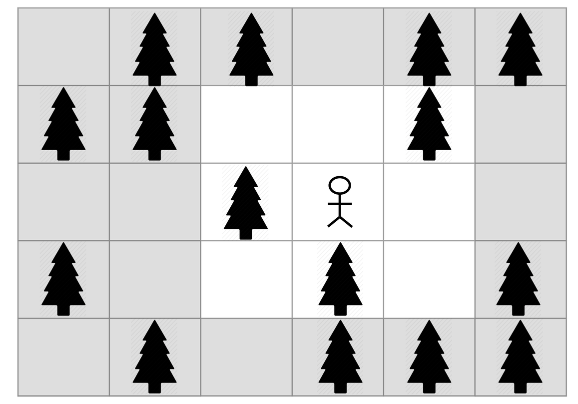
Completing State Representations using Spectral Learning

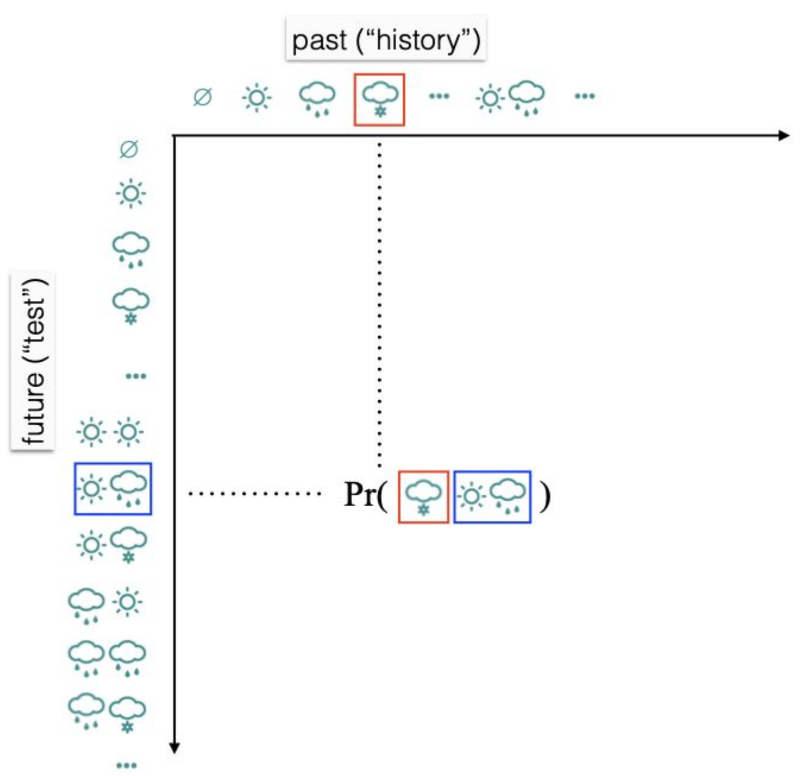
Nan Jiang (UIUC), Alex Kulesza (Google Research), Satinder Singh (U Michigan)

State Representation Discovery

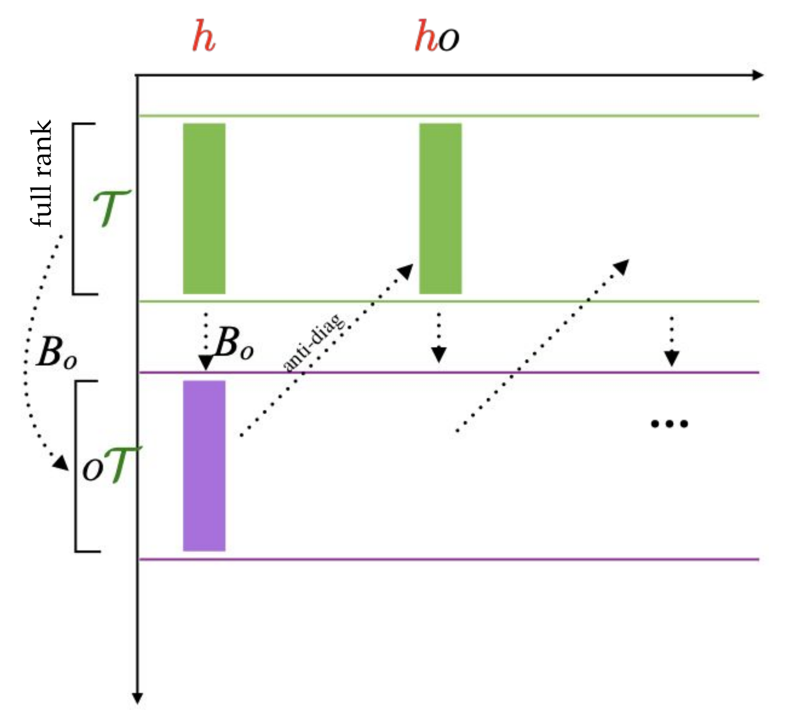
- Model *non-Markov* systems: need to compactly summarize history as *state*
- A plethora of methods: EM for HMMs/POMDPs, **Spectral Learning** for **PSRs**/ OOMs/WFAs, BPTT for RNNs, etc.
- Issue: difficult to incorporate prior knowledge about state representation
  - Many RL domains are *almost* Markov; just need a few bits to resolve ambiguity
  - User can write down an informative but incomplete state representation
- This work: Accelerate spectral learning with a given (& likely imperfect) state representation



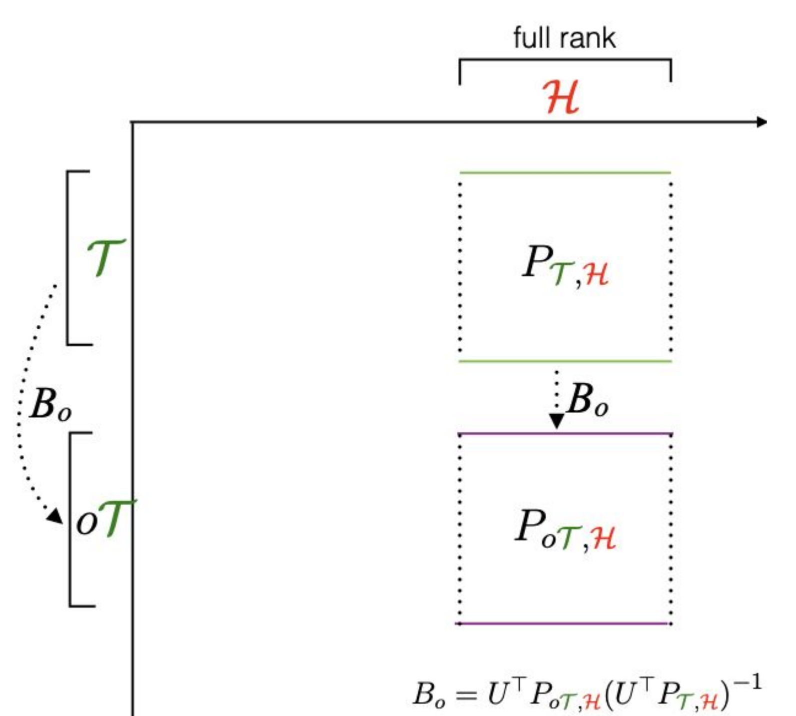
Background: Predictive State Representations (PSRs)



System Dynamics Matrix M
- "Hankelness": repeating anti-diagonal entries
- Low-rankness: rank ≤ # hidden states for HMMs/POMDPs



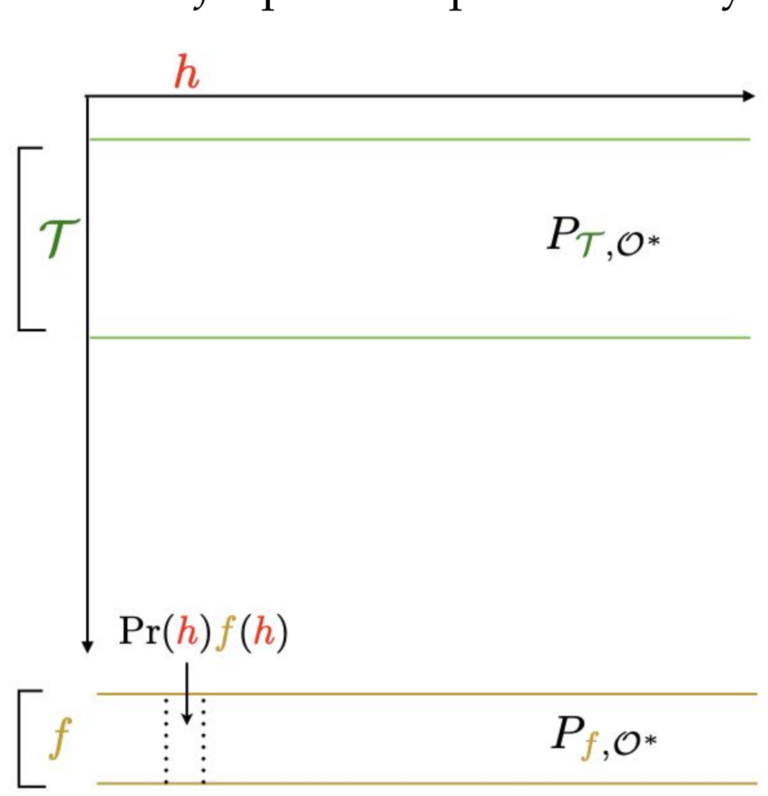
Maintain state during prediction "core" tests
- *State*: $P_{\mathcal{T}|h}$, prob. of a set of future events T
- Model parameter: $\{B_o\}_{o\in\mathcal{O}}$
- Given new obs o, the new state is $P_{\mathcal{T}|ho} \propto P_{o\mathcal{T}|h} = B_o P_{\mathcal{T}|h}$

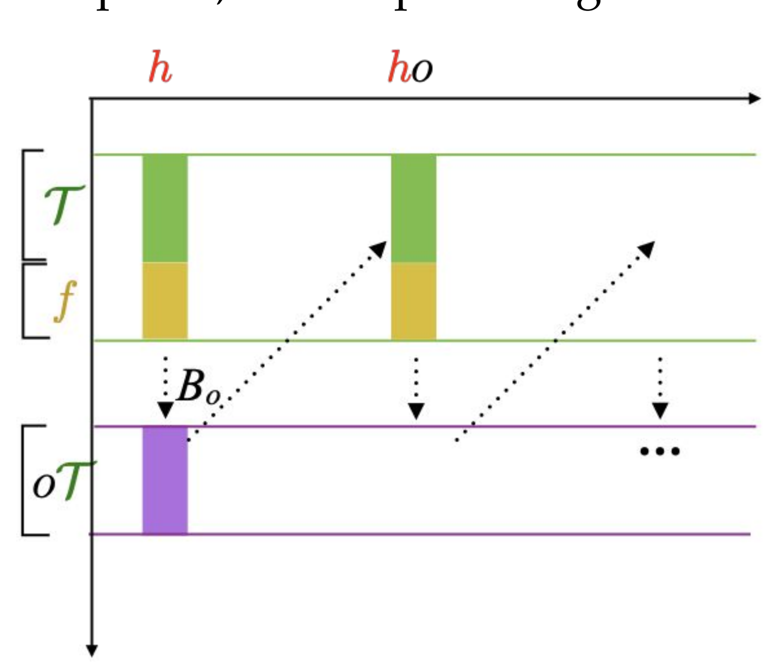


Learning algorithm
- Estimate full-rank submatrices + matrix inv
- "Spectral": use over-complete T and compress by SVD
- Discovered state repr: $U^\top P_{\mathcal{T}|h}$

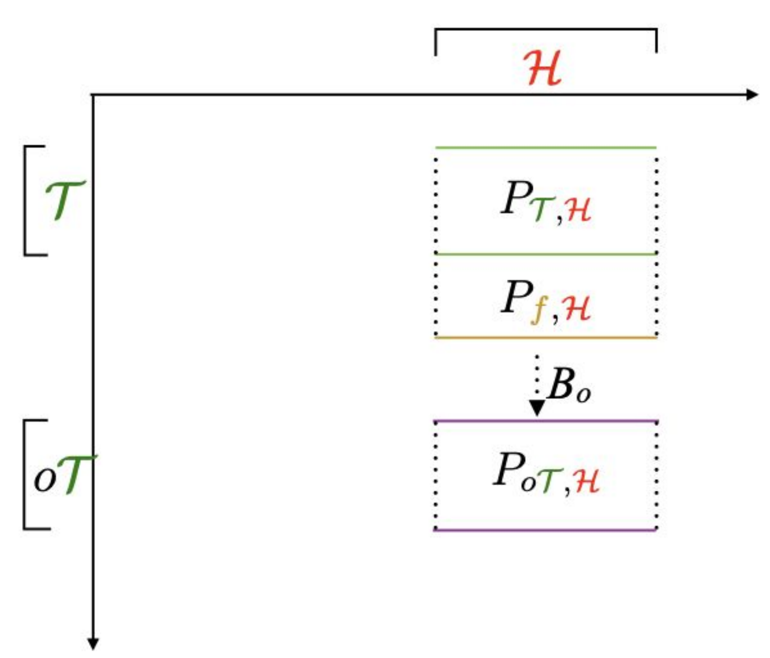
PSR-f : Incorporating a given representation

Setting: Given a manually designed $f : \mathcal{O}^* \to \mathbb{R}^m$. **Ideally:** f is already a state, e.g., $f(h) = P_{\mathcal{T}_f|h}$ for rich enough T_f
More realistically: f is incomplete. **Naive alg:** Combine $P_{f,\mathcal{H}}$ and (a smaller) $P_{\mathcal{T},\mathcal{H}}$ to predict $P_{o\mathcal{T},\mathcal{H}}$
Benefit: "f" part computed directly in state update; no compounding errors.





Maintain state during prediction
- *State*: $P_{\mathcal{T}|h}$ concat. w/ $f(h)$
- Given new obs o, predict $P_{\mathcal{T}|ho}$, then concat. w/ $f(ho)$



Algorithm 1 Template for learning transformed PSR-fs

$U := \begin{bmatrix}U_\mathcal{T}\\ U_f\end{bmatrix}$. $B_o := U_\mathcal{T}^\top \hat{P}_{o\mathcal{T},\mathcal{H}} \left(U^\top \begin{bmatrix}\hat{P}_{\mathcal{T},\mathcal{H}}\\ \hat{P}_{f,\mathcal{H}}\end{bmatrix}\right)^+$

Theoretical Properties

Define: $\mathrm{rank}(f) = \mathrm{rank}(P_{f,\mathcal{O}^*})$, and $\mathrm{rank}(f; M) = \dim(\mathrm{rowspace}(M) \bigcap \mathrm{rowspace}(P_{f,\mathcal{O}^*}))$.

	PSR	PSR-f
Consistency Condition	$\mathrm{rank}(P_{\mathcal{T},\mathcal{H}}) = \mathrm{rank}(M)$	$\mathrm{rank}\left(\begin{bmatrix}P_{\mathcal{T},\mathcal{H}}\\ P_{f,\mathcal{H}}\end{bmatrix}\right) = \mathrm{rank}\left(\begin{bmatrix}M\\ P_{f,\mathcal{O}^*}\end{bmatrix}\right)$
Minimal $\lvert\mathcal{T}\rvert$	rank(M)	rank(M) - rank(f ; M)
Minimal $\lvert\mathcal{H}\rvert$	rank(M)	rank(M) + rank(f) - rank(f ; M)
State dimension	rank(M)	rank(M) + rank(f) - rank(f ; M) (w/ naive alg)
Model size	(# parameters in each B_o :) state dimension * minimal $\lvert\mathcal{T}\rvert$	

- rank(f ; M) : # of "relevant" dimensions. The greater, the smaller $\lvert\mathcal{T}\rvert$ needed.
- rank(f) - rank(f ; M) : # of "irrelevant" dimensions. The greater, the larger $\lvert\mathcal{H}\rvert$ needed.
  - Imagine f that provides useful info on some core H and takes arbitrary values elsewhere
  - Needs more than rank(M) histories to witness the inconsistent behavior

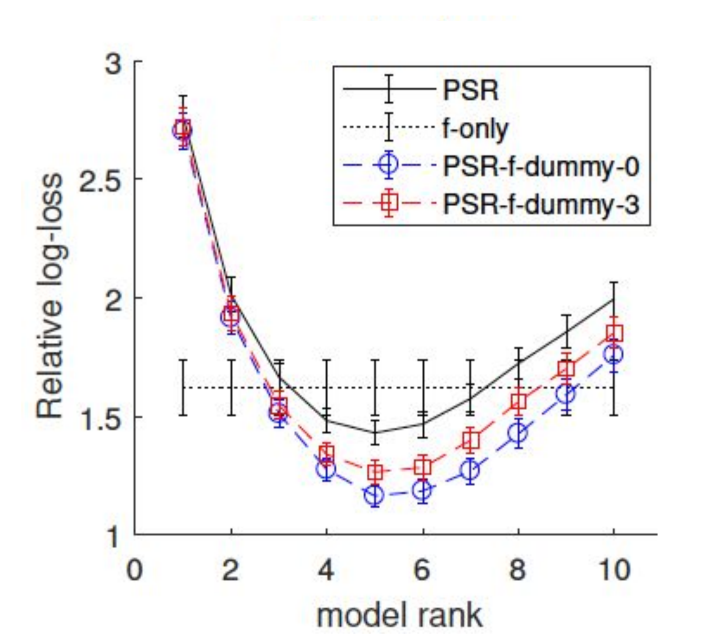
State dim & model size suffer from misspecification of f. Can we improve? Also, how to extend to spectral learning?

Spectral Learning of PSR-f

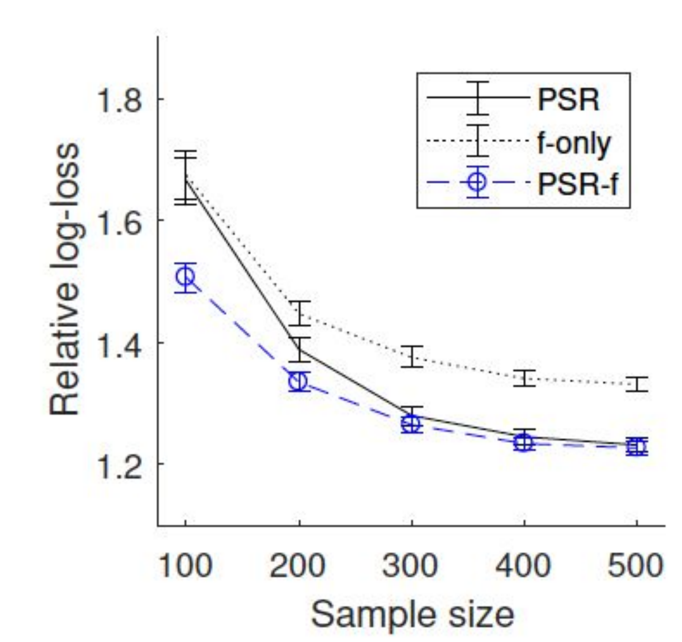
Naive idea Run Alg 1 with U set to the top singular vectors of $\begin{bmatrix}\hat{P}_{\mathcal{T},\mathcal{H}}\\ \hat{P}_{f,\mathcal{H}}\end{bmatrix}$.
Problem Irrelevant dim. might dominate relevant ones spectrally; still need to keep the irrelevant info in state.
Solution Extract relevant dimensions out of f.
Algorithm (Consistent with state dim rank(M) & naturally extends spectral learning.)
Input: f, estimated moments, T, H, k (≈rank(M)) and d (≈rank(f ; M))

- Choose large T and H. Run SVD on $P_{T,H}$ and truncate after k s.v.'s.
- Identify intersection (of dim. d) between rowspaces of truncated $P_{T,H}$ and $P_{f,H}$ robustly via *principal angle*.
- Let U_f be the projection of P_{f,O^*} to the identified subspace. Let $U_f{}^T f(\cdot)$ be part of state.
- Project $P_{T,H}$ onto the orthogonal complement of the subspace; use SVD to identify the rest k-d dim of state as usual.

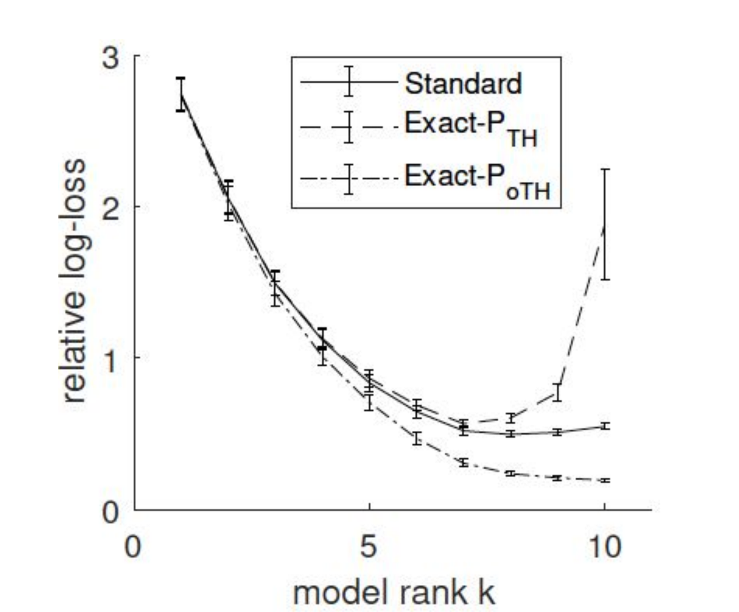
Experiments



Synthetic HMMs
- $f(h) = P_{\mathcal{T}_f|h}$ w/ $\lvert T_f\rvert = 3$, concat. w/ 0 or 3 irrelevant features ("dummy-x")
- Better than vanilla PSR & using f alone



Aircraft Identification domain
- HMM converted from POMDP using uniformly random actions
- Noisy obs. of position + foe/friend
- $f(h)$ is smoothed estimation w/ quadratic features



Negative Results
- Intuition: $f(h) = P_{\mathcal{T}_f|h}$ should always help
- Actually: can hurt a lot sometimes!
- Not specific to our setting; can reproduce in standard spectral learning (see above figure)
- Need new theory to explain