FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

FACULTY OF SCIENCES

CHAIR IN APPLIED ANALYSIS FAU

SYSTEM IDENTIFICATION BY KOOPMAN OPERATORS: QUANTITATIVE ANALYSIS

Christophe Zhang Applied Analysis - AvH Professorship, FAU Erlangen-Nürnberg

Enrique Zuazua Applied Analysis - AvH Professorship, FAU Erlangen-Nürnberg

A system identification problem

We consider the following system identification problem: given an unknown differential equation

$$\dot{x} = f(x), \quad x \in \Omega \subset \mathbb{R}^d, \tag{1}$$

with flow $\Phi_f^t : \Omega \to \Omega$, **recover the vector field $f : \Omega \to \mathbb{R}^d$ from data on the trajectories, and predict the future evolution of the system.**

Koopman operator: exchanging finite dimension for linearity

It is well-known that f can also be seen as the velocity field of the linear transport equation

$$\frac{\partial \varphi}{\partial t}(t,x) = f(x) \cdot \nabla \varphi(t,x)$$

with the associated linear transport operator and semigroup:

$$\mathcal{K} := f(x) \cdot \nabla, \quad e^{t\mathcal{K}} \varphi_0 = \varphi_0 \circ \Phi_f^t,$$

which are sometimes referred to as the **Koopman operators** (after B.O.Koopman, see [4]). Now, applying the Koopman operators on the coordinate functions $g_i(x) = x_i$, we get :

$$\mathcal{K} g_i = f_i, \quad e^{t\mathcal{K}} g_i = g_i \circ \Phi_f^t = \left(\Phi_f^t\right)_i, \quad i = 1 \cdots d. \tag{2}$$

Thus, applying $\mathcal{K}$ to the g_i yields f and its flow Φ_f^t. The system identification problem can then be solved by **using data to recover an approximation of $\mathcal{K}$ and applying it to the g_i.**

Galerkin projections and system identification

Due to the infinite dimensional nature of $\mathcal{K}$, data-driven methods focus on recovering *finite dimensional approximations of* $\mathcal{K}$. A common choice is the **Galerkin projection** $\mathcal{K}_N$ on a N-dimensional space $V_N \subset L^2(\Omega)$ of functions. Noting Π_N the orthogonal projection on V_N:

$$\mathcal{K}_N = \Pi_N \mathcal{K} \Pi_N. \tag{3}$$

Well chosen Galerkin projections can provide **approximate solutions to the system identification problem**. Indeed, (2) can be approximated by applying the Galerkin projection $\mathcal{K}_N$ to the coordinate functions g_i. It is then natural to focus on the cases where $g_i \in V_N$, in which case we get, from (2) and (3),

$$\mathcal{K}_N g_i = \Pi_N(\mathcal{K} g_i) = \Pi_N f_i, \quad i = 1 \cdots d. \tag{4}$$

If V_N is well chosen (see "Quantitative analysis"), one can approximate the semigroup $e^{t\mathcal{K}}$ with $e^{t\mathcal{K}_N}$:

$$e^{t\mathcal{K}_N} g_i \approx \Pi_N(e^{t\mathcal{K}} g_i) = \Pi_N \left(\Phi_f^t\right)_i, \quad i = 1 \cdots d. \tag{5}$$

In turn, for relevant choices of V_N (see "Quantitative analysis"), $\Pi_N f_i$ and $\Pi_N \left(\Phi_f^t\right)_i$ provide good approximations of the vector field components f_i and the flow components $\left(\Phi_f^t\right)_i$ respectively.

Generator Extended Dynamic Mode Decomposition (gEDMD)

It remains now to recover $\mathcal{K}_N$. Data-driven approximations of Galerkin projections $\mathcal{K}_N$ can be computed by the gEDMD (generator Extended Dynamic Mode Decomposition algorithm, see [6, 3]).

Step 1 Draw m random points $\{x^k \in \Omega, 1 \le k \le m\}$ uniformly and independently.

Step 2 From the corresponding solutions $x^k(t)$ of (1), approximate the velocities $y^k \approx \dot{x}^k(0) = f(x^k)$.

Step 3 Choose a basis $\psi_1, \cdots, \psi_N$ of V_N. These functions will be used to enrich the observations $\{x^k, y^k\} = \{x^k, f(x^k)\}$ with additional data, in order to learn the action of $\mathcal{K}$ on V_N.

Step 4 Define the data matrices, sampling the values of the ψ_i and the $\mathcal{K}\psi_i$ on the data set $\{x^k, y^k\}$:

$$A_N^m = \begin{pmatrix} \psi_1(x^1) & \cdots & \psi_1(x^m) \\ \vdots & \ddots & \vdots \\ \psi_N(x^1) & \cdots & \psi_N(x^m) \end{pmatrix}, \; G_N^m = \begin{pmatrix} y^1 \cdot \nabla\psi_1(x^1) & \cdots & y^m \cdot \nabla\psi_1(x^m) \\ \vdots & \ddots & \vdots \\ y^1 \cdot \nabla\psi_N(x^1) & \cdots & y^m \cdot \nabla\psi_N(x^m) \end{pmatrix}.$$

Step 5 Perform the linear regression $K_N^m := \underset{K \in M_N(\mathbb{R})}{\operatorname{argmin}} \left\| K A_N^m - G_N^m \right\|^2 = G_N^m \left(A_N^m\right)^\dagger$.

The matrix K_N^m defines an operator $\mathcal{K}_N^m$ on V_N. We prove the following (see also [6, 5]):

Proposition.[8] *If for $m > N$ large enough, $A_N^m \left(A_N^m\right)^\top$ is invertible with probability 1, then, in any operator norm on V_N, there exists a constant $C(V_N) > 0$ such that*

$$\|\mathcal{K}_N^m - \mathcal{K}_N\| \le C(V_N) m^{-\frac{1}{2}} \tag{6}$$

Implementation of system identification

The system identification problem can thus be solved in a data-driven way:

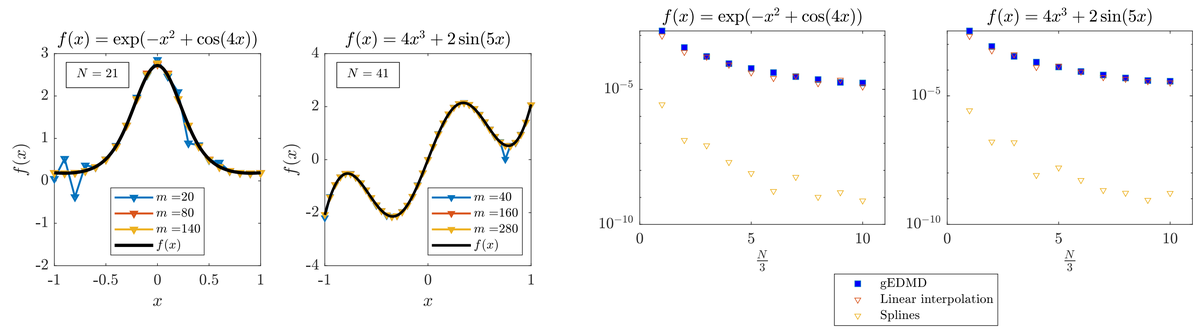
Step 1 Choose a subspace V_N such that $g_i \in V_N$

Step 2 Perform gEDMD with V_N. The matrix K_N^m thus obtained represents an approximation of $\mathcal{K}_N$.

Step 3 Then, recalling (4), (5) and (6),

$$\|\mathcal{K}_N^m g_i - \Pi_N f_i\|_{L^2} \le C_1(V_N) \|g_i\|_{L^2} m^{-\frac{1}{2}}, \quad \|e^{t\mathcal{K}_N^m} g_i - e^{t\mathcal{K}_N} g_i\|_{L^2} \le C_2(V_N) \|g_i\|_{L^2} m^{-\frac{1}{2}}, \quad i = 1 \cdots d. \tag{7}$$

These figures show the recovery of some 1-D functions by gEDMD with linear finite elements (left), and a comparison of the approximation error for different methods on the same data set (right).



Quantitative analysis

The gEDMD algorithm guarantees a good approximation of the Galerkin projection $\mathcal{K}_N$. We have seen above that some Galerkin projections can provide approximations of f and Φ_f^t. Now, we need to quantify this approximation error. We have, from (4) and (5), for all $i = 1 \cdots d$:

$$\|\mathcal{K}_N g_i - f_i\|_{L^2} \le \|(\Pi_N - I) f_i\|_{L^2}, \quad \|e^{t\mathcal{K}_N} g_i - (\Phi_f^t)_i\|_{L^2} \le \|\Pi_N (e^{t\mathcal{K}\Pi_N} - e^{t\mathcal{K}}) g_i\|_{L^2} + \|(\Pi_N - I) e^{t\mathcal{K}} g_i\|_{L^2}. \tag{8}$$

In [8] we outline two typical situations where the right-hand sides of (8) can be estimated:

Koopman invariant subspaces. [1] This ideal situation, where both right-hand sides become 0, seems too rare to be used consistently.

Classical approximation spaces. Classical function spaces such as finite elements or polynomials guarantee estimates on the projection error for any f regular enough. When $\Omega = [0, L]^d$ and V_N is the space of finite elements of degree at most k in each variable, on a uniform rectangular grid of size h, we have, for all $i = 1 \cdots d$:

$$\|\mathcal{K}_N g_i - f_i\|_{L^2} \le C_1 h^{k+1} |f_i|_{k+1}, \quad \|(\Pi_N - I) e^{t\mathcal{K}} g_i\|_{L^2} \le C_2 h^{k+1}, \quad \left\|(e^{t\mathcal{K}\Pi_N} - e^{t\mathcal{K}}) g_i\right\|_{L^2} \le C(f) h^k. \tag{9}$$

In this case we have estimates both on the recovery of f and its flow Φ_f^t, by considering the *linear* operator $\mathcal{K}_N$ and its exponential $e^{t\mathcal{K}_N}$. Theoretically this presents an advantage over direct interpolation methods, which only provide an approximation of f, and would require the integration of a *nonlinear* ODE to recover the flow. However, to guarantee at most $\varepsilon > 0$ error in (9), N must satisfy

$$N \ge C k^d \varepsilon^{-\frac{d}{k+1}},$$

which is exponential in the dimension d. This is the *curse of dimensionality*. *Even in small dimension d the resulting large dimension N makes the recovery of the flow by $e^{t\mathcal{K}_N^m}$ computationally expensive.* Moreover, the 1-D numerical examples above illustrate the superiority of direct interpolation methods over gEDMD with linear finite elements for recovering f. In fact, for a sample $\{x^k, f(x^k),\, k = 1 \cdots m\}$, gEDMD with higher order finite elements can reach comparable accuracy. However, in that case, due to the linear regression performed in Step 5 of the algorithm, the computational cost is of the order $\mathcal{O}(m^3)$. This is *more expensive than the linear complexity of direct interpolation methods.*

To sum up, gEDMD with finite elements produces **reliable approximations** of f and its flow Φ_f^t. However the method quickly becomes **computationally intractable**.

In small dimension d **direct interpolation methods are a better and cheaper option** to recover f.

Perspectives

- Find **optimal lower-dimensional subspaces of functions** V_N.
- Develop a theoretical understanding of **deep-learning methods to find these subspaces** [7].
- Alleviate computations using **kernel methods** [2].

References

[1] Steven L. Brunton et al. "Koopman Invariant Subspaces and Finite Linear Representations of Nonlinear Dynamical Systems for Control". In: *PLOS ONE* 11.2 (2016), pp. 1–19. DOI: 10.1371/journal.pone.0150171.

[2] Stefan Klus, Feliks Nüske, and Boumediene Hamzi. "Kernel-Based Approximation of the Koopman Generator and Schrödinger Operator". In: *Entropy* 22.7 (June 2020), p. 722.

[3] Stefan Klus et al. "Data-driven approximation of the Koopman generator: Model reduction, system identification, and control". In: *Physica D: Nonlinear Phenomena* 406 (2020), p. 132416.

[4] B. O. Koopman. "Hamiltonian Systems and Transformation in Hilbert Space". In: *Proceedings of the National Academy of Sciences* 17.5 (1931), pp. 315–318.

[5] Milan Korda and Igor Mezić. "On Convergence of Extended Dynamic Mode Decomposition to the Koopman Operator". In: *Journal of Nonlinear Science* 28.2 (Apr. 2018), pp. 687–710.

[6] Matthew O. Williams et al. "Extending Data-Driven Koopman Analysis to Actuated Systems". In: *IFAC-PapersOnLine* 49.18 (2016), pp. 704–709.

[7] E. Yeung, S. Kundu, and N. Hodas. "Learning Deep Neural Network Representations for Koopman Operators of Nonlinear Dynamical Systems". In: *2019 American Control Conference (ACC)*. 2019, pp. 4832–4839.

[8] Christophe Zhang and Enrique Zuazua. *A quantitative analysis of Koopman operator methods for system identification and predictions*. in preparation. 2021.

caa-avh.nat.fau.eu 04/2021

DDS Department of DATA SCIENCE

Unterstützt von / Supported by

Alexander von Humboldt Stiftung/Foundation