Extracting Latent State Representations with Linear Dynamics from Rich Observations

Abraham Frandsen [1] **Rong Ge** [1] **Holden Lee** [2]

Abstract

Recently, many reinforcement learning techniques have been shown to have provable guarantees in the simple case of linear dynamics, especially in problems like linear quadratic regulators. However, in practice many tasks require learning a policy from rich, high-dimensional features such as images, which are unlikely to be linear. We consider a setting where there is a hidden linear subspace of the high-dimensional feature space in which the dynamics are linear. We design natural objectives based on forward and inverse dynamics models. We prove that these objectives can be efficiently optimized and their local optimizers extract the hidden linear subspace. We empirically verify our theoretical results with synthetic data and explore the effectiveness of our approach (generalized to nonlinear settings) in simple control tasks with rich observations.

1. Introduction

Reinforcement learning has made tremendous progress recently, achieving strong performance in difficult problems like go (Silver et al., 2017) and Starcraft (Vinyals et al., 2019). A common theme in the recent progress is the use of neural networks to handle the cases when the system dynamics and policy are both highly nonlinear. However, theoretical understanding for reinforcement learning is most thoroughly developed in the tabular setting (where the number of state/actions is small) or when the underlying dynamics of the system is linear (see Section 1.1).

Requiring the dynamics to be linear is especially limiting for problems with *rich*, high dimensional output, e.g. manipulating a robot from video frames or playing a game by observing pixel representations. Consider a simple system where we control an object by applying forces to it. The state of the object (position and velocity) can evolve linearly according to physical laws. However, if the observation is a visual rendering of this object in a 3-d environment, the observation contains a lot of redundant information and doesn't have linear dynamics. Such problems can potentially be solved by learning a *state representation mapping* ϕ that maps the complicated observations to states that satisfy simpler dynamics. State representation learning is popular in practice, but theoretical understanding is still nascent; see the survey by Lesort et al. (2018) and more references in Section 1.1. Many approaches either try to learn a *forward model*, which predicts the next state or an *inverse model*, which predicts the action taken given the states. In this paper we show both approaches can provably extract a state representation that encodes linear dynamics.

We first consider a simple theoretical model where the full observation x does not have linear dynamics, but there exists an unknown subspace $\mathcal{V}$ where the projection $y = \Pi_{\mathcal{V}} x$ has linear dynamics. This corresponds to the case when the state representation mapping is a linear projection. We give two provably correct algorithms for identifying $\mathcal{V}$, one based on learning a linear forward model, and one based on learning a linear inverse model.

In more complicated settings one might need a nonlinear mapping in order to extract a latent space representation that has linear dynamics. We extend our algorithms to the nonlinear setting, and show that if we can find solutions to similar nonconvex optimization problems with 0 loss, then the representations have nontrivial linear dynamics.

We discuss related works in Section 1.1. We next introduce our model and discuss how one can formalize learning a state representation mapping as optimization problems in Section 2. In Sections 3 and 4 we give algorithms based on forward and inverse models, respectively, and prove that these recover the underlying state representation. We then extend these to nonlinear state representations in Section 5. Finally in Section 6, we empirically validate our approach on synthetic data and simple RL environments. All detailed proofs are found in the appendix.

[1]Department of Computer Science, Duke University, Durham, North Carolina, USA [2]Department of Mathematics, Duke University, Durham, North Carolina, USA. Correspondence to: Abraham Frandsen <abef@cs.duke.edu>, Rong Ge <rongge@cs.duke.edu>, Holden Lee <holden.lee@duke.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

1.1. Related Work

State Representation Learning with Rich Observations Several recent papers have addressed the problem of state representation learning (SRL) in control problems. Lesort et al. (2018) survey the recent literature and identify four categories that describe many SRL approaches: reconstructing the observation, learning a forward dynamics model, learning an inverse dynamics model, and using prior knowledge to constrain the state space. Raffin et al. (2019) evaluate many of these SRL approaches on robotics tasks and show how to combine the strengths of the different methods. Several papers adopt the approach of learning forward or inverse models (Hafner et al., 2019b; Pathak et al., 2017; Zhang et al., 2018) and demonstrate practical effectiveness, but they lack a theoretical analysis of the approach. Our work aims to help fill this gap.

Domains with rich observations include raw images or video frames from video games (Anand et al., 2019), robotics environments (Higgins et al., 2017), and renderings of classic control problems (Watter et al., 2015), and deep learning methods have enabled success in this space. Srinivas et al. (2020) use a contrastive learning approach to extract state representations from pixels. Ha & Schmidhuber (2018) learn low-dimensional representations and dynamics which simple linear policies to achieve effective control. Hafner et al. (2019a) utilize latent imagination to learn behaviors that achieve high performance in terms of reward and sample-efficiency on several visual control tasks.

Theoretical work on state representation learning Du et al. (2019a) investigate whether good representations lead to sample-efficient RL in the context of MDPs, showing exponential lower bounds in many settings. Our setting circumvents these negative results because the representations that we learn transform the nonlinear problem into a linear (and hence tractable) control problem. Other works (Du et al., 2019b; Misra et al., 2019) study the Block MDP model, in which a high-dimensional observation space is generated from a finite set of latent states. Our model, by contrast, considers continuous state and action spaces.

Recent work by Mhammedi et al. (2020) studies a similar setting to ours, where a latent LQR control problem generates nonlinear observations. That work is interested in finding a near-optimal controller with respect to the quadratic costs, whereas we focus here on finding the ground-truth representation that has linear dynamics. We give both a forward and an inverse approach while Mhammedi et al. (2020) focuses on learning inverse models. Recent work by Dean et al. (2020) also consider a somewhat similar setting, although their focus is on robust control guarantees, and their state representations are learned through a supervised technique that requires ground-truth representations.

Linear Dynamical Systems and Control Problems Linear dynamical systems and control problems have been extensively studied for many decades and admit efficient, robust, and provably correct algorithms. For the problem of system identification, Qin (2006) gives a review of subspace identification methods; our inverse model approach, while for a different setting, is somewhat similar to the regression approaches described in the review. Other recent works analyze gradient-based methods for system identification (Hardt et al., 2018), policy optimization (Fazel et al., 2018), and online control (Cohen et al., 2018) in the setting of linear dynamical systems and quadratic costs.

2. Hidden Subspace Model and State Representation Learning

In this section we introduce a basic model which admits a linear state representation mapping. Later we show that the linear state representation can be learned efficiently via either a forward or inverse modelling approach.

2.1. Notation and Preliminaries

We follow the notation of a discrete-time control system, where x_t denotes the (observed) state at the t-th step, u_t denotes the control signal (action) at the t-th step, and f denotes the dynamics function, $x_{t+1} = f(x_t, u_t)$. A *state representation mapping* is a function $\phi : \mathbb{R}^d \to \mathbb{R}^r$ that maps x_t to a different space (usually $r \ll d$) such that the dynamics governing the evolution of $\phi(x_t)$ are simpler, e.g., linear.

If $a_1, \ldots, a_k$ are vectors, we let $(a_1, \ldots, a_k)$ denote the concatenation of these vectors. For centered random vectors a and b, let Σ_{ab} denote the matrix $\mathbb{E}[ab^\top]$ and Σ_a denote $\mathbb{E}[aa^\top]$. Likewise, if $A \in \mathbb{R}^{m\times k}$ and $B \in \mathbb{R}^{n\times k}$, let Σ_{AB} denote the empirical cross-covariance matrix $k^{-1}AB^\top$ and let Σ_A denote the empirical covariance matrix $k^{-1}AA^\top$. Let $\rho(a, b)$ denote the canonical correlation between a and b, namely

$$\rho(a, b) = \max_{a', b'} \frac{\mathbb{E}[\langle a, a'\rangle\langle b, b'\rangle]}{\sqrt{\mathbb{E}[\langle a, a'\rangle^2]\mathbb{E}[\langle b, b'\rangle^2]}}.$$

For a matrix A, let A^+ denote its Moore-Penrose pseudoinverse, and let $\mathrm{col}(A)$ denote the column-space of A. For a linear subspace $\mathcal{C} \subset \mathbb{R}^n$, let $\Pi_{\mathcal{C}} \in \mathbb{R}^{n\times n}$ denote the orthogonal projection matrix onto $\mathcal{C}$, and let $\mathcal{C}^\perp$ denote the orthogonal complement of $\mathcal{C}$.

We adapt the standard notion of controllability from control systems theory to be subspace-dependent.

Definition 2.1. Given matrices $A \in \mathbb{R}^{d\times d}$, $B \in \mathbb{R}^{d\times l}$, and an r-dimensional subspace $\mathcal{V} \subset \mathbb{R}^d$, we say that the tuple (A, B) is *$\mathcal{V}$-controllable* if $\mathrm{col}(A), \mathrm{col}(A^\top), \mathrm{col}(B) \subset \mathcal{V}$

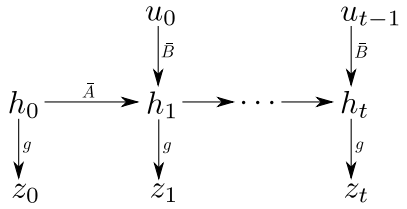

Figure 1. A graphical illustration of our hidden subspace model. The hidden states h_i evolve according to a linear control system and generate nonlinear features z_i.

and the $d \times rl$ matrix

$$\begin{bmatrix} B & AB & \cdots & A^{r-1}B \end{bmatrix}$$

has rank r.

2.2. Hidden Subspace Model

We consider a model with a latent ground truth state $h_t \in \mathbb{R}^r$ and controls $u_t \in \mathbb{R}^l$ that satisfy linear dynamics:

$$h_{t+1} = \bar{A}h_t + \bar{B}u_t.$$

We observe a high-dimensional state $x_t \in \mathbb{R}^d$ with $d > r$ that satisfies $x_t = Vh_t + V^\perp g(h_t)$. Here, $V \in \mathbb{R}^{d\times r}$ and $V^\perp \in \mathbb{R}^{(d-r)\times r}$ are full-rank matrices whose columns respectively form bases for an r-dimensional subspace $\mathcal{V}$ and its orthogonal complement $\mathcal{V}^\perp$, and $g(h_t) \in \mathbb{R}^{d-r}$ is a nonlinear, possibly stochastic function of h_t. We use y_t to denote Vh_t and z_t to denote $V^\perp g(h_t)$, and we call these the *linear* and *nonlinear* parts of x_t, respectively. The model is illustrated in Figure 1.

Observe that y_t also satisfies linear dynamics, namely $y_{t+1} = Ay_t + Bu_{t-1}$, where $A = V\bar{A}V^+$ and $B = V\bar{B}$. On the other hand, z_t is conditionally independent of all other variables $\{u_i : i \geq 0\}$, $\{y_i : i \neq t\}$, and $\{z_i : i \neq t\}$ given h_t. Thus, we can write x_t as a sum of two orthogonal components, $x_t = y_t + z_t$, where y_t evolves linearly and z_t contains the nonlinear, redundant features. The constraint that the linear and nonlinear parts of x_t lie in mutually orthogonal subspaces enables one to project away z_t and recover the linear part, if $\mathcal{V}$ is known. Our task is to extract the latent state h_t (or any invertible linear transformation thereof) from x_t, given observed trajectories $x_0, x_1, x_2, \ldots$ and controls $u_0, u_1, \ldots$. To find this mapping it suffices to recover the hidden subspace $\mathcal{V}$.

Throughout this section and Sections 3 and 4, we assume that the initial latent state h_0 and the controls u_i are independent standard Gaussian random vectors, that $\mathbb{E}[z_i] = 0$ for each i, and that Σ_{x_i} is full rank for each i. All expectations are taken over the randomness induced by h_0, u_i, and z_i.

2.3. Learning Forward and Inverse Models

In order to learn a state representation mapping that induces linear dynamics, a natural approach is to jointly learn the representation and a dynamics model that enforces linearity. Suppose we take random actions $u_0, u_1, \ldots$ from a random initial state x_0, generating a trajectory of observations $x_1, x_2, \ldots$. We could attempt to learn a mapping ϕ and matrices $C \in \mathbb{R}^{r\times r}, D \in \mathbb{R}^{r\times l}$ such that the forward dynamics equation is linear: $\phi(x_{t+1}) = C\phi(x_t) + Bu_t$. Alternatively, we could instead seek to learn ϕ and matrices $P, L \in \mathbb{R}^{l\times d}$ such that the inverse dynamics equation is linear: $u_t = P\phi(x_{t+1}) - L\phi(x_t)$.

We show how both of these ideas can be carefully implemented to identify $\mathcal{V}$ in the hidden subspace model. In particular, we propose the *forward model objective*

$$\min_{P,Q,D} \frac{1}{2}\mathbb{E}\|Px_1 - Qx_0 - Du_0\|_2^2 + \frac{\lambda}{4}\|P\Sigma_{x_1}P^\top - I\|_F^2$$

as well as the *inverse model objective*

$$\min_{P,\{L_i\},\{T_i\}} \frac{1}{2}\mathbb{E}\sum_{i=1}^{r}\|Px_i - L_ix_0 - \sum_{k=1}^{i-1}T_ku_{i-1-k} - u_{i-1}\|_2^2.$$

While both of these approaches are viable in the hidden subspace model, it's worth noting how they differ from each other. The forward model objective is non-convex, only requires one step of the control system, and it immediately yields the state representation map P. By contrast, the inverse model is a convex optimization problem, it considers trajectories of length r, and the final state representation map is constructed from $P, L_1, \ldots, L_r$. The two approaches also require different assumptions for their theoretical guarantees. In Sections 3 and 4 we motivate and analyze these two approaches in more detail.

3. Forward Model

In this section, we focus on the forward model, which tries to predict the next state given the current state and action. We first motivate the design of forward objective function (1) and prove its guarantees. We next explain how the forward model objective is connected to canonical correlation analysis. We finally study the sample complexity of the empirical version of the problem.

Recall the forward model objective

$$\min_{P,Q,D} \frac{1}{2}\mathbb{E}\|Px_1 - Qx_0 - Du_0\|_2^2 + \frac{\lambda}{4}\|P\Sigma_{x_1}P^\top - I\|_F^2 \quad (1)$$

where $P, Q \in \mathbb{R}^{r\times d}$, $D \in \mathbb{R}^{r\times l}$, and $\lambda > 0$. To motivate (1), note that a first attempt to learn a linear forward model is to find matrices P, C, D that satisfy $Px_1 = CPx_0 + Du_0$.

This, however, immediately runs into the problem of trivial solutions – we can choose these matrices to be all zero and the linear dynamics equation holds, but this is clearly not a useful representation and it doesn't recover the subspace $\mathcal{V}$.

A simple way to rule out such trivial solutions is to constrain the state representation Px_1 to have a full-rank covariance matrix. Dealing with non-convex rank constraints directly can be difficult, so we instead introduce a regularizer term $\|\mathbb{E}[Px_1x_1^\top P] - I\|_F^2$ which encourages Px_1 to have spherical covariance. Additionally, we relax the forward dynamics model to a simpler linear model $Px_1 = Qx_0 + Du_0$. This relaxation removes some non-convexity from the objective function and simplifies the analysis. Both of these adjustments lead to (1). This objective is still non-convex due to the regularizer term, but its landscape is benign, as explained in Theorem 3.2.

In order to ensure that solutions to (1) recover $\mathcal{V}$, we specify the nonlinearity of z_1 as follows:

Assumption 3.1 (Forward Nonlinearity)**.** There exists a constant $\rho \in (0, 1)$ such that $\rho(z_1, (x_0, u_0)) \leq \rho$.

This assumption simply asserts that z_1 is not linearly dependent on the initial data x_0, u_0. We can now state the theoretical guarantees for the forward model.

Theorem 3.2. *Set $\lambda \in (0, 1 - \rho^2)$, and let (P, Q, D) be a second-order stationary point of (1). Under Assumption 3.1, col$(P^\top) = \mathcal{V}$.*

A variety of local search algorithms (such as perturbed gradient descent (Jin et al., 2017)) are proven to efficiently find second-order stationary points, and so Theorem 3.2 implies that we can efficiently recover $\mathcal{V}$ by optimizing (1). Intuitively, Theorem 3.2 holds because if the rows of P aren't all in $\mathcal{V}$, then Pz_1 is nonzero. However, Assumption 3.1 implies that z_1 is *not* a linear function of x_0 and u_0, so Pz_1 contributes excess loss in the first term of (1), so it should be removed. Moreover, if P is rank-deficient, we can reduce the second term of (1) by increasing the rank of P while ensuring its rows stay in $\mathcal{V}$. Our proof makes this intuition precise by first showing that Assumption 3.1 implies a gap in the canonical correlations between x_1 and (x_0, u_0), and then utilizing Theorem 3.3. Full details are in the appendix.

3.1. Connection to CCA

We now analyze the forward model objective in a more general setting and draw connections to canonical correlation analysis. Consider two random vectors $u \in \mathbb{R}^n$, $v \in \mathbb{R}^m$ with full-rank covariance matrices Σ_u and Σ_v. Canonical correlation analysis deals with finding the directions of maximal correlation between u and v. To this end, we propose the optimization problem

$$\min_{P,Q} \frac{1}{2}\mathbb{E}_{u,v}\|Pu - Qv\|_2^2 + \frac{\lambda}{4}\|P\Sigma_u P^\top - I\|_F^2 \quad (2)$$

where $P \in \mathbb{R}^{r\times n}$, $Q \in \mathbb{R}^{r\times m}$, and $\lambda \in (0, 1)$ is a hyperparameter. Define $C = \Sigma_u^{-1/2}\Sigma_{uv}\Sigma_v^{-1}\Sigma_{vu}\Sigma_u^{-1/2}$ and write its spectral decomposition as $C = \sum_{i=1}^d \rho_i^2 c_i c_i^\top$ where $1 \geq \rho_1 \geq \cdots \geq \rho_d \geq 0$ and $\|c_i\|_2 = 1$ for each i. According to CCA theory (Borga, 2001), ρ_i are the canonical correlations between u and v, and the vectors $\Sigma_u^{-1/2}c_i$ are the corresponding canonical correlation directions for u, i.e., the directions in which u maximally correlates with v.

For each $i \in \{1, \ldots, d\}$, define $\mathcal{C}_i = \text{span}\{c_1, \ldots, c_i\}$. Let $\mathcal{C}_0$ denote the trivial subspace $\{0\}$ and define $\rho_0 = 1, \rho_{d+1} = 0$. The subspaces $\mathcal{C}_i$ are useful because they allow us to project u to a lower-dimensional subspace while maximally preserving its correlation with v. Solving the optimization problem (2) recovers these subspaces, depending on the value of λ.

Theorem 3.3. *Let $i \in \{0, 1, \ldots, d\}$ satisfy $1 - \rho_i^2 < \lambda < 1 - \rho_{i+1}^2$. Let (P, Q) be a second-order stationary point of (2). Then col$(\Sigma_u^{1/2}P^\top) \subset \mathcal{C}_i$ and rank$(P) = \min\{r, i\}$.*

To prove Theorem 3.3, we first analyze the first-order necessary conditions of (2), which are closely connected to the matrix C. In particular, the gradients vanish only if the rows of $P\Sigma_u^{1/2}$ are contained in $\mathcal{C}_i$. Next, we show that the loss function can be additionally minimized if rank$(P) < \min\{r, i\}$. In particular, by carefully increasing the rank of P we can ensure that the regularizer term decreases more than the linear model term increases.

3.2. Sample Complexity

Since (1) involves an expectation, we can't solve it exactly. Instead, we optimize the empirical objective function

$$\min_\theta \frac{1}{2n}\|PX_1 - QX_0 - DU_0\|_F^2 + \frac{\lambda}{4}\|P\Sigma_{X_1}P^\top - I\|_F^2 \quad (3)$$

where the columns of $X_i \in \mathbb{R}^{d\times n}$ and $U_i \in \mathbb{R}^{l\times n}$ are i.i.d. copies of x_i and u_i, respectively. If n is sufficiently large, the solution to this problem recovers $\mathcal{V}$. We introduce the following assumption that allows us to utilize quantitative concentration results, and then we state the theoretical guarantee for (3).

Assumption 3.4 (Sub-Gaussianity)**.** There exists a constant $C > 0$ such that $P(|\langle q, \Sigma_{\xi_i\xi_i}^{-1/2}\xi_i\rangle| > t) \leq \exp(-Ct^2)$ for any unit vector q, where we define $\xi_i := (z_1, x_0, u_0)$.

Theorem 3.5. *Set $\lambda \in (0, (1-\rho^2)/4)$, and let (P, Q, D) be a second-order stationary point of (3). Under Assumptions 3.1 and 3.4, there exists a constant C_0 such that if $n \geq C_0 \log^2(2d + l)/(1 - \rho)^2$, then with probability at least 0.99, col$(P^\top) = \mathcal{V}$.*

To prove this theorem, we first use a concentration of measure result to establish that the empirical canonical correlations between X_1 and (X_0, U_0) have a sufficient gap, and then the proof of Theorem 3.3 does the rest of the work.

4. The Inverse Model

In this section, we focus on the inverse model, whose goal is to predict action based on the state representations. We show that this approach efficiently learns the linear state representation in our hidden subspace model when certain assumptions are satisfied. We also study the sample complexity of this problem when we only have i.i.d. samples from the model. In the appendix we study a simplified version of the model where there is noise.

Recall the inverse model objective

$$\min_{\theta} \frac{1}{2}\mathbb{E}\sum_{i=1}^{r} \|Px_i - L_i x_0 - \sum_{k=1}^{i-1} T_k u_{i-1-k} - u_{i-1}\|_2^2 \quad (4)$$

Here, θ is the tuple of parameters $(P, L_1, \ldots, L_r, T_1, \ldots, T_{r-1})$ with $P, L_i \in \mathbb{R}^{l\times d}$ and $T_i \in \mathbb{R}^{l\times l}$, and the expectation is taken over the randomness of $x_0, u_0, \ldots, u_{r-1}$. To motivate (4), we start by considering one step of the dynamics:

$$x_1 = y_1 + z_1 = Ay_0 + Bu_0 + z_1 = Ax_0 + Bu_0 + z_1.$$

If B has full column rank, then we have $B^+B = I$ and $B^+z_1 = 0$ since the rows of B^+ are in $\mathcal{V}$. Hence, we have $u_0 = B^+x_1 - B^+Ax_0$. This expression suggests that if we fit a linear model to predict u_0 given x_0 and x_1, the solution may allow us to recover B^+ and B^+A, both of which reveal part of the latent subspace $\mathcal{V}$. Advancing the system up to timestep i, we have a similar relationship:

$$u_{i-1} = B^+x_i - B^+A^ix_0 - \sum_{k=1}^{i-1} B^+A^kBu_{i-1-k}.$$

Once again, if we fit a linear model to predict u_{i-1} from $x_i, x_0, u_0, \ldots, u_{i-2}$, then we can recover more of $\mathcal{V}$.

Trying to solve for A and B directly by minimizing a squared error loss based on the above expression is problematic, given the presence of high powers of A and products between A, B, and B^+. The optimization landscape corresponding to such an objective function is non-convex and ill-conditioned. To circumvent this issue, we propose the *convex relaxation*:

$$u_{i-1} = Px_i - L_ix_0 - \sum_{k=1}^{i-1} T_k u_{i-1-k}$$

Here, P corresponds to B^+, L_i to B^+A^i, and T_k to B^+A^kB. We arrive at (4) by fitting this inverse model over a trajectory of length r, which is chosen so that we can recover the entirety of $\mathcal{V}$.

In order to state our theoretical guarantees for this approach, we introduce a few assumptions.

Assumption 4.1 (No Linear Dependence)**.** Let h_0 and $u_0, \ldots, u_{i-1}$ be independent standard Gaussian vectors. Then $\mathbb{E}[z_i] = 0$ and there is a constant $0 \le \rho < 1$ such that for each $i = 1, \ldots, r$, $\rho((z_i, z_0), (h_i, h_0)) \le \rho$.

Remark 4.2. Assumption 4.1 concretely specifies the nonlinearity of z_i, as it precludes any linear dependence between z_i and the controls. Without this assumption, the inverse model that we learn may use information from $\mathcal{V}^\perp$ to predict the controls, and it is impossible to uniquely recover $\mathcal{V}$.

Assumption 4.3 (Controllability)**.** The tuple $(A^\top, (B^+)^\top)$ is $\mathcal{V}$-controllable.

Remark 4.4. Assumption 4.3 is related to the standard controllability property of linear control systems. Instead of assuming (A, B) controllability, we need the property to hold for $(A^\top, (B^+)^\top)$ since we are learning an inverse model which is related to the matrices $B^+, B^+A, \ldots, B^+A^r$.

Assumption 4.5 (Non-degeneracy)**.** ,The matrix B has linearly independent columns, i.e. rank$(B) = l$.

Remark 4.6. Assumption 4.5 allows us to learn the inverse model. If B is rank-deficient, we could not hope to predict even u_0 from x_0 and x_1, since it is non-identifiable. One interpretation of this assumption is that the control inputs u_i are well-specified, i.e., not redundant.

Observe that (4) is a convex optimization problem, but there may not be a unique global minimizer due to redundancies in the parametrization. While the set of global optimizers is in general a linear subspace, by imposing certain norm preferences we can still recover the intended solutions B^+, B^+A_i and B^+A^kB. We now state the theoretical guarantee for our algorithm.

Theorem 4.7. *Let f be the objective function in (4), and let $\Theta_0^* = \{\theta = (P, \{L_i\}_{i=1}^r, \{T_i\}_{i=1}^{r-1}) \in f^{-1}(0) \mid \|P\|_F$ is minimal$\}$ be the set of optimal solutions to (4) that have minimal norm for P. Let $\theta^* = (P^*, \{L_i^*\}, \{T_i^*\}) \in \Theta_0^*$ be the solution in this set that minimizes $\sum_{i=1}^r \|L_i\|_F^2$. Then under assumptions 4.1, 4.3, and 4.5, $P = B^+$ and $L_i = B^+A^i$ for $i = 1, \ldots, r$. Moreover, $\mathcal{V} = col(P^\top) + col(L_1^\top) + \cdots + col(L_r^\top)$.*

Remark 4.8. To find the desired solution, we can first find the set $\Theta^* = f^{-1}(0)$ of global minimizers. For such linear systems, Θ^* is a subspace, so Θ_0^* can be obtained by optimizing for the norm of P within this subspace.

Intuitively, Theorem 4.7 is correct because by Assumption 4.1, any direction in $\mathcal{V}^\perp$ will not have a perfect linear correlation with the control signal u_i that we are trying to predict. This does not mean that every optimal solution to

Equation (4) has components only in $\mathcal{V}$ – it is still possible that components in $\mathcal{V}^\perp$ cancel each other. However, if any of the matrices P, L_i have components in $\mathcal{V}^\perp$, removing those components will reduce the norm of the matrices while not changing the predictive accuracy. Therefore the minimum norm solution must lie in the correct subspace. Finally, the fact that we recover the entirety of $\mathcal{V}$ follows from Assumption 4.3. Our proof makes this intuition precise by analyzing the first-order optimality conditions of (4) and making use of a spectral characterization of Assumption 4.1. The detailed proof is deferred to the appendix, where we also adapt this result to a simplified case where there is noise in the linear dynamics.

4.1. Sample Complexity

As with the forward model, in practice we can only solve the empirical inverse model objective

$$\min_\theta \frac{1}{2n}\sum_{i=1}^{r}\|PX_i - L_iX_0 - \sum_{k=1}^{i-1}T_kU_{i-1-k} - U_{i-1}\|_F^2 \quad (5)$$

Here, the columns of $X_i \in \mathbb{R}^{d\times n}, U_i \in \mathbb{R}^{l\times n}$ are i.i.d. copies of x_i and u_i, respectively. We again introduce an assumption that allows us to utilize quantitative concentration results, and then we state the sample complexity result.

Assumption 4.9 (Sub-Gaussianity)**.** There exists a constant $C > 0$ such that for each $i \in \{1,\dots,r\}$, $P(|\langle q, \Sigma_{\xi_i\xi_i}^{-1/2}\xi_i\rangle| > t) \le \exp(-Ct^2)$ for any unit vector q, where we define $\xi_i := (z_i, z_0, h_i, h_0)$.

Theorem 4.10. *Let f be the objective function in (5), and let $\Theta_0^* = \{\theta = (P, \{L_i\}_{i=1}^r, \{T_i\}_{i=1}^{r-1}) \in f^{-1}(0) \mid \|P\|_F$ is minimal$\}$ be the set of optimal solutions to (4) that have minimal norm for P. Let $\theta^* = (P^*, \{L_i^*\}, \{T_i^*\}) \in \Theta_0^*$ be the solution in this set that minimizes $\sum_{i=1}^r \|L_i\|_F^2$. Under assumptions 4.1, 4.3, 4.5, and 4.9, there exists a constant C_0 such that if $n \ge C_0(d+rl)\log r \log^2(d+rl)/(1-\rho)^2$, then with probability at least 0.99, $P = B^+$ and $L_i = B^+A^i$ for $i = 1,\dots,r$.*

Our proof for Theorem 4.10 is similar to that of Theorem 4.7 but requires somewhat more care. We use additional concentration of measure results in order to ensure that the empirical canonical correlation $\rho((Z_i, Z_0), (H_i, H_0))$ is close to its population value. The first-order optimality conditions of (5) also contain additional empirical cross-covariance terms that must be handled. Full details are deferred to the appendix.

5. Nonlinear State Representation Learning

In this section, we extend the forward model and inverse model objectives to the setting in which there are no latent linear dynamics in the original state observations. In this case, we try to learn a nonlinear state representation ϕ under which the dynamics are nearly linear. An example of this setting is when the state observations are raw pixels from a camera and ϕ is a convolutional neural network.

For the forward model objective, we introduce an intermediate feature map $\psi : \mathbb{R}^d \to \mathbb{R}^{d'}$ and fit the forward model to the transformed states $\psi(x_0), \psi(x_1)$. The resulting optimization problem is

$$\min_\theta \mathbb{E}\|P\psi(x_1) - Q\psi(x_0) - Du_0\|_2^2 + \lambda\|P\Sigma_{\psi(x_1)}P^\top - I\|_F^2 \quad (6)$$

where θ is the tuple of parameters (ψ, P, Q, D). The final state representation map ϕ is given by $\phi(x) = P\psi(x)$.

For the inverse model objective, we again simply fit the inverse model to the transformed states $\psi(x_i)$:

$$\min_\theta \frac{1}{2}\mathbb{E}\sum_{i=1}^{\tau}\|P\psi(x_i) - L_i\psi(x_0) - \sum_{k=1}^{i-1}T_ku_{i-1-k} - u_{i-1}\|_2^2 \quad (7)$$

Although unlikely to be obtained in practice, we verify that if the 0 loss is achieved, then we can extract a nontrivial linear control system.

Theorem 5.1. *Let $\psi, P, \{L_i, T_i\}, i = 1,\dots,\tau$ be optimal solutions to the optimization problem (7), and assume that these parameters incur zero loss. Define $\mathcal{V} = col(P^\top) + col(L_1^\top) + \cdots + col(L_{\tau-1}^\top)$, and assume that $col(L_\tau^\top) \subset \mathcal{V}$. Let $\phi(x) = \Pi_{\mathcal{V}}\psi(x)$. Then there exist matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times l}$ such that for each x and u,*

$$\phi(x_{t+1}) = A\phi(x_t) + Bu_t.$$

As the theorem indicates, to get the final representation ϕ, we apply ψ followed by the projection $\Pi_{\mathcal{V}}$. The dynamics of ϕ are nontrivial because, as before in the linear case, the control input can be predicted given the initial and current state representations and previous control inputs.

Intuitively, as τ increases, by result of Theorem 4.7 we can expect that one learns a larger and larger subspace with linear dynamics. Theorem 5.1 shows that as soon as the dimension of this linear subspace stops increasing at a certain length τ, the dynamics of $\psi(x)$ are linear on this subspace. To prove this, we use the fact that the loss is 0 to obtain the identity

$$P\psi(x_i) = L_i\psi(x_0) + \sum_{k=1}^{i-1}T_ku_{i-1-k} + u_{i-1}$$

for $i = 1,\dots,\tau$. Notice that if we view the trajectory as starting at x_1, we have

$$P\psi(x_i) = L_{i-1}\psi(x_1) + \sum_{k=1}^{i-2}T_ku_{i-1-k} + u_{i-1}.$$

Combining these identities and simplifying yields $L_{i-1}\psi(x_1) = L_i\psi(x_0) + T_{i-1}u_0$. This shows roughly that ψ has linear dynamics in the directions of L_{i-1} and L_i. We use these facts together with condition $\text{col}(L_\tau^\top) \subset \mathcal{V}$ to show that ψ has (invariant) linear dynamics on all of $\mathcal{V}$. The detailed proof is found in the appendix.

To solve either (6) or (7) in practice, we constrain ψ to be in some parametric function class and minimize the empirical version of the objective function induced by a finite sample. Since these problems now involve optimizing the parameters of ψ, they are non-convex and much more difficult to analyze explicitly. In general, we can only hope to obtain small loss rather than 0 loss. It is natural to ask whether we can get any guarantees when we have small but nonzero loss for (7). This is a challenging question to answer theoretically, but empirically we observe that achieving moderately small loss yields reasonable state representations on two simple nonlinear control environments; see Section 6.

6. Experiments

We conduct simple experiments to numerically validate our theory. We first discuss experiments with synthetic data generated according to our hidden subspace model, and then experiments using standard RL environments with nonlinear dynamics and high-dimensional observations.

6.1. Synthetic Experiments

Forward Model We create synthetic data from the hidden subspace model by first drawing random matrices $\bar{A} \in \mathbb{R}^{3\times3}$ and $\bar{B} \in \mathbb{R}^{3\times2}$ with i.i.d. standard Gaussian entries, and then generating 1000 i.i.d. samples from the model $h_0 \in \mathbb{R}^3 \sim N(0, I)$, $u_0 \in \mathbb{R}^2 \sim N(0, I)$, $h_1 = \bar{A}h_0 + \bar{B}u_0$. We consider two methods for generating nonlinear features: first by adding white noise where $z_i = h_i + \epsilon_i$, with $\epsilon_i \sim N(0, \sigma^2 I)$ and $\sigma = .1$, and second by taking cubic features $z_i = h_i^3$ (where the cubic power is performed entry-wise, and then truncation is applied to prevent numerical blowup). The observations are then constructed by concatenation: $x_i = (h_i, z_i)$. For both the noisy and cubic features, we optimize (3) using gradient descent with $\lambda = .75, \lambda = .5$, learning rates $.001, .0005$, and number of steps $2.5 \times 10^6, 5 \times 10^5$, respectively. In both cases, the correct subspace is successfully identified. In Figure 2 we plot the loss function and error in P for both experiments. We also conduct similar experiments for the CCA objective – details are in the appendix.

Inverse Model We generate data according to our hidden subspace model as follows. Set the system matrices $\bar{A}, \bar{B}$ at random (with i.i.d. Guassian entries) and multiply $\bar{A}$ by a constant to ensure it is well-conditioned (to avoid numerical issues). Sample the initial latent states $h_0 \sim N(0, I)$ and actions $u_i \sim N(0, I)$. The nonlinear components z_i are created either as independent Gaussian noise or low-degree polynomials of h_i. We collect $5(d + rl)$ samples for each run (this is lower than the sample complexity we give in the Theorem 4.10, but it sufficed for our experiments).

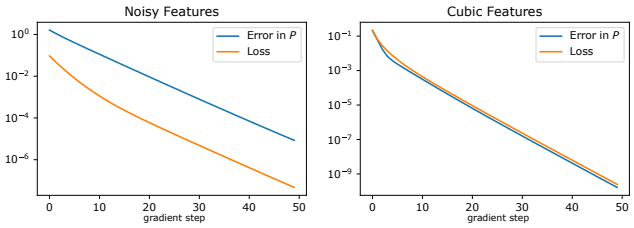

Figure 2. Error in P and loss for the forward model objective

To construct the particular minimal-norm solutions in Theorem 4.10, we optimize (5) in a two-stage linear least squares process using a standard least-square solver ("lstsq" function in SciPy). Further details about this process are given in the appendix. We then check that our constructed solution matches the solution guaranteed by Theorem 4.10. In all of our runs, whenever the computations were numerically stable, we indeed recover the expected solution.

6.2. Nonlinear RL Environments

While our theory doesn't provide guarantees for the setting in which the learned nonlinear state representations incur nonzero loss, we can empirically investigate whether optimizing (6) and (7) lead to reasonable representations. We examine the learned representations visually and explore whether they admit effective control policies. We focus on two standard continuous control tasks from OpenAI Gym (Brockman et al., 2016): 'Pendulum-v0' and 'MountainCarContinuous-v0'.

We implement our learning algorithm in PyTorch (Paszke et al., 2017), and our policy search algorithms use the Stable Baselines library (Hill et al., 2018). We follow the basic approach taken by Lillicrap et al. (2015) in working with pixel observations: modify the environments so that each action is repeated over three consecutive timesteps in the original environment, and concatenate the resultant observations. We parameterize the map ψ as a neural network with two convolutional layers followed by fully-connected layers with ReLu activations. The full details of training are provided in the appendix.

Visual Analysis The pendulum environment has two underlying state variables: the angle of the pendulum from vertically upwards, and its angular velocity. Hence, the slice of the state space corresponding to 0 angular velocity can be viewed as a circle, with the pendulum angle ranging

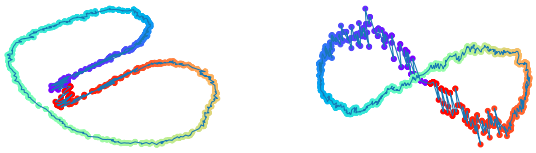

Figure 3. Visualizations of learned pendulum state representations for the forward model (left) and inverse model (right)

cyclically from 0 to 2π. Given a trained representation, we evenly sample this slice of the state space and compute the state representation at each of these points. We then project onto the top two principal components and plot the result in Figure 3, where the color-coding indicates the angle in radians of the pendulum from vertical (red and violet correspond to fully vertical). Both representations capture in distinct ways the symmetry of the state space when reflecting the angle about 0.

Policy Learning After training the state representations, we next explore whether they admit effective policies. For the inverse model, we restrict to linear policies, as these are simpler to optimize and work well for our representation. For the forward model, nonlinear neural network policies are trained, as the linear policies weren't as effective. We also include two baselines that rely on standard RL algorithms implemented in Stable Baselines. Baseline 1 trains nonlinear policies directly from the raw pixel observations (we tune the learning rates and report the best results), and baseline 2 trains policies directly on the low-dimensional state variables using tuned hyperparameters provided by RL Baselines Zoo (Raffin, 2018). The learning curves for all approaches are shown in Figure 6.2.

Note that the forward model representation is *far less sample efficient* than the other methods (its x-axis is scaled up by an order of magnitude compared to the others). The discrepancy between the forward and inverse model representations in this respect may be due in part to the fact that a neural network policy has many more parameters than a linear policy, and hence will naturally train slower. It may also point to an intrinsic difference in the quality of representations produced by the forward model and the inverse model. Perhaps the inverse model representation benefits from being trained over longer trajectories, whereas the forward model just uses a single time step.

For both the pendulum and mountain car environments, each representation admits performant policies that solve the tasks, albeit with less total reward than the tuned Baseline 2 policies. These results show that our state representation learning algorithms capture the relevant structure and dynamics necessary to enable reasonable control policies.

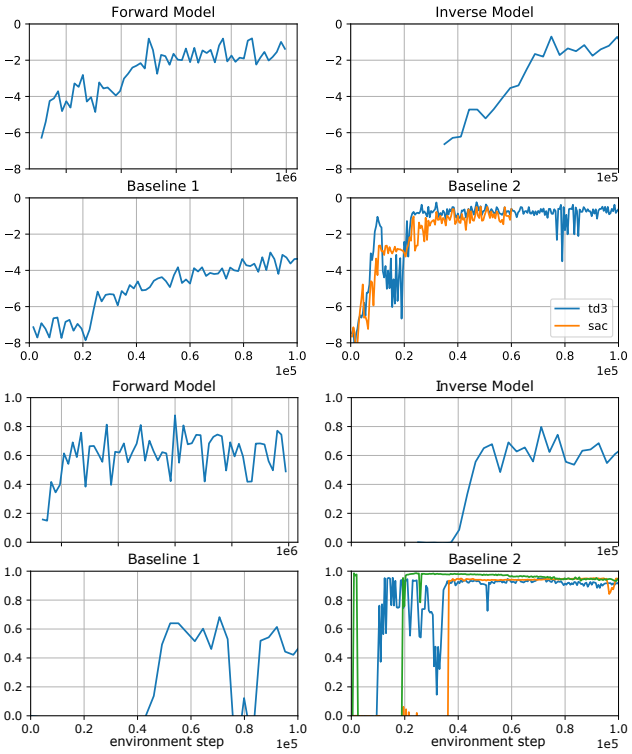

Figure 4. Learning curves for 'Pendulum-v0' (top four) and 'MountainCarContinuous-v0' (bottom four).'

7. Conclusion and Future Work

State representation learning is a promising way to bridge nonlinear, high-dimensional reinforcement learning problems and the simple linear models that have theoretical guarantees. In this paper we study a basic model for state representation learning and show that effective, low-dimensional state representations can be learned efficiently from rich observations using either forward or inverse models. The algorithm inspired by our theory can indeed recover reasonable state representations for simple tasks in OpenAI gym.

There are still many open problems: the nonconvex objectives (6) and (7) can be hard to optimize for the network architectures we tried; is there a way to design the architecture to make the loss go to 0? Additionally, the our algorithms rely on the initial state distribution, which may not sufficiently cover all parts of the state space; can we complement our algorithm with an exploration strategy? Are there more realistic models for state representation learning that can also be learned efficiently? We hope our paper serves as a starting point towards these questions.

8. Acknowledgement

Abraham Frandsen, Rong Ge and Holden Lee are supported in part by NSF Award DMS-2031849, CCF-1704656, CCF-1845171 (CAREER), CCF-1934964 (Tripods), a Sloan Research Fellowship, and a Google Faculty Research Award.

References

Anand, A., Racah, E., Ozair, S., Bengio, Y., Côté, M.-A., and Hjelm, R. D. Unsupervised state representation learning in atari. In *Advances in Neural Information Processing Systems*, pp. 8766–8779, 2019.

Borga, M. Canonical correlation: a tutorial. *On line tutorial http://people. imt. liu. se/magnus/cca*, 4(5), 2001.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Cohen, A., Hassidim, A., Koren, T., Lazic, N., Mansour, Y., and Talwar, K. Online linear quadratic control. *arXiv preprint arXiv:1806.07104*, 2018.

Dean, S., Matni, N., Recht, B., and Ye, V. Robust guarantees for perception-based control. In *Learning for Dynamics and Control*, pp. 350–360. PMLR, 2020.

Du, S. S., Kakade, S. M., Wang, R., and Yang, L. F. Is a good representation sufficient for sample efficient reinforcement learning? *arXiv preprint arXiv:1910.03016*, 2019a.

Du, S. S., Krishnamurthy, A., Jiang, N., Agarwal, A., Dudík, M., and Langford, J. Provably efficient rl with rich observations via latent state decoding. *arXiv preprint arXiv:1901.09018*, 2019b.

Fazel, M., Ge, R., Kakade, S. M., and Mesbahi, M. Global convergence of policy gradient methods for the linear quadratic regulator. *arXiv preprint arXiv:1801.05039*, 2018.

Gao, C., Garber, D., Srebro, N., Wang, J., and Wang, W. Stochastic canonical correlation analysis. *Journal of Machine Learning Research*, 20(167):1–46, 2019.

Ha, D. and Schmidhuber, J. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., and Davidson, J. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*, pp. 2555–2565. PMLR, 2019b.

Hardt, M., Ma, T., and Recht, B. Gradient descent learns linear dynamical systems. *The Journal of Machine Learning Research*, 19(1):1025–1068, 2018.

Higgins, I., Pal, A., Rusu, A., Matthey, L., Burgess, C., Pritzel, A., Botvinick, M., Blundell, C., and Lerchner, A. Darla: Improving zero-shot transfer in reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1480–1490. JMLR. org, 2017.

Hill, A., Raffin, A., Ernestus, M., Gleave, A., Kanervisto, A., Traore, R., Dhariwal, P., Hesse, C., Klimov, O., Nichol, A., Plappert, M., Radford, A., Schulman, J., Sidor, S., and Wu, Y. Stable baselines. https://github.com/hill-a/stable-baselines, 2018.

Jin, C., Ge, R., Netrapalli, P., Kakade, S. M., and Jordan, M. I. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pp. 1724–1732. PMLR, 2017.

Lesort, T., Díaz-Rodríguez, N., Goudou, J.-F., and Filliat, D. State representation learning for control: An overview. *Neural Networks*, 108:379–392, 2018.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Mhammedi, Z., Foster, D. J., Simchowitz, M., Misra, D., Sun, W., Krishnamurthy, A., Rakhlin, A., and Langford, J. Learning the linear quadratic regulator from nonlinear observations. *arXiv preprint arXiv:2010.03799*, 2020.

Misra, D., Henaff, M., Krishnamurthy, A., and Langford, J. Kinematic state abstraction and provably efficient rich-observation reinforcement learning. *arXiv preprint arXiv:1911.05815*, 2019.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 16–17, 2017.

Qin, S. J. An overview of subspace identification. *Computers & chemical engineering*, 30(10-12):1502–1513, 2006.

Raffin, A. Rl baselines zoo. https://github.com/araffin/rl-baselines-zoo, 2018.

Raffin, A., Hill, A., Traoré, K. R., Lesort, T., Díaz-Rodríguez, N., and Filliat, D. Decoupling feature extraction from policy learning: assessing benefits of state representation learning in goal based robotics. *arXiv preprint arXiv:1901.08651*, 2019.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

Srinivas, A., Laskin, M., and Abbeel, P. Curl: Contrastive unsupervised representations for reinforcement learning. *arXiv preprint arXiv:2004.04136*, 2020.

Vershynin, R. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Vinyals, O., Babuschkin, I., Chung, J., Mathieu, M., Jaderberg, M., Czarnecki, W. M., Dudzik, A., Huang, A., Georgiev, P., Powell, R., et al. Alphastar: Mastering the real-time strategy game starcraft ii. *DeepMind blog*, pp. 2, 2019.

Watter, M., Springenberg, J., Boedecker, J., and Riedmiller, M. Embed to control: A locally linear latent dynamics model for control from raw images. In *Advances in neural information processing systems*, pp. 2746–2754, 2015.

Zhang, A., Satija, H., and Pineau, J. Decoupling dynamics and reward for transfer learning. *arXiv preprint arXiv:1804.10689*, 2018.

In this appendix, we first give proofs of the theoretical results stated in the main paper. Next, we discuss synthetic experiments that validate our theoretical results. Finally, we give the additional details for the experiments in Section 6 in the main paper.

A. Deferred Proofs from Sections 3, 4, and 5

In this section we re-state and prove the theorems in the main paper.

We begin by stating a result about empirical canonical correlation that will be used later on. Define empirical canonical correlation in the natural way: for random vectors y and z, let Y and Z be the corresponding sample matrices and define

$$\rho(Y, Z) = \max_{a,b} \frac{a^\top \Sigma_{YZ} b}{\sqrt{a^\top \Sigma_Y a}\sqrt{b^\top \Sigma_Z b}}.$$

We need to control the difference between $\rho(Y, Z)$ and $\rho(y, z)$ when the number of samples is large enough. We utilize a concentration result stated in (Gao et al., 2019) that quantifies this.

Lemma A.1 (Adapted from Corollary 7 of (Gao et al., 2019))**.** *Assume that $y \in \mathbb{R}^{k_1}$ and $z \in \mathbb{R}^{k_2}$ are sub-Gaussian, set $k = k_1 + k_2$, and let $\epsilon \in (0, 1)$. There exists a constant C such that for any $t \geq 1$, if $n \geq Ct^2 k \log^2 k/\epsilon^2$ then $|\rho(Y, Z) - \rho(y, z)| \leq \epsilon$ with probability at least $1 - \exp(-t^2 k)$.*

Note that the statement of this result in (Gao et al., 2019) is slightly different since they don't specify the dependence of the sample complexity on the failure probability parameter t. Our version here is easily obtained by using Corollary 5.50 from (Vershynin, 2010) to include the parameter t.

A.1. Forward Model

We begin by proving Theorem 3.3, which we state again here.

Theorem. Let $i \in \{0, 1, \ldots, d\}$ satisfy $1 - \rho_i^2 < \lambda < 1 - \rho_{i+1}^2$. Let (P, Q) be a second-order stationary point of (2). Then $\text{col}(\Sigma_u^{1/2} P^\top) \subset \mathcal{C}_i$ and $\text{rank}(P) = \min\{r, i\}$.

Proof. Define

$$f(P, Q) = \frac{1}{2}\mathbb{E}_{u,v}\|Pu - Qv\|_2^2,\ r(P) = \frac{1}{4}\|P\Sigma_u P^\top - I\|_F^2.$$

Let (P, Q) be a second-order stationary point. We first show that $\text{col}(\Sigma_u^{1/2} P^\top) \subset \mathcal{C}_i$. The gradients of $g := f + \lambda r$ are as follows:

$$\nabla_Q g = Q\Sigma_v - P\Sigma_{uv}$$
$$\nabla_P g = P\Sigma_u - Q\Sigma_{vu} + \lambda(P\Sigma_u P^\top - I)P\Sigma_u$$

Since these gradients vanish at (P, Q), we have $Q = P\Sigma_{uv}\Sigma_v^{-1}$. Plugging this into the other gradient expression, we have

$$0 = P\Sigma_u^{1/2}((1 - \lambda)I - C)\Sigma_u^{1/2} + \lambda P\Sigma_u P^\top P\Sigma_u \tag{8}$$

Set $\tilde{P} = P\Sigma_u^{1/2}$ and $\hat{P} = \tilde{P}(I - \Pi_{\mathcal{C}_i})$. Let $a \in \mathbb{R}^r$ and $\eta > 0$ satisfy $\hat{P}\hat{P}^\top a = \eta a$. Note that

$$((1 - \lambda)I - C)(I - \Pi_{\mathcal{C}_i}) = (I - \Pi_{\mathcal{C}_i})((1 - \lambda)I - C) = \sum_{j=i+1}^{d} (1 - \lambda - \rho_j^2) c_j c_j^\top$$

is positive definite since $1 - \lambda - \rho_j^2 > 1 - (1 - \rho_{i+1}^2) - \rho_j^2 = \rho_{i+1}^2 - \rho_j^2 \geq 0$ for $j > i$. In particular, this implies that

$$a^\top \tilde{P}((1 - \lambda)I - C)\hat{P}^\top a = a^\top \hat{P}((1 - \lambda)I - C)\hat{P}^\top a \geq 0.$$

Now by left-multiplying (8) with $a^\top$ and right-multiplying with $\Sigma_u^{-1/2}\hat{P}^\top a$ we have

$$\begin{aligned} 0 &= a^\top \tilde{P}((1 - \lambda)I - C)\hat{P}^\top a + \lambda a^\top \tilde{P}\tilde{P}^\top \tilde{P}\hat{P}^\top a \\ &\geq \lambda a^\top \tilde{P}\tilde{P}^\top \hat{P}\hat{P}^\top a \\ &= \lambda\eta a^\top \tilde{P}\tilde{P}^\top a \\ &\geq \lambda\eta^2 \|a\|_2^2. \end{aligned}$$

This means that a must be 0, and so $\hat{P}\hat{P}^\top$ has no nonzero eigenvectors, i.e., $\hat{P} = 0$. Thus, the rows of $\tilde{P}$ are contained in $\mathcal{C}_i$, which is precisely what we wanted to show.

Next, we show that $\text{rank}(P) = \min\{i, r\}$. Assume to the contrary that $\text{rank}(P) < \min\{i, r\}$ and let $b \in \mathbb{R}^d$, $a \in \mathbb{R}^r$ be unit vectors satisfying $b \in \mathcal{C}_i$, $\tilde{P}b = 0$, and $\tilde{P}^\top a = 0$. We claim that for sufficiently small $\epsilon > 0$, the point (P', Q'), where $P' = P + \epsilon ab^\top \Sigma_u^{-1/2}$ and $Q' = Q + \epsilon ab^\top \Sigma_u^{-1/2}\Sigma_{uv}\Sigma_v^{-1}$ yields a strictly smaller loss. First observe that

$$(P + \epsilon ab^\top \Sigma_u^{-1/2})\Sigma_u(P + \epsilon ab^\top \Sigma_u^{-1/2})^\top = P\Sigma_u P^\top + \epsilon^2 aa^\top.$$

Using this and the fact that $P^\top a = \Sigma_u^{-1/2}\tilde{P}^\top a = 0$, we have

$$\begin{aligned} 4r(P') &= \|P'\Sigma_u P'^\top - I\|_F^2 \\ &= \|P\Sigma_u P^\top - I\|_F^2 + \|\epsilon^2 aa^\top\|_F^2 - 2\epsilon^2 \text{tr}(aa^\top) \\ &= \|P\Sigma_u P^\top - I\|_F^2 + \epsilon^4 - 2\epsilon^2 \\ &< 4r(P) \end{aligned}$$

for sufficiently small ϵ, so r indeed decreases.

Next we inspect the change in f on this step. We have the following calculation:

$$\begin{aligned} \mathbb{E}\|\epsilon ab^\top \Sigma_u^{-1/2}(u - \Sigma_{uv}\Sigma_v^{-1}v)\|_F^2 &= \epsilon^2 \mathbb{E}\langle ab^\top \Sigma_u^{-1/2}(u - \Sigma_{uv}\Sigma_v^{-1}v), ab^\top \Sigma_u^{-1/2}(u - \Sigma_{uv}\Sigma_v^{-1}v)\rangle \\ &= \epsilon^2 \langle \Sigma_u^{-1/2} ba^\top ab^\top \Sigma_u^{-1/2}, \Sigma_u - \Sigma_{uv}\Sigma_v^{-1}\Sigma_{vu}\rangle \\ &= \epsilon^2 \langle bb^\top, I - C\rangle \\ &\leq \epsilon^2(1 - \rho_i^2) \end{aligned}$$

since $b \in \mathcal{C}_i$ and $\|b\|_2 = 1$. Furthermore, we have $\mathbb{E}\langle Pu - Qv, ab^\top \Sigma_u^{-1/2}(u - \Sigma_{uv}\Sigma_v^{-1}v)\rangle = 0$ since $a^\top P = 0$ and $a^\top Q = a^\top P\Sigma_{uv}\Sigma_v^{-1} = 0$. Putting these together, we have

$$\begin{aligned} 2f(P', Q') &= \mathbb{E}\|(P + \epsilon ab^\top \Sigma_u^{-1/2})u - (Q + \epsilon ab^\top \Sigma_u^{-1/2}\Sigma_{uv}\Sigma_v^{-1})v\|_2^2 \\ &= \mathbb{E}\|Pu - Qv + \epsilon ab^\top \Sigma_u^{-1/2}(u - \Sigma_{uv}\Sigma_v^{-1}v)\|_2^2 \\ &= 2f(P, Q) + 2\epsilon \mathbb{E}\langle Pu - Qv, ab^\top \Sigma_u^{-1/2}(u - \Sigma_{uv}\Sigma_v^{-1}v)\rangle + 2\mathbb{E}\|\epsilon ab^\top \Sigma_u^{-1/2}(u - \Sigma_{uv}\Sigma_v^{-1}v)\|_2^2 \\ &\leq 2f(P, Q) + \epsilon^2(1 - \rho_i^2). \end{aligned}$$

Therefore, we conclude that

$$(f(P, Q) + \lambda r(P)) - (f(P', Q') + \lambda r(P')) \geq \frac{\lambda(2\epsilon^2 - \epsilon^4)}{4} - \frac{\epsilon^2(1 - \rho_i^2)}{2} = \frac{(\lambda - (1 - \rho_i^2))\epsilon^2}{2} - \frac{\lambda\epsilon^4}{4}.$$

Since $\lambda > 1 - \rho_i^2$, this implies that the Hessian has a negative eigenvalue at (P, Q). This contradicts the fact that (P, Q) is a second-order stationary point, so we conclude that $\text{rank}(P) = \min\{i, r\}$. □

Now we work toward proving Theorem 3.2. We start with a lemma that connects Assumption 3.1 to the canonical correlations between x_1 and (x_0, u_0).

Lemma A.2. *Let w denote (x_0, u_0) and let x denote x_1. Define $C = \Sigma_x^{-1/2}\Sigma_{xw}\Sigma_w^{-1}\Sigma_{wx}\Sigma_x^{-1/2}$. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$ be the eigenvalues of C (with multiplicity) and let $c_1, \ldots, c_d$ be corresponding orthonormal eigenvectors. Then under Assumption 3.1, $\lambda_i = 1$ for $i = 1, \ldots, r$ and $\lambda_{r+1} \leq \rho^2$. Moreover, span$\{\Sigma_x^{-1/2}c_i : 1 \leq i \leq r\} = \mathcal{V}$.*

Proof. We first show that $\lambda_{r+1} \leq \rho^2$. Define $\tilde{\mathcal{V}} = \{\Sigma_x^{1/2}v : v \in \mathcal{V}\}$. Noting that $\Pi_{\mathcal{V}}x = z_1$ and $\dim(\tilde{\mathcal{V}}) = 1$, by the

variational characterization of eigenvalues of symmetric matrices and Assumption 3.1, we have that

$$\begin{aligned}
\lambda_{r+1} &= \min_{\mathcal{U}} \left\{ \max_{a \in \mathcal{U}} \frac{a^\top C a}{\|a\|_2^2} \ : \ \dim(\mathcal{U}) = d - r \right\} \\
&\leq \min_{\mathcal{U}} \left\{ \max_{a \in \mathcal{U}, b \in \mathbb{R}^{d+l}} \frac{(a^\top \Sigma_x^{-1/2} \Sigma_{xw} \Sigma_w^{-1/2} b)^2}{\|a\|_2^2 \|b\|_2^2} \ : \ \dim(\mathcal{U}) = d - r \right\} \\
&\leq \max_{a \in \tilde{\mathcal{V}}, b \in \mathbb{R}^{d+l}} \frac{(a^\top \Sigma_x^{-1/2} \Sigma_{xw} \Sigma_w^{-1/2} b)^2}{\|a\|_2^2 \|b\|_2^2} \\
&= \max_{a \in \mathcal{V}, b \in \mathbb{R}^{d+l}} \frac{(a^\top \Sigma_{xw} b)^2}{(a^\top \Sigma_x a)(b^\top \Sigma_w b)} \\
&= \max_{a \in \mathbb{R}^d, b \in \mathbb{R}^{d+l}} \frac{(a^\top \Sigma_{z_1 w} b)^2}{(a^\top \Sigma_{z_1} a)(b^\top \Sigma_w b)} \\
&= \rho(z_1, w)^2 \\
&\leq \rho^2,
\end{aligned}$$

as desired.

Now let $T : \mathbb{R}^{d+l} \to \mathbb{R}^d$ be the linear transformation satisfying $y_1 = Tw$. Let $v \in \mathcal{V}$. Note that $v^\top x = v^\top y_1 = v^\top T w$, so we have

$$\begin{aligned}
C \Sigma_x^{1/2} v &= \Sigma_x^{-1/2} \Sigma_{xw} \Sigma_w^{-1} \Sigma_{wx} v \\
&= \Sigma_x^{-1/2} \Sigma_{xw} \Sigma_w^{-1} \Sigma_w T^\top v \\
&= \Sigma_x^{-1/2} \Sigma_{xw} T^\top v \\
&= \Sigma_x^{-1/2} \Sigma_x v \\
&= \Sigma_x^{1/2} v.
\end{aligned}$$

Hence, $\Sigma_x^{1/2} v$ is an eigenvector of C with eigenvalue equal to 1. Since $\mathcal{V}$ is r-dimensional, we conclude that the top r eigenvalues of C are all equal to 1, and the corresponding r-dimensional eigenspace is precisely $\tilde{\mathcal{V}}$. It then follows that $\text{span}\{\Sigma_x^{-1/2} c_i \ : \ 1 \leq i \leq r\} = \mathcal{V}$. □

We are now ready to prove Theorem 3.2.

Proof. Let (P, Q, D) be a second-order stationary point of (1). Let x, w, C be as in Lemma A.2. Let $\mathcal{C}_r$ be as Theorem 3.3 where u is identified with x and v is identified with w. By Lemma A.2, we have that $\mathcal{V} = \Sigma_x^{-1/2} \mathcal{C}_r$. By Theorem 3.3, we have $\text{rank}(P) = r$ and $\text{col}(\Sigma_x^{1/2} P^\top) = \mathcal{C}_r$. Thus, we conclude that $\text{col}(P^\top) = \Sigma_x^{-1/2} \mathcal{C}_r = \mathcal{V}$, as desired. □

Finally, we address the sample complexity of the forward model objective. We begin with the following lemma.

Lemma A.3. *Let w denote (x_0, u_0). Let Z and W be the sample matrices of z_1 and w, respectively. Let $\mathcal{E}$ denote the event that $|\rho(Z, W) - \rho(z, w)| \leq (1 - \rho)/2$. Under Assumption 3.4, there exists a constant C_0 such that if $n \geq C_0 \log(2d + l)/(1 - \rho)^2$, then $P(\mathcal{E}) \geq 0.99$.*

Proof. Set $t = \sqrt{2 \log(10)/(2d + l)}$ and note that $1 - \exp(-t^2(2d + l)) = 0.99$. Let $\epsilon = (1 - \rho)/2$ and note that

$$t^2 (2d + l) \log^2(2d + l)/\epsilon^2 = 8 \log 10 \log^2(2d + l)/(1 - \rho)^2.$$

Then by Lemma A.1 applied to z_1 and w with the above ϵ, there exists a constant C such that if $n \geq (8 \log 10) C \log^2(2d + l)/(1 - \rho)^2$, then $|\rho(Z, W) - \rho(z_1, w)| \leq (1 - \rho)/2$ with probability at least 0.99. □

We can now prove Theorem 3.5.

Proof. As before, let x denote x_1, and let X denote the corresponding sample matrix. We also have w, W, and Z as before. Condition on the event that $|\rho(Z,W) - \rho(z_1,w)| \leq (1-\rho)/2$, which has probability at least 0.99 by Lemma A.3. Then $\rho(Z,W) \leq \rho(z_1,w) + (1-\rho)/2 \leq (1+\rho)/2$.

Define $C = \Sigma_X^{-1/2}\Sigma_{XW}\Sigma_W^{-1}\Sigma_{WX}\Sigma_X^{-1/2}$, and let $\lambda_1 \geq \cdots \geq \lambda_d$ be its eigenvalues with corresponding orthonormal eigenvectors $c_1, \ldots, c_d$. Just as in the population case, the eigenvalues of C are the squared empirical canonical correlations between X and W, and the eigenvectors give the corresponding canonical correlation directions. Note that $\Pi_{\mathcal{V}} X$ has perfect linear correlation with W. Using the same arguments from Lemma A.2, we have that the top r eigenvalues of C are equal to 1 and $\lambda_{r+1} \leq (1+\rho)^2/4$. Moreover, $\text{span}\{\Sigma_X^{-1/2}c_i : 1 \leq i \leq r\} = \mathcal{V}$.

Note that $(1-\rho^2)/4 \leq (3-2\rho-\rho^2)/4 = 1-(1+\rho)^2/4$. Hence, we have $\lambda \leq 1-(1+\rho)^2/4$. Now let (P,Q,D) be a second-order stationary point of (3). We claim that the proof of Theorem 3.3 carries through exactly the same if we replace the population objective function with the corresponding finite sample version (the argument is identical, we just need to replace every covariance and cross-covariance matrix with the appropriate empirical version – there are no spurious correlations to deal with). Hence, we can make use of that result here, and conclude that P has rank r and $\text{col}(\Sigma_X^{1/2}P^\top) = \text{span}\{c_1, \ldots, c_r\}$. Combining this with the fact that $\text{span}\{\Sigma_X^{-1/2}c_i : 1 \leq i \leq r\} = \mathcal{V}$ completes the proof. □

A.2. Inverse Model

We first give a proof of Theorem 4.7, which shows that our objective function can recover the unknown subspace V. The theorem is restated below.

Theorem A.4. *Let f be the objective function in (4), and let $\Theta_0^* = \{\theta = (P, \{L_i\}_{i=1}^r, \{T_i\}_{i=1}^{r-1}) \in f^{-1}(0) \mid \|P\|_F$ is minimal$\}$ be the set of optimal solutions to (4) that have minimal norm for P. Let $\theta^* = (P^*, \{L_i^*\}, \{T_i^*\}) \in \Theta_0^*$ be the solution in this set that minimizes $\sum_{i=1}^r \|L_i\|_F^2$. Then under assumptions 4.1, 4.3, and 4.5, $P = B^+$ and $L_i = B^+A^i$ for $i = 1, \ldots, r$. Moreover, $V = col(P^\top) + col(L_1^\top) + \cdots + col(L_r^\top)$.*

Recall the objective (4) was:

$$\min_\theta \frac{1}{2}\mathbb{E}\sum_{i=1}^r \|Px_i - L_i x_0 - \sum_{k=1}^{i-1} T_k u_{i-1-k} - u_{i-1}\|_2^2$$

The main difficulty of proving this theorem lies in a mismatch between Assumption 4.1 and our objective function (4): in the objective (4), we try to enforce a linear relationship between $x_i, x_0, u_1, u_2, ..., u_{i-1}$, while Assumption 4.1 is about (h_i, h_0) and (z_i, z_0). The following lemma helps relate the two.

Lemma A.5. *Let $i \in \{1, \ldots, r\}$. Let $\tilde{h}_i = (h_i, h_0, u_0, \ldots, u_{i-2})$ and let $\tilde{z}_i = (z_i, z_0)$. Then*

$$\rho(\tilde{h}_i, \tilde{z}_i) \leq \rho((h_i, h_0), (z_i, z_0)).$$

Proof. Note that the definition of $\tilde{h}_i$ doesn't make sense for $i = 1$. In that case, define $\tilde{h}_1 = (h_1, h_0)$. Let $u = (u_0, u_1, \ldots, u_{i-1})$. Observe that the coordinates of $\tilde{h}_i$ are a subset of the coordinates of (h_i, h_0, u), so $\rho(\tilde{h}_i, \tilde{z}_i) \leq \rho((h_i, h_0, u), \tilde{z}_i)$. Note that there exist matrices P and Q such that $h_i = Ph_0 + Qu$. Let $a_1, a_2 \in \mathbb{R}^r$, $b_1, b_2 \in \mathbb{R}^d$, and $a_3 \in \mathbb{R}^{il}$. Write $a_3 = Q^\top v_1 + v_2$, where $Qv_2 = 0$. Note that u is independent of h_0 and $\langle v_2, u\rangle$ is independent of each coordinate of h_i (as these are Gaussian random vectors). Then we have

$$\begin{aligned}
\mathbb{E}[(\langle a_1, h_i\rangle + \langle a_2, h_0\rangle + \langle a_3, u\rangle)^2] &= \mathbb{E}[(\langle a_1, h_i\rangle + \langle a_2, h_0\rangle + \langle Q^\top v_1, u\rangle + \langle v_2, u\rangle + \langle P^\top v_1, h_0\rangle - \langle P^\top v_1, h_0\rangle)^2] \\
&= \mathbb{E}[(\langle a_1 + v_1, h_i\rangle + \langle a_2 - P^\top v_1, h_0\rangle + \langle v_2, u\rangle)^2] \\
&= \mathbb{E}[(\langle a_1 + v_1, h_i\rangle + \langle a_2 - P^\top v_1, h_0\rangle)^2] + \mathbb{E}[\langle v_2, u\rangle^2]
\end{aligned}$$

Now u is independent of z_0, so $\mathbb{E}[\langle v_2, u\rangle\langle b_2, z_0\rangle] = 0$. Moreover, u and z_i are conditionally independent given h_i, so we

have

$$\begin{aligned}\mathbb{E}[\langle v_2, u\rangle(\langle b_1, z_i\rangle + \langle b_2, z_0\rangle)] &= \mathbb{E}[\langle v_2, u\rangle\langle b_1, z_i\rangle] \\ &= \mathbb{E}[\mathbb{E}[\langle v_2, u\rangle\langle b_1, z_i\rangle|h_i]] \\ &= \mathbb{E}[\mathbb{E}[\langle v_2, u\rangle|h_i]\mathbb{E}[\langle b_1, z_i\rangle|h_i]] \\ &= \mathbb{E}[\mathbb{E}[\langle v_2, u\rangle]\mathbb{E}[\langle b_1, z_i\rangle|h_i]] \\ &= 0.\end{aligned}$$

Then we have

$$\begin{aligned}&\frac{\mathbb{E}[(\langle a_1, h_i\rangle + \langle a_2, h_0\rangle + \langle a_3, u\rangle)(\langle b_1, z_i\rangle + \langle b_2, z_0\rangle)]}{\sqrt{\mathbb{E}[(\langle a_1, h_i\rangle + \langle a_2, h_0\rangle + \langle a_3, u\rangle)^2]\mathbb{E}[(\langle b_1, z_i\rangle + \langle b_2, z_0\rangle)^2]}} \\ &\quad\leq \frac{\mathbb{E}[(\langle a_1 + v_1, h_i\rangle + \langle a_2 - P^\top v_1, h_0\rangle)(\langle b_1, z_i\rangle + \langle b_2, z_0\rangle)}{\sqrt{\mathbb{E}[(\langle a_1 + v_1, h_i\rangle + \langle a_2 - P^\top v_1, h_0\rangle)^2]\mathbb{E}[(\langle b_1, z_i\rangle + \langle b_2, z_0\rangle)^2]}} \\ &\quad\leq \rho((h_i, h_0), (z_i, z_0)).\end{aligned}$$

□

We are ready to prove Theorem 4.7:

Proof. The main idea of the proof is to derive conditions for the variables based on first-order optimality conditions. We first prove that the optimal variables have support only on the linearizing subspace. As a consequence, we can then show that these variables equal the true model parameters.

To start, fix $i \in \{1, \ldots, r\}$, define $\theta_i = [P\ L_i\ T_1 \cdots T_{i-1}]$, let $\tilde{y}_i = (y_i, -y_0, -u_{i-2}, \ldots, -u_0)$, $\tilde{h}_i = (h_i, -h_0, -u_{i-2}, \ldots, -u_0)$, and $\tilde{z}_i = (z_i, -z_0)$. Define $K = [I\ 0]^\top$ to be the block matrix that satisfies $\theta_i K = [P\ L_i]$. Define $\tilde{V} = \mathrm{diag}(V, V, I, \ldots, I)$ to be the block diagonal matrix that satisfies $\tilde{y}_i = \tilde{V}\tilde{h}_i$, and note that $\tilde{V}$ has full column rank. Observe that there exists a matrix M such that $u_{i-1} = M\tilde{h}_i$.

We can now express the objective function as $f(\theta) = \sum_{i=1}^{\tau} f_i(\theta_i)$, where $f_i(\theta_i) = \frac{1}{2}\mathbb{E}\|\theta_i(K\tilde{z}_i + \tilde{V}\tilde{h}_i) - u_{i-1}\|_2^2$. Since each f_i has minimal value 0, any optimal point for f must simultaneously optimize each f_i. Hence, $\nabla f(\theta_i) = 0$ is a necessary condition for optimality. To this end, we compute the gradient of f_i as

$$\nabla f_i(\theta_i) = \theta_i(K\Sigma_{\tilde{z}_i\tilde{z}_i}K^\top + \tilde{V}\Sigma_{\tilde{h}_i\tilde{h}_i}\tilde{V}^\top + \tilde{V}\Sigma_{\tilde{h}_i\tilde{z}_i}K^\top + K\Sigma_{\tilde{z}_i\tilde{h}_i}\tilde{V}^\top) - \Sigma_{u_{i-1}\tilde{z}_i}K^\top - \Sigma_{u_{i-1}\tilde{h}_i}\tilde{V}^\top$$

We split the optimality condition according to orthogonal subspaces V and $V^\perp$ to obtain

$$0 = \theta_i(\tilde{V}\Sigma_{\tilde{h}_i\tilde{h}_i}\tilde{V}^\top + K\Sigma_{\tilde{z}_i\tilde{h}_i}\tilde{V}^\top) - \Sigma_{u_{i-1}\tilde{h}_i}\tilde{V}^\top \tag{9}$$

$$0 = \theta_i(K\Sigma_{\tilde{z}_i\tilde{z}_i}K^\top + \tilde{V}\Sigma_{\tilde{h}_i\tilde{z}_i}K^\top) - \Sigma_{u_{i-1}\tilde{z}_i}K^\top \tag{10}$$

From (9), we have $\theta_i\tilde{V}\Sigma_{\tilde{h}_i\tilde{h}_i} = \Sigma_{u_{i-1}\tilde{h}_i} - \theta_i K\Sigma_{\tilde{z}_i\tilde{h}_i}$, and plugging this into (10) (while also clearing $K^\top$ by right-multiplying the equation by K) gives

$$\begin{aligned}0 &= \theta_i K\Sigma_{\tilde{z}_i\tilde{z}_i} + \theta_i\tilde{V}\Sigma_{\tilde{h}_i\tilde{h}_i}\Sigma^+_{\tilde{h}_i\tilde{h}_i}\Sigma_{\tilde{h}_i\tilde{z}_i} - \Sigma_{u_{i-1}\tilde{z}_i} \\ &= \theta_i K\Sigma_{\tilde{z}_i\tilde{z}_i} - \theta_i K\Sigma_{\tilde{z}_i\tilde{h}_i}\Sigma^+_{\tilde{h}_i\tilde{h}_i}\Sigma_{\tilde{h}_i\tilde{z}_i} - \Sigma_{u_{i-1}\tilde{z}_i} + \Sigma_{u_{i-1}\tilde{h}_i}\Sigma^+_{\tilde{h}_i\tilde{h}_i}\Sigma_{\tilde{h}_i\tilde{z}_i} \\ &= \theta_i K(\Sigma_{\tilde{z}_i\tilde{z}_i} - \Sigma_{\tilde{z}_i\tilde{h}_i}\Sigma^+_{\tilde{h}_i\tilde{h}_i}\Sigma_{\tilde{h}_i\tilde{z}_i}) - M\Sigma_{\tilde{h}_i\tilde{z}_i} + M\Sigma_{\tilde{h}_i\tilde{h}_i}\Sigma^+_{\tilde{h}_i\tilde{h}_i}\Sigma_{\tilde{h}_i\tilde{z}_i} \\ &= \theta_i K\Sigma_{\tilde{z}_i\tilde{z}_i}(I - \Sigma^+_{\tilde{z}_i\tilde{z}_i}\Sigma_{\tilde{z}_i\tilde{h}_i}\Sigma^+_{\tilde{h}_i\tilde{h}_i}\Sigma_{\tilde{h}_i\tilde{z}_i}).\end{aligned}$$

By Lemma A.5 and Assumption 4.1, we have that $I - \Sigma^+_{\tilde{z}_i\tilde{z}_i}\Sigma_{\tilde{z}_i\tilde{h}_i}\Sigma^+_{\tilde{h}_i\tilde{h}_i}\Sigma_{\tilde{h}_i\tilde{z}_i}$ is nonsingular, so we conclude that $\theta_i K\Sigma_{\tilde{z}_i\tilde{z}_i} = 0$. In particular, this implies that $\theta_i K\Sigma_{\tilde{z}_i\tilde{y}_i} = 0$.

We can now simplify (9) as $0 = \theta_i\tilde{V}\Sigma_{\tilde{h}_i\tilde{h}_i}\tilde{V}^\top - \Sigma_{u_{i-1}\tilde{h}_i}\tilde{V}^\top = \theta_i\Sigma_{\tilde{y}_i\tilde{y}_i} - \Sigma_{u_{i-1}\tilde{y}_i}$. This matrix equation can be naturally partitioned into blocks according to the block partition of θ_i and $\tilde{y}_i$. Reading out the second block column gives $0 =$

$-PA^i + L_i VV^\top$. Reading out the $(k+1)$-st block column (for $k \geq 1$) gives $0 = -PA^kB + T_k$. The first block column gives

$$\begin{aligned}
0 &= P\Sigma_{y_i y_i} - L_i \Sigma_{y_0 y_i} - \sum_{k=1}^{i-1} T_k \Sigma_{u_{i-1-k} y_i} - \Sigma_{u_{i-1} y_i} \\
&= P(A^i(A^i)^\top + \sum_{k=1}^{i-1} A^k B(A^k B)^\top + BB^\top) - L_i(A^i)^\top - \sum_{k=1}^{i-1} T_k (A^k B)^\top - B^\top \\
&= (PB - I)B^\top + (PA^i - L_i)(A^i)^\top + \sum_{k=1}^{i-1} (PA^k B - T_k)(A^k B)^\top \\
&= (PB - I)B^\top.
\end{aligned}$$

Using Assumption 4.5, we right-multiply by $(B^+)^\top$ to obtain $PB = I$. Since P is the minimal-norm optimal solution, we conclude that $P = B^+$. Then $L_i VV^\top = B^+ A^i$ and $T_k = B^+ A^k B$. Since we are also minimizing the norm of L_i, we see that L_i must vanish on $V^\perp$, so that $L_i = L_i VV^\top$, and $L_i = B^+ A^i$. That we recover all of V is a consequence of Assumption 4.3. □

Now we address the finite sample complexity of the inverse model objective. In this section, for each random vector z_i, y_i, h_i, u_i involved in the model, we consider corresponding sample matrices $Z_i, Y_i \in \mathbb{R}^{d \times n}, H_i \in \mathbb{R}^{r \times n}, U_i \in \mathbb{R}^{l \times n}$. For sample covariance matrices we use the notation $\Sigma_{U_i U_i} = \frac{1}{n} U_i U_i^\top, \Sigma_{Y_i Z_i} = \frac{1}{n} Y_i Z_i^\top$, and so on.

More precisely, let $H_0 \in \mathbb{R}^{r \times n}$ be a random matrix whose columns are independent standard Gaussian vectors. Likewise, for $k = 0, \ldots, r-1$, let $U_k \in \mathbb{R}^{l \times n}$ be a matrix whose columns are independent standard Gaussian vectors. For $i = 0, \ldots, r$, let $H_i = \bar{A}^i H_0 + \sum_{k=0}^{i-1} \bar{A}^k B U_{i-1-k}$, and let $X_i = VH_i + Z_i$, where the columns of X_i are the observed states and the columns of Z_i are the nonlinear parts.

We also need to ensure that a certain Gaussian empirical covariance matrix is invertible. We use the following standard matrix concentration inequality.

Lemma A.6 (From Corollary 5.35 of (Vershynin, 2010))**.** *Let $Y \in \mathbb{R}^{k \times n}$ be a matrix whose entries are independent standard Gaussian random variables. Then for every $t \geq 0$, with probability at least $1 - 2\exp(-t^2/2)$ it holds that*

$$\sqrt{n} - \sqrt{k} - t \leq \sigma_{min}(Y).$$

We now use these two concentration results to prove our main lemma for this section.

Lemma A.7. *Let $\tilde{Z}_i$ and $\tilde{H}_i$ be the sample matrices of $\tilde{z}_i$ and $\tilde{h}_i$, respectively (from the proof of Theorem 4.7). Further define $\hat{H}_i$ to be the sample matrix for the random vector $\hat{h}_i := (h_0, u_{i-1}, u_{i-2}, \ldots, u_0)$. Let $\mathcal{E}_i$ denote the event that $|\rho(\tilde{H}_i, \tilde{Z}_i) - \rho(\tilde{h}_i, \tilde{z}_i)| \leq (1-\rho)/2$. Let $\mathcal{F}_i$ denote the event that $\sigma_{min}(\hat{H}_i) \geq 1/2$. There exists a constant C_0 such that if $n = C_0(d + rl) \log r \log^2(d + rl)/(1-\rho)^2$, then*

$$P\left(\bigcap_{i=1}^{r} \mathcal{E}_i \cap \mathcal{F}_i\right) \geq 0.99.$$

Proof. Set the failure probability parameter $t = C'\sqrt{\log r}$, where C' is a large enough constant such that

$$r(\exp(-t^2(2d + 2r)) + 2\exp(-t^2/2)) \leq 0.01.$$

Let C be the constant from Lemma A.1 applied to $\tilde{h}_i$ and $\tilde{z}_i$ with $\epsilon = (1-\rho)/2$ – we can take the same C for each i since we assume each (h_i, z_i) satisfy the same sub-Gaussian property. Set C_0 large enough so that when $n = C_0(d + rl) \log r \log^2(d + rl)/(1-\rho)^2$, the following hold for $i = 1, \ldots, r$:

$$\begin{aligned}
n &\geq 4Ct^2(2d + 2r + (i-2)l) \log^2(2d + 2r + (i-2)l)/(1-\rho)^2, \\
\sqrt{n} &\geq 1/2 + \sqrt{r + (i-1)l} + t
\end{aligned}$$

We first analyze $P(\mathcal{E}_i)$. Apply Lemma A.1 to $\tilde{h}_i \in \mathbb{R}^{2r+(i-2)l}$ and $\tilde{z}_i \in \mathbb{R}^{2d}$ with $\epsilon = (1-\rho)/2$ and the specified value of t. Then we see that n is large enough to ensure that $P(\mathcal{E}_i) \geq 1 - \exp(-t^2(2d+2r+(i-2)l)) \geq 1 - \exp(-t^2(2d+2r))$.

Next, consider $P(\mathcal{F}_i)$. Apply Lemma A.6 to $\hat{H}_i$ with the specified value of t. Again it is clear that n is large enough to ensure that $P(\mathcal{F}_i) \geq 1 - 2\exp(-t^2/2)$.

Finally, by the union bound,

$$\begin{aligned} P\left(\bigcap_{i=1}^{r} \mathcal{E}_i \cap \mathcal{F}_i\right) &\geq 1 - \sum_{i=1}^{r}(2 - P(\mathcal{E}_i) + P(\mathcal{F}_i)) \\ &\geq 1 - r(\exp(-t^2(2d+2r)) + 2\exp(-t^2/2)) \\ &\geq 0.99. \end{aligned}$$

□

We now prove Theorem 4.10.

Proof. Lemma A.7 provides the sample complexity and success probability – all that's left is to analyze the empirical loss assuming that the conclusion of Lemma A.7 holds. Our analysis of the empirical loss is close to that of the population loss. We use the same notation as in the proof of Theorem 4.7, e.g. $\tilde{Y}_i, \tilde{H}_i, \tilde{U}_i, \tilde{Z}_i$ are the sample matrices of $\tilde{y}_i, \tilde{h}_i, \tilde{u}_i, \tilde{z}_i$, respectively. Likewise, define θ_i, K, and $\tilde{V}$ as before. We additionally define $\hat{H}_i$ to be the sample matrix for $(h_0, u_{i-1}, u_{i-2}, \ldots, u_0)$.

By the same argument as in the proof of Theorem 4.7, we have that

$$0 = \theta_i K \Sigma_{\tilde{Z}_i \tilde{Z}_i}(I - \Sigma^{+}_{\tilde{Z}_i \tilde{Z}_i} \Sigma_{\tilde{Z}_i \tilde{H}_i} \Sigma^{+}_{\tilde{H}_i \tilde{H}_i} \Sigma_{\tilde{H}_i \tilde{Z}_i}).$$

The spectral norm of $-\Sigma^{+}_{\tilde{Z}_i \tilde{Z}_i} \Sigma_{\tilde{Z}_i \tilde{H}_i} \Sigma^{+}_{\tilde{H}_i \tilde{H}_i} \Sigma_{\tilde{H}_i \tilde{Z}_i}$ is $\rho(\tilde{H}_i, \tilde{Z}_i)$, and by assumption and Lemma A.7, we have

$$\rho(\tilde{H}_i, \tilde{Z}_i) \leq \rho(\tilde{h}_i, \tilde{z}_i) + (1-\rho)/2 \leq (1+\rho)/2 < 1.$$

Hence, $(I - \Sigma^{+}_{\tilde{Z}_i \tilde{Z}_i} \Sigma_{\tilde{Z}_i \tilde{H}_i} \Sigma^{+}_{\tilde{H}_i \tilde{H}_i} \Sigma_{\tilde{H}_i \tilde{Z}_i})$ is robustly nonsingular, so we conclude that $\theta_i K \Sigma_{\tilde{Z}_i \tilde{Z}_i} = 0$ and likewise $\theta_i K \Sigma_{\tilde{Z}_i \tilde{Y}_i} = 0$.

Using this fact, we can continue to follow the proof of Theorem 4.7 to obtain

$$0 = \theta_i \tilde{V} \Sigma_{\tilde{H}_i \tilde{H}_i} \tilde{V}^\top - \Sigma_{U_{i-1} \tilde{H}_i} \tilde{V}^\top.$$

Analyzing this equation is slightly more complicated now due to the fact that sample cross-covariance terms like $\Sigma_{H_0 U_j}$ are nonzero (whereas the corresponding population covariances vanish due to independence). By splitting the equation into block columns, grouping terms, and simplifying the terms that cancel, it is straightforward to see that

$$0 = [(PA^i - L_i)\ (PB - I)\ (PAB - T_1)\ \cdots\ (PA^{i-1}B - T_{i-1})]\tilde{V} \Sigma_{\hat{H}_i \hat{H}_i} \tilde{V}^\top.$$

By assumption, $\sigma_{min}(\hat{H}_i) \geq 1/2$, so $\Sigma_{\hat{H}_i \hat{H}_i}$ is robustly nonsingular. Hence, we have that

$$0 = [(PA^i - L_i)\ (PB - I)\ (PAB - T_1)\ \cdots\ (PA^{i-1}B - T_{i-1})]\tilde{V},$$

which implies that $PB = I$ and $(PA^i - L_i)V = 0$ for all i. Since we assume P has minimal norm, we conclude that $P = B^+$. Thus, $L_i V = B^+ A^i V$, i.e. $L_i = B^+ A^i$ on the subspace V. By the construction of our minimal norm solution, we know that L_i must vanish on $V^\perp$, and this completes the proof. □

A.3. Handling Noise in the Model

We now consider a simple version of our model with noise, and show that our algorithm identifies the correct subspace (up to an error proportional to the noise) in this setting as well. We consider a one-step trajectory where the initial state $x_0 = 0$, and we assume that our observation is corrupted by independent centered noise. In particular, we can write the state observation as $x = Bu + z + \xi$, where ξ is a random vector in $\mathbb{R}^d$ that is independent of both u and z. Assume the noise

covariance matrix $\Sigma_{\xi\xi}$ splits orthogonally along the subspace V and $V^\perp$, that is, we can write $\Sigma_{\xi\xi} = \Sigma_1 + \Sigma_2$, where Σ_1 is the covariance of the noise projected onto V and Σ_2 is the covariance of the noise projected onto the column-span of $V^\perp$. This orthogonal splitting is satisfied when ξ is a spherical Gaussian random vector, for example.

Given this noisy state observation x and control input u, the task is to recover the column-span of B by learning a linear inverse model:

$$\min_P \frac{1}{2}\mathbb{E}_{u,\xi}\|Px - u\|_2^2 \tag{11}$$

Due to the noise term, this linear model will not achieve zero error. However, we can bound the error of our solution as a function of the noise magnitude and correlation bound.

Theorem A.8. *Let $u \in \mathbb{R}^l$ and $\xi \in \mathbb{R}^d$ be independent spherical Gaussian random vectors, with $\Sigma_{\xi\xi} = \sigma^2 I$. Let P be the minimal norm optimal solution to the optimization problem (11). Write $P = P_1 + P_2$, where P_1 is the projection of P onto V, and P_2 is its projection onto $V^\perp$. In the noisy setting described above, we have $P_1 = B^+$ and*

$$\|P_2\|_2 \leq \frac{\sigma\rho}{2\sqrt{1-\rho^2}}\|B^+\|_2\|P_1\|_2$$

where $\sigma = \lambda_{max}(\Sigma_{\xi\xi})$ and $\rho := \rho(u, z)$.

Note that ideally we want $P_2 = 0$, since its rows are in $V^\perp$. This theorem says that the spectral norm of P_2 is small compared to P_1, which allows us to approximately recover B^+.

Proof. In this setting, the optimality conditions of (11) take the form

$$0 = B\Sigma_{uu}(B^\top P_1 - I) + B\Sigma_{uz}P_2 + \sigma^2 P_1 \tag{12}$$

$$0 = \Sigma_{zu}(B^\top P_1 - I) + \Sigma_{zz}P_2 + \sigma^2 P_2, \tag{13}$$

Multiplying (12) by $\Sigma_{zu}(B\Sigma_{uu})^+$ and subtracting (13) yields the following identity (after simplification):

$$(\sigma^2 I + \Sigma_{zz} - \Sigma_{zu}\Sigma_{uu}^{-1}\Sigma_{uz})P_2 = \sigma^2\Sigma_{zu}(B\Sigma_{uu})^+ P_1. \tag{14}$$

Let Q_z be the (orthogonal) projection onto the column-span of Σ_{zz}, and note that we can write $Q_z = (\Sigma_{zz}^{1/2})^+\Sigma_{zz}^{1/2}$. Define $C = Q_z - (\Sigma_{zz}^{1/2})^+\Sigma_{zu}\Sigma_{uu}^{-1}\Sigma_{uz}(\Sigma_{zz}^{1/2})^+$. Note that $(\Sigma_{zz}^{1/2})^+\Sigma_{zu}\Sigma_{uu}^{-1}\Sigma_{uz}(\Sigma_{zz}^{1/2})^+$ has maximal eigenvalue ρ^2. Then C has column-span equal to that of Σ_{zz}, with minimal nonzero singular value equal to $1 - \rho^2$.

Set $\Gamma = (\sigma^{-1}\Sigma_{zz}^{1/2}C^{1/2})^+ + (\sigma^{-1}\Sigma_{zz}^{1/2}C^{1/2})^\top$. Based on the properties of C that we established, it is evident that Γ has column-span equal to that of Σ_{zz}, and it has minimal singular value bounded below by 2 by Lemma A.9. We have

$$\begin{aligned}
P_2 &= (C^{1/2}\Gamma)^+ C^{1/2}\Gamma P_2 \\
&= (C^{1/2}\Gamma)^+\sigma^{-1}(\Sigma_{zz}^{1/2})^+\left(\sigma^2 I + \Sigma_{zz} - \Sigma_{zu}\Sigma_{uu}\Sigma_{uz}\right)P_2 \\
&= (C^{1/2}\Gamma)^+\sigma^{-1}(\Sigma_{zz}^{1/2})^+\sigma^2\Sigma_{zu}\Sigma_{uu}^{-1}B^+P_1 \\
&= \sigma\Gamma^+(C^{1/2})^+((\Sigma_{zz}^{1/2})^+\Sigma_{zu}\Sigma_{uu}^{-1/2})\Sigma_{uu}^{-1/2}B^+P_1.
\end{aligned}$$

Now $(\Sigma_{zz}^{1/2})^+\Sigma_{zu}\Sigma_{uu}^{-1/2}$ must have maximal singular value equal to ρ, since it gives a symmetric low-rank factorization of $(\Sigma_{zz}^{1/2})^+\Sigma_{zu}\Sigma_{uu}^{-1}\Sigma_{uz}(\Sigma_{zz}^{1/2})^+$. Hence, we finally have the bound

$$\|P_2\|_2 \leq \frac{\sigma\rho}{2\sqrt{1-\rho^2}}\|\Sigma_{uu}^{-1/2}B^+\|_2\|P_1\|_2.$$

□

Lemma A.9. *For any matrix A, the minimal nonzero singular value of $A^+ + A^\top$ is at least 2.*

Proof. Write the compressed SVD of A^+ as $U\Sigma V^\top$, and note that we can write $A^\top = U\Sigma^{-1}V^\top$. It is then evident that the non-zero singular values of $A^+ + A^\top$ are of the form $x + x^{-1}$ for $x > 0$. But $x + x^{-1} \geq 2$ for all $x > 0$. □

A.4. Nonlinear Representation Learning

We here give a proof of Theorem 5.1. For this section, instead of writing $\phi(x)$ to denote the state representation of x, we simply drop explicit reference to ϕ and agree that any system state we discuss has already been mapped to its representation via ϕ. This will simplify notation but doesn't change any of the analysis.

Theorem A.10. *Let $\phi, P, \{L_i, T_i\}, i = 1, \ldots, \tau$ be optimal solutions to the optimization problem (7), and assume that these parameters incur zero loss. Define $V = col(P^\top) + col(L_1^\top) + \cdots + col(L_{\tau-1}^\top)$, and assume that $col(L_\tau^\top) \subset V$. Let Q be the orthogonal projection matrix onto V. Then there exist matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times l}$ such that*

$$Qf(x,u) = Qf(Qx,u) = AQx + Bu.$$

Proof. Zero loss in the objective function implies that

$$Pf(x, \{u_0, \ldots, u_{i-1}) = L_i x + \sum_{k=0}^{i-2} T_{i-1-k} u_k + u_{i-1}$$

for all $x \in \phi(\mathbb{R}^d)$ and $u_j \in \mathbb{R}^l$, $j = 0, \ldots, i-1$.

Fix $2 \le i \le \tau + 1$. By assumption,

$$Pf(x, \{u_0, \ldots, u_{i-1}\}) = L_i x + \sum_{k=0}^{i-2} T_{i-1-k} u_k + u_{i-1}.$$

But we can also express this as follows:

$$\begin{aligned} Pf(x, \{u_0, \ldots, u_{i-1}\}) &= Pf(f(x,u_0), \{u_1, \ldots, u_{i-1}\}) \\ &= L_{i-1} f(x,u_0) + \sum_{k=0}^{i-3} T_{i-2-k} u_{k+1} + u_{i-1} \end{aligned}$$

Equating these two expressions and eliminating like terms gives

$$L_{i-1} f(x,u_0) = L_i x + T_{i-1} u_0.$$

Note that here it is crucial that we couple the T_i matrices. Without the coupling we would not be able to eliminate the terms relating u_i for $i > 0$.

Next, let $\{v_1, \ldots, v_r\}$ be an orthonormal basis for V. Then we can write $Q = \sum_{j=1}^r v_j v_j^\top$. Furthermore, by construction, for each v_j, there exist vectors $y_{j,0}, y_{j,1}, \ldots, y_{j,\tau}$ such that $v_j = P^\top y_{j,0} + \sum_{i=1}^\tau L_i^\top y_{j,i}$. Notice that for $i = 1, \ldots, \tau+1$, since $\text{col}(L_i^\top) \subset V$, it holds that $L_i = L_i Q$. Then we have

$$\begin{aligned} Qf(x,u) &= \sum_{j=1}^r v_j v_j^\top f(x,u) \\ &= \sum_{j=1}^r v_j \left(y_{j,0}^\top Pf(x,u) + \sum_{i=1}^\tau y_{j,i}^\top L_i f(x,u) \right) \\ &= \sum_{j=1}^r v_j \left(y_{j,0}^\top (L_1 x + u) + \sum_{i=1}^\tau y_{j,i}^\top (L_{i+1} x + T_i u) \right) \\ &= \left(\sum_{j=1}^r \sum_{i=0}^\tau v_j y_{j,i}^\top L_{i+1} \right) Qx + \left(\sum_{j=1}^r \sum_{i=0}^\tau v_j y_{j,i}^\top T_i \right) u \end{aligned}$$

where we let $T_0 = I$. Now set $A = \sum_{j=1}^r \sum_{i=0}^\tau v_j y_{j,i}^\top L_{i+1}$ and $B = \sum_{j=1}^r \sum_{i=0}^\tau v_j y_{j,i}^\top T_i$, and we have our result. □

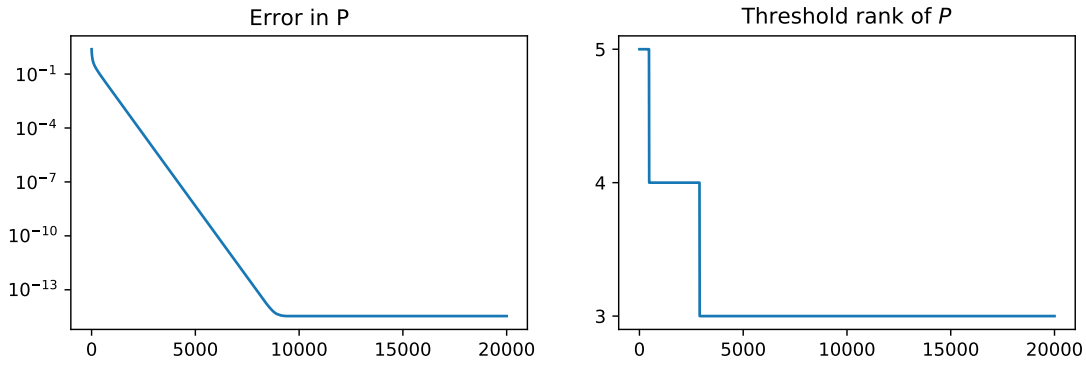

Figure 5. Error and threshold rank of P during training

B. Synthetic Experiments

B.1. CCA Objective

We generate 1000 i.i.d. samples from the model $u \in \mathbb{R}^5 \sim N(0, I)$ and $v = u + \epsilon$, where $\epsilon \sim N(0, \text{diag}(0, .25, .5, .75, 1))$. We compute the empirical canonical correlations $\rho_1, \ldots, \rho_5$ and the corresponding subspaces $\mathcal{C}_1, \ldots, \mathcal{C}_5$, and then set regularizer weights $\lambda_i = (2 - \rho_i^2 - \rho_{i+1}^2)/2$ for $i = 1, \ldots, 4$. For each λ_i, we optimize the finite-sample version of (2) using gradient descent on the parameters $P, Q \in \mathbb{R}^{5\times 5}$ with a learning rate of 0.1 for 20000 steps. We measure the error in P given by $\|P\Sigma_u^{1/2}\Pi_{\mathcal{C}_i^\perp}\|_F$ and the threshold rank of P (the number of singular values of P greater than 10^{-5}). In every case, the error in P converges to 0 and the threshold rank converges to i, thus confirming the conclusion of Theorem 3.3. In Figure 5, we plot of these quantities for $i = 3$ as a function of the gradient step.

B.2. Constructing the Solution to the Inverse Model

In this section we discuss in detail how to obtain the particular minimal-norm solution to (5) that we require in Theorem 4.10. We then discuss synthetic numerical experiments that we conducted to validate the correctness of this result.

It simplifies things to consider the least squares problem

$$\min_{x,y} \|Ax + By - c\|_2^2,$$

where A and B are arbitrary matrices and c is an arbitrary vector. Assume the space of solutions $\{x, y\}$ that have zero error is nonempty (i.e. it is an entire linear space of solutions). We want to select the optimal solution (x^*, y^*) such that for any other optimal solution (x', y'), we have $\|x^*\|_2 \leq \|x'\|_2$ and if $\|x^*\|_2 = \|x'\|_2$ then $\|y^*\|_2 \leq \|y'\|_2$.

We can obtain such a solution by splitting the problem into two stages. First, let x^* be the minimal norm solution of

$$\min_x \|(I - P_B)Ax - (I - P_B)c\|_2^2,$$

where P_B is the orthogonal projection onto the column-span of B. We can compute x^* using standard least squares techniques such as using the singular value decomposition. Then, let y^* be the minimal norm solution to

$$\min_y \|By - P_B c + P_B A x^*\|_2^2.$$

Let us verify that (x^*, y^*) has the desired properties. Let (x', y') be any solution, i.e. $Ax' + By' = c$. Left-multiplying the equation by $I - P_B$, we see that $(I - P_B)Ax' = (I - P_B)c$. By construction, we have that $\|x^*\|_2 \leq \|x'\|_2$. Now assume that $\|x^*\|_2 = \|x'\|_2$. This implies that $x^* = x'$ (the minimum-norm solution is unique). Then we have $By' = P_B c - P_B A x' = P_B c - P_B A x^*$. Again, by construction we have that $\|y^*\|_2 \leq \|y'\|_2$, as desired.

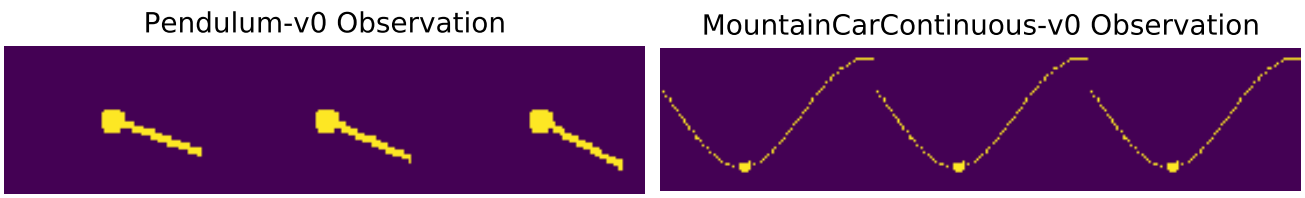

Figure 6. Pixel observations for the environments tested.

C. Experimental Details for Section 6

We here provide further details about the experiments discussed in Section 6.

For each environment, we re-implement the scene renderings to reduce render time and make it compatible with our computing environment. We also add to our environments the ability to reset the system in a desired state (rather than a random state). This allows us to sample initial states from a desired distribution on the state space. We repeat the action three times for both environments, and the resulting concatenated pixel observations have sizes $(64, 192)$ and $(80, 360)$ for the pendulum and mountain car environments, respectively. Figure 6 displays examples of these state observations.

The state representation map ϕ is a basic neural network with two convolutional layers (each with 16 output channels, the first layer with kernel size 8 and stride 4, the second layer with kernel size 4 and stride 2) followed by two fully connected layers each of width 50 for the inverse model, and two fully connected layers of width 1000 and 8 for the forward model. All layers use ReLu activation with no other nonlinearities. For the inverse model, after the final layer, we project to the top 4 right singular directions of the matrix $[P^\top \ L_1^\top \ \cdots \ L_\tau^\top]^\top$, so that in the end, we have a 4-dimensional representation.

To train the representations, we solve (6) and (7) using the Adam optimizer using minibatches of data. We observed that the loss function converged to a nonzero value, which means there may be room to better learn the forward and inverse models if we explore different architecture or training options.

Policy Training The forward and inverse model representations used in Figure 6.2 were trained on a fixed set of batches of trajectories. The inverse model naturally trains over longer trajectories. With the forward model we found it necessary to not just collect many length-1 trajectories, but instead collect longer trajectories and then extract the 1-step trajectories embedded within.

We use the Stable Baselines implementation of TRPO to learn a linear policy for our trained inverse model representations, and we learn a basic fully-connected neural network policy for the forward model representations. For the linear policies, we use all of the default parameters except for the stepsize parameter "vf_stepsize", which we tested over the range of values $[0.00005, 0.0001, 0.0005, 0.001, 0.01, 0.1, 0.5]$. We observed similar performance for all of these choices, but reported the best results in Figure 6.2. For the nonlinear policies, we tested over a larger set of hyperparameters and again reported the best results.