = \operatorname*{argmin}_{\|x\|_1=1, x\geq 0} \left\| \hat{\boldsymbol{\xi}} - Kx \right\|_{1/\hat{\boldsymbol{\xi}}}^2. \tag{15}$$

In analogy to the discrete case, we assume $\operatorname{rank}(K) = n$ so (15) has a unique solution. Its asymptotic consistency and accuracy are discussed in Section 4.

Estimating the transition matrix A. Next, following the same paradigm we derive an analogue of (4). Bayes rule implies that for stationary chains,

$$P(k \,|\, Y) = \frac{f_k(Y)\pi_k}{\sum_{l=1}^{n} f_l(Y)\pi_l}. \tag{16}$$

We define the matrices $\boldsymbol{\eta} \in \mathbb{R}^{n\times n}$ and $F \in \mathbb{R}^{n\times n}$ (analogues of $\boldsymbol{\sigma}$ and B) as follows. Let Y and Y' be two *consecutive* observations of the HMM, then

$$\begin{aligned} \eta_{kk'} &\equiv \mathbf{E}\left[P(k\,|\,Y)P(k'\,|\,Y')\right] \\ F_{kj} &\equiv \mathbf{E}[P(k\,|\,Y)\,|\,j] = \int_{\mathcal{Y}} P(k\,|\,y)P(y\,|\,j)dy. \end{aligned} \tag{17}$$

A simple calculation shows that, as in (4),

$$\eta_{kk'} = \sum_{i,j=1}^{n} \pi_j A_{ij} F_{kj} F_{k'i}. \tag{18}$$

Since here F plays the role of B, we may call it an *effective* observation matrix. This suggests estimating A with the same tools used in the discrete case. Thus, given an observed sequence $(y_t)_{t=0}^{T-1}$ we construct an empirical estimate $\hat{\boldsymbol{\eta}}$ by

$$\hat{\eta}_{kk'} = \frac{1}{T-1}\sum_{t=1}^{T-1} \hat{P}(k\,|\,y_{t-1})\hat{P}(k'\,|\,y_t), \tag{19}$$

where $\hat{P}$ is given by (16) but with π replaced by $\hat{\pi}$. Consequently we solve the following QP

$$\hat{A} = \operatorname*{argmin}_{A_{ij}\ge 0, \sum_i A_{ij}=1, A\hat{\boldsymbol{\pi}}=\hat{\boldsymbol{\pi}}} \left\| \hat{\boldsymbol{\eta}} - (\hat{C}A)\right\|^2_{1/\hat{\boldsymbol{\eta}}}, \tag{20}$$

where $\hat{C}_{ij}^{kk'} = \hat{\pi}_j F_{kj} F_{k'i}$ and $(\hat{C}A)_{kk'} = \sum_{ij}\hat{C}_{ij}^{kk'}A_{ij}$. As for the matrix B in the discrete case, to ensure a unique solution to Eq. (20) we assume $\operatorname{rank}(F) = n$.

Remark 1. *Instead of (18), we could estimate* $\eta'_{k,k'} \equiv \mathbf{E}[f_k(Y)f_{k'}(Y')]$*, and recover* A *from the relation*

$$\eta'_{k,k'} = \sum_{i,j=1}^{n} K_{k'i}K_{kj}A_{ij}\pi_j.$$

This has the advantage that for many distributions the matrix K *can be cast in a closed analytic form. For example, in the Gaussian case, we have*

$$K_{ij} = \frac{1}{\sqrt{2\pi}}\frac{1}{\sqrt{\sigma_i^2+\sigma_j^2}}\exp\left(-\frac{1}{2}\frac{(\mu_i-\mu_j)^2}{\sigma_i^2+\sigma_j^2}\right),$$

whereas F *needs to be calculated numerically. Additionally,* K *does not depend on the stationary distribution. The drawback is that in principle, and as simulations suggest, accurately estimating* $\boldsymbol{\eta}'$ *may require many more samples, see Kontorovich et al. (2013).*

In summary, given approximate output parameters $(\hat{\theta}_1,\ldots,\hat{\theta}_n)$, we first calculate the $n\times n$ matrix K. Next, we construct the vector $\hat{\boldsymbol{\xi}}$ by a single pass over the data $(Y_t)_{t=0}^{T-1}$. Then the stationary distribution $\boldsymbol{\pi}$ is estimated via (15). Given $\hat{\boldsymbol{\pi}}$, we calculate the $n\times n$ matrix F, construct the empirical estimate $\hat{\boldsymbol{\eta}}$, and estimate A via (20). As in the discrete observation case, the time complexity of this scheme is $O(T+n^6)$ with additional terms for calculating K and F.

4. Error analysis

First, we study the statistical properties of our estimators under the assumption that the output parameters, $(\theta_1,\ldots,\theta_n)$ in the continuous case, or the matrix B in the discrete case, are known *exactly*. Later on we show that our estimators are stable to perturbations in these parameters. For simplicity, throughout this section we assume that the initial hidden state X_0 is sampled from the stationary distribution $\boldsymbol{\pi}$. This assumption is not essential and omitting it would not qualitatively change our results. Due to space constraints, the proofs appear in Kontorovich et al. (2013).

To provide bounds on the error and required sample size we make the following additional assumptions:

(2a) In the discrete case, there exists an $a_1 > 0$ such that $\min_j \rho_j \ge a_1$.

(2b) In the continuous case, all f_{θ_i} are bounded:

$$\max_{i\in[n]}\sup_{y\in\mathbb{R}} f_{\theta_i}(y) \le L < \infty.$$

Finally, for ease of notation we define $g_\psi \equiv \frac{2G}{1-\psi}$.

Asymptotic Strong Consistency. Our first result shows that with perfectly known output probabilities, as $T\to\infty$, our estimates $\hat{\boldsymbol{\pi}}, \hat{A}$ are strongly consistent.

Theorem 1. *Let* $(Y_t)_{t=0}^{T-1}$ *be an observed sequence of an HMM, whose Markov chain satisfies Assumptions (1a,1b). Assume* $\operatorname{rank}(B) = n$ *in the discrete case, or* $\operatorname{rank}(F) = \operatorname{rank}(K) = n$ *in the continuous case. Then, both estimators,* $\hat{\boldsymbol{\pi}}$ *of (9) and* $\hat{A}$ *of (11) in the discrete case, or (15) and (20) in the continuous case, are asymptotically strongly consistent. Namely, as* $T\to\infty$*, with probability one,* $\hat{\boldsymbol{\pi}}\to\boldsymbol{\pi}$ *and* $\hat{A}\to A$.

Error analysis for the stationary distribution $\boldsymbol{\pi}$**.** Recall that to estimate $\boldsymbol{\pi}$ in the discrete case, we argued that for sufficiently large sample size T, the positivity constraints can be ignored, which amounts to solving an $n\times n$ system of linear equations, Eq. (9). The following theorem provides a lower bound on the required sample size T for this condition to hold with high probability, and a bound on the error $\hat{\boldsymbol{\pi}} - \boldsymbol{\pi}$.

Theorem 2. Discrete case*: Let* $\hat{\boldsymbol{\rho}}$ *be given by (5), and* $\hat{\boldsymbol{\pi}}$ *be the solution of (9). Let* $\tilde{B} = \operatorname{diag}(1/\sqrt{\boldsymbol{\rho}})B$, *and $\sigma_1(\tilde{B})$ be its smallest singular value. Under Assumption (2a), a sequence of length*

$$T \gtrsim \frac{g_\psi \sqrt{\log n}}{a_0 a_1 \sigma_1(\tilde{B})}, \tag{21}$$

suffices to ensure that with high probability, all entries in $\hat{\boldsymbol{\pi}}$ are strictly positive. Furthermore, as $T \to \infty$,

$$\|\hat{\boldsymbol{\pi}} - \boldsymbol{\pi}\|_2 \lesssim_P \sqrt{\frac{g_\psi^2}{T a_1^2 \sigma_1^2(\tilde{B})}}. \tag{22}$$

Next we consider the errors in the estimate $\hat{\boldsymbol{\pi}}$ for the continuous observations case. For simplicity, instead of analyzing the quadratic program (15) with a weighted ℓ_2 norm, we consider the following quadratic program, whose solution is also asymptotically consistent:

$$\min_{x \geq 0, \sum_i x_i = 1} \|\hat{\boldsymbol{\xi}} - Kx\|_2^2. \tag{23}$$

This allows for a cleaner analysis, without changing the qualitative flavor of the results.

Theorem 3. Continuous case*: Let $\hat{\boldsymbol{\xi}}$ be given by (14), $\hat{\boldsymbol{\pi}}$ be the solution of (23), and $\tilde{K} = \operatorname{diag}(1/\sqrt{\boldsymbol{\xi}})K$. Under Assumption (2b), as $T \to \infty$,*

$$\|\hat{\boldsymbol{\pi}} - \boldsymbol{\pi}\|_2 \lesssim_P \sqrt{\frac{(n^3 \ln n) g_\psi^2 L^4}{T \sigma_1^4(\tilde{K})}}, \tag{24}$$

Error Analysis for the Matrix A. Again, for simplicity, instead of analyzing the quadratic programs (11) and (20) with a weighted ℓ_2 norm, we consider the following quadratic programs, whose solutions are also asymptotically consistent for $\hat{\boldsymbol{\nu}} \in \{\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\eta}}\}$:

$$\min_{A_{ij} \geq 0, \sum_i A_{ij} = 1} \|\hat{\boldsymbol{\nu}} - \hat{C}A\|_2^2. \tag{25}$$

Note that this QP is applicable even if $\nu_{kk'} = 0$ for some k, k', which implies that $\hat{\nu}_{kk'} = 0$ as well.

Theorem 4. Discrete case. *Let $\hat{A}$ be the solution of (25) with $\hat{\boldsymbol{\nu}} = \hat{\boldsymbol{\sigma}}$ given in (5). Then, as $T \to \infty$,*

$$\left\|\hat{A} - A\right\|_F \lesssim_P \sqrt{\frac{n^3 g_\psi^2}{T a_0^4 a_1^2 \sigma_1^{10}(B)}} \tag{26}$$

and thus an observed sequence length

$$T \gtrsim \frac{n^3 g_\psi^2}{a_0^4 a_1^2 \sigma_1^{10}(B)} \tag{27}$$

suffices for accurate estimation.

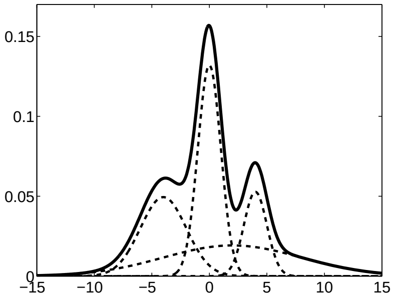

Figure 1. The mixture and its components.

Theorem 5. Continuous case. *Let $\hat{A}$ be the solution of (25) with $\hat{\boldsymbol{\nu}} = \hat{\boldsymbol{\eta}}$ given in (19). Then, as $T \to \infty$,*

$$\left\|\hat{A} - A\right\|_F \lesssim_P \sqrt{\frac{(n^7 \ln n) g_\psi^2 L^4}{T a_0^6 \sigma_1^8(F) \sigma_1^4(K)}} \tag{28}$$

and thus an observed sequence length

$$T \gtrsim \frac{(n^7 \ln n) g_\psi^2 L^4}{a_0^4 \sigma_1^8(F) \sigma_1^4(K)} \tag{29}$$

suffices for accurate estimation.

Remarks. Note the key role of the smallest singular value σ_1, in the error bounds in the theorems above: Two hidden states with very similar output probabilities drive σ_1 toward zero, thus requiring many more observations to resolve the properties of the underlying hidden sequence.

Inaccuracies in the output parameters. In practice we only have approximate output parameters, found for example, via an EM algorithm. For simplicity, we study the effect of such inaccuracies only in the continuous case. Similar results hold in the discrete case. To this end, assume the errors in the matrices K and F of Eqs. (12) and (17) are of the form

$$\hat{K} = K + \epsilon L^2 Q, \qquad \hat{F} = F + \epsilon P, \tag{30}$$

with $\|Q\|_F, \|P\|_F \leq 1$. The following theorem shows our estimators are *stable* w.r.t. errors in the estimated output parameters. Note that if K, F are estimated by a sequence of length T, then typically $\epsilon = O(T^{-1/2})$.

Theorem 6. *Given an error of $O(\epsilon)$ in the output parameters as in Eq. (30), the estimators given in Theorems 3 and 5, incur an additional error of $O\left(\frac{n^r \epsilon}{a_0^2 \sigma_1^4}\right)$, with $r = 1$ for estimating $\boldsymbol{\pi}$, and $r = \frac{3}{2}$ for estimating A, and where σ_1 is the smallest singular value of K/L^2 when estimating $\boldsymbol{\pi}$, and of F when estimating A.*

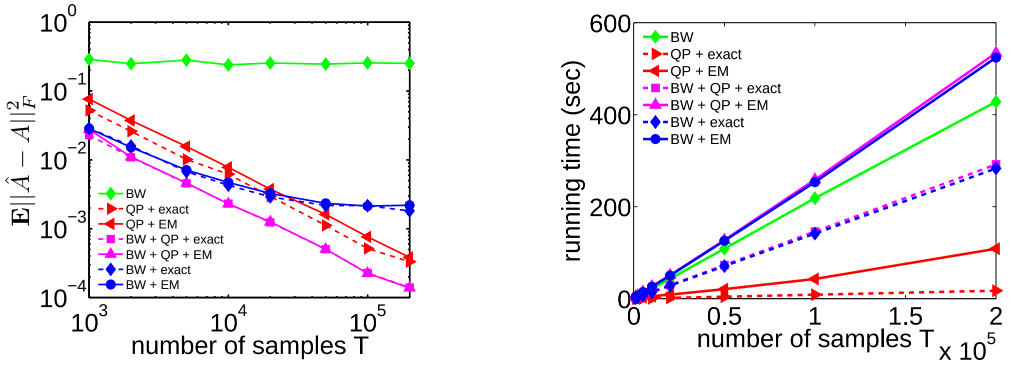

Figure 2. Average Error $\mathbf{E}\|\hat{A}-A\|_F^2$ and runtime comparison of different algorithms vs. sample size T.

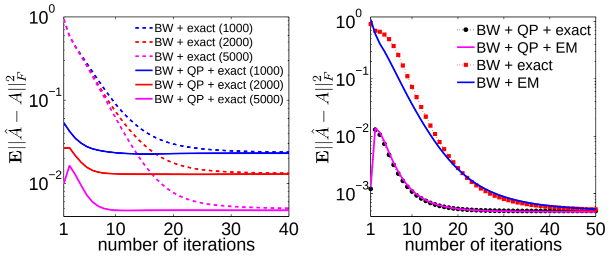

Figure 3. Convergence of the BW iterations.

5. Simulation Results

We illustrate our algorithm by some simulation results, executed in MATLAB with the help of the HMM and EM toolboxes[3]. We consider a toy example with $n = 4$ hidden states, whose outputs are univariate Gaussians, $\mathcal{N}(\mu_i, \sigma_i^2)$, with A, $\mathcal{F}_n^\theta$ and $\boldsymbol{\pi}$ given by

$$A = \begin{pmatrix} 0.7 & 0.0 & 0.2 & 0.5 \\ 0.2 & 0.6 & 0.2 & 0.0 \\ 0.1 & 0.2 & 0.6 & 0.0 \\ 0.0 & 0.2 & 0.0 & 0.5 \end{pmatrix}, \quad \begin{aligned} f_1 &= \mathcal{N}(-4, 4) \\ f_2 &= \mathcal{N}(0, 1) \\ f_3 &= \mathcal{N}(2, 36) \\ f_4 &= \mathcal{N}(4, 1) \end{aligned}$$

$$\boldsymbol{\pi}^\intercal = (0.3529, 0.2941, 0.2353, 0.1176).$$

Fig.1 shows the mixture and its four components.

To estimate A we considered the following methods:

	method	initial θ	initial A
1	BW	random	random
2	none	exactly known	QP
3	none	EM	QP
4	BW	exactly known	QP
5	BW	EM	QP
6	BW	exactly known	random
7	BW	EM	random

Fig. 2 (left) shows on a logarithmic scale $\mathbf{E}\|\hat{A}-A\|_F^2$ vs. sample size T, averaged over 100 independent realizations. Fig. 2 (right) shows the running time as a function of T. In these two figures, the number of iterations of the BW step was set to 20.

Fig. 3 (left) shows the convergence of $\mathbf{E}\|\hat{A}-A\|_F^2$ as a function of the number of BW iterations, with known output parameters, but either with or without the QP results. Fig. 3 (right) gives $\mathbf{E}\|\hat{A}-A\|_F^2$ as a function of the number of BW iterations for both known and EM-estimated output parameters with 10^5 samples.

The simulation results highlight the following points: (i) BW with a random guess of both A and the parameters $\theta_j = (\mu_j, \sigma_j^2)$ is useless if run for only 20 iterations. It often requires hundreds of iterations to converge, in some cases to a highly inaccurate solution (results not shows due to lack of space); (ii) For a small number of samples the accuracy of QP+EM (method 3) is comparable to BW+EM (method 5) but requires only a fraction of the computation time. (iii) When the number of samples becomes large, the QP+EM is not only faster, but (surprisingly) also more accurate than BW+EM. As Fig. 3 suggests, this is due to the slow convergence of the BW algorithm, which requires more than 20 iterations for convergence. (iv) Starting the BW iterations with (μ_i, σ_i^2) estimated by EM and A estimated by QP as its initial values significantly accelerated the convergence giving a superior accuracy after only 20 iterations. These results show the (well known) importance of initializing the BW algorithm with sufficiently accurate starting values. Our QP approach provides such an initial value for A by a computationally fast algorithm.

Acknowledgments. This research was supported by ISF grants #1141/12 and #328/10 and by the Frankel Center for Computer Science.

[3] Available at http://www.cs.ubc.ca/~murphyk and http://www.mathworks.com/ (under EM_GM_Fast).

References

Abe, N. and Warmuth, M.K. On the computational complexity of approximating distributions by probabilistic automata. *Machine Learning*, 9:205–260, 1992.

Anandkumar, A., Hsu, D., and Kakade, S.M. A method of moments for mixture models and hidden markov models. In *COLT*, 2012.

Bailly, Raphael. Quadratic weighted automata: Spectral algorithm and likelihood maximization. *Journal of Machine Learning Research*, 20:147–162, 2011.

Balle, Borja, Quattoni, Ariadna, and Carreras, Xavier. Local loss optimization in operator models: A new insight into spectral learning. *arXiv preprint arXiv:1206.6393*, 2012.

Baum, L.E., Petrie, T., Soules, G., and Weiss, N. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Stat.*, 41(1):pp. 164–171, 1970.

Belkin, M. and Sinha, K. Polynomial learning of distribution families. In *Foundations of Computer Science (FOCS)*, pp. 103–112, 2010.

Bickel, P.J., Ritov, Y., and Rydén, T. Asymptotic normality of the maximum-likelihood estimator for general hidden Markov models. *Ann. Statist.*, 26(4):1614–1635, 1998.

Cappé, O., Moulines, E., and Rydén, T. *Inference in hidden Markov models*. Springer Series in Statistics. Springer, New York, 2005.

Chang, J.T. Full reconstruction of Markov models on evolutionary trees: identifiability and consistency. *Math. Biosci.*, 137(1):51–73, 1996.

Cybenko, G. and Crespi, V. Learning hidden Markov models using nonnegative matrix factorization. *IEEE Trans. Information Theory*, 57(6):3963 –3970, 2011.

Daniel, J.W. Stability of the solution of definite quadratic programs. *Mathematical Programming*, 5:41–53, 1973.

Dantzig, G.B., Folkman, J., and Shapiro, N. On the continuity of the minimum sets of a continuous function. *J. Math. Anal. Appl.*, 17:519–548, 1967.

den Hertog, D. *Interior point approach to linear, quadratic and convex programming*, volume 277 of *Mathematics and its Applications*. Kluwer, Dordrecht, 1994.

Douc, R. and Matias, C. Asymptotics of the maximum likelihood estimator for general hidden Markov models. *Bernoulli*, 7(3):pp. 381–420, 2001.

Faragó, A. and Lugosi, G. An algorithm to find the global optimum of left-to-right hidden Markov model parameters. *Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform.*, 18(6):435–444, 1989.

Hsu, D., Kakade, S.M., and Zhang, T. A spectral algorithm for learning hidden markov models. In *COLT*, 2009.

Kontorovich, A. and Weiss, R. Uniform Chernoff and Dvoretzky-Kiefer-Wolfowitz-type inequalities for Markov chains and related processes, arxiv:1207.4678. 2012.

Kontorovich, Aryeh, Nadler, Boaz, and Weiss, Roi. On learning parametric-output hmms, arxiv:1302.6009. 2013.

Lakshminarayanan, B. and Raich, R. Non-negative matrix factorization for parameter estimation in hidden markov models. In *Machine Learning for Signal Processing (MLSP)*, pp. 89 –94, 2010.

Lyngsø, R. B. and Pedersen, C. N. Complexity of comparing hidden markov models. In *Proceedings of the 12th International Symposium on Algorithms and Computation*, pp. 416–428. Springer-Verlag, 2001.

Mohri, M. and Rostamizadeh, A. Stability bounds for stationary φ-mixing and β-mixing processes. *The Journal of Machine Learning Research*, 11:789–814, 2010.

Moitra, Ankur and Valiant, Gregory. Settling the polynomial learnability of mixtures of gaussians. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pp. 93–102. IEEE, 2010.

Mossel, E. and Roch, S. Learning nonsingular phylogenies and hidden Markov models. *Ann. Appl. Probab.*, 16(2): 583–614, 2006.

Nesterov, Y. and Nemirovskii, A. *Interior-point polynomial algorithms in convex programming.* SIAM, Philadelphia, PA, 1994.

Rabiner, L. R. Readings in speech recognition. chapter A tutorial on hidden Markov models and selected applications in speech recognition, pp. 267–296. Morgan Kaufmann, 1990.

Roweis, S. and Ghahramani, Z. A unifying review of linear gaussian models. *Neural Comput.*, 11:305–345, February 1999. ISSN 0899-7667.

Siddiqi, S. M., Boots, B., and Gordon, G. J. Reduced-rank Hidden Markov Models. In *AISTAT*, 2010.

Song, Le, Boots, Byron, Siddiqi, Sajid, Gordon, Geoffrey, and Smola, Alex. Hilbert space embeddings of hidden markov models. 2010.

Terwijn, S. On the learnability of Hidden Markov Models. In *Proceedings of the 6th International Colloquium on Grammatical Inference: Algorithms and Applications*, ICGI '02, pp. 261–268, London, UK, 2002. Springer-Verlag.