Proceedings of Machine Learning Research vol 144:1–11, 2021

A Data Driven, Convex Optimization Approach to Learning Koopman Operators

M. Sznaier MSZNAIER@COE.NEU.EDU

Abstract

Koopman operators provide tractable means of learning linear approximations of non-linear dynamics. Many approaches have been proposed to find these operators, typically based upon approximations using an a-priori fixed class of models. However, choosing appropriate models and bounding the approximation error is far from trivial. Motivated by these difficulties, in this paper we propose an optimization based approach to learning Koopman operators from data. Our results show that the Koopman operator, the associated Hilbert space of observables and a suitable dictionary can be obtained by solving two rank-constrained semi-definite programs (SDP). While in principle these problems are NP-hard, the use of standard relaxations of rank leads to convex SDPs.

Keywords: Koopman Operators, Learning Nonlinear Dynamics, Nonlinear Identification.

1. Introduction and motivation

Many scenarios involve predicting the output of an unknown non-linear system based on past measurements and some a-priori information. Recently, substantial interest has been devoted to the use of Koopman operator based methods to solve this problem, as a tractable alternative to nonlinear identification. An excellent introduction to the topic is given in Mezić (2013), and more recent references can be found in Lusch et al. (2018); Otto and Rowley (2019). Given a non-linear discrete time system of the form:

$$\boldsymbol{\xi}_{k+1} = f(\boldsymbol{\xi}_k) \text{ where } \boldsymbol{\xi}_k = \begin{bmatrix} \mathbf{x}_{k-r+1}^T & \dots & \mathbf{x}_k^T \end{bmatrix}^T, \; \mathbf{x}_j \in \mathbb{R}^n \tag{1}$$

let $\mathbb{H}$ denote a Hilbert space of functions $\boldsymbol{\psi}(\boldsymbol{\xi})\colon \mathbb{R}^{nr} \to \mathbb{R}^{mr}$(the so called observables). The Koopman $\mathcal{K}$ operator acts on the elements of $\mathbb{H}$, by propagating their values one step into the future:

$$(\mathcal{K} \circ \boldsymbol{\psi})(\boldsymbol{\xi}_k) = (\boldsymbol{\psi} \circ f)(\boldsymbol{\xi}_k) = \boldsymbol{\psi}(\boldsymbol{\xi}_{k+1}) \tag{2}$$

$\mathcal{K}$ is a linear operator, albeit typically infinite dimensional. When it has a countable set of eigenfunctions $\boldsymbol{\phi}_i(.)$ with eigenvalues μ_i, the observables $\boldsymbol{\psi}(.)$ can be propagated as follows. Let $\mathbf{a} = \begin{bmatrix} a_1 \dots \end{bmatrix}^T$ denote the coordinates of $\boldsymbol{\psi}(.)$ in the basis spanned by $\boldsymbol{\phi}(.)$, that is

$$\boldsymbol{\psi}(.) = \sum a_i \boldsymbol{\phi}_i(.) \doteq \boldsymbol{\Phi}(.)\mathbf{a}, \text{ where: } \boldsymbol{\Phi}(.) = \begin{bmatrix} \boldsymbol{\phi}_1(.) \dots \end{bmatrix}$$

Then

$$(\mathcal{K} \circ \boldsymbol{\psi})(.) = \sum a_i \mu_i \boldsymbol{\phi}_i(.) = \boldsymbol{\Phi}(.)\mathbf{M}\mathbf{a}, \text{where } \mathbf{M} = \text{diag}(\mu_i)$$

In particular, if the state $\boldsymbol{\xi} \in \text{span}\{\boldsymbol{\phi}_i\}$, then $\boldsymbol{\xi}_{k+1} = \boldsymbol{\Phi}(\boldsymbol{\xi}_k)\mathbf{M}\mathbf{a}$. While this approach leads for to linear representations of (1), identifying the Koopman eigenfunctions from data is not trivial.

Extended Dynamical Mode Decomposition (EDMD) type approaches seek to identify approximations to Koopman operators over a restricted subspace, defined by the span of a given dictionary $\mathcal{D}(.) \doteq \left[\psi_1(.) \ldots \psi_N(.)\right]$. In this subspace, the Koopman operator can then be approximated by a matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ that propagates the coefficients of the expansion, that is, for $\psi(.) = \mathcal{D}(.)\mathbf{a}$, then $(\mathcal{K} \circ \psi)(.) = \mathcal{D}(.)\mathbf{K}\mathbf{a}$. Typically, given experimental data $\mathbf{X} \doteq \left[\boldsymbol{\xi}_1 \quad \boldsymbol{\xi}_2 \quad \ldots \quad \boldsymbol{\xi}_T\right]$, $\mathbf{K}$ is found by minimizing the one-step prediction error over a set of observables. Specifically, this approach considers m observables $\psi^{(j)}(.) \doteq \mathcal{D}(.)\mathbf{a}_j$, each defined by a coordinate vector $\mathbf{a}_j$, and solves:

$$\mathbf{K} = \underset{K}{argmin} \sum_{j=1}^{m} \sum_{k=1}^{T-1} \|[\mathbf{D}(\boldsymbol{\xi}_{k+1}) - \mathbf{D}(\boldsymbol{\xi}_k)\mathbf{K}]\mathbf{a}_j\|_2^2 \tag{3}$$

where $\mathbf{D}(\boldsymbol{\xi}_k)$ is the matrix obtained by evaluating the dictionary a the point $\boldsymbol{\xi}_k$. EDMD often works well, but requires choosing a suitable dictionary, with the approximation error strongly hinging on this choice. This approximation error can be reduced by considering larger dictionaries, but this may lead to overfitting of the data and poor generalization capabilities Otto and Rowley (2019).

Deep learning motivated approaches use a neural network parameterized by a set of weights W as dictionary. The Koopman operator $\mathbf{K}$ is found by alternatively minimizing the prediction error over W and $\mathbf{K}$. Alternating minimization methods can get trapped in local minima. Further, the issue of which architectures are best suited to represent dynamical systems is largely open. Recent work Lusch et al. (2018); Otto and Rowley (2019) proposed encoder/decoder type architectures that map states $\boldsymbol{\xi}$ to latent variables $\mathbf{y}$ and impose approximately linear dynamics for the evolution of the latter. A salient feature of these approaches is that the states $\boldsymbol{\xi}$ are no longer required to be in the span of the Koopman eigenfuctions. As shown in Otto and Rowley (2019), the use of a nonlinear decoder to map $\mathbf{y}$ back to $\boldsymbol{\xi}$ (as opposed to a linear one if $\boldsymbol{\xi} \in$ span $\{\mathcal{D}\}$) results in substantially smaller dictionaries. Still, these methods require ad-hoc parameter selection (dimension of the latent variables, order of the dynamics) and, as before, can lead to local minima.

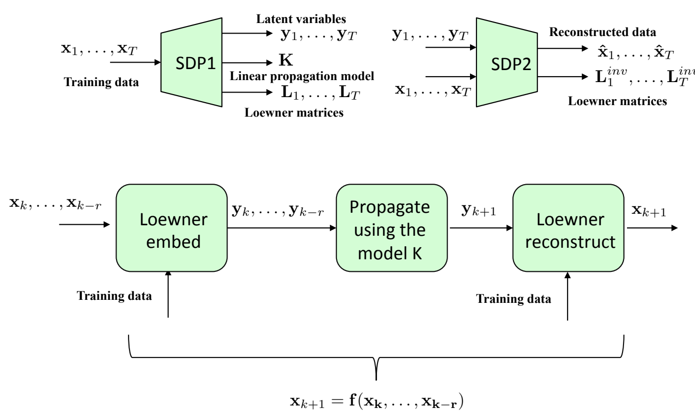

Figure 1: Top: Finding Koopman operators via Semi-Definite Programs. The first SDP (Section 3.3) finds the observables $\mathbf{y}_k$ corresponding to given data $\mathbf{x}_k$, the Koopman operator $\mathbf{K}$, and the Loewner matrices that encode the mapping $\mathbf{x}_k \to \mathbf{y}_k$. The second SDP (Section 3.4) finds the inverse mapping $\mathbf{y}_k \to \mathbf{x}_k$. Bottom: The pipeline to predict $\mathbf{x}_{k+1}/\mathbf{x}_k \ldots, \mathbf{x}_{k-r}$ uses explicit expressions for the predictions of the model $\mathbf{K}$. Thus, it only requires $\mathcal{O}(r)$ operations.

An alternative approach, HAVOK Brunton et al. (2017), rooted in Takens embedding theorem Takens (1981), seeks to model the trajectories of (1) by considering a forced linear system, whose dynamics are precisely the Koopman operator. The states and forcing term are obtained from the singular value decomposition of a Hankel matrix $\mathbf{H}_\mathbf{x}$, formed by delayed measurements of $\mathbf{x}_k$. As shown in Brunton et al. (2017) this approach successfully recovers the trajectories of nonlinear chaotic systems, as linear combinations of a given basis. However, this linear reconstruction, combined with the difficulty of identifying the linear dynamics from the svd of $\mathbf{H}_\mathbf{x}$ Brunton et al. (2017) can lead to high order models (e.g. a 14^{th} order model for the third order Lorentz system).

In this paper, motivated by Fei Xiong et al. (2011); Xiong et al. (2013); Brunton et al. (2017); Lusch et al. (2018); Otto and Rowley (2019), we propose an alternative, convex optimization based, approach to the problem of data-driven identification of Koopman operators. The philosophy, illustrated in Fig. 1, uses delay coordinates, but, as in Lusch et al. (2018); Otto and Rowley (2019) does not impose that the state of the system belongs to span of the Koopman eigenfunctions. Rather, we identify a manifold of latent variables where the dynamics are linear and map back to state-space via a non-linear transformations. The problems of finding the embedding manifold, the associated Koopman operators and the mapping back to state-space are all recast as rank-constrained semi-definite programs (SDPs). In turn, these can be relaxed to convex optimizations using the standard weighted nuclear norm surrogate for rank. Advantages of the proposed approach include:

- A simple rank check allows for certifying that the solution to these convex SDPs is indeed the Koopman operator underlying the given data.
- Does not specify a priory the dimension of the embedding or the order of the dynamics. Rather, both of these can be obtained from the solution to the SDPs.
- Minimizing the order of the linear dynamics leads to simpler models than competing methods.
- In cases where the spectrum of the Koopman operator is not finite, it allows for obtaining finite dimensional approximations with guaranteed approximation error.
- These SDPs have an underlying structure, chordal sparsity, that can be exploited to substantially reduce computational complexity, leading to algorithms that scale linearly with the number of data points.

The paper is organized as follows. In section 2 we formally state the problem under consideration and summarize some needed results on rational interpolation. Section 3 contains the main results of the paper. It shows that a Hilbert space $\mathbb{H}$ of observables, its associated Koopman dictionary and eigenfunctions, and the mapping back to state-space can be found by solving rank-constrained SDPs. Section 4 illustrates the proposed approach with some simple examples. Finally, Section 5 summarizes the paper and points out to directions for extending its results. Due to space constraints all technical proofs are omitted. They can be found in the ArXiV version of the paper https://arxiv.org/pdf/2102.03934.pdf.

2. Preliminaries

For ease of reference, next we summarize our notation and recall some results on interpolation.

2.1. Notation

$\|\mathcal{S}\|$	cardinality of the set $\mathcal{S}$
$\mathbf{x}, \mathbf{M}$	a vector in $\mathbb{R}^n$ (matrix in $\mathbb{R}^{n\times m}$)
$\otimes$	Matrix Kronecker product
$\mathbf{M} \succeq 0$	the matrix $\mathbf{M}$ is positive semidefinite.
$\\|\mathbf{M}\\|_*$	nuclear norm: $\\|\mathbf{M}\\|_* = \Sigma$ singular values of $\mathbf{M}$.
$\mathbf{H}_y^m$	Hankel matrix with m columns associated with a vector sequence $\mathbf{y}(.)$, with block elements $(\mathbf{H}_y^m)_{i,j} = \mathbf{y}_{i+j-1}$
$\textbf{svec}(\mathbf{M})$	(column-wise) vectorization of the unique elements of a symmetric matrix $\mathbf{M}$.
$\textbf{smat}(\mathbf{v})$	create a symmetric matrix $\mathbf{M}$ from the elements of $\mathbf{v}$ such that $\textbf{svec}(\mathbf{M}) = \mathbf{v}$

2.2. Rational Interpolants and Loewner Matrices

Given $2n$ scalar pairs (x_i, y_i), consider the problem of finding a rational function $g(x) \doteq \frac{\sum_{k=1}^m a_k x^k}{\sum_{k=1}^m b_k x^k}$ such that $y_i = g(x_i),\ i = 1, \ldots 2n$. Define the Loewner matrix

$\mathbf{L} \in \mathbb{R}^{n\times n}$ as :

$$\mathbf{L}(x,y) = \begin{bmatrix} \frac{y_1 - y_{n+1}}{x_1 - x_{n+1}} & \frac{y_1 - y_{n+2}}{x_1 - x_{n+2}} & \cdots & \frac{y_1 - y_{2n}}{x_1 - x_{2n}} \\ \frac{y_2 - y_{n+1}}{x_2 - x_{n+1}} & \frac{y_2 - y_{n+2}}{x_2 - x_{n+2}} & \cdots & \frac{y_2 - y_{2n}}{x_2 - x_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{y_n - y_{n+1}}{x_n - x_{n+1}} & \frac{y_n - y_{n+2}}{x_n - x_{n+2}} & \cdots & \frac{y_n - y_{2n}}{x_n - x_{2n}} \end{bmatrix} \tag{4}$$

Then, there exists a rational function of order at most m that interpolates the given data points if and only if $\text{rank}(\mathbf{L}) \leq m - 1$ Antoulas and Anderson (1986); Ionita (2013).

2.3. Statement of the problem

Consider the nonlinear dynamical system:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \ldots, \mathbf{x}_{k-r+1}) \qquad \mathbf{x}_j \in \mathbb{R}^n \tag{5}$$

where both the dynamics $f(.)$ and its order r are unknown. Our goal is to identify its associated Koopman operator, over a suitable space of observables, from experimental data $\mathbf{x}$. Specifically:

Problem 1 *Given a set of N trajectories $\{\mathbf{x}_k^{(i)}\}_{k=1}^{T_i},\ i = 1, \ldots, N$, $\mathbf{x}_k^{(i)} \in \mathbb{R}^n$, find a (functional) dictionary $\mathcal{D}(.)$, a Hilbert space $\mathbb{H}$ of observables $\boldsymbol{\psi}(.)$ of the form:*

$$\begin{aligned} &\boldsymbol{\psi}(\boldsymbol{\xi}_k) \doteq \left[\mathbf{y}_{k-r+1}^T \cdots \mathbf{y}_k^T\right]^T \in \textit{span}\ \{\mathcal{D}(\boldsymbol{\xi}_k)\}\ \textit{with}\ \mathbf{y}_j \in \mathbb{R}^m \\ &\textit{where}\ \boldsymbol{\xi}_k \doteq \left[\mathbf{x}_{k-r+1} \cdots \mathbf{x}_k\right]^T \end{aligned} \tag{6}$$

and an operator $\mathcal{K} \colon \mathbb{H} \to \mathbb{H}$ such that $(\mathcal{K} \circ \boldsymbol{\psi})(\boldsymbol{\xi}_k) = \boldsymbol{\psi}(\boldsymbol{\xi}_{k+1})$.

Problem 1 is reminiscent of EDMD approaches. However, the main difference is that here we seek to learn the dictionary $\mathcal{D}$ and the dimensions of the space $\mathbb{H}$ directly from the data, rather than postulating a fixed dictionary and dimension. Further, if Problem 1 has a solution, the resulting operator $\mathcal{K}$ is indeed the exact Koopman operator in $\mathbb{H}$.

Remark 1 *As stated, Problem 1 is ill posed, since $\|\boldsymbol{\psi}(.)\|$ can be arbitrarily small or large. To avoid this, and with an eye towards reconstruction of $\boldsymbol{\xi}$ from $\boldsymbol{\psi}$, we will impose the additional constraints:*

$$\frac{1}{M_\ell(\boldsymbol{\xi}_i)}\|\boldsymbol{\psi}(\boldsymbol{\xi}_i)-\boldsymbol{\psi}(\boldsymbol{\xi}_j)\|_2 \leq \|\boldsymbol{\xi}_i-\boldsymbol{\xi}_j\|_2 \leq M_u(\boldsymbol{\xi}_i)\|\boldsymbol{\psi}(\boldsymbol{\xi}_i)-\boldsymbol{\psi}(\boldsymbol{\xi}_j)\|_2 \; \forall \boldsymbol{\xi}_j \text{ such that } \|\boldsymbol{\xi}_i-\boldsymbol{\xi}_j\|_2 \leq \delta$$
$$\mathbf{y}(\boldsymbol{\xi}_k)_j^T\mathbf{y}(\boldsymbol{\xi}_k)_j = (\boldsymbol{\xi}_k)_j^T(\boldsymbol{\xi}_k)_j \; j=1,\ldots r, \text{ for all } k \text{ in a given set of ``anchor'' points } \mathbb{I}$$

where $\mathbf{y}_j$ denotes the j^{th} block component of $\boldsymbol{\psi}(\boldsymbol{\xi}_k)$ and the scalar δ and the set of anchor points $\mathbb{I}$ are design hyperparameters. That is, we impose that (a) the mapping $\phi\colon \boldsymbol{\xi} \to \boldsymbol{\psi}$ and its inverse are locally Lipschitz continuous, with Lipschitz constants $M_\ell(\boldsymbol{\xi}_i)$ and $M_u(\boldsymbol{\xi}_i)$; and (b) the function $\boldsymbol{\psi}(.)$ is normalized to have components with unity gain at some given "anchor" points.

3. Learning Koopman Operators via Semi Definite Optimization

In this section we present the main theoretical result of the paper: a reformulation of Problem 1 as a rank minimization subject to a positive semi-definite constraint. Since this problem is generically NP hard, we then develop a tractable convex relaxation, along with optimality certificates.

3.1. Finding Koopman operators as a constrained rank minimization

Consider the following feasibility problem (in $\mathbf{y}, r, m$):

Problem 2 *Given a set of N trajectories $\{\mathbf{x}_\ell^{(i)}\}_{\ell=1}^{T_i},\ i=1,\ldots N$, $\mathbf{x}_\ell^{(i)} \in \mathbb{R}^n$, find scalars r, m and N trajectories $\mathbf{y}_k^{(i)} \in \mathbb{R}^m, k=1,\ldots,T_i$, such that the following holds:*

$$\text{rank }({\mathbf{H}_\mathbf{y}}^{(r+1)}) \leq r, \text{ where } {\mathbf{H}_\mathbf{y}}^{(r+1)} \doteq \begin{bmatrix} \mathbf{H}_{\mathbf{y}^{(1)}}^{(r+1)} \\ \vdots \\ \mathbf{H}_{\mathbf{y}^{(N)}}^{(r+1)} \end{bmatrix} \text{ and } \mathbf{H}_{\mathbf{y}^{(i)}}^{(r+1)} \doteq \begin{bmatrix} \mathbf{y}_1^{(i)} & \mathbf{y}_2^{(i)} & \cdots & \mathbf{y}_{r+1}^{(i)} \\ \mathbf{y}_2^{(i)} & \mathbf{y}_3^{(i)} & \cdots & \mathbf{y}_{r+2}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}_{T_i-r}^{(i)} & \mathbf{y}_{T_i-r+1}^{(i)} & \cdots & \mathbf{y}_{T_i}^{(i)} \end{bmatrix} \tag{7}$$

$$\left.\begin{array}{l} \|\mathbf{x}_s-\mathbf{x}_t\|_2 \leq M_u(\mathbf{x}_s)\|\mathbf{y}_s-\mathbf{y}_t\|_2 \\ \|\mathbf{y}_s-\mathbf{y}_t\|_2 \leq M_\ell(\mathbf{x}_s)\|\mathbf{x}_s-\mathbf{x}_t\|_2 \end{array}\right\} \forall (s,t) \text{ such that } \|\mathbf{x}_s-\mathbf{x}_t\|_2 \leq \delta \tag{8}$$

$$\|\mathbf{y}_s\|_2 = \|\mathbf{x}_s\|_2 \text{ for all } s \in \mathbb{I} \tag{9}$$

As shown next, the solution to Problem 1 (e.g the dictionary $\mathcal{D}$, the embedding Hilbert space $\mathbb{H}$ and the associated Koopman operator) can be constructed from any feasible solution to (7)-(9).

Theorem 1 *Let $(\mathbf{y}_k^{(i)}, r, m)$ denote a feasible solution to (7)-(9) with $\mathbf{y}_k^{(i)} \in \mathbb{R}^m$ and rank$(\mathbf{H}_\mathbf{y}) = r* \leq r$. Let ${\mathbf{H}_\mathbf{y}}^{(r^*+1)}$ and $\mathcal{N}_R({\mathbf{H}_\mathbf{y}}^{(r^*+1)})$ denote the Hankel matrix obtained by rearranging the elements of ${\mathbf{H}_\mathbf{y}}^{(r+1)}$ into r^*+1 columns, and its right null space, respectively. Note that by construction rank$({\mathbf{H}_\mathbf{y}}^{(r^*+1)}) = r^*$ and thus $dim(\mathcal{N}_R({\mathbf{H}_\mathbf{y}}^{(r^*+1)})) \geq 1$. Consider a vector $\mathbf{p} \in \mathcal{N}_R({\mathbf{H}_\mathbf{y}}^{(r^*+1)})$, of the form $\mathbf{p} = \begin{bmatrix} a_0 & \ldots & a_{(r^*-1)} & -1\end{bmatrix}^T$. Let $\rho_j,\ j=1,\ldots,r^*$ denote the roots of the polynomial $\mathcal{P}(\rho) \doteq \rho^{r^*} - \sum_{i=0}^{r^*-1} a_i\rho^i$ and define the r^* vectors*

$$\mathbf{v}_j \doteq \begin{bmatrix} 1 & \rho_j & \rho_j^2 & \ldots \rho_j^{r^*}\end{bmatrix}^T$$

Finally, let $\mathbf{V}$ *denote the Vandermonde matrix* $\mathbf{V} = \begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_{r^*}\end{bmatrix}$. *Then:*

1. *The desired dictionary* $\mathcal{D}(.)$ *has the matrix representation* $\mathbf{D} = \mathbf{V} \otimes \mathbf{I}_m$.
2. *The Hilbert space* $\mathbb{H}$ *of observables is given by span*($\mathcal{D}$)*, with the usual inner product.*
3. *The operator* $\mathcal{K}\colon \mathbb{H} \to \mathbb{H}$ *with the matrix representation* $\mathbf{\Lambda} = diag(\rho_i) \otimes \mathbf{I}_m$ *in the basis defined by the columns of* $\mathbf{V}$ *is the Koopman operator associated with* (1) *in the space* $\mathbb{H}$.

Theorem 1 provides the foundation for constructing the Koopman operator from the solution of an optimization problem, but is of limited practical value, due to several reasons: (i) It does not indicate how to find m, the dimension of $\mathbf{y}_k$, or r, the "memory" of the system, and (ii) it leads to a difficult, non-convex problem. Motivated by Fei Xiong et al. (2011), next we show that Problem 2 is equivalent to a SDP constrained rank-minimization. The starting point is to consider the Kernel matrix with entries $\mathbf{K}_{r,s} \doteq \mathbf{y}_r^T \mathbf{y}_s$, where $\mathbf{y}_r, \mathbf{y}_s$ denote the observables corresponding to points $\mathbf{x}_r, \mathbf{x}_s$ drawn from (not necessarily the same) training trajectories. Let $\mathbf{y}_s^{(i)},\ s = 1, \dots T_i$ denote the observables corresponding to the i$^{\text{th}}$ trajectory and define the $(r+1) \times (r+1)$ Gram matrix $\mathbf{G}^{(i)} \doteq (\mathbf{H}_{\mathbf{y}^{(i)}}^{(r+1)})^T \mathbf{H}_{\mathbf{y}^{(i)}}^{(r+1)}$. The key observation is that both the entries of $\mathbf{G}^{(i)}$ and the argument of the constraints (8)–(9) are *affine* functions of entries of $\mathbf{K}$, leading to the following result:

Theorem 2 *Define the family of Gram matrices:* $\mathbf{G}^{(i)} = (\mathbf{H}_{\mathbf{y}^{(i)}}^{(r+1)})^T \mathbf{H}_{\mathbf{y}^{(i)}}^{(r+1)} = \sum_{\ell=0}^{T_i - r} \mathbf{K}_{\ell,r}^{(i)}$ *where*

$$\mathbf{K}_{\ell,r}^{(i)} = \begin{bmatrix} (\mathbf{y}_\ell^{(i)})^T \mathbf{y}_\ell^{(i)} & (\mathbf{y}_\ell^{(i)})^T \mathbf{y}_{\ell+1}^{(i)} & \cdots & (\mathbf{y}_\ell^{(i)})^T \mathbf{y}_{\ell+r}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ (\mathbf{y}_{\ell+r}^{(i)})^T \mathbf{y}_\ell^{(i)} & (\mathbf{y}_{\ell+r}^{(i)})^T \mathbf{y}_{\ell+1}^{(i)} & \cdots & (\mathbf{y}_{\ell+r}^{(i)})^T \mathbf{y}_{\ell+r}^{(i)} \end{bmatrix}$$

(note that $\mathbf{K}_{\ell,r}^{(i)}$ *are submatrices of* $\mathbf{K}$*). Consider the following rank minimization problem:*

$$r^* = \min_{\mathbf{K} \succeq 0} rank(\mathbf{G} \doteq \begin{bmatrix}(\mathbf{G}^{(1)})^T & \dots & (\mathbf{G}^{(N)})^T\end{bmatrix}^T) \text{ subject to:} \tag{10}$$

$$\left.\begin{array}{l} \frac{1}{M_u^2(\mathbf{x}_s)} \|\mathbf{x}_s - \mathbf{x}_t\|_2^2 \leq K_{s,s} - 2K_{s,t} + K_{t,t} \\ K_{s,s} - 2K_{s,t} + K_{t,t} \leq M_\ell^2(\mathbf{x}_s) \|\mathbf{x}_s - \mathbf{x}_t\|_2^2 \end{array}\right\} \forall (s,t) \text{ such that } \|\mathbf{x}_s - \mathbf{x}_t\|_2 \leq \delta \tag{11}$$

$$K_{s,s} = \|\mathbf{x}_s\|_2^2 \text{ for all } s \in \mathbb{I} \tag{12}$$

Denote by $\mathbf{K}^{(i)}$ *the submatrix of* $\mathbf{K}$ *with entries* $(\mathbf{K}^{(i)})_{\ell,j} = (\mathbf{y}_\ell^{(i)})^T \mathbf{y}_j^{(i)}$*, and let* $m = \max_i \left\{rank(\mathbf{K}^{(i)})\right\}$*. Consider the factorizations* $(\mathbf{Y}^{(i)})^T \mathbf{Y}^{(i)} = \mathbf{K}^{(i)}$ *with* $\mathbf{Y}^{(i)} \in \mathbb{R}^{m \times T_i}$*. Then, if* $r* < r + 1$*, the columns* $\mathbf{y}_k^{(i)}$ *of* $\mathbf{Y}^{(i)}$ *solve Problem 2.*

3.2. Adding a regularization

Theorems 1 indicates how to find the observables $\boldsymbol{\psi}(.) \in \mathbb{H}$ by solving a constrained optimization problem. Further, these constraints guarantee that the mapping $\boldsymbol{\psi}(.)\colon \mathbb{R}^{rn} \to \mathbb{H}$ locally satisfies some Lipschiz and gain constraints. However these constraints alone do not guarantee that $\boldsymbol{\psi}(.)$ is not arbitrarily complex, or even has the same functional form for all $\boldsymbol{\xi}$. These issues can complicate the task of finding an explicit form for the mapping, if one is needed. Next, we briefly indicate how to use additional degrees of freedom available in the problem to guarantee that $\psi(.)$ is the simplest possible mapping, in a sense precisely defined below, and has the same functional form for all $\boldsymbol{\xi}$.

Consider a point $\boldsymbol{\xi}_k \doteq \begin{bmatrix}\mathbf{x}_{k-r+1}^T & \dots & \mathbf{x}_k^T\end{bmatrix}^T$, and for each (block) component $\mathbf{x}_j$, denote by $\mathbb{N}_{\mathbf{x}_j}$ the indexes of its nearest neighbors. Let $\mathbf{K}_{\mathbf{x}_j}$ be matrix with elements $(\mathbf{K}_{\mathbf{x}_j})_{r,s} = \mathbf{x}_r^T\mathbf{x}_s$ for all $r, s \in \left\{j \cup \mathbb{N}_{\mathbf{x}_j}\right\}$. Similarly, given $\psi(\boldsymbol{\xi}_k) \doteq \begin{bmatrix}\mathbf{y}_{k-r+1} \dots \mathbf{y}_k\end{bmatrix}^T$, let $\mathbf{K}_{\mathbf{y}_j}$ be the submatrix of $\mathbf{K}$ with elements $(\mathbf{K}_{\mathbf{y}_j})_{r,s} = \mathbf{y}_r^T\mathbf{y}_s$ for all $r, s \in \left\{j \cup \mathbb{N}_{\mathbf{x}_j}\right\}$. For ease of notation, let $\boldsymbol{\kappa}_{\mathbf{x}}^{(j)} \doteq \mathbf{svec}(\mathbf{K}_{\mathbf{x}_j}) \in \mathbb{R}^q$, $\boldsymbol{\kappa}_{\mathbf{y}}^{(j)} \doteq \mathbf{svec}(\mathbf{K}_{\mathbf{y}_j}) \in \mathbb{R}^q$, where $q \doteq \frac{(|\mathbb{N}_{\mathbf{x}_j}|+1)(|\mathbb{N}_{\mathbf{x}_j}|+2)}{2}$. Note that these vectors contain the unique elements of the matrices $\mathbf{K}_{\mathbf{x}_j}, \mathbf{K}_{\mathbf{x}_j}$. Finally, let $p = \lfloor \frac{q}{2} \rfloor$ and define the Loewner matrix

$$\mathbf{L}_{\mathbf{x}_j} \doteq \begin{bmatrix} \frac{\kappa_{\mathbf{y}_1} - \kappa_{\mathbf{y}_{p+1}}}{\kappa_{\mathbf{x}_1} - \kappa_{\mathbf{x}_{p+1}}} & \frac{\kappa_{\mathbf{x}_1} - \kappa_{\mathbf{x}_{p+2}}}{\kappa_{\mathbf{y}_1} - \kappa_{\mathbf{x}_{p+2}}} & \dots & \frac{\kappa_{\mathbf{y}_1} - \kappa_{\mathbf{y}_q}}{\kappa_{\mathbf{x}_1} - \kappa_{\mathbf{x}_q}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\kappa_{\mathbf{y}_p} - \kappa_{\mathbf{y}_{p+1}}}{\kappa_{\mathbf{x}_p} - \kappa_{\mathbf{x}_{p+1}}} & & \dots & \frac{\kappa_{\mathbf{y}_p} - \kappa_{\mathbf{y}_q}}{\kappa_{\mathbf{x}_p} - \kappa_{\mathbf{x}_q}} \end{bmatrix} \tag{13}$$

where $\boldsymbol{\kappa}_{\mathbf{x}_i}, \boldsymbol{\kappa}_{\mathbf{y}_i}$ denote the i^{th} component of $\boldsymbol{\kappa}_{\mathbf{x}}^{(j)}$ and $\boldsymbol{\kappa}_{\mathbf{y}}^{(j)}$ respectively. From the results in section 2.2, it follow that if $\text{rank}(\mathbf{L}_{\mathbf{x}_j}) < p$, then there exists a rational mapping of degree up to $p-1$ that maps the elements of $\mathbf{K}_{\mathbf{x}_j}$ to those of $\mathbf{K}_{\mathbf{y}_j}$. Further, the degree of this mapping can be minimized by minimizing the rank of $\mathbf{L}_{\mathbf{x}_j}$ with respect to the variables $\boldsymbol{\kappa}_{\mathbf{y}_i}$, leading (locally) to the lowest order rational mapping $\psi(\boldsymbol{\xi}_k)$. If a global, rather than local, rational mapping is desired, a similar idea can be using involving all pairs $\mathbf{x}, \mathbf{y}$, rather than just the nearest neighbors of each point.

3.3. A Convex Relaxation

Theorem 2 allows for reducing Problem 1 to a constrained rank minimization problem. However, this problem is still NP-hard. In order to obtain a tractable relaxation, we will replace the objective (10) by $\sum_{i=1}^{N} \text{rank}(\mathbf{G}^{(i)})$ and add a term of the form $\lambda_1 \sum_{j=1}^{T} \text{rank}(\mathbf{L}_{\mathbf{x}_j})$, where $T = \sum T_i$ is the total number of points. Then, proceeding as in Mohan and Fazel (2012), we will replace rank with a convex surrogate, a weighted nuclear norm, where the weights are updates as each step of the algorithm. Finally, in order to handle outliers, we will consider a "soft" version of (11)-(12), where these are added to the objective as penalties. The complete algorithm is outlined in Algorithm 1. It is worth noting that if the algorithm yields a solution $\mathbf{G}$ with $\text{rank}(\mathbf{G}) < r$, this certifies that m is indeed the Koopman operator. On the other hand, if the algorithm yields a solution $\mathbf{G}$ with minimum singular value σ_{min}, then an r^{th} order approximate model m_r can be obtained by performing PCA on $\mathbf{G}$. In this case the approximation error is bounded (in the Hankel norm sense) by $\sqrt{\sigma_{\text{min}}}$.

3.4. Mapping observables to states

The approach presented in Section 3 finds the observables $\psi(\boldsymbol{\xi}_k)$ corresponding to a given trajectory $\boldsymbol{\xi}_k,\ k = 1, \dots T$. However, it does not explicitly provide a method for mapping a given $\psi(\boldsymbol{\xi})$, obtained for instance by using the Koopman operator to propagate a trajectory in observable space, back to the corresponding point $\boldsymbol{\xi}$ in state space. Motivated by Roweis and Saul (2000) we propose to find (pointwise) the mapping $\psi \rightarrow \boldsymbol{\xi}$ by locally approximating the mapping between the embedded space and ambient space kernels, $\mathbf{K}_{\mathbf{y}}$ and $\mathbf{K}_{\mathbf{x}}$, with a rational function. Specifically, given a point $\mathbf{y}^* \in \mathbb{R}^m$, let $\mathcal{N}_{\mathbf{y}^*} \doteq \left\{\mathbf{y}_k \colon \|\mathbf{y}^* - \mathbf{y}_k\|_2^2 \le \delta\right\}$ and denote by $\mathcal{X}$ its preimage. We propose to estimate $\mathbf{x}^*$ by first finding $K_{\mathbf{x}_i,\mathbf{x}^*}$, the elements of $\mathbf{K}_x$ corresponding to $\mathbf{x}_i^T\mathbf{x}^*,\ \forall \mathbf{x}_i \in \mathcal{X}$ and then

Algorithm 1 Reweighted $\|.\|_*$ based Koopman Identification

initialize: $iter = 0, \mathbf{W}_0 = \mathbf{I}; \mathbf{V}_0^{(j)} = \mathbf{I}, j = 1, \dots, T; \lambda_1, \lambda_2, \lambda_3, \delta \leftarrow$ hyperparameters, $\mathbb{NN} = \{(s,t)\colon \|\mathbf{x}_s - \mathbf{x}_t\|_2 \leq \delta\}$, $\sigma \leftarrow$small number, $r \leftarrow$ upper bound on system order.

Repeat: Solve

$$\begin{array}{l} \min_{\mathbf{K}^{(i)} \succeq 0} \quad \|\mathbf{W}_{iter}\mathbf{G}\|_* + \lambda_1 \sum_{j=1}^{T} \|\mathbf{V}_{iter}^{(j)} \mathbf{L}_{\mathbf{x}_j}\|_* + \lambda_2 \sum_{s \in \mathbb{I}} (K_{s,s} - \|\mathbf{x}_s\|_2^2)^2 \\ + \lambda_3 \sum_{r,t \in \mathbb{NN}} \max \left\{ 0, \frac{1}{M_u^2(\mathbf{x}_s)} \|\mathbf{x}_s - \mathbf{x}_t\|_2^2 - K_{s,s} + 2K_{s,t} - K_{t,t} \right\} \\ + \lambda_3 \sum_{r,t \in \mathbb{NN}} \max \left\{ 0, K_{s,s} - 2K_{s,t} + K_{t,t} - M_\ell^2(\mathbf{x}_s) \|\mathbf{x}_s - \mathbf{x}_t\|_2^2 \right\} \end{array}$$

Update

$$\mathbf{W}_{(iter+1)} = \left(\frac{\mathbf{G} + \sigma \mathbf{I}}{\|\mathbf{G} + \sigma \mathbf{I}\|} \right)^{-1}, \ \mathbf{V}_{(iter+1)}^{(j)} = \left(\frac{\mathbf{L}_{\mathbf{x}_j} + \sigma \mathbf{I}}{\|\mathbf{L}_{\mathbf{x}_j} + \sigma \mathbf{I}\|} \right)^{-1}, \ iter = iter + 1$$

Until: rank$(\mathbf{G}) < r$.

$[\mathbf{U}^{(i)}, \mathbf{S}^{(i)}, (\mathbf{U}^{(i)})^T] \leftarrow$ svd$(\mathbf{K}^{(i)})$, $\mathbf{S}^{(i)} \leftarrow \frac{\mathbf{S}^{(i)}}{\|\mathbf{S}^{(i)}\|}$, $r_k^{(i)} \leftarrow \min r\colon \sum_{j=1}^{r} \mathbf{S}_{jj}^{(i)} \geq 0.99$

$\mathbf{Y}^{(i)} \leftarrow [\mathbf{U}^{(i)}(:, 1 : r_k)]^T$

$[\mathbf{U}_\mathbf{G}, \mathbf{R}, \mathbf{V}_\mathbf{G}^T] \leftarrow$ svd$(\mathbf{G})$, $\mathbf{m} \leftarrow \mathbf{V}_\mathbf{G}(:, r+1)$

Output: embeddings $\mathbf{Y}^{(\mathbf{i})}$, model $\mathbf{m}$.

finding $\mathbf{x}^*$ by factorizing $\mathbf{K}_x$. Note that, in order to get a valid kernel compatible with the priors, the elements $K_{\mathbf{x}_i, \mathbf{x}^*}$ should be such that the completed matrix $\mathbf{K}_\mathbf{x} \succeq 0$, rank$(\mathbf{K}) \leq n$, and the constraints (8)-(9) are satisfied. As shown next, under the assumption that the mapping $G\colon \mathbf{K}_\mathbf{y} \to \mathbf{K}_\mathbf{x}$ is rational, then $\mathbf{x}^*$ can be found by solving a rank minimization problem subject to semi-definite constraints.

Consider the Kernel matrices $\mathbf{K}_\mathbf{x}, \mathbf{K}_\mathbf{y} \in \mathbb{R}^{(|\mathcal{X}|+1) \times (|\mathcal{X}|+1)}$, where the entries have been ordered so that the elements of the form $\mathbf{y}_i^T \mathbf{y}^*$ and $\mathbf{x}_i^T \mathbf{x}^*$ appear in the first row and column. As before, for ease of notation, let $\boldsymbol{\kappa}_\mathbf{x} = \mathbf{svec}(\mathbf{K}_\mathbf{x})$, $\boldsymbol{\kappa}_\mathbf{y} = \mathbf{svec}(\mathbf{K}_\mathbf{y})$. Note that $\boldsymbol{\kappa}_\mathbf{x}, \boldsymbol{\kappa}_\mathbf{y} \in \mathbb{R}^q$, with $q \doteq \frac{(|\mathcal{X}|+1)(|\mathcal{X}|+2)}{2}$, and that all inner products involving $\mathbf{y}^*$ and $\mathbf{x}^*$ appear in the first $|\mathcal{X}| + 1$ elements of $\boldsymbol{\kappa}_\mathbf{y}$ and $\boldsymbol{\kappa}_\mathbf{x}$. Let $p = \lfloor \frac{q}{2} \rfloor$ and consider the following rank minimization problem:

$$\min_{\boldsymbol{\kappa}_\mathbf{x}} \text{rank } (\mathbf{L}) \text{subject to:} \tag{14}$$

$$\left. \begin{array}{l} \kappa_{\mathbf{x}_1} - 2\boldsymbol{\kappa}_{\mathbf{x}_i} + \boldsymbol{\kappa}_{\mathbf{x}_j} \leq \delta^2 \\ \boldsymbol{\kappa}_{\mathbf{x}_1} - 2\boldsymbol{\kappa}_{\mathbf{x}_i} + \boldsymbol{\kappa}_{\mathbf{x}_j} \leq M_u^2(\mathbf{x}_i)(\kappa_{\mathbf{y}_1} - 2\boldsymbol{\kappa}_{\mathbf{y}_i} + \boldsymbol{\kappa}_{\mathbf{y}_j}) \\ \boldsymbol{\kappa}_{\mathbf{y}_1} - 2\boldsymbol{\kappa}_{\mathbf{y}_i} + \boldsymbol{\kappa}_{\mathbf{y}_j} \leq M_\ell^2(\mathbf{x}_i)(\kappa_{\mathbf{x}_1} - 2\boldsymbol{\kappa}_{\mathbf{x}_i} + \boldsymbol{\kappa}_{\mathbf{x}_j}) \end{array} \right\} \begin{array}{l} i = 2, \dots, |\mathcal{X}| + 1 \\ j = \frac{(2|\mathcal{X}|+4-i)(i-1)}{2} + 1 \end{array} \tag{15}$$

$$\mathbf{K}_\mathbf{x} \doteq \mathbf{smat}(\boldsymbol{\kappa}_\mathbf{x}) \succeq 0, \ \text{rank}(\mathbf{K}_\mathbf{x}) \leq n, \boldsymbol{\kappa}_{\mathbf{x}_1} = \boldsymbol{\kappa}_{\mathbf{y}_1} \tag{16}$$

$$\mathbf{L} = \begin{bmatrix} \frac{\boldsymbol{\kappa}_{\mathbf{x}_1} - \boldsymbol{\kappa}_{\mathbf{x}_{p+1}}}{\boldsymbol{\kappa}_{\mathbf{y}_1} - \boldsymbol{\kappa}_{\mathbf{y}_{p+1}}} & \frac{\boldsymbol{\kappa}_{\mathbf{x}_1} - \boldsymbol{\kappa}_{\mathbf{x}_{p+2}}}{\boldsymbol{\kappa}_{\mathbf{y}_1} - \boldsymbol{\kappa}_{\mathbf{y}_{p+2}}} & \cdots & \frac{\boldsymbol{\kappa}_{\mathbf{x}_1} - \boldsymbol{\kappa}_{\mathbf{x}_q}}{\boldsymbol{\kappa}_{\mathbf{y}_1} - \boldsymbol{\kappa}_{\mathbf{y}_q}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\boldsymbol{\kappa}_{\mathbf{x}_p} - \boldsymbol{\kappa}_{\mathbf{x}_{p+1}}}{\boldsymbol{\kappa}_{\mathbf{y}_p} - \boldsymbol{\kappa}_{\mathbf{y}_{p+1}}} & & \cdots & \frac{\boldsymbol{\kappa}_{\mathbf{x}_p} - \boldsymbol{\kappa}_{\mathbf{x}_q}}{\boldsymbol{\kappa}_{\mathbf{y}_p} - \boldsymbol{\kappa}_{\mathbf{y}_q}} \end{bmatrix}$$

Theorem 3 *Let $\boldsymbol{\kappa}^*_{\mathbf{x}}, \mathbf{L}^*$ denote the solution to (14)-(16). If rank$(\mathbf{L}^*) < p$, then (i) there exist a rational function $g(.)$ of degree at most p such that $g(\kappa_{\mathbf{y}_i}) = \kappa_{\mathbf{x}_i}$; and (ii) the vector $\mathbf{x}^*$ defined by the first row of $\mathbf{X}$, where $\mathbf{X}^T\mathbf{X} = \mathbf{K_x}$ satisfies constraints (8)-(9) in Problem 2.*

Relaxing the rank in (14) and (16) to a weighed nuclear norm, leads to an algorithm similar to Algorithm 1, based on solving a sequence of SDPs until rank deficient matrices $\mathbf{L}, \mathbf{K_x}$ are obtained.

4. Illustrative Examples

Example 1: Lorentz Attractor. In this example we consider the Lorentz chaotic system:

$$\dot{x}_1 = \sigma(x_2 - x_1); \dot{x}_2 = x_1(\rho - x_3) - x_2; \dot{x}_3 = x_1 x_2 - \beta x_3; \tag{17}$$

with parameters $\sigma = 28, \rho = 10, \beta = \frac{8}{3}$. We used 400 points of the trajectory starting at $[-10.38\ -4.5366\ 35.1640]^T$, uniformly sampled every 0.0271 seconds to find the embeddings, and matlab's command `ssest` to estimate an 7$^{\text{th}}$ order model. Fig 2(a) shows the training and one step ahead reconstructed data, that is the results of applying the encoder/decoder illustrated on the top of Fig. 1 to (i) train, (ii) project the training data, (iii) perform a one step ahead prediction and (iv) lift back. Figure 2(b) shows the predictions obtained using the pipeline at the bottom of Fig. 1, for points not part of the training data. As shown there, the proposed pipeline is indeed able to predict with reasonable accuracy the one step ahead value of the trajectory, using a 7th order Koopman operator. For comparison, Brunton et al. (2017) uses a 14th order model.

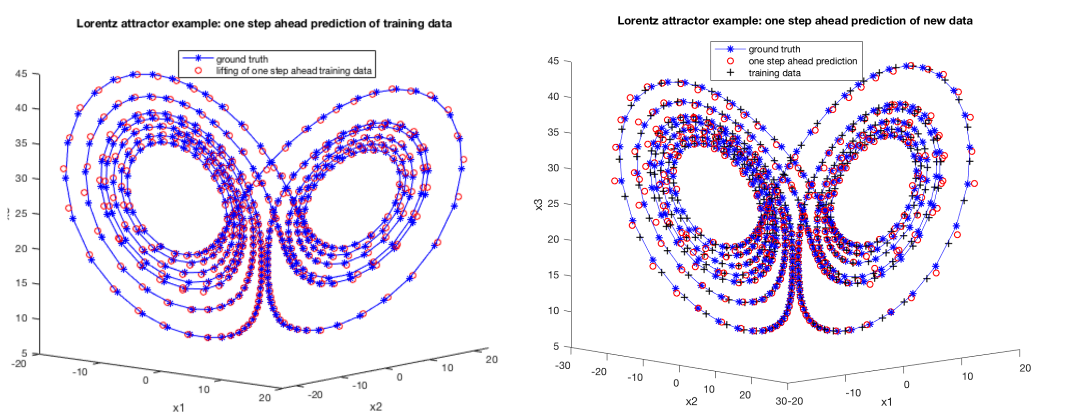

Figure 2: Lorentz attractor: Left: one step ahead prediction of training data. Right: one step ahead prediction of new data.

Example 2: The Duffing Oscillator. Here we consider the system[1]:

$$\dot{x}_1 = x_2; \dot{x}_2 \ \ = -0.5x_2 - x_1 - x_1^3 + 0.42x_3;\ \dot{x}_3 = x_4;\ \dot{x}_4 \ \ = -x_3 \tag{18}$$

1. The conventional Duffing equation is a forced oscillator. Here we use the last two equations to generate the forcing term $sin(t)$.

In this case, Algorithm 1 yielded an embedding $\mathbf{y} \in \mathbb{R}^3$. We then used matlab's command `ssest` to estimate a second order model for each component of $\mathbf{y}$. Fig 3 (left) shows the one step ahead prediction of the training data. The right panel in Fig. 3 shows the predictions obtained using the pipeline at the bottom of Fig. 1, for points not part of the training data. As before, the proposed pipeline successfully predicts the next point in the trajectory.

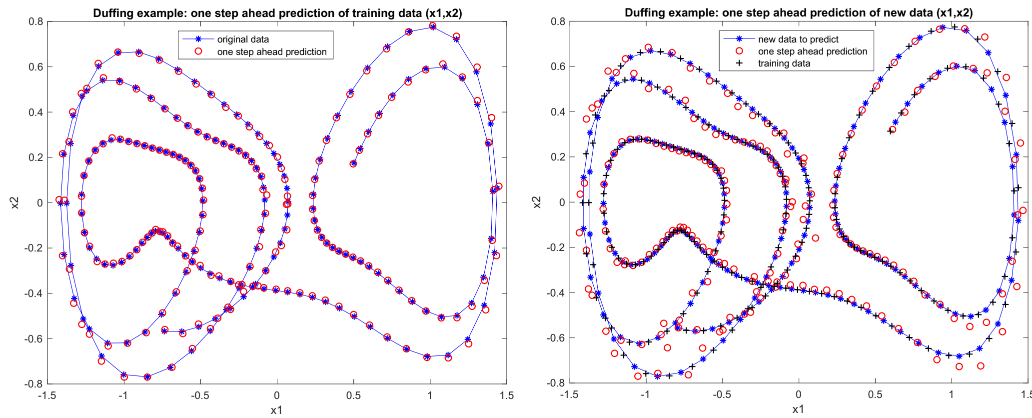

Figure 3: Duffing oscillator one step ahead predictions of (left) training data and (right) new data.

5. Conclusions

This paper proposes a convex optimization approach to learning Koopman operators from data. The main idea is to use delay coordinates and nonlinear, kernel based embeddings to recast the problem as a rank-constrained optimization. In turn, this optimization can be relaxed to a tractable semi-definite program. Salient features of this approach are its ability to certify that the solution to this SDP indeed solves the original problem, and the fact that neither the order of the embedding nor of the dynamics governing their evolution need to be specified a-priori. Further, by seeking embeddings that minimize the order of these dynamics, it leads to simpler models than those obtain for instance by simply factoring the Hankel matrix of the observed data. The effectiveness of the proposed technique was illustrated with two examples that exhibit chaotic behavior. In principle the approach proposed here requires solving a large SDP, and it is well known that SDPs have poor scaling properties. However, as shown in the Appendix, the specific optimization arising in this paper exhibits an underlying sparse structure (chordal sparsity) than can be exploited to obtain algorithms whose complexity scales linearly with the number of data points, when these SDPs are solved using an ADMM based method such as the one proposed in Zheng et al. (2020). This extension along with an extension to piecewise linear dynamics on the manifold, is currently being explored.

6. Acknowledgements

This work was supported in part by NSF grants IIS–1814631, ECCS–1808381 and CNS–2038493, and AFOSR grant FA9550-19-1-0005.

References

A. C. Antoulas and B. D. O. Anderson. On the scalar rational interpolation problem. *IMA J. of Mathematical Control and Information*, 3:61–88, 1986.

Steven L. Brunton, Bingni W. Brunton, Joshua L. Proctor, Eurika Kaiser, and J. Nathan Kutz. Chaos as an intermittently forced linear system. *Nature Communications*, 8(1):19, 2017.

Jerome Dancis. Positive semidefinite completions of partial hermitian matrices. *Linear Algebra and its Applications*, 175:97 – 114, 1992.

Fei Xiong, O. I. Camps, and M. Sznaier. Low order dynamics embedding for high dimensional time series. In *2011 International Conference on Computer Vision*, pages 2368–2374, 2011.

Robert Grone, Charles R Johnson, Eduardo M Sá, and Henry Wolkowicz. Positive definite completions of partial hermitian matrices. *Linear Algebra Appl.*, 58:109–124, 1984.

A. C. Ionita. *Lagrange Rational Interpolation and Its Applications to Approximation of Large-Scale Dynamical System*. PhD thesis, Rice University, 2013.

Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1), 2018.

Igor Mezić. Analysis of Fluid Flows via Spectral Properties of the Koopman Operator. *Annual Review of Fluid Mechanics*, 45:357–378, January 2013.

Karthik Mohan and Maryam Fazel. Iterative reweighted algorithms for matrix rank minimization. *Journal of Machine Learning Research*, 13(110):3441–3473, 2012. URL http://jmlr.org/papers/v13/mohan12a.html.

Samuel E. Otto and Clarence W. Rowley. Linearly recurrent autoencoder networks for learning dynamics. *SIAM Journal on Applied Dynamical Systems*, 18(1):558–593, 2019.

Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

Floris Takens. Detecting strange attractors in turbulence. In David Rand and Lai-Sang Young, editors, *Dynamical Systems and Turbulence, Warwick 1980*, pages 366–381, Berlin, Heidelberg, 1981. Springer Berlin Heidelberg.

F. Xiong, Y. Cheng, O. Camps, M. Sznaier, and C. Lagoa. Hankel based maximum margin classifiers: A connection between machine learning and wiener systems identification. In *52nd IEEE Conference on Decision and Control*, pages 6005–6010, 2013.

Y. Zheng, G. Fantuzzi, A. Papachristodoulou, P. Goulart, and A. Wynn. Chordal decomposition in operator-splitting methods for sparse semidefinite programs. *Mathematical Programming*, 180 (1):489–532, 2020.