Reduced-Rank Hidden Markov Models

Sajid M. Siddiqi
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
siddiqi@google.com

Byron Boots
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213
beb@cs.cmu.edu

Geoffrey J. Gordon
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213
ggordon@cs.cmu.edu

Abstract

Hsu et al. (2009) recently proposed an efficient, accurate *spectral* learning algorithm for Hidden Markov Models (HMMs). In this paper we relax their assumptions and prove a tighter finite-sample error bound for the case of *Reduced-Rank HMMs*, i.e., HMMs with low-rank transition matrices. Since rank-k RR-HMMs are a larger class of models than k-state HMMs while being equally efficient to work with, this relaxation greatly increases the learning algorithm's scope. In addition, we generalize the algorithm and bounds to models where multiple observations are needed to disambiguate state, and to models that emit multivariate real-valued observations. Finally we prove consistency for learning Predictive State Representations, an even larger class of models. Experiments on synthetic data and a toy video, as well as on difficult robot vision data, yield accurate models that compare favorably with alternatives in simulation quality and prediction accuracy.

1 Introduction and Related Work

Models of stochastic discrete-time dynamical systems have important applications in a wide range of fields. Hidden Markov Models (HMMs) (Rabiner, 1989) and Linear Dynamical Systems (LDSs) (Kalman, 1960) are two examples of *latent variable models* which assume that sequential data points are noisy, incomplete observations of a latent state that evolves over time. The distributional assumptions of HMMs and LDSs result in important differences in the evolution of their belief over time. The discrete state of HMMs is good for modeling systems with *competitive inhibition*: e.g., an HMM can predict that our next observation will be either image A or image B, while disallowing blends of A and B. In an LDS, the joint predictive distribution over observations is log-concave, and thus cannot represent competitive inhibition. However, LDSs model *smooth state evolution*, which HMMs can only model by discretizing the state space very finely. Ideally we would like to model *both* competitive inhibition *and* smooth evolution, but few models display both of these properties. Those that do, e.g. Switching State Space Models (Ghahramani & Hinton, 2000), typically rely on on approximations for inference and learning.

HMMs are typically learned using Expectation-Maximization (EM) (Rabiner, 1989), which is prone to local optima, especially in large state spaces. On the other hand, LDSs are often learned using Subspace Identification (Subspace ID) (Van Overschee & De Moor, 1996). The latter is a *spectral* method: it finds an approximate factorization of the estimated covariance between past and future observations. And, it learns an *observable representation*, whose parameters can be simply related to directly-measurable quantities. In part because of these qualities, subspace ID is free of local optima and statistically consistent, though (unlike EM) it does not typically find even a local optimum of the log-likelihood for finite sample sizes.

More recently, researchers have proposed two generalizations of HMMs: Observable Operator Models (OOMs) (Jaeger, 2000) and Predictive State Representations (PSRs) (Singh et al., 2004). These models represent states as vectors of predicted probabilities of future events (called tests or characteristic events) conditioned on past events (called histories or indicative events). This representation is observable, and so we might hope to discover efficient, consistent spectral learning algorithms for PSRs and OOMs. But, despite much research in this area (summarized in Wingate (2008)), there is still a lack of provably accurate learning algorithms that have been demonstrated to work well in practice. (Since PSRs and OOMs are largely

Appearing in Proceedings of the 13^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2010, Chia Laguna Resort, Sardinia, Italy. Volume 9 of JMLR: W&CP 9. Copyright 2010 by the authors.

equivalent, we will refer to both as PSRs below.)

Recently, Hsu, Kakade and Zhang (HKZ for short) proposed a spectral algorithm which learns observable representations of HMMs (Hsu et al., 2009). The HKZ algorithm is free of local optima and statistically consistent, with a finite-sample bound on L_1 error in joint probability estimates. However, learning large-state-space HMMs is still difficult: the number of parameters grows prohibitively with the size of the state space.

In this paper, we extend the advantages of spectral learning to a larger class of models. We propose a variant of HMMs called *Reduced-Rank HMMs (RR-HMMs)* (Section 2), which have a large implicit state space but a low-rank transition matrix. RR-HMMs admit a compact observable representation (Definition 1) whose dimensionality depends on the rank rather than the state space size, allowing inference and learning on rank-k RR-HMMs with arbitrarily large state spaces in $\mathcal{O}(Nk^2)$ time, where $N > k$ is the number of training samples. The large implicit state space allows RR-HMMs to model smooth state trajectories, and the compact representation allows efficient learning. We generalize the HKZ spectral algorithm and bounds to the RR-HMM, deriving tighter bounds that depend on the rank and not the number of states (Section 3). Since rank-k RR-HMMs are much more expressive than k-state HMMs, our generalization greatly increases the scope of the spectral learning algorithm. Further, we prove that the spectral method is statistically consistent for a class of uncontrolled PSRs (Section 3.3).

We also generalize spectral HMM learning in other important ways. HKZ assumes that single observations are informative about the latent state (1-*step observability*) and that observations are discrete. In Section 3.4 we describe and test a method for relaxing the former by combining observations to make them more informative. In Section 3.5 we show how to handle high-dimensional real-valued observations with Kernel Density Estimation (KDE) (Silverman, 1986).

Experiments (Section 4) show that our learning algorithm can recover the underlying RR-HMM in a variety of synthetic domains, and show that RR-HMMs perform well compared to LDSs and HMMs on difficult video simulation and prediction tasks. Finally, we demonstrate that RR-HMMs are able to compactly model smooth evolution *and* competitive inhibition, both in a clock pendulum video and in real-world mobile robot vision data (see videos at `http://www.select.cs.cmu.edu/projects/RRHMM`).

2 Reduced-Rank HMMs

Let $h_t \in 1, \ldots, m$ denote the discrete hidden state of an HMM at time t, and $x_t \in 1, \ldots, n$ denote the discrete observation. Assume for now that $m \leq n$ (we relax this assumption in Section 3.4). Let $T \in \mathbb{R}^{m \times m}$ be the state transition probability matrix with $T_{ij} = \Pr[h_{t+1} = i \mid h_t = j]$. Let $O \in \mathbb{R}^{n \times m}$ be the observation probability matrix, with $O_{ij} = \Pr[x_t = i \mid h_t = j]$. Write O_x for the column of O corresponding to observation x, and $\mathrm{diag}(O_x)$ for a diagonal matrix with O_x on the diagonal. Let $\vec{\pi} \in \mathbb{R}^m$ be the initial state distribution, $\vec{\pi}_i = \Pr[h_1 = i]$. Let $\vec{h}_t \in \mathbb{R}^m$ denote the system's *belief*, i.e., the distribution over hidden states at time t given all observations up to time t.

In addition to the standard HMM notation above, assume T has rank k and let $T = RS$ where $R \in \mathbb{R}^{m \times k}$ and $S \in \mathbb{R}^{k \times m}$. Assume also that the initial state distribution lies in the low dimensional space, i.e., $\vec{\pi} = R\vec{\pi}_l$ for some vector $\vec{\pi}_l \in \mathbb{R}^k$. As we show below, these assumptions imply that the dynamics of the system can be expressed in $\mathbb{R}^k$ rather than $\mathbb{R}^m$. We denote the k-dimensional projection of the hidden state as $\vec{l}_t$, which is simply a vector of real numbers rather than a stochastic vector. Figure 1(A) shows the RR-HMM graphical model. Figure 1(B) illustrates the matrices R, S and O, the spaces they act on (discrete latent space, low-rank continuous latent space, and observation space), and the random variables within those spaces. By convention, we think of S as projecting h_t to l_t and R as propagating l_t to h_{t+1}.

To see how the probability of a sequence can be computed using these parameters, define $A_x = RS\ \mathrm{diag}(O_x)$, so that $A_x \in \mathbb{R}^{m \times m}$, and define $W_x = S\ \mathrm{diag}(O_x)R$, so that $W_x \in \mathbb{R}^{k \times k}$. With these definitions, the joint probability of $x_1, ..., x_t$, can be written using $\{A_x\}$, but also using $\{W_x\}$ (Jaeger, 2000), as

$$\Pr[x_1, ..., x_t] = \vec{1}_m^{\mathsf{T}} A_{x_t} \ldots A_{x_1} \vec{\pi} = \vec{1}_m^{\mathsf{T}} R W_{x_t} \ldots W_{x_1} \vec{\pi}_l$$

The latter parametrization casts a rank-k RR-HMM as a k-dimensional PSR or transformed PSR (Rosencrantz et al., 2004). Inference can be carried out in $\mathcal{O}(Nk^2)$ time in this representation. However, since every HMM is trivially a PSR, this leads to the question of how expressive rank-k RR-HMMs are in comparison to k-state full-rank HMMs.

2.1 How Expressive are RR-HMMs?

We give an example of an RR-HMM whose set of possible predictive distributions is easy to visualize and describe. Our example has rank 3, 10 states, and 4 observations. The observation probabilities in each state are of the form $O_i = [\ p_i q_i \quad p_i \overline{q}_i \quad \overline{p}_i q_i \quad \overline{p}_i \overline{q}_i]^{\mathsf{T}}$ for some $0 \leq p_i, q_i \leq 1$, $\overline{p} = 1 - p_i$ and $\overline{q} = 1 - q_i$. That is, there are 4 discrete observations, factored as

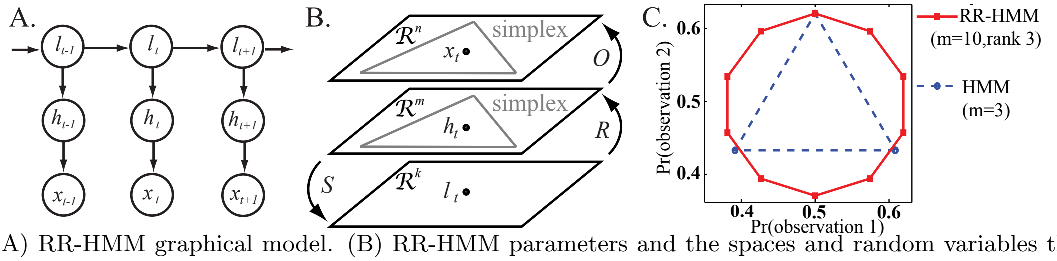

Figure 1: (A) RR-HMM graphical model. (B) RR-HMM parameters and the spaces and random variables they act on. (C) Projection of sets of predictive distributions of a rank-3 RR-HMM and a 3-state HMM with similar parameters.

two binary components which are independent given the state. T and p_i, q_i are chosen to place the vertices of the set of possible predictive distributions on evenly spaced points along a circle in (p,q)-space: $p_i = [\sin(2\pi i/m)+1]/2$, $q_i = [\cos(2\pi i/m)+1]/2$ and

$$T_{ij} = \frac{1}{2m}\left[2 + \sin\frac{2\pi i}{m}\cdot\sin\frac{2\pi j}{m} + \cos\frac{2\pi i}{m}\cdot\cos\frac{2\pi j}{m}\right]$$

In Figure 1(C) we plot the marginal probability of each component of the observation, for all achievable values of the latent state vector, for the $m = 10$ case. These distributions are marginals of the rows of $O^\top T$. We also plot the corresponding marginals for the $m = 3$ case, which is a full-rank 3-state HMM. In general, for an m-state HMM of any rank, the set of possible predictive distributions is a polyhedron with at most m vertices. So, rank-k RR-HMMs (which can have $m \gg k$ states) can model sets of predictive distributions which k-state HMMs cannot. For more on PSR expressivity, see Jaeger (2000) and Singh et al. (2004).

2.2 The Observable Representation

We define a representation of an RR-HMM based only on observable quantities, which makes it easier to learn. This idea is analogous to the HMM observable representation of HKZ. The observable representation depends on the initial probability vector $\vec{P}_1 \in \mathbb{R}^n$ and on the covariance and "trivariance" matrices $P_{2,1} \in \mathbb{R}^{n\times n}$ and $P_{3,x,1} \in \mathbb{R}^{n\times n}$ for $x = 1\ldots n$:

$$[\vec{P}_1]_i = \Pr[x_1 = i]$$
$$[P_{2,1}]_{i,j} = \Pr[x_2 = i, x_1 = j]$$
$$[P_{3,x,1}]_{i,j} = \Pr[x_3 = i, x_2 = x, x_1 = j]$$

for $i, j = 1,\ldots,n$. They can be expressed in terms of RR-HMM parameters, e.g.

$$P_{2,1} = ORS\ \mathrm{diag}(\pi)O^\top \tag{1a}$$
$$P_{3,x,1} = ORW_xS\ \mathrm{diag}(\pi)O^\top \tag{1b}$$

Note that $P_{2,1}$ and $P_{3,x,1}$ each have rank k because of the factor R. The primary intuition is that the distributions of tuples of observations reveal the low-rank structure in the transition matrix, and hence can be used to infer the observable parameters.

Definition 1 *The observable representation of an RR-HMM comprises the parameters* $b_1, b_\infty, \{B_x\}_{x=1}^n$*:*

$$\vec{b}_1 = U^\top\vec{P}_1 \tag{2a}$$
$$\vec{b}_\infty = (P_{2,1}^\top U)^+\vec{P}_1 \tag{2b}$$
$$B_x = (U^\top P_{3,x,1})(U^\top P_{2,1})^+ \quad x = 1,\ldots,n \tag{2c}$$

where $U \in \mathbb{R}^{n\times k}$ *is such that* $U^\top OR$ *is invertible.*

Note that the RR-HMM dimensionality is determined by k, not m: $b_1 \in \mathbb{R}^k$, $b_\infty \in \mathbb{R}^k$ and $\forall x \quad B_x \in \mathbb{R}^{k\times k}$. Though these definitions seem arbitrary, they relate closely to the original RR-HMM parameters:

$$\vec{b}_1 = (U^\top OR)\pi_l = (U^\top O)\pi \tag{3a}$$
$$\vec{b}_\infty^\top = 1_m^\top R(U^\top OR)^{-1} \tag{3b}$$
$$B_x = (U^\top OR)W_x(U^\top OR)^{-1} \quad x = 1,\ldots,n \tag{3c}$$

Hence B_x is a similarity transform of the RR-HMM parameter matrix $W_x = S\ \mathrm{diag}(O_x)R$, and $\vec{b}_1$ and $\vec{b}_\infty$ are the corresponding linear transformations of the initial state distribution and the normalization vector. To illustrate, we prove equation 3(c); proofs of 3(a–b) are similar (Siddiqi et al., 2009).

$$\begin{aligned}
B_x &= U^\top P_{3,x,1}(U^\top P_{2,1})^+ \\
&= (U^\top OR)W_xS\ \mathrm{diag}(\vec{\pi})O^\top(U^\top P_{2,1})^+ \quad \text{(eq. 1(b))} \\
&= (U^\top OR)W_x(U^\top OR)^{-1}(U^\top OR)S\ \mathrm{diag}(\vec{\pi})O^\top(U^\top P_{2,1})^+ \\
&= (U^\top OR)W_x(U^\top OR)^{-1}(U^\top P_{2,1})(U^\top P_{2,1})^+ \ \text{(eq. 1(a))} \\
&= (U^\top OR)W_x(U^\top OR)^{-1} \qquad (4)
\end{aligned}$$

2.3 RR-HMM Inference in the Observable Representation

For inference in the RR-HMM using the observable representation, we define the *internal state* $\vec{b}_t$. Just as the parameter $\vec{b}_1$ is a linear transform of the initial RR-HMM belief state, $\vec{b}_t$ is a linear transform of the belief state of the RR-HMM at time t:

$$\vec{b}_t = (U^\top OR)\vec{l}_t(x_{1:t-1}) = (U^\top O)\vec{h}_t(x_{1:t-1}) \tag{5}$$

$\vec{b}_t$ can be updated to condition on observations and evolve over time, just as we can update $\vec{l}_t$ for RR-HMMs and $\vec{h}_t$ for regular HMMs. Given a set of observable parameters, we can now easily compute sequence probabilities (eqn. 6(a)), update the internal state (eqn. 6(b)), and compute conditional probabilities (eqn. 6(c)) (proofs in Siddiqi et al. (2009)):

$$\widehat{\Pr}[x_1, \ldots, x_t] = \widehat{b}_\infty^\top \widehat{B}_{x_t} \ldots \widehat{B}_{x_1} \widehat{b}_1 \tag{6a}$$

$$\widehat{b}_{t+1} = \frac{\widehat{B}_{x_t} \widehat{b}_t}{\widehat{b}_\infty^\top \widehat{B}_{x_t} \widehat{b}_t} \tag{6b}$$

$$\widehat{\Pr}[x_t \mid x_{1:t-1}] = \frac{\widehat{b}_\infty^\top \widehat{B}_{x_t} \widehat{b}_t}{\sum_x \widehat{b}_\infty^\top \widehat{B}_x \widehat{b}_t} \tag{6c}$$

3 Learning Reduced-Rank HMMs

We estimate the parameters of the RR-HMM observable representation from data using Singular Value Decomposition (SVD) (Golub & Van Loan, 1996). The basic algorithm for estimating rank-k RR-HMMs is equivalent to the spectral learning algorithm of HKZ for learning k-state HMMs. However, our relaxation of their conditions (e.g., HKZ assume a full-rank transition matrix, without which their bounds are vacuous) leads to finite-sample performance guarantees for rank-k RR-HMMs. In addition, we provide (and analyze) generalizations to t-step observable RR-HMMs, to RR-HMMs with continuous observations, to indicative and characteristic *features* (rather than events), and to general PSRs. So, our new results allow us to learn a much larger class of models.

3.1 The Algorithm

The algorithm takes as input the desired rank k rather than the number of states m. Alternatively, given a threshold, the algorithm can choose the rank of the HMM by examining the singular values of $\widehat{P}_{2,1}$ (whose rank is k in the absence of noise) in Step 2. It assumes that we are given N independently sampled observation triples $\langle x_1, x_2, x_3 \rangle$ from the HMM. In practice, we can use a single long sequence of observations as long as we discount the bound on the number of samples based on the mixing rate of the HMM (i.e. $(1 -$ second eigenvalue of $T)$), in which case π must correspond to the stationary distribution of the HMM to allow estimation of $\vec{P}_1$. The algorithm results in an *estimated observable representation* of the RR-HMM:

```
Algorithm: Learn-RR-HMM($k, N$)
  1. Compute empirical estimates $\widehat{P}_1, \widehat{P}_{2,1}, \widehat{P}_{3,x,1}$ of
     $\vec{P}_1, P_{2,1}, P_{3,x,1}$ (for $x = 1, ..., n$).
  2. Use SVD on $\widehat{P}_{2,1}$ to compute $\widehat{U}$, the matrix
     of left singular vectors corresponding to the $k$
     largest singular values.
  3. Compute model parameter estimates:
     (a) $\widehat{b}_1 = \widehat{U}^\top \widehat{P}_1$,
     (b) $\widehat{b}_\infty = (\widehat{P}_{2,1}^\top \widehat{U})^+ \widehat{P}_1$,
     (c) $\widehat{B}_x = \widehat{U}^\top \widehat{P}_{3,x,1} (\widehat{U}^\top \widehat{P}_{2,1})^+$ ( $x = 1, \ldots, n$)
```

Estimated RR-HMM parameters can, in theory, lead to negative probability estimates, which is an intrinsic aspect of linear PSRs (Wiewiora, 2007). These are most harmful when they cause the normalizers $\widehat{b}_\infty^\top \widehat{B}_{x_t} \widehat{b}_t$ or $\sum_x \widehat{b}_\infty^\top \widehat{B}_x \widehat{b}_t$ to be negative. However, in our experiments, the latter was never negative and the former was very rarely negative; and, using real-valued observations (Section 3.5) makes negative normalizers even less likely, since in this case the normalizer is a weighted sum of several estimated probabilities. In practice we recommend thresholding the normalizers with a small positive number, and not trusting probability estimates for a few steps if the normalizers fall below the threshold.

3.2 Theoretical Guarantees

Theorem 2 bounds the L_1 error in joint probability estimates from the learned RR-HMM, generalizing Theorem 6 from HKZ to the case of low-rank T. This bound shows the consistency of the algorithm in learning a correct observable representation of the underlying RR-HMM, without ever needing to recover the high-dimensional parameters R, S, O of the latent representation. Note that the number of samples needed to achieve a certain error level is *independent* of m, the number of hidden states; instead, it depends on k, the rank of the transition matrix, which can be much smaller than m. Since HKZ explicitly assumes a full-rank HMM transition matrix, and their bounds become vacuous otherwise, generalizing their guarantees involves relaxing this condition.

Define $\sigma_k(M)$ to denote the k^{th} largest singular value of a matrix M. The sample complexity bounds depend polynomially on $1/\sigma_k(P_{2,1})$ and $1/\sigma_k(OR)$. The larger $\sigma_k(P_{2,1})$ is, the more well-separated are the dynamics from noise. The larger $\sigma_k(OR)$ is, the more informative the observation is regarding state. For both these quantities, the larger the magnitude, the fewer samples we need to learn a good model. The bounds also depend on a term $n_0(\epsilon)$, which is the minimum number of observations that account for $(1 - \epsilon)$ of probability mass, i.e. the number of "important" observations.

Theorem 2 *There exists a constant $C > 0$ such that the following holds. Pick any $0 \le \epsilon, \eta \le 1$ and $t \ge 1$. Assume $\vec{\pi} > 0$ everywhere,* rank$(T) = k$, rank$(U^\top OR) \ge k$ *and* rank$(O) \ge k$. *In addition, assume* rank$(S \operatorname{diag}(\vec{\pi}) O^\top) \ge k$, *$\|R\|_1 \le 1$, and for some column c of R, $\|R_c\|_2 \le \sqrt{k/m}$. Let $\varepsilon = \sigma_k(OR)\sigma_k(P_{2,1})\epsilon/(4t\sqrt{k})$. Assume*

$$N \ge C \cdot \frac{t^2}{\epsilon^2} \left(\frac{k}{\sigma_k(OR)^2 \sigma_k(P_{2,1})^4} + \frac{k \cdot n_0(\varepsilon)}{\sigma_k(OR)^2 \sigma_k(P_{2,1})^2} \right) \log \frac{1}{\eta}$$

With probability $\ge 1 - \eta$, the model returned by LearnRR-HMM*(k, N) satisfies*

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

Some of the assumptions above are similar to conditions in HKZ. Others (starting with "in addition") are unique to the low-rank setting. The condition on rank(S diag$(\vec{\pi})O^\top$) is needed to ensure invertibility. The condition on $\|R_c\|_2$ can be satisfied by choosing one column of R to be a near-uniform distribution. The condition on $\|R\|_1$ can be satisfied by scaling down R and scaling up S accordingly; however, this increases one of the terms in our bound, $1/\sigma_k(OR)$, so we pay a price by increasing the number of samples needed to attain a particular error bound. For details and proofs, see Siddiqi et al. (2009).

3.3 Consistent Learning of PSRs

Observable representations of rank-k RR-HMMs are a subset of k-dimensional PSRs. In the finite data case we are not guaranteed that the observable representation learned by the spectral method corresponds to any finite-state HMM or any RR-HMM. However, we can show that the spectral method is a statistically consistent PSR learning algorithm as well. This means that we can apply it to situations where the underlying model is not necessarily an RR-HMM and learn something sensible in the limit. This is a first step towards deriving error bounds to general PSRs.

PSRs have no transition or observation matrices. There are only *observable operators* (Jaeger, 2000), which account for an observation and a transition simultaneously; we will call these M_x, analogous to the W_x observable parameters of the RR-HMM. In addition, a PSR needs a normalization vector $\vec{m}_\infty$ and an initial prediction vector $\vec{m}_1$. Here we only prove consistency for the observable operators M_x in discrete-observation uncontrolled PSRs where one-step tests and histories are sufficient. We show consistency for other parameters, and for general discrete and continuous-observation controlled PSRs with arbitrary tests and histories, in Boots et al. (2009).

Denote tests by q and histories by h. PSR dimensionality is determined by the size of the *minimal* set of *core tests* that can represent it. Assume the PSR can be represented by a minimal set of k core tests $q_1, \ldots, q_k$. Assume all histories and tests are indicator functions of single-step observations, so that the expected value of a test or history is the probability of seeing an observation. The *prediction vector* $\vec{s}_h \in \mathbb{R}^k$ is defined as the vector of core test probabilities given the history h: $\vec{s}_h = [\Pr[q_i \mid h]]_{i=1}^k$. For PSRs, the prediction vector $\vec{s}_h$ is a sufficient statistic for the state of the system (Singh et al., 2004), and the probability of any other test τ can be computed linearly from $\vec{s}_h$ using some vector $\vec{r}_\tau$: $\Pr[\tau \mid h] = \vec{r}_\tau^\top \vec{s}_h$.

Minimal core test set discovery is a hard problem which many researchers have worked on (Jaeger et al., 2006; Wingate, 2008). Here, we take an approach that is closely related to Subspace ID and spectral HMM learning: we discover a minimal set of core tests by SVD of a matrix of probabilities of a larger set of tests, choosing dimensionality based on the singular values. This approach learns a representation called a *Transformed PSR* (TPSR) (Rosencrantz et al., 2004), which is simply a similarity transform of an ordinary PSR. Let $\mathcal{T} = \{\tau_1, \ldots, \tau_n\}$ and $\mathcal{H} = \{h_1, \ldots, h_n\}$ be the initial sets of $n > k$ core tests and histories (in general their sizes may differ, but for simplicity of notation we assume they are the same). Let $S \in \mathbb{R}^{k\times n}$ be the matrix whose columns $\vec{s}_h$ are prediction vectors for the unknown set of k core tests given all histories $h \in \mathcal{H}$. Let $R \in \mathbb{R}^{n\times k}$ be the matrix whose rows $\vec{r}_\tau$ are linear coefficients for computing probabilities of all tests $\tau \in \mathcal{T}$ from the unknown set of k core tests. Now define $\vec{P}_1$, $P_{2,1}$ and $P_{3,x,1}$ as

$$\begin{aligned}[\vec{P}_1]_i &= \Pr[h_i] \\ [P_{2,1}]_{i,j} &= \Pr[\tau_i, h_j] \\ [P_{3,x,1}]_{i,j} &= \Pr[\tau_i, x, h_j] \quad x = 1, \ldots, n\end{aligned}$$

for $i, j = 1, \ldots, n$. For brevity we will denote $\vec{P}_1$ by $\vec{\pi}$. It is now straightforward to show that $[P_{2,1}]_{i,j} = \vec{r}_{\tau_i}^\top \vec{s}_{h_j} \vec{\pi}_{h_j}$ and $[P_{3,x,1}]_{i,j} = \vec{r}_{\tau_i}^\top M_x \vec{s}_{h_j} \vec{\pi}_{h_j}$, and hence

$$P_{2,1} = RS \text{ diag}(\vec{\pi}) \tag{7a}$$

$$P_{3,x,1} = RM_xS \text{ diag}(\vec{\pi}) \tag{7b}$$

The parallels to the RR-HMM case are evident (see eqn. 1(a–b)). If we now fix a matrix $U \in \mathbb{R}^{n\times k}$ such that $U^\top R$ is invertible, we can define a k-dimensional parameterization $\langle \vec{b}_1, \vec{b}_\infty, \{B_x\}\rangle$ of the TPSR in terms of observable quantities; in particular, the observable operators are $B_x = U^\top P_{3,x,1}(U^\top P_{2,1})^+$ for observations x. B_x is related to M_x by a similarity transform, which we prove in a manner similar to eqn. (4) under similar conditions:

$$\begin{aligned}B_x &= U^\top P_{3,x,1}(U^\top P_{2,1})^+ \\ &= U^\top RM_xS \text{ diag}(\vec{\pi})(U^\top P_{2,1})^+ \quad \text{(by eq. 7(b))} \\ &= U^\top RM_x(U^\top R)^{-1}(U^\top R)S \text{ diag}(\vec{\pi})(U^\top P_{2,1})^+ \\ &= (U^\top R)M_x(U^\top R)^{-1}(U^\top P_{2,1})(U^\top P_{2,1})^+ \text{ (by eq. 7(a))} \\ &= (U^\top R)M_x(U^\top R)^{-1}\end{aligned}$$

Just as above, we can estimate B_x by plugging in empirical estimates $\widehat{P}_{2,1}, \widehat{P}_{3,x,1}$. Since $\widehat{P}_{2,1} \to P_{2,1}$ and $\widehat{P}_{3,x,1} \to P_{3,x,1}$, and since multiplication and pseudoinverse are continuous in a neighborhood of the limit due to our rank assumptions, we see that B_x is a statistically consistent estimator of M_x up to a similarity transform. Note that this argument depends on assuming a fixed U matrix, since SVD is not continuous in its inputs; in practice, however, we can use SVD to pick U for PSRs as well.

3.4 Learning with Sequences and Features

The probability matrix $P_{2,1}$ acts as a correlation matrix relating one past timestep to one future timestep. It is useful under the assumption that the vector of observation probabilities at a single step is sufficient to disambiguate state ($n \geq m$ and $\mathrm{rank}(O) = m$). In system identification theory, this corresponds to assuming 1-*step observability* (Van Overschee & De Moor, 1996). This assumption is unduly restrictive for many real-world dynamical systems of interest. More complex sufficient statistics of past and future may need to be modeled, such as the *block Hankel matrix* formulations for subspace methods (Van Overschee & De Moor, 1996), to identify linear systems that are not 1-step observable. It is possible to consider *sequences* of observations in the past and future and estimate larger versions of $P_{2,1}$ and $P_{3,x,1}$ accordingly (for single observations x). These matrices will have one row for each distinct sequence of past observations, and one column for each distinct sequence of future observations. As long as past and future sequences never overlap, these matrices still have rank equal to that of the dynamics model, and we can learn a k-dimensional RR-HMM representation with as many parameter matrices B_x as the number of distinct single observations, unlike the sequence-modeling method suggested in HKZ which was more complex and did not necessarily preserve rank. The consistency and error bounds of the algorithm hold as well.

Another interpretation of $P_{2,1}, P_{3,x,1}$ is as matrices containing expected values of products of indicator functions of observations, which correspond to simple indicative and characteristic events (Jaeger, 2000), or histories and tests (Singh et al., 2004). More generally, these matrices can contain expected values of statistics computed from the observations, which we call *indicative* and *characteristic features*. The consistency results still hold in this case; however extending the bounds to this case is an area of current research.

3.5 Learning with Real-Valued Observations

It is straightforward to model multivariate real-valued data sequences with RR-HMMs using KDE. Assume for ease of notation that the training data consists of N tuples of three consecutive continuous observation vectors each, i.e., $\langle \vec{x}_{1,1}, \vec{x}_{1,2}, \vec{x}_{1,3} \rangle, \langle \vec{x}_{2,1}, \vec{x}_{2,2}, \vec{x}_{2,3} \rangle, \ldots, \langle \vec{x}_{N,1}, \vec{x}_{N,2}, \vec{x}_{N,3} \rangle$. Also assume for now that each observation vector contains a single raw observation, though this technique can easily be combined with the more sophisticated sequence-based learning and feature-based learning methods described above. Pick a kernel function $K(\cdot)$ and n kernel centers $\vec{c}_1 \ldots \vec{c}_n$. (In general we can use different kernels and centers for different feature vectors.) Let λ be a bandwidth parameter that goes to zero at the appropriate rate in the limit. First compute $n \times 1$ feature vectors $\langle \vec{\phi}_j \rangle_{j=1}^N$, $\langle \vec{\psi}_j \rangle_{j=1}^N$, $\langle \vec{\xi}_j \rangle_{j=1}^N$ and $\langle \vec{\zeta}_j \rangle_{j=1}^N$, and normalize each to sum to 1:

$$[\vec{\phi}_j]_i \propto K(\vec{x}_{j,1} - \vec{c}_i) \qquad [\vec{\psi}_j]_i \propto K(\vec{x}_{j,2} - \vec{c}_i)$$
$$[\vec{\xi}_j]_i \propto K(\vec{x}_{j,3} - \vec{c}_i) \qquad [\vec{\zeta}_j]_i \propto K\left((\vec{x}_{j,2} - \vec{c}_i)/\lambda\right)$$

Note that, in $\vec{\zeta}_j$ only, we scale the kernel by the bandwidth λ. Then, estimate the vector $\vec{P}_1$ and matrices $P_{2,1}$ and $P_{3,x,1}$ (for $\vec{x} = \vec{c}_1, \ldots, \vec{c}_n$) from data:

$$\widehat{P}_1 = \frac{1}{N} \sum_{j=1}^{N} \vec{\phi}_j \qquad \widehat{P}_{2,1} = \frac{1}{N} \sum_{j=1}^{N} \vec{\psi}_j \vec{\phi}_j^{\mathsf{T}}$$

$$\text{For } x = c_1, \ldots, c_n\text{:} \;\; \widehat{P}_{3,x,1} = \frac{1}{N} \sum_{j=1}^{N} [\vec{\zeta}_j]_x \vec{\xi}_j \vec{\phi}_j^{\mathsf{T}}$$

We compute n "base" observable operators $B_{c_1}, \ldots, B_{c_n}$ from the estimated probability matrices, as well as vectors $\vec{b}_1$ and $\vec{b}_\infty$, using algorithm Learn-RR-HMM (Section 3.1). Filtering for a sequence $\langle \vec{x}_1, \ldots, \vec{x}_\tau \rangle$ now proceeds as follows:

For $t = 1, \ldots, \tau$:

1. Compute & normalize$[\vec{\sigma}_t]_i \propto K\left((\vec{x}_t - \vec{c}_i)/\lambda\right)$.

2. $B_{\sigma_t} = \sum_{j=1}^{n} [\vec{\sigma}_t]_j B_{c_j}$ 3. $\vec{b}_{t+1} = \dfrac{B_{\sigma_t} \vec{b}_t}{\vec{b}_\infty B_{\sigma_t} \vec{b}_t}$

Many of our theoretical results carry over to the KDE case (Siddiqi et al., 2009). Essentially, the bound holds for predicting functions of $\vec{\sigma}_1, \vec{\sigma}_2, \ldots, \vec{\sigma}_t$, though we cannot yet connect this bound to the error in estimating probabilities of raw observations.

4 Experimental Results

We designed several experiments to evaluate the properties of RR-HMMs and the learning algorithm on synthetic and real-world data. The first set of experiments (Sec. 4.1) tests the ability of the spectral learning algorithm to recover the correct RR-HMM. The second (Sec. 4.2) evaluates the representational capacity of the RR-HMM by learning a model of a video that requires both competitive inhibition and smooth state evolution. The third (Sec. 4.3) tests the model's ability to learn, filter and predict video captured from a robot moving in an indoor office environment.

4.1 Learning Synthetic RR-HMMs

First we evaluate the consistency of the spectral learning algorithm for RR-HMMs on 3 synthetic examples. In each case, we build an RR-HMM, sample observations from the model, and estimate the model with the spectral learning algorithm described in Section 3. In Figure 2 we compare the eigenvalues of $B = \sum_x B_x$ in the learned model to the eigenvalues of the transition

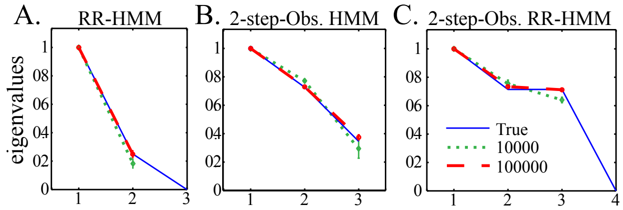

Figure 2: Learning discrete RR-HMMs. The three figures depict the actual eigenvalues of three different RR-HMM transition matrices, and the estimated eigenvalues with 95% error bars, for two different training set sizes.

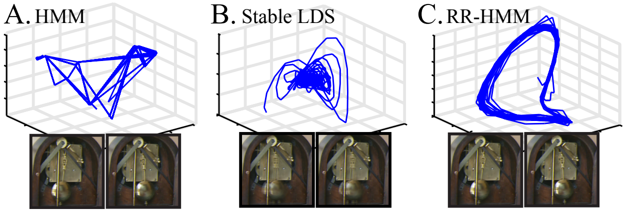

Figure 3: State space manifold and video frames simulated by a HMM, a stable LDS, and a RR-HMM learned using clock pendulum video (manifold scales are arbitrary). (A) 10-state HMM. (B) 10-dim LDS. (C) Rank 10 RR-HMM.

matrix T of the true model. B is a similarity transform of SR, which has the same non-zero eigenvalues as $T = RS$, so the estimated eigenvalues should converge to the true eigenvalues with enough data. See Siddiqi et al. (2009) for the HMM parameters.

Example 1: An RR-HMM [$m = 3$ hidden states, $n = 3$ observations, $k = 2$ rank]. In this example the RR-HMM is low-rank. See Figure 2(A).
Example 2: A 2-step-Observable HMM [$m = 3$ hidden states, $n = 2$ observations]. In this example, the HMM violates the $m \leq n$ condition of HKZ. The parameters of this HMM cannot be estimated with the original learning algorithm, since a single observation does not provide enough information to disambiguate state. However, by considering 2 consecutive observations (see Section 3.4), the spectral learning algorithm can be applied successfully. See Figure 2(B).
Example 3: A 2-step-Observable RR-HMM [$m = 4$ hidden states, $n = 2$ observations, $k = 3$ rank]. In this example, the HMM is low rank, *and* it violates the $m \leq n$ condition of HKZ. See Figure 2(C).

4.2 Competitive Inhibition+Smooth Dynamics

We model a clock pendulum video consisting of 55 frames (with a period of ∼ 22 frames) as a 10-state HMM, a 10-dimensional LDS, and a rank 10 RR-HMM. Note that we could easily learn models with more than 10 latent states/dimensions; we limited the dimensionality in order to demonstrate the relative expressive power of the different models. For the HMM, we convert the continuous data to discrete observations by 1-nearest neighbor on 25 kernel centers sampled sequentially from the training data. We trained the resulting discrete HMM using EM. We learned the LDS directly from the video using subspace ID with stability constraints (Siddiqi et al., 2007) using a Hankel matrix of 10 observations. We trained the RR-HMM by considering sequences of 4 continuous multivariate observations, choosing an approximate rank of 10 dimensions, and learning 25 observable operators corresponding to 25 Gaussian kernel centers. We simulate a series of 500 observations from the model and compare the manifolds in the 10-dimensional space and the observations and frames from the simulated videos (Figure 3). The small number of states in the HMM is not sufficient to capture the smooth evolution of the clock: the simulated video is characterized by realistic looking frames, but exhibits jerky irregular motion. For the LDS, although the 10-dimensional subspace captures smooth evolution of the simulated video, the system quickly degenerates and individual frames of video are modeled poorly (resulting in superpositions of pendulums in generated frames). For the RR-HMM, the simulated video benefits from both smooth state evolution *and* competitive inhibition. The low-dimensional manifold is smooth and structured and the video is realistic. The results demonstrate that the RR-HMM has the benefits of smooth state evolution and compact state space of a LDS and the benefit of competitive inhibition of a HMM.

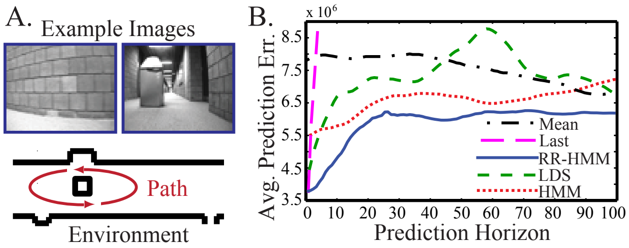

Figure 4: (A) Sample images from the robot's camera. The figure below depicts the hallway environment with a central obstacle (black) and the path that the robot took through the environment while collecting data (the red counter-clockwise ellipse) (B) Squared error for prediction $(1, \ldots, 100$ steps out in future) with different estimated models and baselines, averaged over different initial filtering durations $(1, \ldots, 250)$.

4.3 Filtering, Prediction, and Simulation

We compare HMMs, LDSs, and RR-HMMs on the problem of modeling video data from a mobile robot in an indoor environment. A video of 2000 frames was collected at 6 Hz from a Point Grey Bumblebee2 stereo camera mounted on a Botrics Obot d100 mobile robot platform circling a stationary obstacle (Figure 4(A)) and 1500 frames were used as training data for each model. Each frame from the training data was reduced to 100 dimensions via SVD on single observations. Using this training data, we trained an RR-HMM ($k = 50, n = 1500$) using spectral learning with sequences of 20 continuous observations and

KDE with Gaussian kernels with 1500 centers; a 50-dimensional LDS using Subspace ID with Hankel matrices of 20 timesteps; and a 50-state HMM with 1500 discrete observations using EM run until convergence. For each model, we performed filtering for different extents $t_1 = 100, 101, \ldots, 250$, then predicted an image which was a further t_2 steps in the future, for $t_2 = 1, 2 \ldots, 100$. The squared error of this prediction in pixel space was recorded, and averaged over all the different filtering extents t_1 to obtain means which are plotted in Figure 4(B). As baselines, we plot the error obtained by using the mean of filtered data as a predictor ('Mean'), and the error obtained by using the last filtered observation ('Last').

Both baselines perform worse than any of the more complex algorithms (though as expected, the 'Last' predictor is a good one-step predictor), indicating that this is a nontrivial prediction problem. The LDS does well initially (due to smoothness), and the HMM does well in the longer run (due to competitive inhibition), while the RR-HMM performs as well or better at both time scales since it models both the smooth state evolution and competitive inhibition in its predictive distribution. In particular, the RR-HMM yields lower prediction error consistently for the duration of the prediction horizon (100 steps, or $16\frac{2}{3}$ seconds).

5 Conclusion

We have generalized the spectral learning algorithm and bounds of Hsu et al. (2009) to accurately learn a larger class of sequential data models (RR-HMMs) under a larger class of observation models (non-1-step-observable domains and multivariate continuous observations). RR-HMMs combine desirable properties of HMMs and LDSs, allowing them to model a larger class of dynamical systems. We have also shown that the algorithm is consistent for learning a simple class of PSRs. The generalization of this algorithm to consistent learning of general controlled PSRs yields accurate, compact models that facilitate successful planning and control (Boots et al., 2009). The most important task for future work is to extend the performance bounds from RR-HMMs to general PSRs, as well as to explore practical applications in sequential data modeling and planning.

Acknowledgements

We acknowledge helpful conversations with Sham Kakade and comments from anonymous reviewers. Julian Ramos assisted with gathering robot vision data. SMS was supported by the NSF under grant number 0000164, the USAF under grant number FA8650-05-C-7264, the USDA under grant number 4400161514, and a project with MobileFusion/TTC. BEB was supported by the NSF under grant number EEEC-0540865. GJG was supported by DARPA under grant number HR0011-07-10026, the Computer Science Study Panel program, and DARPA/ARO under MURI grant number W911NF-08-1-0301. BEB and GJG were supported by ONR MURI grant number N00014-09-1-1052. SMS is now at Google.

References

Boots, B., Siddiqi, S., & Gordon, G. J. (2009). Closing the learning-planning loop with predictive state representations. *http://arxiv.org/abs/0912.2385.*

Ghahramani, Z., & Hinton, G. E. (2000). Variational learning for switching state-space models. *Neural Comp.*, *12*.

Golub, G. H., & Van Loan, C. F. (1996). *Matrix computations.* The Johns Hopkins University Press.

Hsu, D., Kakade, S., & Zhang, T. (2009). A spectral algorithm for learning hidden markov models. *COLT.*

Jaeger, H. (2000). Observable operator models for discrete stochastic time series. *Neural Computation*, *12*, 1371–1398.

Jaeger, H., Zhao, M., Kretzschmar, K., Oberstein, T., Popovici, D., & Kolling, A. (2006). Learning observable operator models via the es algorithm. In *New directions in statistical signal processing: from systems to brain.* MIT Press.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering.*

Rabiner, L. R. (1989). A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. IEEE.*

Rosencrantz, M., Gordon, G., & Thrun, S. (2004). Learning low dimensional predictive representations. *Proc. ICML.*

Siddiqi, S., Boots, B., & Gordon, G. (2007). A constraint generation approach to learning stable linear dynamical systems. *Proc. NIPS.*

Siddiqi, S., Boots, B., & Gordon, G. (2009). Reduced-rank hidden markov models. *http://arxiv.org/abs/0910.0902.*

Silverman, B. W. (1986). *Density estimation for statistics and data analysis.* Chapman & Hall.

Singh, S., James, M., & Rudary, M. (2004). Predictive state representations: A new theory for modeling dynamical systems. *Proc. UAI.*

Van Overschee, P., & De Moor, B. (1996). *Subspace identification for linear systems: Theory, implementation, applications.* Kluwer.

Wiewiora, E. (2007). *Modeling Probability Distributions with Predictive State Representations.* Doctoral dissertation.

Wingate, D. (2008). *Exponential family predictive representations of state.* Doctoral dissertation.