Representation Learning for Dynamical Systems

Pietro Novelli

4th Symposium on ML and Dynamical Systems Fields Institute — Jul 9, 2024

From the previous episodes

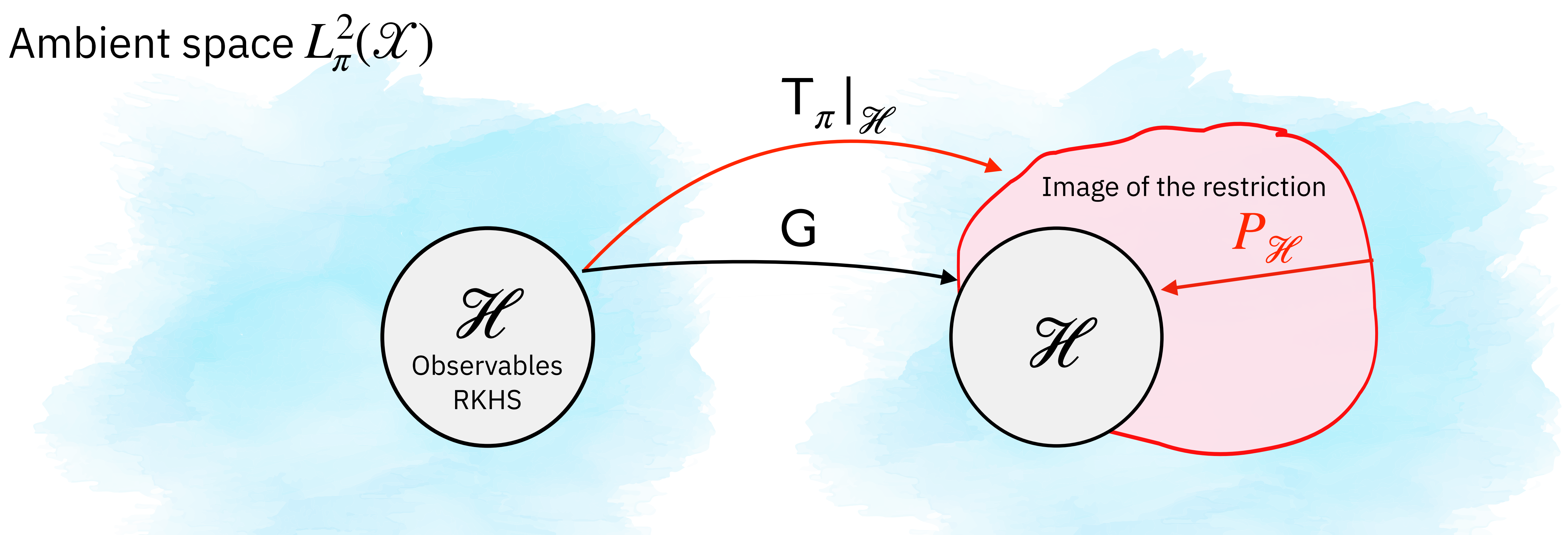
- The **transfer operator** T describes the evolution of any scalar function of the state in a suitable set $\mathscr{F}$.

$$(\mathsf{T}f)(x) = \mathbb{E}[\, f(X_{t+1}) \mid X_t = x\,], \quad f \in \mathscr{F}$$

- If $\mathscr{F}$ it is large enough, the transfer operator offers a comprehensive characterization of a (stochastic) dynamical system *as a whole*.
- Provides a **global** linearization of the dynamics.
- Its spectral decomposition yield dynamic modes, for interpretability and control.

Subspace approach

- Idea: approximate T_π at least on a subset $\mathscr{H} \subset L^2_\pi$.
- We choose $\mathscr{H}$ to be a **Reproducing Kernel Hilbert Space**.
- Linearly parametrized functions $\langle w, \phi(x) \rangle$ for some $w \in \mathscr{H}$.
- $\phi : \mathscr{X} \to \mathscr{H}$ is called **feature map**. $\mathscr{H}$ can be finite or infinite dim.

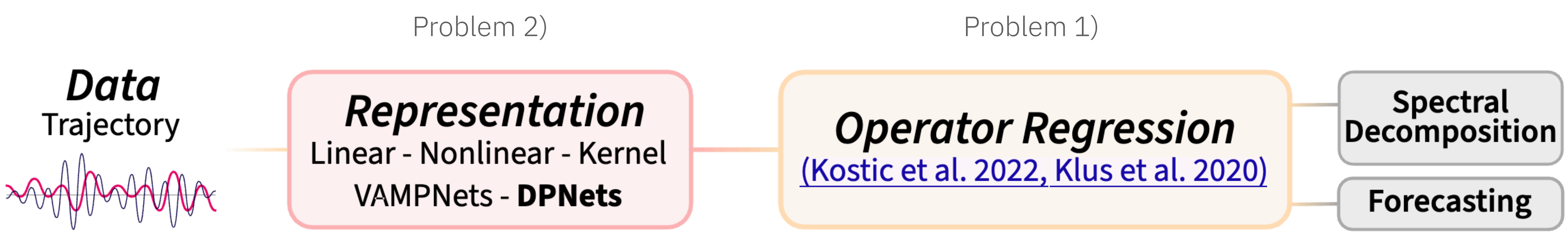


Two *learning* problems

1) Assuming somebody gave us a "good" hypothesis space $\mathscr{H}$, learn an estimator $\hat{\mathrm{G}}$ of the restriction $\mathrm{T}_{\pi}|_{\mathscr{H}}$ from data.

2) When no off-the-shelf $\mathscr{H}$ does the job*, learn it as well.

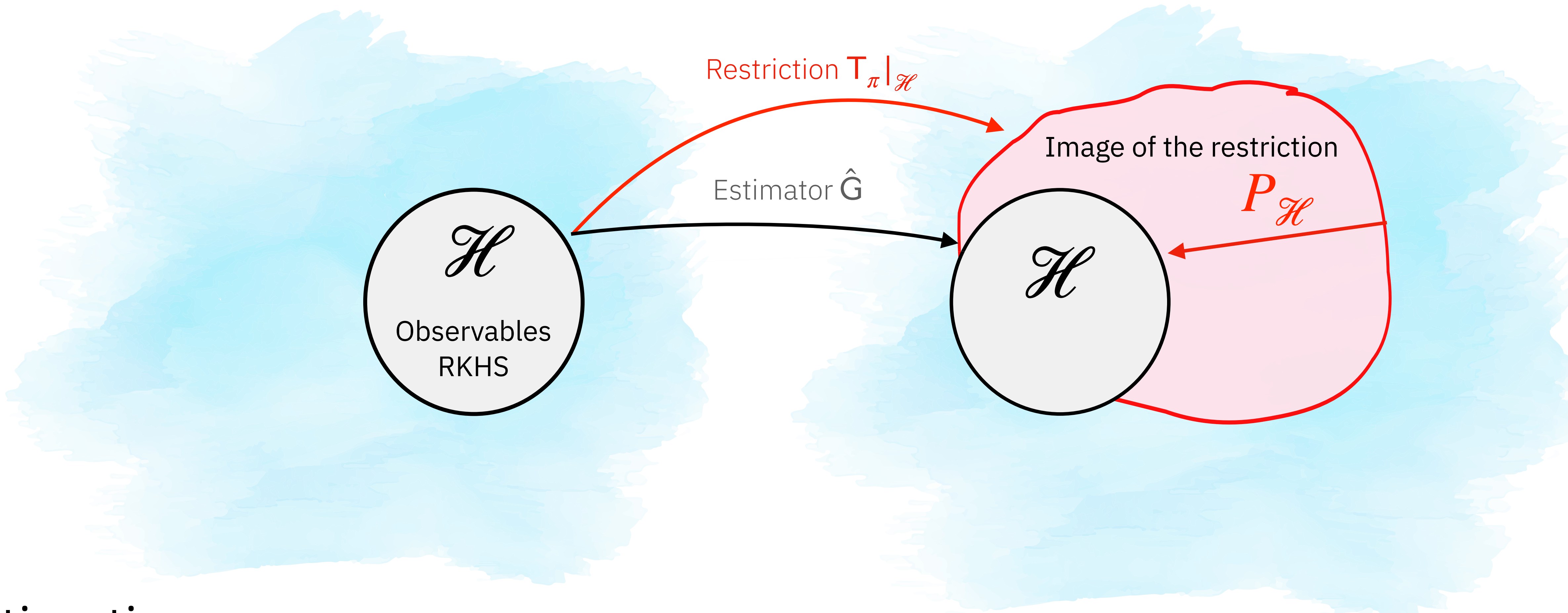
* e.g. when dealing with structured data like graphs, images, or signals.



Cool! How?

Statistical learning to the rescue

Ambient space $L^2_\pi(\mathcal{X})$



Estimation error

$$\|\mathsf{T}_{\pi|_{\mathcal{H}}} - \hat{\mathsf{G}}\| \leq \underbrace{\|(I - P_{\mathcal{H}})\mathsf{T}_{\pi|_{\mathcal{H}}}\|}_{\text{Representation error}} + \underbrace{\|P_{\mathcal{H}}\mathsf{T}_{\pi|_{\mathcal{H}}} - \mathsf{G}\|}_{\text{Estimator bias}} + \underbrace{\|\mathsf{G} - \hat{\mathsf{G}}\|}_{\text{Estimator variance}}$$

All norms are the operator norm $\|\cdot\| = \|\cdot\|_{\mathcal{H}\to L^2_\pi}$

Representation Learning

Kostic, Novelli, Grazzi, Lounici, and Pontil — ICLR '24

$$\|\mathsf{T}_{\pi|_{\mathscr{H}}} - \hat{\mathsf{G}}\|_{\mathscr{H}\to L^2_\pi} \leq \underbrace{\|(I-P_{\mathscr{H}})\mathsf{T}_{\pi|_{\mathscr{H}}}\|}_{\text{Representation error}} + \underbrace{\|P_{\mathscr{H}}\mathsf{T}_{\pi|_{\mathscr{H}}} - \mathsf{G}\|}_{\text{Estimator bias}} + \underbrace{\|\mathsf{G} - \hat{\mathsf{G}}\|}_{\text{Estimator variance}}$$

Our approach looks for an empirical estimator of the **representation error** via the following upper and lower bounds (consequence of the norm change from $\mathscr{H}$ to L^2_π)

$$\|(I-P_{\mathscr{H}})\mathsf{T}_\pi P_{\mathscr{H}}\|^2 \lambda^+_{\min}(C_{\mathscr{H}}) \leq \|(I-P_{\mathscr{H}})\mathsf{T}_{\pi|_{\mathscr{H}}}\|^2 \leq \|(I-P_{\mathscr{H}})\mathsf{T}_\pi P_{\mathscr{H}}\|^2 \lambda_{\max}(C_{\mathscr{H}})$$

Representation Learning

Kostic, Novelli, Grazzi, Lounici, and Pontil — ICLR '24

$$\|\mathsf{T}_{\pi|_{\mathcal{H}}} - \hat{\mathsf{G}}\|_{\mathcal{H}\to L^2_\pi} \leq \underbrace{\|(I - P_{\mathcal{H}})\mathsf{T}_{\pi|_{\mathcal{H}}}\|}_{\text{Representation error}} + \underbrace{\|P_{\mathcal{H}}\mathsf{T}_{\pi|_{\mathcal{H}}} - \mathsf{G}\|}_{\text{Estimator bias}} + \underbrace{\|\mathsf{G} - \hat{\mathsf{G}}\|}_{\text{Estimator variance}}$$

Our approach looks for an empirical estimator of the **representation error** via the following upper and lower bounds (consequence of the norm change from $\mathcal{H}$ to L^2_π)

$$\|(I - P_{\mathcal{H}})\mathsf{T}_\pi P_{\mathcal{H}}\|^2 \lambda^+_{\min}(C_{\mathcal{H}}) \leq \|(I - P_{\mathcal{H}})\mathsf{T}_{\pi|_{\mathcal{H}}}\|^2 \leq \|(I - P_{\mathcal{H}})\mathsf{T}_\pi P_{\mathcal{H}}\|^2 \lambda_{\max}(C_{\mathcal{H}})$$

If the population covariance $C_{\mathcal{H}} = I$ upper and lower bounds match, and the Eckart-Young-Mirsky theorem on $P_{\mathcal{H}}\mathsf{T}_\pi P_{\mathcal{H}}$ provides a way to minimize the **representation error**.

Fast forward a couple of lemmas:

The **representation error** can be minimized by optimizing

$$\mathscr{P}[\phi] := \| C_{X_t}^{\dagger/2} C_{X_t X_{t+1}} C_{X_{t+1}}^{\dagger/2} \|_{\mathrm{HS}}^2 - \gamma \left[\mathscr{R}(C_{X_t}) + \mathscr{R}(C_{X_{t+1}}) \right]$$

Where $C_{X_t} = \mathbb{E}_{x \sim X_t} \left[\phi(x) \otimes \phi(x) \right]$ and $C_{X_t X_{t+1}} = \mathbb{E}_{(x,y) \sim (X_t, X_{t+1})} \left[\phi(x) \otimes \phi(y) \right]$, and $\mathscr{R}$ is a regularization term encouraging $C_{X_t} \simeq I$

Fast forward a couple of lemmas:

The **representation error** can be minimized by optimizing

$$\mathscr{P}[\phi] := \|C_{X_t}^{\dagger/2} C_{X_t X_{t+1}} C_{X_{t+1}}^{\dagger/2}\|_{\mathrm{HS}}^2 - \gamma \left[\mathscr{R}(C_{X_t}) + \mathscr{R}(C_{X_{t+1}})\right]$$

Where $C_{X_t} = \mathbb{E}_{x \sim X_t}\left[\phi(x) \otimes \phi(x)\right]$ and $C_{X_t X_{t+1}} = \mathbb{E}_{(x,y) \sim (X_t, X_{t+1})}\left[\phi(x) \otimes \phi(y)\right]$, and $\mathscr{R}$ is a regularization term encouraging $C_{X_t} \simeq I$

- $\mathscr{P}[\phi]$ is a functional of the feature map through the covariances $C_{X_t}, C_{X_t X_{t+1}}$.
- By learning ϕ we learn $\mathscr{H} = \overline{\mathrm{span}\left(\phi(x) \,|\, x \in \mathscr{X}\right)}$.
- The pseudo-inverses make $\mathscr{P}[\phi]$ it **nasty** to optimize with gradient descent.

Fast forward a couple of lemmas:

The **representation error** can be minimized by optimizing

$$\mathscr{P}[\phi] := \|C_{X_t}^{\dagger/2} C_{X_t X_{t+1}} C_{X_{t+1}}^{\dagger/2}\|_{\mathrm{HS}}^2 - \gamma \left[\mathscr{R}(C_{X_t}) + \mathscr{R}(C_{X_{t+1}})\right]$$

Where $C_{X_t} = \mathbb{E}_{x \sim X_t}\left[\phi(x) \otimes \phi(x)\right]$ and $C_{X_t X_{t+1}} = \mathbb{E}_{(x,y)\sim(X_t, X_{t+1})}\left[\phi(x) \otimes \phi(y)\right]$, and $\mathscr{R}$ is a regularization term encouraging $C_{X_t} \simeq I$

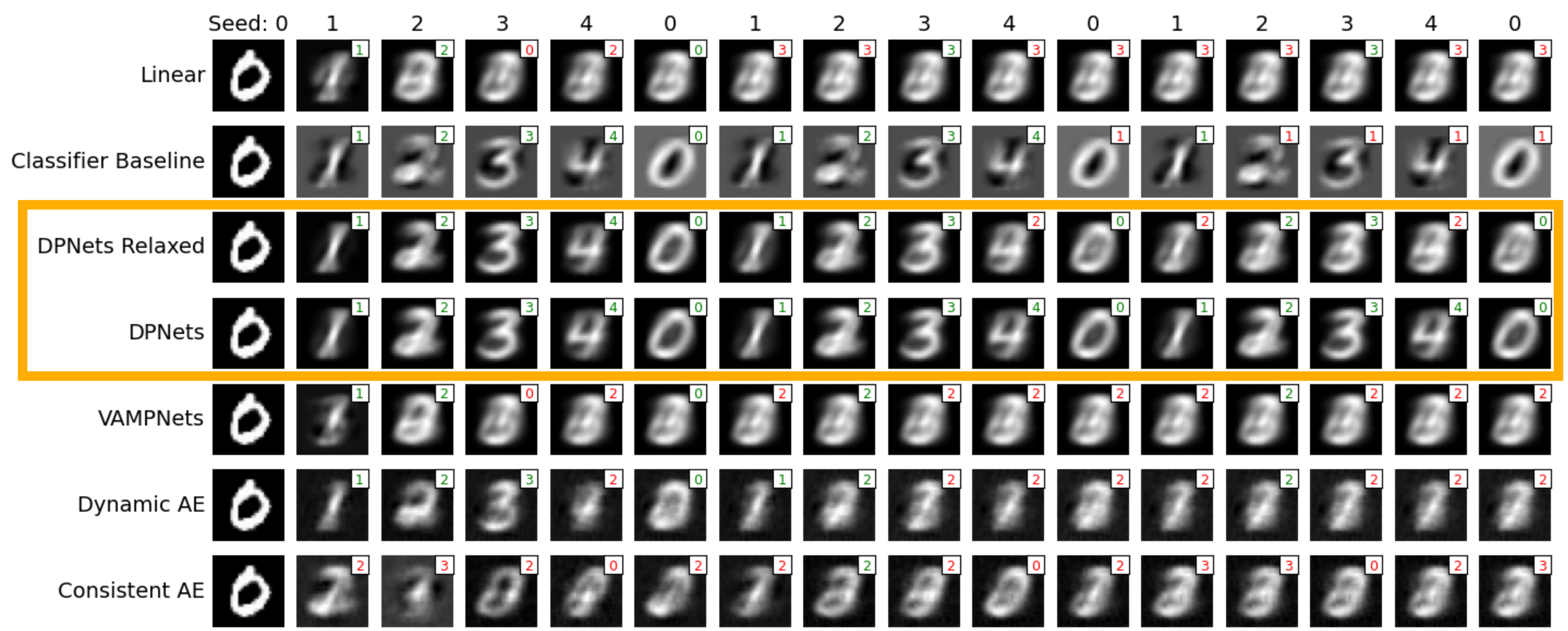
- $\mathscr{P}[\phi]$ is a functional of the feature map through the covariances $C_{X_t}, C_{X_t X_{t+1}}$.
- By learning ϕ we learn $\mathscr{H} = \overline{\mathrm{span}\left(\phi(x) \,|\, x \in \mathscr{X}\right)}$.
- The pseudo-inverses make $\mathscr{P}[\phi]$ it **nasty** to optimize with gradient descent.

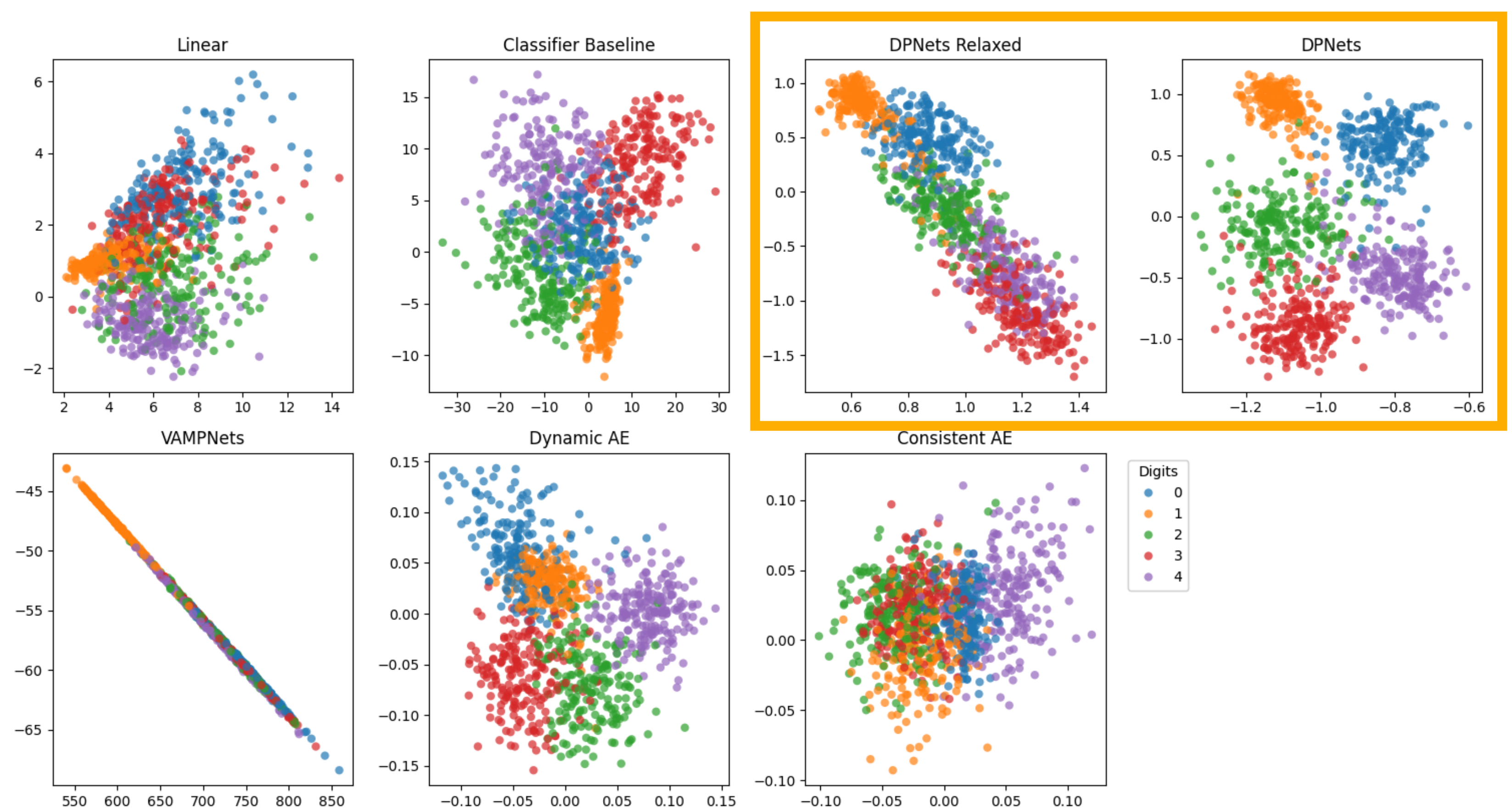
$$\mathscr{S}[\phi] := \frac{\|C_{X_t X_{t+1}}\|_{\mathrm{HS}}^2}{\|C_{X_t}\| \, \|C_{X_{t+1}}\|} - \gamma \left[\mathscr{R}(C_{X_t}) + \mathscr{R}(C_{X_{t+1}})\right]$$

Enough math, an example: MNIST

We randomly sample images from the MNIST dataset according to the rule that X_t should be an image of the digit t (mod 5) for all $t \in \mathbb{N}_0$.

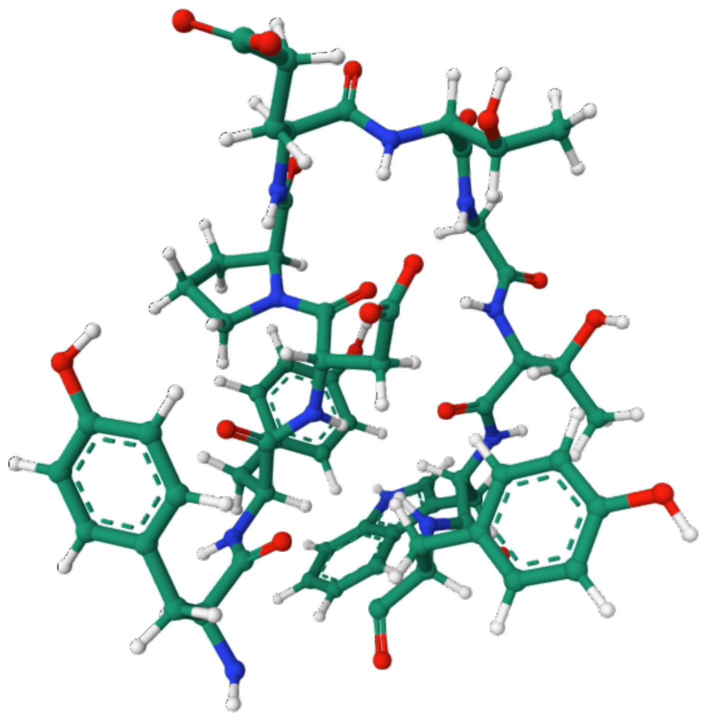
Given an image from the dataset with label c, a model for the transfer operator T of this system should then be able to produce an MNIST-alike image of the next digit in the cycle.





Example: metastable states of Chignolin

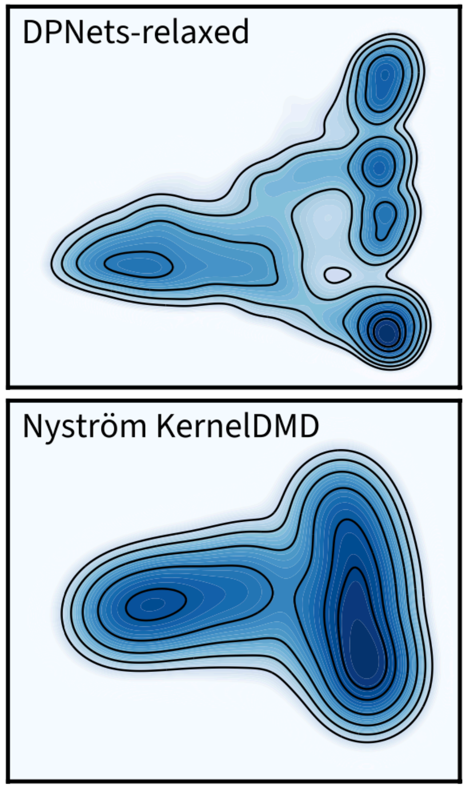
The leading eigenfunctions of T capture the long-term behavior of atomistic dynamics.



A better representation of the data allows a more accurate physical understanding.

Trained DPNets on a Graph Neural Network appropriate for the problmem vs. Fixing $\mathscr{H}$ to be the Gaussian RKHS.

Model	$\mathcal{P}$	Transition	Enthalpy ΔH
DPNets	**12.84**	**17.59 ns**	**-1.97 kcal/mol**
Nys-PCR	7.02	5.27 ns	-1.76 kcal/mol
Nys-RRR	2.22	0.89 ns	-1.44 kcal/mol
Reference	-	40 ns	-6.1 kcal/mol



Free Energy Surface

Conclusions

Additional works

- Estimating Koopman operators with sketching to provably learn large-scale dynamical systems. (NeurIPS’23)
- A randomized algorithm to solve reduced rank operator regression. (Submitted)
- Consistent Long-Term Forecasting of Ergodic Dynamical Systems. (ICML 2024)
- Learning the Infinitesimal Generator of Stochastic Diffusion Processes. (Submitted)
- Operatorial formulation of Reinforcement Learning.
- Neural Conditional Probability models.

kooplearn

Vladimir Kostic

Karim Lounici

Massi Pontil

And also:

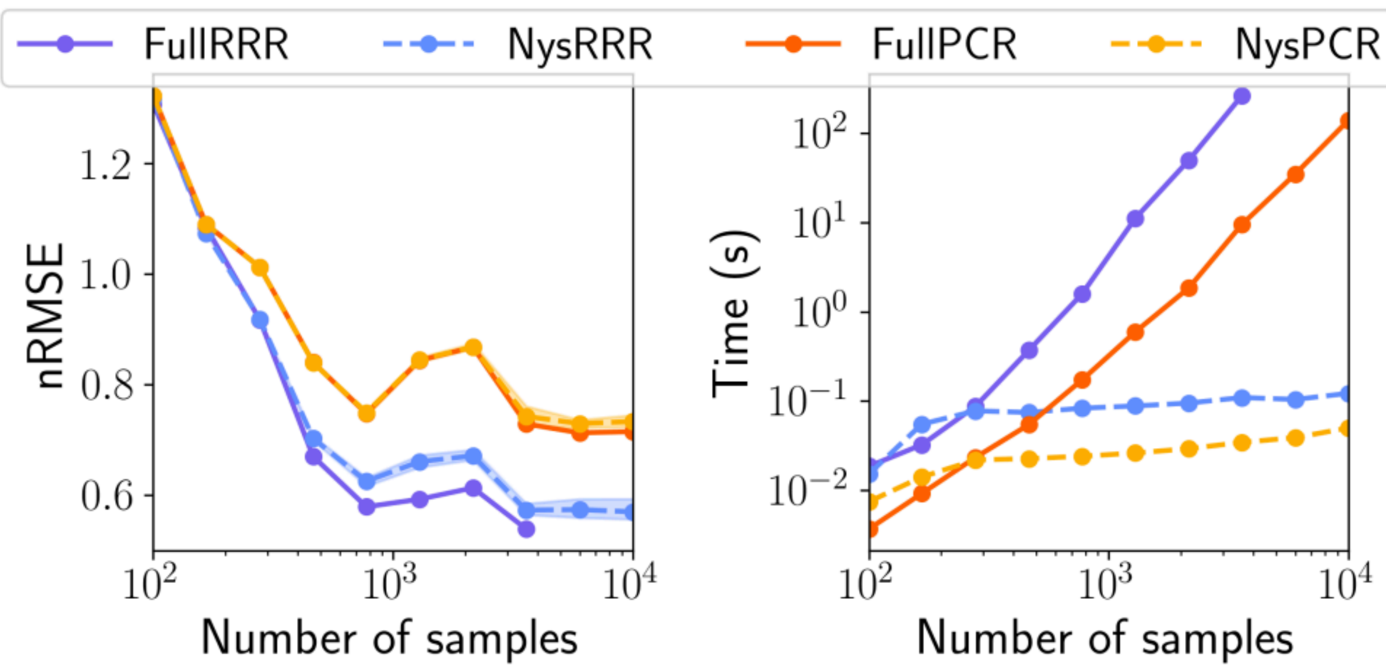
- Riccardo Grazzi
- Giacomo Turri
- Daniel Ordoñez-Apraez
- Prune Inzerilli
- Carlo Ciliberto
- Marco Pratticò
- Andreas Maurer
- Lorenzo Rosasco
- Giacomo Meanti
- Antoine Chatalic

Thank you!

Extra Slides

Large Scale Algorithms

Meanti et al. NeurIPS ’23 — Turri et al. (submitted)



Nyström (left) and **Randomized SVD** (bottom) estimators: 1-2 orders of magnitude faster, same statistical optimality.