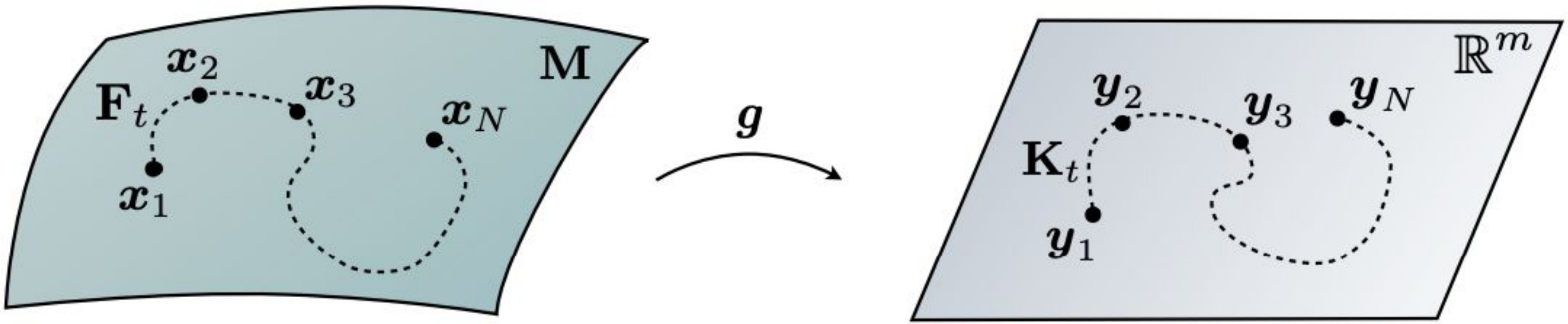
Koopman Dynamics

Varun Varanasi
S&DS 631: Optimization and Computation

Motivation: Dynamical Systems

- Examples:
  - Predator-Prey Models
  - Neuron Activity
  - Lorenz Systems
- Goals:
  - Predict
  - Control
  - Governing behavior
- Difficulties:
  - Nonlinear
  - Many sources of uncertainty
  - General solutions are still an open problem

Solution: Koopman Dynamics



- In higher dimensions we can unpack nonlinearities
- Extend regions where linear approximations are appropriate
- Provide physical interpretability to complex systems

Background: Dynamical System Theory

Dynamical System Definition

- System relating the position of a point in ambient space over time

Continuous

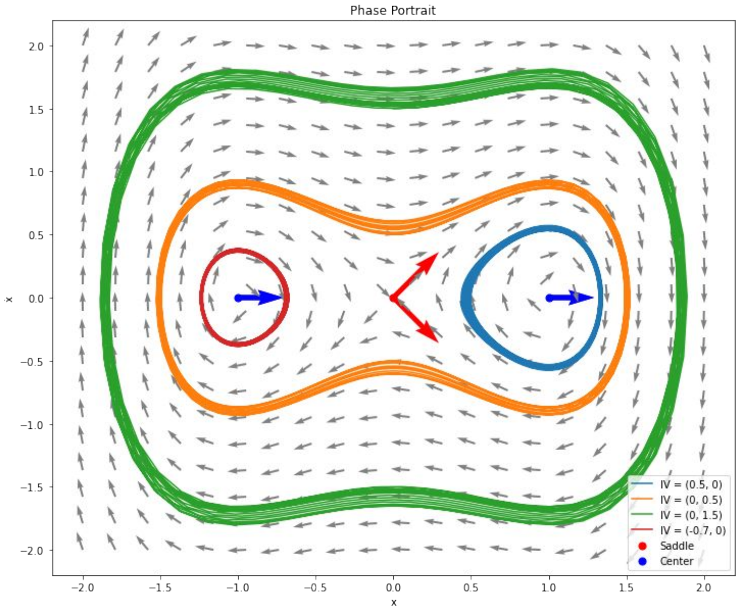
$$\frac{d}{dt}\boldsymbol{x}(t) = \boldsymbol{f}(\boldsymbol{x}(t), t; \beta)$$

Discrete

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}(\boldsymbol{x}_k)$$

Dynamical System Example

$$\frac{d^2x}{dt^2} = -\frac{dV(x)}{dx} \qquad V(x) = \frac{x^4}{4} - \frac{x^2}{2}$$



Linear Dynamical Systems

Natural System

General Form:

$$\frac{d}{dt}\boldsymbol{x} = \boldsymbol{A}\boldsymbol{x}$$

Closed Form:

$$x(t + t_0) = e^{\boldsymbol{A}t}\boldsymbol{x}(t_0)$$

Uncoupled System

Decoupled Form:

$$\frac{d}{dt}z_i = \lambda_i z_j$$

Spectral Decomposition Form:

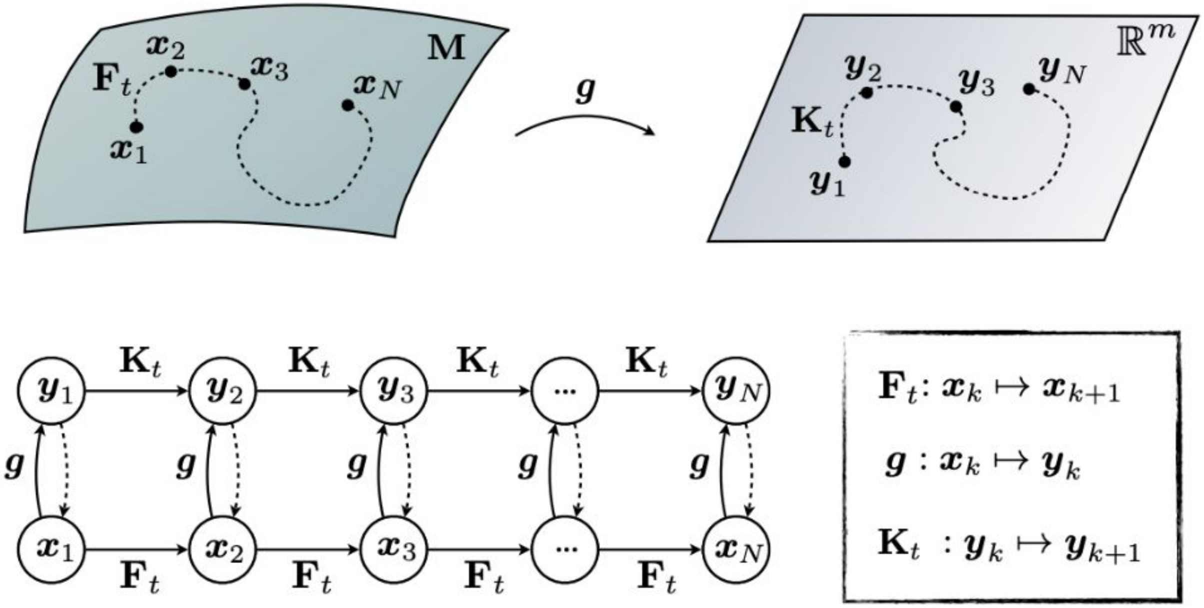
$$x(t + t_0) = \boldsymbol{Q}e^{\boldsymbol{\Lambda}t}\boldsymbol{Q}^{-1}\boldsymbol{x}(t_0)$$

Koopman Operator Theory

High Level Idea

1. Extend phase space of system to a Hilbert Space
2. Use spectral decomposition on the operator in this space to find a basis of eigenfunctions
3. Use these functions to decouple and linearize the system

Mathematical Foundations



Eigenfunctions

Continuous

$$\frac{d}{dt}\psi(\boldsymbol{x}) = \mathcal{K}\psi(\boldsymbol{x}) = \lambda\psi(\boldsymbol{x})$$

Discrete

$$\psi(\boldsymbol{x}_{k+1}) = \mathcal{K}\psi(\boldsymbol{x}_k) = \lambda\psi(\boldsymbol{x}_k)$$

Partial Differential Equation Formulation

$$\lambda\psi(\boldsymbol{x}) = \frac{d}{dt}\psi(\boldsymbol{x}) = \nabla\psi(\boldsymbol{x}) \cdot \dot{\boldsymbol{x}} = \nabla\psi(\boldsymbol{x}) \cdot \boldsymbol{f}(\boldsymbol{x})$$

Construction of Eigenfunctions

Continuous

$$\mathcal{K}(\psi_1, \psi_2) = \frac{d}{dt}(\psi_1\psi_2)$$
$$= \psi_1\dot{\psi_2} + \dot{\psi_1}\psi_2$$
$$= \lambda_2\psi_1\psi_2 + \lambda_1\psi_1\psi_2$$
$$= (\lambda_1 + \lambda_2)\psi_1\psi_2$$

Discrete

$$\mathcal{K}_t\left(\psi_1(\boldsymbol{x})\psi_2(\boldsymbol{x})\right) = \psi_1(\boldsymbol{F}_t(\boldsymbol{x}))\psi_2(\boldsymbol{F}_t(\boldsymbol{x}))$$
$$= \lambda_1\lambda_2\psi_1(\boldsymbol{x})\psi_2(\boldsymbol{x})$$

We can create eigenfunctions from other eigenfunctions!

Koopman Mode Decomposition

General Observation

$$\boldsymbol{g}(\boldsymbol{x}) = \begin{bmatrix} g_1(\boldsymbol{x}) \\ g_2(\boldsymbol{x}) \\ \cdots \\ g_p(\boldsymbol{x}) \end{bmatrix}$$

Eigenvalue Decomposition

$$g_i(\boldsymbol{x}) = \sum_j^{\infty} v_{ij}\psi_j$$

Koopman Mode Decomposition

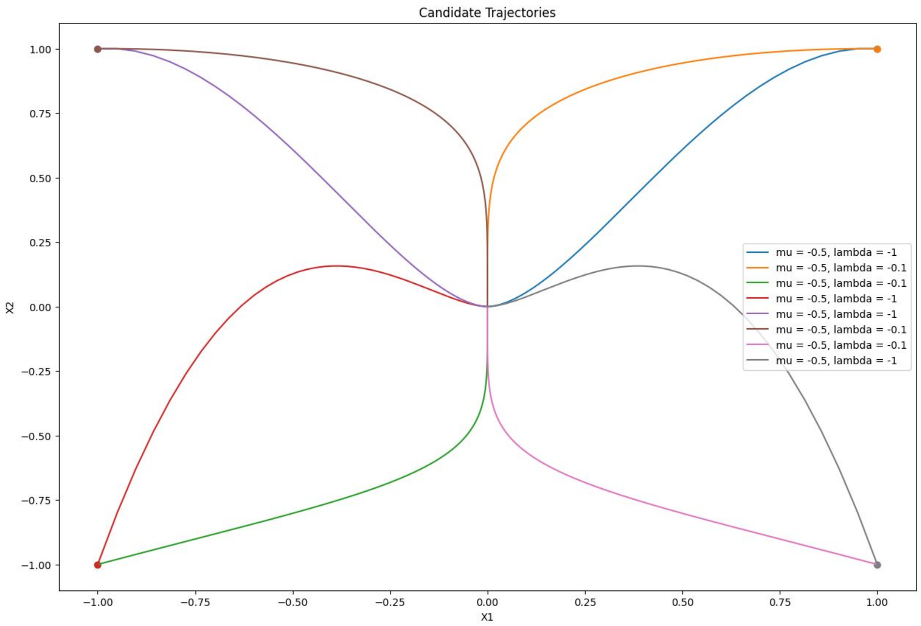
$$\boldsymbol{g}(\boldsymbol{x}) = \begin{bmatrix} \sum_j^{\infty} v_{1j}\psi_j \\ \sum_j^{\infty} v_{2j}\psi_j \\ \cdots \\ \sum_j^{\infty} v_{pj}\psi_j \end{bmatrix} = \sum_j^{\infty} \psi_j(\boldsymbol{x})\boldsymbol{v}_j$$

Dynamics via Koopman Modes

$$\begin{aligned}
\boldsymbol{g}(x(t)) &= \mathcal{K}^t \boldsymbol{g}(\boldsymbol{x}_0) \\
&= \mathcal{K}^t \sum_j^{\infty} \psi_j(\boldsymbol{x}_0)\boldsymbol{v}_j \\
&= \sum_j^{\infty} \mathcal{K}^t \psi_j(\boldsymbol{x}_0)\boldsymbol{v}_j \\
&= \sum_j^{\infty} \lambda_j^t \psi_j(\boldsymbol{x}_0)\boldsymbol{v}_j
\end{aligned}$$

Koopman Embedding Example

$$\dot{x}_1 = \mu x_1 \qquad \dot{x}_2 = \lambda(x_2 - x_1^2)$$



Koopman Embedding Example

Koopman Embedding

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \end{bmatrix} = \begin{bmatrix} \mu & 0 & 0 \\ 0 & \lambda & -\lambda \\ 0 & 0 & 2\mu \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \end{bmatrix}$$

Eigenfunctions

$$\psi_\mu = x_1 \qquad \psi_\lambda = x_2 - b x_1^2 \qquad b = \frac{\lambda}{\lambda - 2\mu}$$

Dynamic Mode Decomposition

High Level Summary

- Calculate spatio-temporal relationships between data
- Find best fit matrix between data at adjacent time steps
- Simplify best fit matrix calculation by projecting it onto a reduced dimensional space

Problem Setup

Data:

$$\boldsymbol{X} = \begin{bmatrix} x(t_1) & x(t_2) & \dots & x(t_m) \end{bmatrix} \qquad \boldsymbol{X}' = \begin{bmatrix} x(t_1') & x(t_2') & \dots & x(t_m') \end{bmatrix}$$

Best Fit Matrix:

$$\boldsymbol{X}' = \boldsymbol{A}\boldsymbol{X}$$

Optimization Problem:

$$\boldsymbol{A} = \arg\min_{A} ||\boldsymbol{X}' - \boldsymbol{A}\boldsymbol{X}||_F = \boldsymbol{X}'\boldsymbol{X}^\dagger$$

DMD Algorithm

- Step 1: Singular Value Decomposition

$$X \approx \tilde{U}\tilde{\Sigma}\tilde{V}^*$$

- Step 2: Pseudo-Inverse

$$A = X'\tilde{V}\tilde{\Sigma}^{-1}\tilde{U}^* \quad \longrightarrow \quad \tilde{A} = \tilde{U}^* A \tilde{U} = \tilde{U}^* X' \tilde{V}\tilde{\Sigma}^{-1}$$

- Step 3: Spectral Decomposition

$$\tilde{A} = W\Lambda W^{-1}$$

- Step 4: Reconstructing DMD Modes

$$\Phi = X'\tilde{V}\tilde{\Sigma}^{-1}W$$

- Step 5: Recovering System State

$$x_k = \sum_{j=1}^{r} \phi_j \lambda_j^{k-1} b_j = \Phi\Lambda^{k-1}b$$

Recent Advances

- Connection between Koopman eigenfunctions and partitions of phase space
- Connections to Ergodic Theory
- Deep Learning applications for Koopman Eigenfunction discovery
- Applications of Koopman Dynamics to Control Systems

References

- S. L. Brunton and J. N. Kutz. Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control. Cambridge University Press, 2019.
- S. L. Brunton, M. Budisi, E. Kaiser, J. N. Kutz Modern Koopman Theory for Dynamical Systems.
- S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz. Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. PLoS ONE, 11(2):e0150171, 2016
- G. Snyder and Z. Song. Koopman Operator Theory for Nonlinear Dynamic Modeling using Dynamic Mode Decomposition