Finite Sample Analysis of Tensor Decomposition for Learning Mixtures of Linear Systems

Maryann Rui MRUI@MIT.EDU and **Munther A. Dahleh** DAHLEH@MIT.EDU
Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

Editors: N. Ozay, L. Balzano, D. Panagou, A. Abate

Abstract

We study the problem of learning mixtures of linear dynamical systems (MLDS) from input-output data. The mixture setting allows us to leverage observations from related dynamical systems to improve the estimation of individual models. Building on spectral methods for mixtures of linear regressions, we propose a moment-based estimator that uses tensor decomposition to estimate the impulse response parameters of the mixture models. The estimator improves upon existing tensor decomposition approaches for MLDS by utilizing the entire length of the observed trajectories. We provide sample complexity bounds for estimating MLDS in the presence of noise, in terms of both the number of trajectories N and the trajectory length T, and demonstrate the performance of the estimator through simulations.

Keywords: system identification, mixture model, tensor decomposition

1. Introduction

In many domains of learning time series, such as in healthcare, social sciences, and biological sciences (Ernst et al., 2005), there are often a large number of data sources (e.g., patients, systems, cells), but a limited amount of data from each individual source. Without additional assumptions, it may be impossible to identify individual models for each data source. However, when the data is actually generated from a few underlying models, we can leverage the collective observations to learn these models, which can then be used to improve estimates of individual systems. The setting of mixture models, in particular, allows for tractability in learning multiple models from data.

In this paper, we propose and study a moment-based estimator that uses tensor decomposition to learn mixtures of linear dynamical systems (MLDS) from input-output data. Compared to existing methods, which we detail in the following related work section, the estimator allows us to utilize the full length T of the observed trajectories. We also provide explicit sample complexity bounds for estimating stable MLDS in the presence of process and observation noise, showing how the error depends on system parameters and how increasing both the number of trajectories, N, and the individual trajectory length, T, can be leveraged to improve estimation.

In the remainder of this section, we review related work. In Section 2, we formalize our MLDS model. In Section 3, we introduce the moment-based MLDS estimator and an estimator for mixtures of linear regression (MLR), on which the MLDS estimator is built. In Section 4, we provide sample complexity bounds for the estimator in Proposition 4.1, where as a key step we derive finite sample error bounds for learning mixtures of linear regressions in the presence of independent noise and bounded perturbations. Finally, in Section 5, we demonstrate the performance of the tensor decomposition approach to MLDS through simulations. Auxiliary results and proofs may be found in the Appendix of the extended version of this paper (Rui and Dahleh, 2024).

© 2025 M. Rui & M.A. Dahleh.

1.1. Related Work

Our work lies at the intersection of spectral methods for mixtures of linear regression and system identification for partially observed linear systems. The most relevant work is Bakshi et al. (2023b), which also sits at this intersection, and which inspired us to derive an alternate moment estimator with explicit sample complexity guarantees for the MLDS problem. While recent work has also studied other forms of shared structure between multiple linear dynamical systems, such as a shared low dimensional representation of the transition matrix (Modi et al., 2021; Zhang et al., 2023), these are largely restricted to fully observed systems. Thus, we choose to focus our review of related work on *mixture* models of static and dynamical linear systems.

Mixtures of linear regression. In mixtures of linear regressions, data of the form $\{(x_i, y_i)\}_{i\in[N]}$ is observed, with a generating model given by $y_i = \langle x_i, \beta_i \rangle$, where the parameter β_i is sampled from a given distribution over the K mixture components $\{\beta_k\}_{k\in[K]}$. The goal is to learn the K mixture components and their respective mixture weights. Approaches to solving MLR can generally be grouped into those based on tensor decomposition (Anandkumar et al., 2014), alternating minimization (Yi et al., 2014), and gradient methods (Li and Liang, 2018), or a combination of these. Both Zhong et al. (2016) and Yi et al. (2016) apply tensor decomposition on sixth-order moments to initialize iterative algorithms based on gradient descent and alternating minimization, respectively. While they provide sample complexity guarantees for MLR in the *noiseless* setting, we extend their estimator and analyses to the setting of noisy observations of linear system trajectories.

Mixtures of linear dynamical systems. Chen and Poor (2022) also study learning mixtures of dynamical systems, though restricted to the fully-observed setting. Most relevant to our work is Bakshi et al. (2023b), which introduces a moment-based estimator that uses tensor decomposition to prove that under minimal assumptions, mixtures of linear dynamical systems (MLDS) can be learned with polynomial sample and computational complexity. However, they do not provide explicit sample complexity bounds and their algorithm only uses a fixed number of samples from each observed trajectory, forfeiting possibly useful information in longer trajectories. In this work, we provide a different moment-based estimator that utilizes the entire length of observed trajectories and derive explicit finite-sample error bounds for mixtures of stable systems with a sharper $\mathrm{poly}(\ln(1/\delta))$ dependence (versus $\mathrm{poly}(1/\delta)$ in Bakshi et al. (2023b)), where bounds are given with high probability $1-\delta$, and include the effects of process and measurement noise. A detailed comparison of the estimators is given in Section A.2 of the full version of this paper.

Finite sample bounds for linear system identification. There is a large body of work on the identification of partially observed linear systems from input-output data, with recent works providing finite-sample error bounds for learning from a single trajectory, or rollout. A standard approach is to estimate Markov parameters of the system and then use the Ho-Kalman, or eigensystem realization, algorithm (Ho and Kálmán, 1966) to obtain a state space realization of the system. Sarkar et al. (2021) and Oymak and Ozay (2021) estimate Markov parameters from a single trajectory of strictly stable systems using an ordinary least squares estimator. Bakshi et al. (2023a) derive a moment-based estimator for the Markov parameters, though the estimator coefficients must be computed via a separate convex program. Estimating single partially observed systems from *multiple* rollouts has also recently been studied. Zheng and Li (2020) provide error bounds for an OLS estimator on N independent length T trajectories, for both stable and unstable systems. However, the error in estimating the first T Markov parameters grows superlinearly in the trajectory length T, which is a suboptimal trend for strictly stable systems.

2. Setup

2.1. Notation

For any natural number $N \in \mathbb{N}$, we define the set $[N] := \{1, 2, \ldots, N\}$. For a $d_1 \times d_2$ matrix A, we denote its trace $\mathrm{Tr}(A)$, transpose A', Moore-Penrose pseudoinverse $A^\dagger$, Frobenius norm $\|A\|_{\mathrm{F}}$, and operator (spectral) norm $\|A\|_2$. For $i \in [\min(d_1, d_2)]$, $\sigma_i(A)$ is the i-th largest singular value of A. The identity matrix in $\mathbb{R}^{d\times d}$ is denoted I_d. A real-valued random variable X is subgaussian with variance proxy σ_x^2 if $\mathbb{P}[|X| \geq t] \leq 2\exp(-t^2/(2\sigma_x^2))$ for $t > 0$. If in addition, X is zero mean, we write $X \sim \mathrm{subG}(0, \sigma^2)$. Similarly, X is subexponential with parameter λ if $\mathbb{P}[|X| \geq t] \leq 2\exp(-t/\lambda)$. A random vector X is subgaussian if for all fixed vectors $v \in \mathbb{R}^n$, $\langle X, v\rangle$ is subgaussian (Vershynin, 2018). We use c to denote a universal positive constant, which may vary from line to line. For real-valued functions a, b, the inequality $a \lesssim b$ implies $a \leq cb$ for some c. Unless otherwise specified, all random variables are defined on the same probability space.

Tensors. A K-th order tensor in a Euclidean space is an element of the tensor product of K Euclidean spaces. The tensor product, or outer product, of K vectors $\{v_k \in \mathbb{R}^{d_k}\}_{k\in[K]}$ is denoted $v_1 \otimes v_2 \otimes \cdots \otimes v_K$ and is a rank 1 K-th order tensor with $(i_1, i_2, \ldots, i_K)$-th entry equal to $\prod_{k=1}^K v_k(i_k)$. For a vector $v \in \mathbb{R}^d$, $v^{\otimes K} = v \otimes v \otimes \cdots \otimes v$ (K times) is its K-th tensor power. In general, the rank of a tensor M is the smallest number of rank-one tensors such that M can be expressed as their sum. A third-order $d_1 \times d_2 \times d_3$ tensor M of rank r may thus be written as $M = \sum_{i=1}^r a_i \otimes b_i \otimes c_i$ for some $a_i \in \mathbb{R}^{d_1}, b_i \in \mathbb{R}^{d_2}, c_i \in \mathbb{R}^{d_3}$. Viewing such a tensor M as a multilinear map, we have the mapping for matrices $A \in \mathbb{R}^{d_1\times l_1}, B \in \mathbb{R}^{d_2\times l_2}$ and $C \in \mathbb{R}^{d_3\times l_3}$, $M(A, B, C) = \sum_{i=1}^r A'a_i \otimes B'b_i \otimes C'c_i \in \mathbb{R}^{l_1} \otimes \mathbb{R}^{l_2} \otimes \mathbb{R}^{l_3}$.

A symmetric third-order tensor M is invariant under permutations of its arguments (A, B, C). Its operator norm is defined as $\|M\|_2 = \sup_{a\in\mathcal{S}^{d-1}} |M(a, a, a)|$. For simplicity of notation, given a $d \times K$ matrix W, we sometimes use the shorthand $M^W := M(W, W, W)$. See Kolda and Bader (2009) for an introductory reference on tensors and tensor decomposition.

2.2. Model

Mixture model. A partially-observed, strictly causal, linear-time invariant (LTI) system can be represented in terms of its impulse response $g = (g(1),\ g(2),\ \ldots)$, which captures the input-output mapping of the system. Assuming for simplicity m-dimensional inputs and single-dimensional outputs, the jth impulse response, or Markov, parameter, $g(j)$ is an $m \times 1$ vector, for $j \in \mathbb{N}$. Given an input trajectory $\{u_t \in \mathbb{R}^m\}_{t\in\mathbb{N}}$, the output of the system at each time $t \in \mathbb{N}$ is given by

$$y_t = \sum_{j=1}^{t} \left\langle g(j), u_{t-j} + w_{t-j}^{(1)} \right\rangle + w_t^{(2)}, \tag{2.1}$$

where $w_t^{(1)} \in \mathbb{R}^m$ and $w_t^{(2)} \in \mathbb{R}$ represent process and measurement noise, respectively, at time t. We assume zero inputs $u_t = 0$ for $t < 0$ and zero feedthrough (i.e., y_t does not depend on u_t).

Consider a mixture of $K \geq 2$ LTI models given by $\mathcal{G} = \{(g_k, p_k)\}_{k\in[K]}$, where the model with impulse response sequence g_k has associated probability $p_k > 0$, with $\sum_{k=1}^K p_k = 1$. We observe N input-output trajectories of length T in the data set $\mathcal{D} = \{(u_{i,t-1}, y_{i,t}) \mid i \in [N], t \in [T]\}$, which are generated from the mixture model in the following way: For each trajectory $i \in [N]$, a system model $g_i = g_{k_i}$ is drawn from $\mathcal{G}$, where the index $k_i = k$ is drawn with probability

p_k, for $k \in [K]$. A trajectory is then rolled out with randomly generated inputs $\{u_{i,t-1}\}_{t\in[T]}$ and corresponding outputs $\{y_{i,t}\}_{t\in[T]}$ generated according to (2.1).

Remark 2.1 *Partially-observed LTI systems corresponding to (2.1) are often represented by the following input-state-output dynamics with a state variable $x_t \in \mathbb{R}^n$, where n is the minimal order of the system:*

$$x_{t+1} = Ax_t + B(u_t + w_t^{(1)}), \quad y_t = Cx_t + w_t^{(2)}.$$

$A \in \mathbb{R}^{n\times n}$ is the state transition matrix, $B \in \mathbb{R}^{n\times m}$ the control matrix, and $C \in \mathbb{R}^{1\times n}$ the measurement matrix. While the parameters (C, A, B) representing the system are only identifiable up to a similarity transformation, they correspond to the representation-independent Markov parameters by $g(t) = CA^{t-1}B$ for $t \geq 1$. Because the crux of most time-domain system identification methods, including ours, lies in estimating Markov parameters, we focus on the impulse-response representation and provide pointers to state-space estimation when relevant.

Objective. Given an input-output data set $\mathcal{D}$ of length-T trajectories from N systems, we aim to estimate the generating mixture $\mathcal{G}$ comprising the component models g_k and their weights p_k. To do so, it suffices to learn just a finite number of Markov parameters to identify the infinite impulse response sequence. If an LTI system given by g_k has finite order bounded by $n > 0$, the sequence g_k is completely determined by its first $2n + 1$ elements (Gragg and Lindquist, 1983). Thus, it suffices to learn the first $L \geq 2n + 1$ Markov parameters of each of the K models in mixture. Further, even if $L < 2n + 1$, the first L Markov parameters can still be very informative of the system behavior. To this end, for a fixed L such that $1 \leq L \leq T$, let us define the truncated impulse response vector $g_i^{(L)} = [g_i(1)', \ldots, g_i(L)']' \in \mathbb{R}^{Lm}$ for the system generating the ith trajectory, $i \in [N]$. We focus on learning the first L Markov parameters and weights $\{(g_k^{(L)}, p_k)\}_{k\in[K]}$ of the mixture.

2.3. Assumptions

Dynamics and distributional assumptions. We assume that each of the K models in the mixture are strictly stable. Under this assumption, define the finite quantity $\Gamma(g_k) := 1 + \sum_{t=1}^{\infty}\|g_k(t)\|_2^2$ capturing the energy of each system $k \in [K]$, and $\Gamma_{\max} := \max_{k\in[K]}\Gamma(g_k) < \infty$. Furthermore, let $\rho > 0$ and $C_\rho > 0$ be such that for every $t \in \mathbb{N}$, $\max_{k\in[K]}\|g_k(t)\|_2 \leq C_\rho\rho^t$. For example, we can take any $\rho < 1$ greater than the largest spectral radius of the K models, by Gelfand's formula (Kozyakin, 2009).

For each trajectory $i \in [N]$, for $t \geq 0$, we assume that the inputs $u_{i,t}$ are i.i.d. zero-mean isotropic Gaussian random vectors in $\mathbb{R}^m$ with variance $\sigma_u^2 I_m$, and that the process noise $w_t^{(1)} \in \mathbb{R}^m$ and measurement noise $w_t^{(2)} \in \mathbb{R}$ are independent zero-mean subgaussian random vectors with variance proxies $\sigma^2_{w^{(1)}}$, and $\sigma^2_{w^{(2)}}$, respectively. Let $\sigma_w := \max(\sigma_{w^{(1)}}, \sigma_{w^{(2)}})$.

Mixture assumptions. Let $p_{\min} := \min_{k\in[K]} p_k > 0$ be a lower bound on the mixture weights. We also assume a non-degeneracy condition on the vectors of Markov parameters $g_k^{(L)}$. Let $M_2 := \sum_{k=1}^K p_k g_k^{(L)} \otimes g_k^{(L)}$ be the weighted sum of outer products of the mixture components. Then we assume that $\sigma_K(M_2) > 0$. With abuse of notation (as it will be clear from context), we let σ_K denote $\sigma_K(M_2)$. Note that we do not assume a minimum separating distance between pairs of mixture components, but that the non-degeneracy condition does require the number of components $K \leq d$, which is a reasonable setting in many practical applications.

3. Method

Recall that we aim to estimate the parameters $\{(g_k^{(L)}, p_k)\}_{k\in[K]}$ of the mixture model $\mathcal{G}$, which are sufficient to identify $\mathcal{G}$ and to yield minimal state-space realizations of the models when $L \geq 2n+1$. We first identify our problem of estimating Markov parameters $g_i^{(L)} = g_{k_i}^{(L)}$ in MLDS with elements of a linear regression model, but with additional noise and correlated perturbations. To do so, we express y_{it} in the form of a linear regression with coefficients $g_i^{(L)}$. Define the parameter vector $f_i^{(L)} := \left[1, g_i^{(L)\prime}\right]' \in \mathbb{R}^{1+Ln}$, the vector of concatenated inputs from $t-L$ to $t-1$, $\bar{u}_{i,t} := \left[u'_{i,t-1},\ u'_{i,t-2},\ \ldots,\ u'_{i,t-L}\right]' \in \mathbb{R}^{Lm}$, and the vector of concatenated noise variables $\bar{w}_{i,t} = \left[w_{i,t}^{(2)},\ (w_{i,t-1}^{(1)})',\ \cdots,\ (w_{i,t-L}^{(1)})'\right]' \in \mathbb{R}^{1+Ln}$. Then the output y_{it} can be written as

$$y_{it} = \left\langle g_i^{(L)}, \bar{u}_{it} \right\rangle + \left\langle f_i^{(L)}, \bar{w}_{it} \right\rangle + \xi_{it}, \tag{3.1}$$

where we collect the remainder due to the length L truncation of the impulse response in the term

$$\xi_{it} = \sum_{j=L+1}^{t} \left\langle g_i(j), u_{i,t-j} + w_{i,t-j}^{(1)} \right\rangle. \tag{3.2}$$

Since the covariates $\{\bar{u}_{i,t}\}$ in (3.1) are vectors of lagged inputs, covariates that are close in time (e.g., $\bar{u}_{i,t}$ and $\bar{u}_{i,t+1}$) have overlapping entries and are thus dependent. In order to work with independent covariates across observations, which simplifies the later analysis, we simply take every L-th sample starting at time index L. Assume without loss of generality that L divides T (otherwise discard at most $L-1$ samples at the end of the trajectory), and let $\mathcal{J} = \{L, 2L, 3L, ..., T\}$ be an index set of size T/L. Then the vectors of lagged inputs and noise terms indexed by $\mathcal{J}$, $\{\bar{u}_{i,t}, \bar{w}_{it} \mid i \in [N], t \in \mathcal{J}\}$, are mutually independent random vectors. We can thus view the MLDS data set $\{(\bar{u}_{it}, y_{it}) \mid t \in \mathcal{J}, i \in [N]\}$ as being sampled from a mixture of linear regressions with noise and perturbations as formulated in Definition 3.1, with an effective sample size of NT/L, mapping each index $(i,t) \in [N] \times \mathcal{J}$ to a linear index $j \in [NT/L]$.

Algorithm 1: Mixtures of Linear Dynamical Systems Estimator

Input: $\{(u_{i,t-1}, y_{i,t}) \mid t \in [T], i \in [N]\}$ — Input-output trajectories
$\mathcal{N}_2 \cup \mathcal{N}_3 = [N]$ — Index set partition for estimating moments
L — number of Markov parameters to estimate ($L \leq T$)
K — number of mixture components

Output: $\{(\widehat{p}_k, \widehat{g}_k^{(L)})\}_{|k\in[K]}$ — Markov parameters and weights of mixture

$\mathcal{J} \leftarrow \{L, 2L, ..., \lfloor T/L \rfloor\}$

for $i \in [N], t \in \mathcal{J}$ **:**

  $\bar{u}_{i,t} \leftarrow \left[u'_{i,t-1} \quad \cdots \quad u'_{i,t-L}\right]'$ `// Stack inputs`

$\{(\widehat{p}_k, \widehat{g}_k^{(L)})\}_{k\in[K]} \leftarrow$ Algorithm 2$(\{(\bar{u}_{i,t}, y_{i,t}) \mid i \in [N], t \in \mathcal{J}\}, \mathcal{N}_2 \times \mathcal{J}, \mathcal{N}_3 \times \mathcal{J})$
`// Mixture of Linear Regressions`

Definition 3.1 (Mixture of Linear Regression) *Data* $\{(x_i, \tilde{y}_i) \in \mathbb{R}^d \times \mathbb{R} \mid i \in [N]\}$ *is generated by a mixture of linear regressions with noise and perturbations if the output* $\tilde{y}_i$ *can be expressed as*

$$\tilde{y}_i = \langle x_i, \beta_{k_i} \rangle + \eta_i + \xi_i,$$

where the covariates $x_i \overset{iid}{\sim} \mathcal{N}(0, I_d)$ *are independent zero-mean isotropic gaussians, the term* $\eta_i \sim \mathrm{subG}(0, \sigma_\eta^2)$ *represents independent subgaussian noise, and the term* $\xi_i \sim \mathrm{subG}(0, \sigma_\xi^2)$ *represents an additional subgaussian perturbation which may be correlated with the covariates and noise terms* $\{x_j, \eta_j\}_{i \in [N]}$*. The latent variable* k_i *indicates the mixture component with coefficient* $\beta_{k_i} \in \{\beta_k \mid k \in [K]\}$ *that the* i*-th observation belongs to, where* $\mathbb{P}[k_i = k] = p_k$ *for* $k \in [K]$*.*

Note that the noise term $\left\langle f_i^{(L)}, \bar{w}_{it} \right\rangle$ in (3.1) is zero-mean subgaussian with variance proxy $\sigma_w^2 \left\| f_i^{(L)} \right\|^2 \leq \sigma_w^2 \Gamma_{\max}$. The perturbations ξ_{it} are not necessarily independent of the covariates or noise, but they are zero-mean subgaussian and thus can be bounded with high probability. Indeed, when the linear models in the mixture are strictly stable, the effect of past inputs on the present output decreases exponentially in L, the rate of which is captured by ρ. Thus, if L is large enough we can treat the contributions of the remaining Markov parameters and past inputs in $\xi_{i,t}$ as bounded noise.

We complete the mapping of the MLDS problem to the MLR problem in Definition 3.1 by assigning the covariates $x_j \leftarrow \bar{u}_{i,t}/\sigma_u = \begin{bmatrix} u'_{i,t-1} & \cdots & u'_{i,t-L} \end{bmatrix}' / \sigma_u \in \mathbb{R}^{Lm}$, outputs $\tilde{y}_j \leftarrow y_{i,t}$ coefficients $\beta_j \leftarrow \sigma_u g_i^{(L)} \in \mathbb{R}^{Lm}$, independent zero-mean subgaussian noise $\eta_j \leftarrow \left\langle f_i^{(L)}, \bar{w}_{it} \right\rangle$, and subgaussian perturbations $\xi_j \leftarrow \xi_{it}$ as defined in (3.2), again with the mapping of indices $(i, t) \mapsto j \in [NT/L]$. Algorithm 1 constructs the MLR problem in this way, and then uses Algorithm 2 as the key subroutine to obtain Markov parameter estimates of the mixture components. From there, the Ho-Kalman algorithm can be used to obtain state-space realizations for the mixture.

3.1. Mixtures of Linear Regression with Noise and Perturbations

In this section we detail the tensor decomposition approach for solving MLR (Definition 3.1), which is the workhorse of Algorithm 1 for solving MLDS.

Motivation for tensor decomposition. While a matrix, or second-order tensor, M_2 of rank K can be expressed as a sum of K rank-1 matrices, e.g., $M_2 = \sum_{k=1}^K a_k \otimes b_k$, this decomposition is not unique. On the other hand, under mild assumptions, a third-order tensor M_3 of rank K does have a unique decomposition as a sum of K rank-1 tensors (up to scaling and ordering of factors). In the case of a symmetric tensor $M_3 = \sum_{k=1}^K p_k \beta_k^{\otimes 3}$, a sufficient condition for uniqueness of the decomposition is when $\{\beta_k\}_{k \in [K]}$ are linearly independent. Then the set of summands $p_k \beta_k^{\otimes 3}$ is unique, though the scaling between p_k and β_k needs to be resolved separately. If $\{(p_k, \beta_k) \mid k \in [K]\}$ represent the parameters of a mixture, then knowing M_3 would allow us to recover the mixture model through tensor decomposition. Additionally, if we have a noisy estimate of M_3, results on the robustness of tensor decomposition for non-degenerate tensors (Anandkumar et al., 2014) assure us that the estimated components are not too far from the true components.

Estimating MLR. We now extend the moment-based tensor decomposition approach to estimating mixtures of linear regressions that was presented in (Yi et al., 2016; Zhong et al., 2016) and given in Algorithm 2. While these works provide estimation error bounds in the *noiseless* case, in

Section 4 we provide performance guarantees under both i.i.d. noise η_i and bounded perturbations ξ_i, which may be correlated with other variables. To begin our analysis of the MLR algorithm, we examine the moments estimated by Algorithm 2 on the noisy, perturbed linear regression data:

$$\widetilde{M}_2 = \frac{1}{2N_2} \sum_{i \in \mathcal{N}_2} \tilde{y}_i^2 (x_i \otimes x_i - I_d), \text{ and } \widetilde{M}_3 = \frac{1}{6N_3} \sum_{i \in \mathcal{N}_3} \tilde{y}_i^3 \big(x_i^{\otimes 3} - \mathcal{E}(x_i)\big), \tag{3.3}$$

where $\mathcal{E}(x_i) = \sum_{j=1}^d x_i \otimes e_j \otimes e_j + e_j \otimes x_i \otimes e_j + e_j \otimes e_j \otimes x_i$, with e_j the j-th coordinate vector in $\mathbb{R}^d$. Here, $\mathcal{N}_2 \cup \mathcal{N}_3$ is a partition of the set of N trajectories into two disjoint sets of size N_2 and N_3 respectively, which enables us to obtain independent estimates of the matrix M_2 from $\mathcal{N}_2$ and of the third order tensor M_3 from $\mathcal{N}_3$.

Let $y_i := \tilde{y}_i - \xi_i$ be "cleaned" observations. If $\{(y_i, x_i) \mid i \in [N]\}$ were observed, we would be solving a mixture of linear regressions with i.i.d. noise η_i and no perturbations ξ_i: $y_i = \langle \beta_{k_i}, x_i \rangle + \eta_i$. We define the moments estimated with unperturbed y_i:

$$\widehat{M}_2 = \frac{1}{2N_2} \sum_{i \in \mathcal{N}_2} y_i^2 (x_i \otimes x_i - I_d) \text{ and } \widehat{M}_3 = \frac{1}{6N_3} \sum_{i \in \mathcal{N}_3} y_i^3 \big(x_i^{\otimes 3} - \mathcal{E}(x_i)\big). \tag{3.4}$$

It can be verified by multiple applications of Stein's identity (Janzamin et al., 2014) using that x_i is isotropic gaussian and uncorrelated with the zero-mean noise η_i, that $\widehat{M}_2$ and $\widehat{M}_3$ are unbiased estimators of the two mixtures of moment tensors:

$$\mathbb{E}[\widehat{M}_2] = M_2 := \sum_{k=1}^K p_k \beta_k \otimes \beta_k \text{ and } \mathbb{E}[\widehat{M}_3] = M_3 := \sum_{k=1}^K p_k \beta_k \otimes \beta_k \otimes \beta_k.$$

Empirical estimates $\widetilde{M}_2$ and $\widetilde{M}_3$ differ from the unbiased estimators $\widehat{M}_2$ and $\widehat{M}_3$ only by factors related to the perturbations ξ_i. When ξ_i have bounded norm, as in our application for stable dynamical systems, then $\widetilde{M}_2$ and $\widetilde{M}_3$ provide good estimates of the target moments M_2 and M_3.

Whitening factors. Although it is possible to run tensor decomposition on the original estimates of M_3, a useful intermediate step is to whiten the set of d-dimensional tensor factors $\{\beta_k \mid k \in [K]\}$ by projecting them onto the K-dimensional subspace of $\mathbb{R}^d$ spanned by the factors themselves, to yield a set of orthonormal K-dimensional vectors $\{W'\beta_k \mid k \in [K]\}$. Here, $W \in \mathbb{R}^{d \times K}$ is a whitening matrix derived from the singular value decomposition (SVD) of the moment matrix $M_2 = \sum_{k=1}^K p_k \beta_k \otimes \beta_k$. This whitening step is a form of dimensionality reduction; estimating and decomposing the whitened third-order tensor $M_3^W = \sum_{k=1}^K p_k (W'\beta_k)^{\otimes 3} \in (\mathbb{R}^K)^{\otimes 3}$ has lower computational and statistical demands than for the original M_3. Additionally, since the transformed factors $W'\beta_k$ have unit norm, it is possible to disentangle the scaling between p_k and β_k through the dewhitening step (c.f., Lines 8-9 in Algorithm 2). Finally, if the number of mixture components K were unknown, K could also be estimated the SVD of empirical estimates of M_2.

In Algorithm 2, the estimated whitening matrix $\widetilde{W} \in \mathbb{R}^{d \times K}$ is obtained from the SVD of the estimate $\widetilde{M}_2$, such that $\widetilde{W}'\widetilde{M}_2\widetilde{W} = I_K$. The whitened third order tensor $\widetilde{M}_3^{\widetilde{W}}$, which estimates M_3^W, then has orthonormal K-dimensional components, making it amenable to the standard robust tensor power iteration method (Anandkumar et al. (2014)) for orthogonal tensor decomposition. In the last step, the output $\{(\tilde{w}_k, \tilde{\beta}_k) \mid k \in [K]\}$ of the decomposition is dewhitened using $\widetilde{W}$ to return estimates $\widehat{p}_k$ and $\widehat{\beta}_k$ of the original mixture weights and coefficients, for each component $k \in [K]$.

Algorithm 2: Mixture of Linear Regressions Estimator

Input: $\{(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R} \mid i \in [N]\}$ — Regression data
$\mathcal{N}_2 \cup \mathcal{N}_3 = [N]$ — Index set partition for estimating moments
K — number of mixture components

Output: $\{(\widehat{p}_k, \widehat{\beta}_k)\}_{k\in[K]}$ — Estimated mixture parameters and weights.

Whitening:

$\widetilde{M}_2 \leftarrow \frac{1}{2N_2}\sum_{i\in\mathcal{N}_2}\left[y_i^2(x_i^{\otimes 2} - I_d)\right]$ `// 2nd order tensor`

$U\Sigma U^T \leftarrow \text{SVD}(\widetilde{M}_2, K)$ `// Rank-`K` approximation`

$\widetilde{W} \leftarrow U\Sigma^{-1/2}$ `// Whitening matrix`

Tensor estimation and decomposition:

$\widetilde{M}_3^{\widetilde{W}} \leftarrow \frac{1}{6N_3}\sum_{i\in\mathcal{N}_3}\left[y_i^3(\widetilde{W}'x_i)^{\otimes 3} - \mathcal{E}(x_i)(\widetilde{W}, \widetilde{W}, \widetilde{W})\right]$ `// 3rd order tensor`

$\{(\tilde{p}_k, \tilde{\beta}_k) \in \mathbb{R}\times\mathbb{R}^K \mid k \in [K]\} \leftarrow$ Orthogonal Tensor Decomposition$\left(\frac{1}{6}\widetilde{M}_3^{\widetilde{W}}, K\right)$

for $k \in [K]$ **:**

$\quad \widehat{p}_k \leftarrow 1/\tilde{p}_k^2, \quad \widehat{\beta}_k \leftarrow \tilde{p}_k(\widetilde{W}')^{\dagger}\tilde{\beta}_k$ `// Dewhiten`

4. Analysis

Proposition 4.1 provides our main result of finite sample error bounds for learning MLDS via Algorithm 1. Using the mixtures of linear regression subroutine, we essentially run tensor decomposition on a whitened (orthonormal) version of the third-order tensor

$$\frac{L}{6NT}\sum_{i\in[N]}\sum_{t\in\mathcal{J}}\left(y_{i,t}^3\bar{u}_{i,t}^{\otimes 3} - y_{i,t}^3\sum_{k\in[mL]}(e_k\otimes\bar{u}_{i,t}\otimes e_k + \bar{u}_{i,t}\otimes e_k\otimes e_k + e_k\otimes e_k\otimes\bar{u}_{i,t})\right),$$

which estimates $\sum_{k=1}^K p_k(g_k)^{\otimes 3}$. Here, e_k is the k-th coordinate vector in $\mathbb{R}^{mL}$.

Proposition 4.1 *Let data $\mathcal{D} = \{(u_{i,t-1}, y_{i,t}) \mid i \in [N], t \in [T]\}$ be generated from a mixture of linear dynamical systems with parameters $\{(p_k, g_k)\}_{k\in[K]}$, and let L be an integer such that $1 \le L \le T$. Let $\{(\widehat{p}_k, \widehat{g}_k^{(L)})\}_{k\in[K]}$ be the estimated mixture parameters obtained from running Algorithm 1 on the data $\mathcal{D}$. Let $\sigma_y^2 := (\sigma_u^2 + \sigma_w^2)\Gamma_{\max}$. For any $\varepsilon > 0, \delta \in (0,1)$, when*

$$N_2T \gtrsim \frac{\sigma_y^4 L\Gamma_{\max}^3}{\varepsilon^2 p_{\min}^2}\cdot\left(\sigma_K^5\ln^4\left(\frac{N_2T\cdot 9^{Lm}}{\delta L}\right)\ln\left(\frac{9^{Lm}}{\delta}\right) + \frac{\delta}{9^{Lm}}\cdot\frac{\sigma_y^4\Gamma_{\max}^3}{\varepsilon^2 p_{\min}^2}\right),$$

$$N_3T \gtrsim \frac{\sigma_y^6 L}{\varepsilon^2 p_{\min}^2\sigma_K^3}\cdot\left(\ln^6\left(\frac{33^K\cdot N_3T}{\delta L}\right)\ln\left(\frac{33^K}{\delta}\right) + \frac{\delta}{33^K}\cdot\frac{\sigma_y^6}{\varepsilon^2 p_{\min}^2\sigma_K^3}\right),$$

$$\varepsilon \lesssim \frac{\sigma_y^3}{\sigma_K^{3/2}p_{\min}}, \text{ and}$$

$$L \ge \ln\left(\frac{\sigma_y^4\Gamma_{\max}}{\varepsilon^2 p_{\min}^2\sigma_K^3}\cdot\frac{C_\rho\rho}{1-\rho}\left[\Gamma_{\max}^{3/2}\ln^2\left(\frac{9^{Lm}N_2}{\delta}\right) + \sigma_y\ln^3\left(\frac{33^K N_3}{\delta}\right)\right]\right)\cdot\frac{1}{\ln(1/\rho)},$$

it holds that with probability at least $1-\delta$*, there exists a permutation* $\pi : [K] \to [K]$ *such that*

$$\left\| \widehat{g}^{(L)}_{\pi(k)} - g^{(L)}_k \right\|_2 \leq \varepsilon \cdot \frac{\sigma_K^{1/2}}{p_{\min}^{3/2}}, \quad \left| \widehat{p}_{\pi(k)} - p_k \right| < \varepsilon p_k^{3/2}, \quad \textit{for } k \in [K]. \tag{4.1}$$

The proof of Proposition 4.1 is found in Section A.3 of the extended version of this paper (Rui and Dahleh, 2024). It proceeds by first bounding estimation error for MLR with noise and perturbations, and then translates those bounds to estimation error in Markov parameters for MLDS. In more detail, we bound first the deviation of $\widetilde{M}_2$ from M_2, then the estimation error of the whitening matrix $\widetilde{W}$ derived from $\widetilde{M}_2$, and finally the estimation error of $\widetilde{M}_3^{\widetilde{W}}$ from M_3^W. In each step, we use various concentration results and control the effect of the perturbations ξ_i. Note that $\widetilde{M}_2$ and $\widetilde{M}_3$ involve 4th and 6th order moments of the regressor random variables which are effectively d- and K-dimensional, respectively, leading to polynomial dependence on the dimensions and variance parameters.

Next, results on the robustness of the tensor power method (Anandkumar et al., 2014) are applied to transfer bounds on $\left\| \widetilde{M}_3^{\widetilde{W}} - M_3^W \right\|_2$ to the orthonormalized components of the tensor $\widetilde{M}_3^{\widetilde{W}}$, i.e., the whitened projections of the mixture components and their corresponding mixture weights $\{(\tilde{p}_k, \tilde{\beta}_k) \mid k \in [K]\}$. The estimation error of the whitened mixture components is propagated through a dewhitening step, yielding estimates for $\{(\widehat{p}_k, \widehat{\beta}_k) \mid k \in [K]\}$. We obtain Proposition 4.1 by adapting this analysis to estimating $\{(\widehat{p}_k, \widehat{g}^{(L)}_k) \mid k \in [K]\}$ from input-output data.

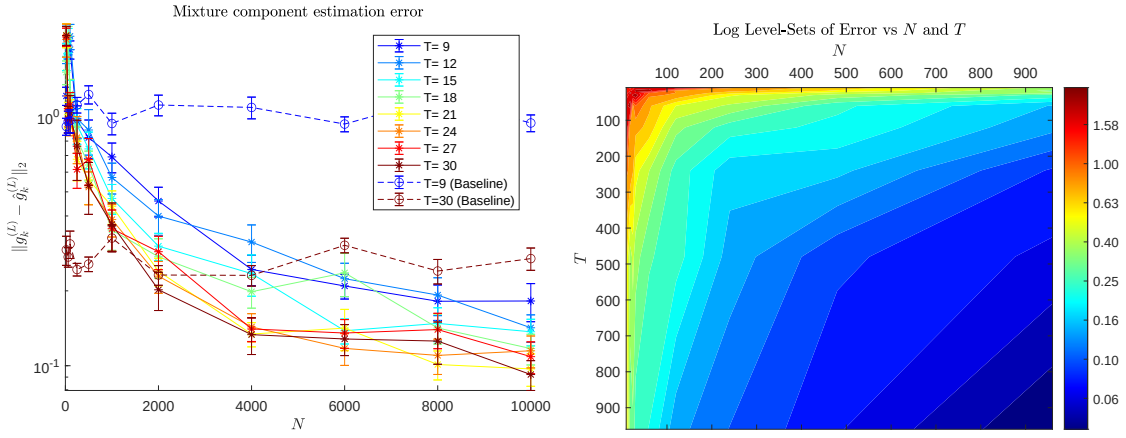
Interpretation of results. Let us rewrite above sample complexity results in Proposition 4.1 in terms of upper bounds on estimation error, for simplicity setting $N_2 = N_3$, ignoring log factors, and keeping only the dependence on N, T, L and ρ. Then we have that given N, T, and L, the estimation error ε of the L impulse response parameters of the mixture components roughly scales as

$$\varepsilon \gtrsim \frac{L^3}{\sqrt{NT}} + L\rho^{L/2}.$$

The first term of the error decreases as $1/\sqrt{NT}$ which is to be expected from solving a linear regression with a sample size of NT. However, to circumvent the dependency structure in the covariates $\bar{u}_{it}$, we take every Lth sample of each trajectory, cutting the effective sample size to NT/L. Furthermore, as we increase L, the dimension Lm of the estimated parameters increases, which enters polynomially into the estimation error bound. The second term of the error, $L\rho^{L/2}$, is due to the tail of the truncated impulse response sequence, corresponding to the perturbations ξ_i in the MLR model (Definition 3.1), and decreases exponentially with L for stable systems. By growing L at a rate of $(NT)^{1/(6+a)}$ as $NT \to \infty$, for $a > 0$, the two components of the estimation error asymptotically decrease to zero.

A particularly interesting property of the tensor decomposition approach to mixtures of linear systems, and which arises in other cases of learning multiple models with latent structure, is the tradeoff between N and T in finite sample error bounds. The setting of observing just a few trajectories, but where each trajectory is long (small N, large T), may yield the same estimation error levels as the setting of observing many short trajectories (large N, small T) from the mixture. The flexibility in sample complexity from assuming and learning a latent structure can prove useful in a wide range of data sets with varying compositions of individual versus collective sample sizes.

5. Simulations



(*a*) (*b*)

Figure 1: Results for estimating the first $L = 7$ Markov parameters of $K = 3$ mixture components. (a) Average parameter estimation error vs. N for various T. Standard errors for 15 trials shown. (b) Level sets of estimation error as a function of N and T.

We evaluate the performance of Algorithm 1 in estimating mixtures of linear systems through a series of simulations. In each trial, $K = 3$ single-input single-output linear models of order $n = 3$ were generated, with spectral radii varying between 0.6 and 0.9. N unlabeled trajectories of length T were sampled from the resulting mixture of K models, for $T \in [9, 960]$ and $N \in [9, 10,000]$. The first $L = 7$ Markov parameters of each mixture component were estimated.

Figure 1 plots the estimation error $(1/K)\sum_{k=1}^{K} \left\|g_k^{(L)} - g_k^{(L)}\right\|_2$ for Algorithm 1, both (a) as a function of N for various T, and (b) as level sets on the (N, T) plane. Additionally, in Figure 1(a), we plot for comparison the error of the "baseline estimator" which estimates Markov parameters individually for each observed trajectory using ordinary least squares (Oymak and Ozay, 2021). The error for the baseline estimator is calculated as $(1/N)\sum_{i=1}^{N} \left\|g_{k_i}^{(L)} - \hat{g}_i^{(L)}\right\|_2$. Although the tensor approach initially has higher error in the small N regime, likely due to the use of higher-order moments, it is able to leverage shared structure across N trajectories to achieve lower estimation error for larger N versus the baseline estimates. This effect is particularly apparent for smaller T, which is a common regime in practical applications.

Figure 1(b) further shows how the performance of the MLDS estimator improves with both N and T. Empirically, we find that the tensor decomposition approach is quite sensitive to the conditioning of the matrix M_2 of Markov parameters, which is related to the degree of non-degeneracy of the mixture parameters. In particular, the norm of the whitening matrix W depends on the smallest singular value of the estimated M_2, which affects the downstream estimation of the whitened third-order tensor M_3^W and the accuracy of the final dewhitened estimates. For future work, it would be interesting to combine the MLDS estimator with iterative mixture estimation methods, which may improve the accuracy and sample complexity of the approach.

References

Animashree Anandkumar, Rong Ge, Daniel J Hsu, Sham M Kakade, Matus Telgarsky, et al. Tensor decompositions for learning latent variable models. *J. Mach. Learn. Res.*, 15(1):2773–2832, 2014.

Brett W. Bader, Tamara G. Kolda, et al. Tensor toolbox for matlab, version 3.6, 9 2023. URL https://www.tensortoolbox.org/.

Ainesh Bakshi, Allen Liu, Ankur Moitra, and Morris Yau. A new approach to learning linear dynamical systems. *arXiv preprint arXiv:2301.09519*, 2023a.

Ainesh Bakshi, Allen Liu, Ankur Moitra, and Morris Yau. Tensor decompositions meet control theory: learning general mixtures of linear dynamical systems. In *International Conference on Machine Learning*, pages 1549–1563. PMLR, 2023b.

Yanxi Chen and H Vincent Poor. Learning mixtures of linear dynamical systems. In *International Conference on Machine Learning*, pages 3507–3557. PMLR, 2022.

Jason Ernst, Gerard J Nau, and Ziv Bar-Joseph. Clustering short time series gene expression data. *Bioinformatics*, 21(suppl_1):i159–i168, 2005.

William B Gragg and Anders Lindquist. On the partial realization problem. *Linear Algebra and its Applications*, 50:277–319, 1983.

BL Ho and Rudolf E Kálmán. Effective construction of linear state-variable models from input/output functions: Die konstruktion von linearen modeilen in der darstellung durch zustandsvariable aus den beziehungen für ein-und ausgangsgrößen. *at-Automatisierungstechnik*, 14 (1-12):545–548, 1966.

Majid Janzamin, Hanie Sedghi, and Anima Anandkumar. Score function features for discriminative learning: Matrix and tensor framework. *arXiv preprint arXiv:1412.2863*, 2014.

Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3): 455–500, 2009.

Victor Kozyakin. On accuracy of approximation of the spectral radius by the gelfand formula. *Linear Algebra and its Applications*, 431(11):2134–2141, 2009.

Yuanzhi Li and Yingyu Liang. Learning mixtures of linear regressions with nearly optimal complexity. In *Conference On Learning Theory*, pages 1125–1144. PMLR, 2018.

Aditya Modi, Mohamad Kazem Shirani Faradonbeh, Ambuj Tewari, and George Michailidis. Joint learning of linear time-invariant dynamical systems. *arXiv preprint arXiv:2112.10955*, 2021.

Samet Oymak and Necmiye Ozay. Revisiting ho–kalman-based system identification: Robustness and finite-sample analysis. *IEEE Transactions on Automatic Control*, 67(4):1914–1928, 2021.

Maryann Rui and Munther Dahleh. Finite sample analysis of tensor decomposition for learning mixtures of linear systems. *arXiv preprint arXiv:2412.10615*, 2024.

Tuhin Sarkar, Alexander Rakhlin, and Munther A Dahleh. Finite time lti system identification. *The Journal of Machine Learning Research*, 22(1):1186–1246, 2021.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Xinyang Yi, Constantine Caramanis, and Sujay Sanghavi. Alternating minimization for mixed linear regression. In *International Conference on Machine Learning*, pages 613–621. PMLR, 2014.

Xinyang Yi, Constantine Caramanis, and Sujay Sanghavi. Solving a mixture of many random linear equations by tensor decomposition and alternating minimization. *arXiv preprint arXiv:1608.05749*, 2016.

Thomas T Zhang, Katie Kang, Bruce D Lee, Claire Tomlin, Sergey Levine, Stephen Tu, and Nikolai Matni. Multi-task imitation learning for linear dynamical systems. In *Learning for Dynamics and Control Conference*, pages 586–599. PMLR, 2023.

Yang Zheng and Na Li. Non-asymptotic identification of linear dynamical systems using multiple trajectories. *IEEE Control Systems Letters*, 5(5):1693–1698, 2020.

Kai Zhong, Prateek Jain, and Inderjit S Dhillon. Mixed linear regression with multiple components. *Advances in neural information processing systems*, 29, 2016.