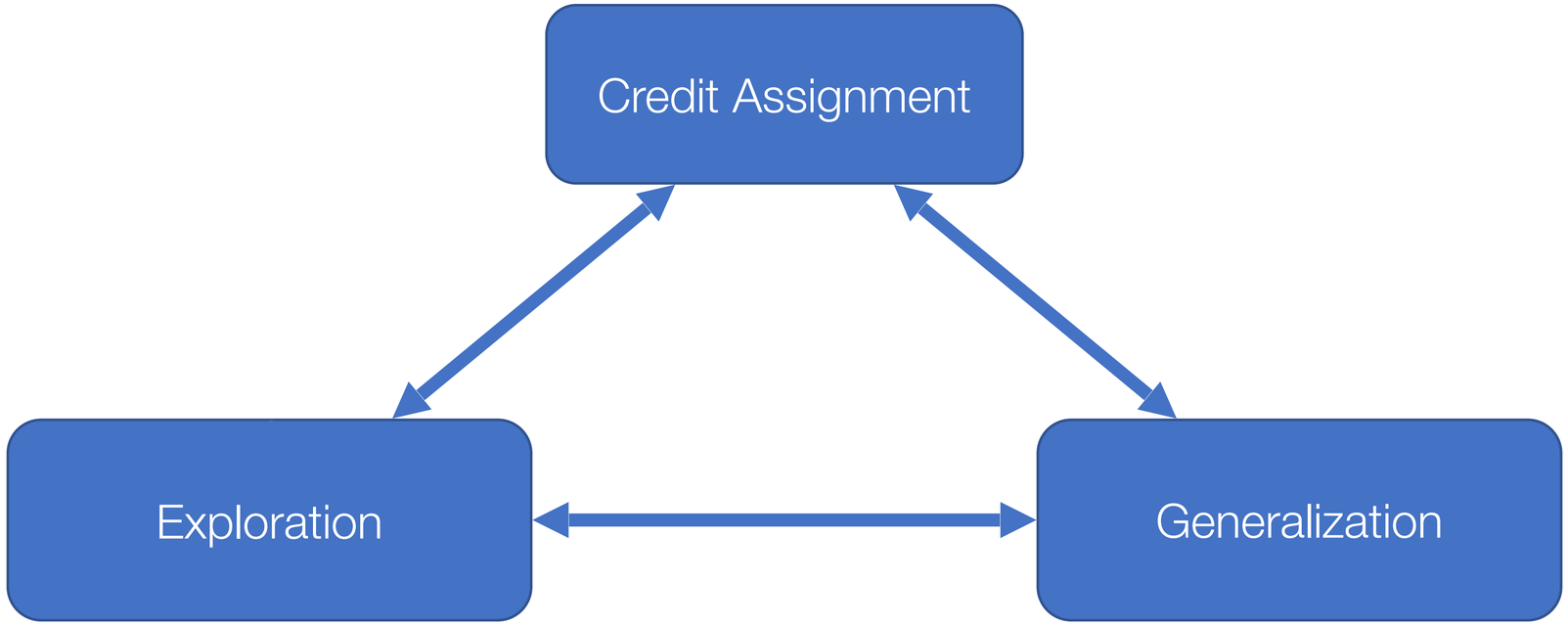
Statistical Foundations of Reinforcement Learning: III

COLT 2021

Akshay Krishnamurthy (MSR, akshaykr@microsoft.com)

Wen Sun (Cornell, ws455@cornell.edu)

Function approximation goal



Focus on episodic setting, with horizon H

Given function class $\mathcal{F}$, find ϵ sub-optimal policy in poly(comp($\mathcal{F}$), $|A|$, H, $1/\epsilon$) samples

Function approximation approaches

Policy search: Policy class $\Pi \subset \{\mathcal{X} \to \mathcal{A}\}$

- Realizability: optimal policy $\pi^\star \in \Pi$

Value-based: Class $\mathcal{F} \subset \{\mathcal{X} \times \mathcal{A} \to \mathbb{R}\}$ of candidate Q functions

- Realizability: $Q^\star \in \mathcal{F}$
- Recall:

Main focus

$$Q_h^\star(x,a) = \mathbb{E}[\sum_{\tau=h}^{H} r_\tau \mid x_h = x, a_h = a, \pi^\star] = \mathbb{E}[r + \max_{a'} Q_{h+1}^\star(x', a') \mid x_h = x, a_h = a]$$

Model-based: Class $\mathcal{M} \subset \{\mathcal{X} \times \mathcal{A} \to \mathbb{R} \times \Delta(\mathcal{X})\}$ of dynamics models

A key challenge: Distribution shift

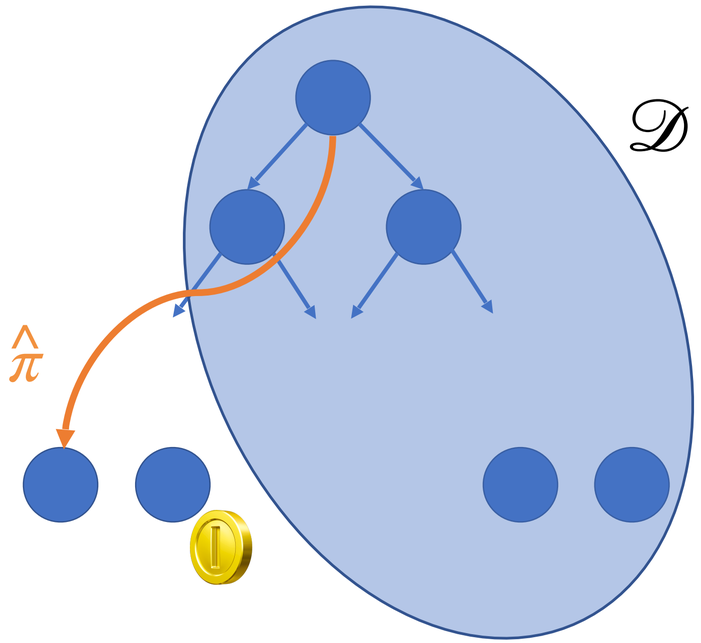
Theorem [General lower bound]: With finite class $\mathcal{F}$ of Q functions that realize $Q^\star$, $\Omega(\min(A^H \log|\mathcal{F}|, |\mathcal{F}|)/\epsilon^2)$ samples are necessary

Distribution shift

- Predicting $Q^\star$ accurately on previous data does not directly imply a good policy (unlike supervised learning)

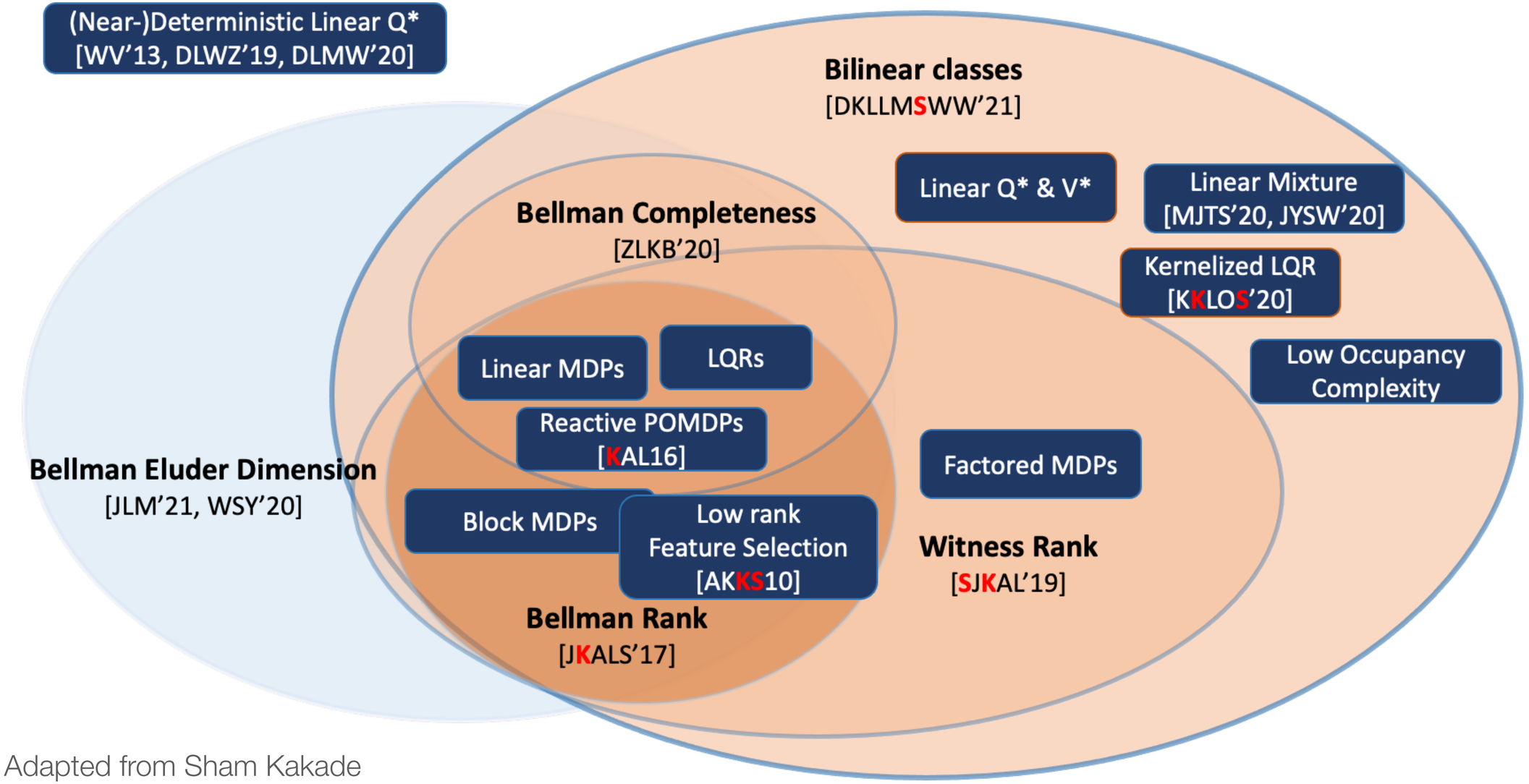
Conceptual solutions

1. Assume function class supports “extrapolation”
2. Assume environment only has “a few” distributions



[Kearns-Mansour-Ng-02] [Kakade-03]

Function approximation landscape



Adapted from Sham Kakade

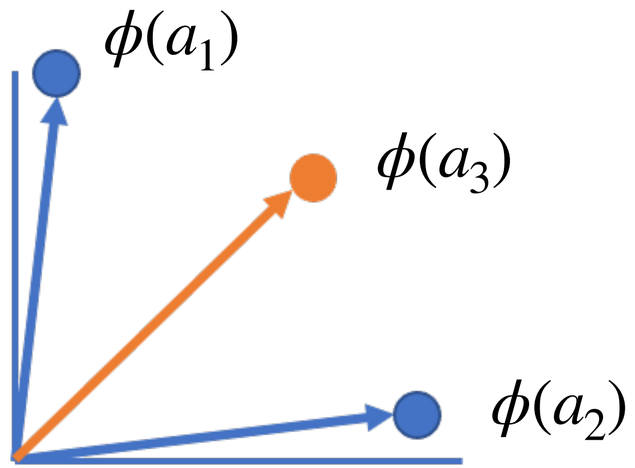
Part 3A: Linear methods

Most basic question

Given feature map $\phi : \mathcal{X} \times \mathcal{A} \to \mathbb{R}^d$ *such that* $Q^\star(x,a) = \langle \theta^\star, \phi(x,a) \rangle$

Is poly$(d, H, 1/\epsilon)$ *sample complexity possible?*

- Yes for supervised learning and bandits
  - Query on basis/spanner, then extrapolate



What about for RL?

Linear RL arms race

Assumption	Setting	Notes	Reference
Linear Q*, deterministic	Online Exploration	Constraint propagation	Wen-van-Roy-13
Linear Q*, low var., gap	Online Exploration	Rollout based	Du-Luo-Wang-Zhang-19
Linear Q^\pi for all \pi	Sample-based planning	API + Exp. Design	Lat-Sze-Wei-20
Linear Q^\pi for all \pi	Batch/offline setting	poly(d) actions	Wang-Fos-Kak-20
Linear Q*	Sample-based planning	exp(d) actions	Wei-Amo-Sze-20
Linear Q* + gap	Sample-based planning	Rollout + Exp. Design	Du-Kak-Wang-Yang-20
Linear Q* + gap	Online Exploration	exp(d) actions	Wang-Wang-Kak-21
Linear V*	Sample-based planning	$(dH)^A$ sample comp.	Wei-Amo-Jan-Abb-Jia-Sze-21

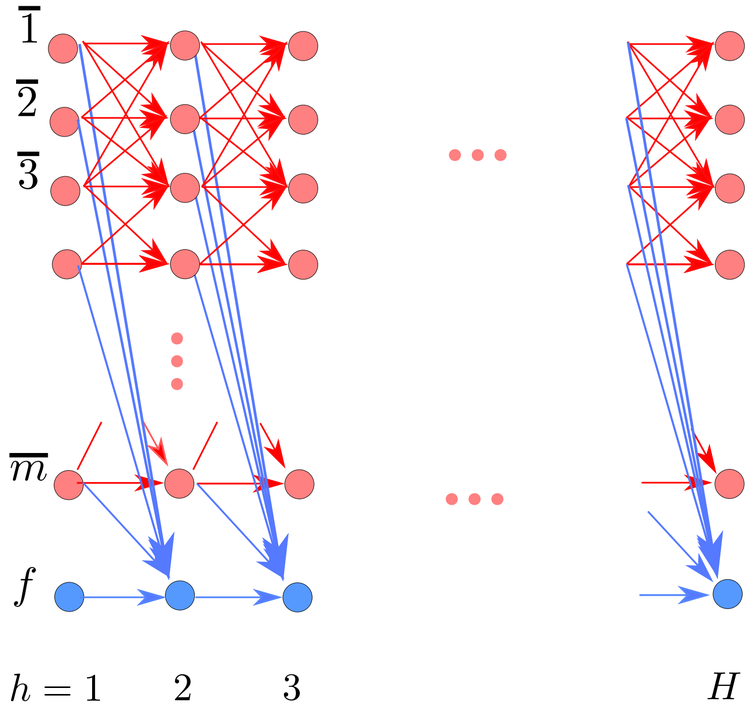
Challenge: Error amplification in dynamic programming

Adapted from Gellert Weisz

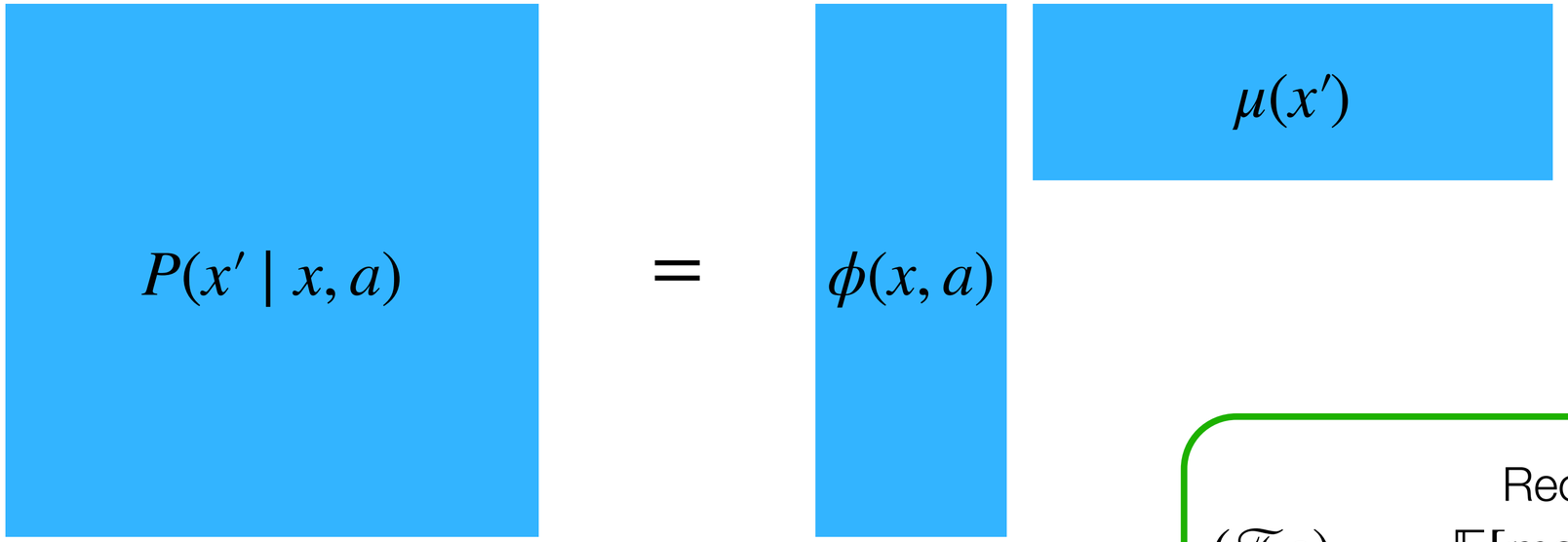
A linear lower bound

Theorem [Wang-Wang-Kakade-21]: There exists a class of linearly realizable MDPs (with constant gap) s.t. any online RL algorithm requires $\min(\Omega(2^d), \Omega(2^H))$ samples to obtain a near optimal policy.

- Extends argument of Weisz-Amortilla-Szepesvari-20 from the planning setting
  - Idea: exp(d) states and actions with near-orthogonal features (JL lemma)
- Fundamentally different from SL and bandits
- RL indeed requires strong assumptions!



Linear upper bound: Low rank/Linear MDP



Transitions and rewards are linear in feature map $\phi(x, a)$

Recall:

$$(\mathcal{T} g)_{x,a} = \mathbb{E}[\max_{a'} g(x', a') \mid x, a]$$

Lemma: For any function $g : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$, $\exists \theta \in \mathbb{R}^d$ such that

$$\langle \theta, \phi(x, a) \rangle = (\mathcal{T} g)_{x,a}$$

LSVI-UCB

Algorithm

- Optimistic dynamic programming

$$\hat{\theta}_h := \arg\min_{\theta} \sum \left(\langle \theta, \phi(x_h, a_h) \rangle - r_h - \max_{a'} \hat{Q}_{h+1}(x_{h+1}, a') \right)^2$$

- Define $\hat{Q}_h(x, a) = \langle \hat{\theta}_h, \phi(x, a) \rangle + \text{bonus}_h(x, a)$

Elliptical bonus: $\|\phi(x, a)\|_{\Sigma_h^{-1}}$

- Collect data with greedy policy $\hat{\pi}_h(x) = \arg\max_a \hat{Q}_h(x, a)$

Theorem [Jin-Yang-Wang-Jordan-19]: In low rank MDP, LSVI-UCB has regret $\tilde{O}(\sqrt{d^3 H^3 N})$ over N episodes with high probability

LSVI-UCB: Analysis

- Similar to UCB-VI [1]: If bonus dominates regression (prediction) error

$$\text{Regret} \lesssim \sum_t \sum_h \text{bonus}_h(x_{t,h}, a_{t,h})$$

Covering argument incurs extra $\sqrt{d}$ factor

- Linear MDP property prevents error amplification (controls regression error)

$$\exists \tilde{\theta}_h \;\; \text{s.t.,} \;\; \langle \tilde{\theta}_h, \phi(x,a) \rangle = (\mathcal{T}\hat{Q}_{h+1})_{x,a}$$

- Elliptical potential lemma (from online learning): If $x_1, \ldots, x_T \in B_2(d)$ and $\Sigma_0 = \lambda I, \Sigma_t \leftarrow \Sigma_{t-1} + x_t x_t^\top$ then

$$\sum_t \|x_t\|_{\Sigma_{t-1}^{-1}} \lesssim \sqrt{Td \log(T/d)}$$

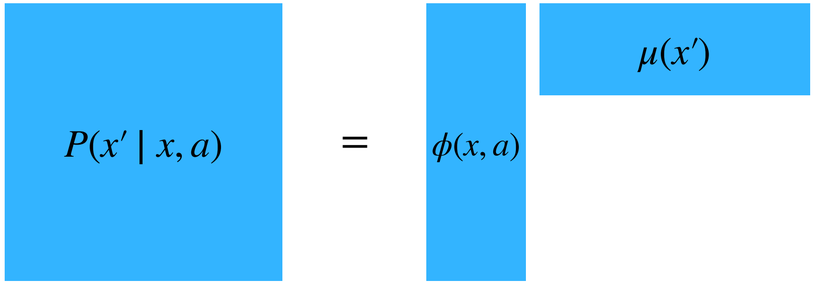
1. See [Neu-Pike-Burke-20]

Linear RL recap + discussion

- Linear function approximation enables extrapolation: elliptical potential lemma
  - Different potential: Eluder dimension [Russo-van Roy 13, Dong-Yang-Ma-21, Li-Kamath-Foster-Srebro-21]
- Challenge is error amplification in dynamic programming
  - Avoided in linear MDPs and with “linear bellman completeness” (more in next part)
- Takeaway: RL is not like SL, much stronger assumptions are necessary
- **Open problem**: Sample-efficient RL with linear $Q^\star$ and poly(d) actions?
- **Open problem**: Efficient alg with optimal dimension dependence for linear MDP?
- **Open problem**: Efficient alg for linear bellman complete setting?

Part 3B: Information Theory

Revisiting linear MDPs



Lemma: For any function $\ell : \mathcal{X} \to \mathbb{R}$, $\exists \theta_\ell \in \mathbb{R}^d$ such that

$$\forall \pi : \mathbb{E}_\pi[\ell(x_h)] = \langle \mathbb{E}_\pi[\phi(x_{h-1}, a_{h-1})], \theta_\ell \rangle$$

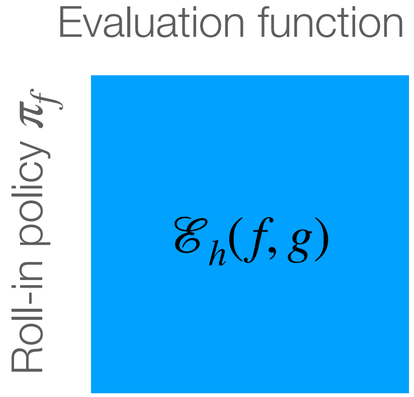
All expectations admit d-dimensional parametrization $\Rightarrow$ only a few distributions!

Natural to define a loss function $\ell : \mathcal{F} \times (\mathcal{X} \times \mathcal{A} \times \mathcal{X} \times \mathbb{R}) \to \mathbb{R}$ and examine

$$\mathcal{E}_h(f, g) := \mathbb{E}[\ell(g, (x_h, a_h, x_{h+1}, r_h)) \mid x_h \sim \pi_f, a_h \sim \pi_g]$$

Linear MDP: For any ℓ of this type, $\text{rank}(\mathcal{E}_h) \leq d$

Question: What loss function?



Bellman rank

$$\mathscr{E}_h(f,g) := \mathbb{E}_{x_h\sim\pi_f, a_h\sim\pi_g}[\ell(g,(x_h,a_h,x_{h+1},r_h))]$$

Bellman rank (V-version): Choose $\ell(g,(x,a,x',r)) := g(x,a) - r - \max_{a'} g(x',a')$

Bellman optimality equation: $\mathscr{E}_h(f,Q^\star) = 0 \forall f$

Theorem [Jiang-Krishnamurthy-Agarwal-Langford-Schapire-17]:

If $Q^\star \in \mathscr{F}$ and $\max_h$ rank$(\mathscr{E}_h) \leq M$ then can learn ϵ suboptimal policy in

$$\tilde{O}(M^2AH^3\text{comp}(\mathscr{F})/\epsilon^2) \text{ samples}$$

OLIVE: Algorithm

Version space algorithm: repeat

1. Select surviving $\hat{f} \in \mathscr{F}$ that maximizes $\mathbb{E}[f(x_1, \pi_f(x_1))]$
2. Collect data with $\hat{\pi} = \pi_{\hat{f}}$ and estimate actual value
3. If actual value $\approx$ guess, terminate and output $\hat{\pi}$
4. Otherwise, eliminate all $g \in \mathscr{F}$ with $\mathscr{E}_h(\hat{f}, g) \neq 0$ at some h

Optimistic guess for $V^\star$

Achieve optimistic guess

Loss minimization

OLIVE: Analysis

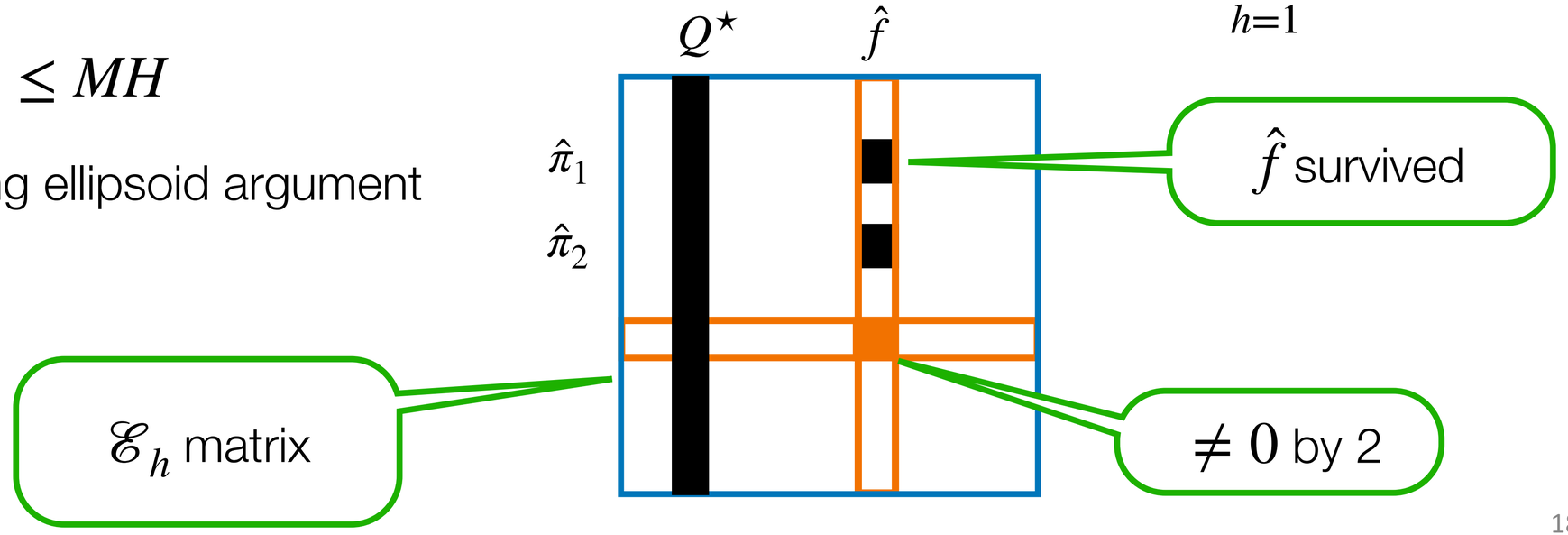
Claim 1: $Q^\star$ never eliminated (by bellman equation)

Claim 1 + Optimism: Final policy is near optimal

Claim 2: Telescoping performance decomposition $\mathbb{E}[f(x_1, \pi_f(x_1))] - V(\pi_f) = \sum_{h=1}^{H} \mathscr{E}_h(f, f)$

Claim 3: Iterations $\leq MH$

"Robust" proof using ellipsoid argument



Bilinear classes

Ingredients: Function class $\mathscr{H}$, Loss class $\{\ell_f : f \in \mathscr{H}\}$, policies $\{\pi_{\text{est}}(f) : f \in \mathscr{H}\}$

(Unknown) Embedding functions $W_h, X_h : \mathscr{H} \to \mathscr{V}$ (a Hilbert space)

1. **Realizability**: $f^\star \in \mathscr{H}$ induces optimal Q function
2. **Bellman domination**:

$$|\mathbb{E}_{\pi_f}[Q_f(x_h, a_h) - r_h - V_f(x_{h+1})]| \le |\langle W_h(f), X_h(f)\rangle|$$

3. **Loss decomposition**:

$$|\mathbb{E}_{\pi_f \circ \pi_{\text{est}(f)}}[\ell_f(g, x_h, a_h, x_{h+1}, r_h)]| = |\langle W_h(g), X_h(f)\rangle|$$

Bellman rank:

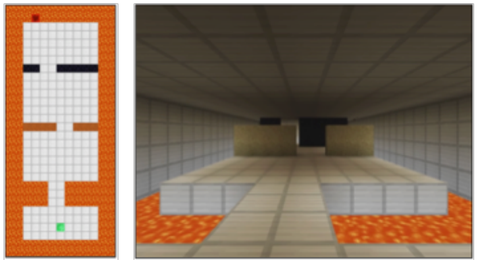
$\mathscr{H}$ = Q functions

ℓ_f = (IW) bellman errors

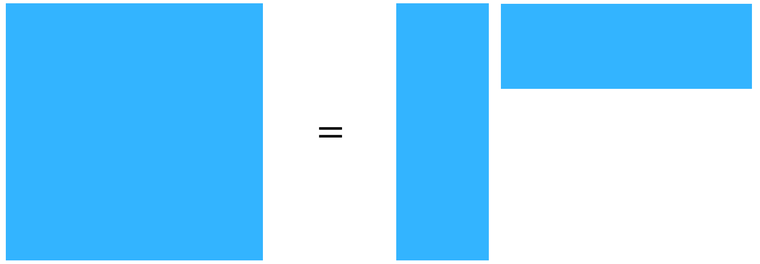
$\pi_{\text{est}(f)}$ = Unif($\mathscr{A}$)

[Du-Kakade-Lee-Lovett-Mahajan-Sun-Wang-21] also see [Jin-Liu-Miryoosefi-21]

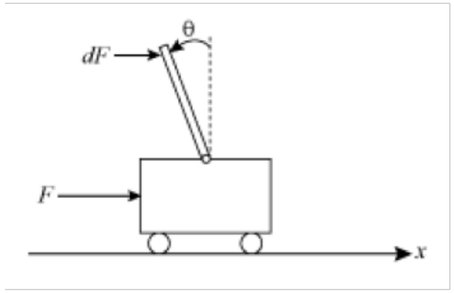
Examples



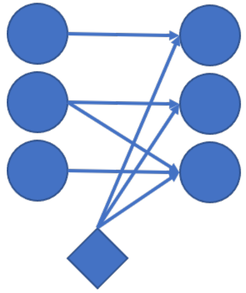
Block MDP [DKJADL-19, MHKL-20]



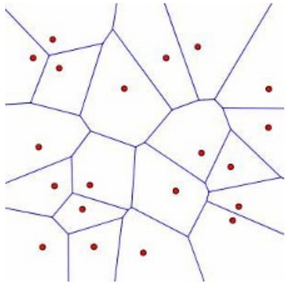
Low rank/Linear MDP [JYWJ-20,…,AKKS-20]



Linear quadratic control



Factored MDP



$Q^\star$ state abstraction

Uses $\pi_{\text{est}(f)}$

Linear bellman complete [ZLKB-20]

Linear mixture MDP [YW-20,MJTS-20, AJSWY-20]

Linear $Q^\star/V^\star$

Kernelized Nonlinear Regulator [MJR-20, KKLOS-20]

Some models with memory: PSR, etc.

Uses model-based $\mathscr{H}$

Uses f-adapted loss

Example: Linear bellman complete

Assumption: for any θ exists w such that[1]

$$(\mathcal{T}\theta)_{x,a} := \mathbb{E}[r + \max_{a'} \langle \theta, \phi(x', a') \rangle \mid x, a] = \langle w, \phi(x, a) \rangle$$

- Standard assumption in analysis of dynamic programming algorithms[2]
- Unclear if LSVI-UCB works: misspecification when backing up quadratic bonus

$$\mathcal{E}(\pi_f, \theta_g) := \mathbb{E}_{\pi_f}[\langle \theta_g, \phi(x, a) \rangle - r - \max_{a'} \langle \theta_g, \phi(x', a') \rangle]$$

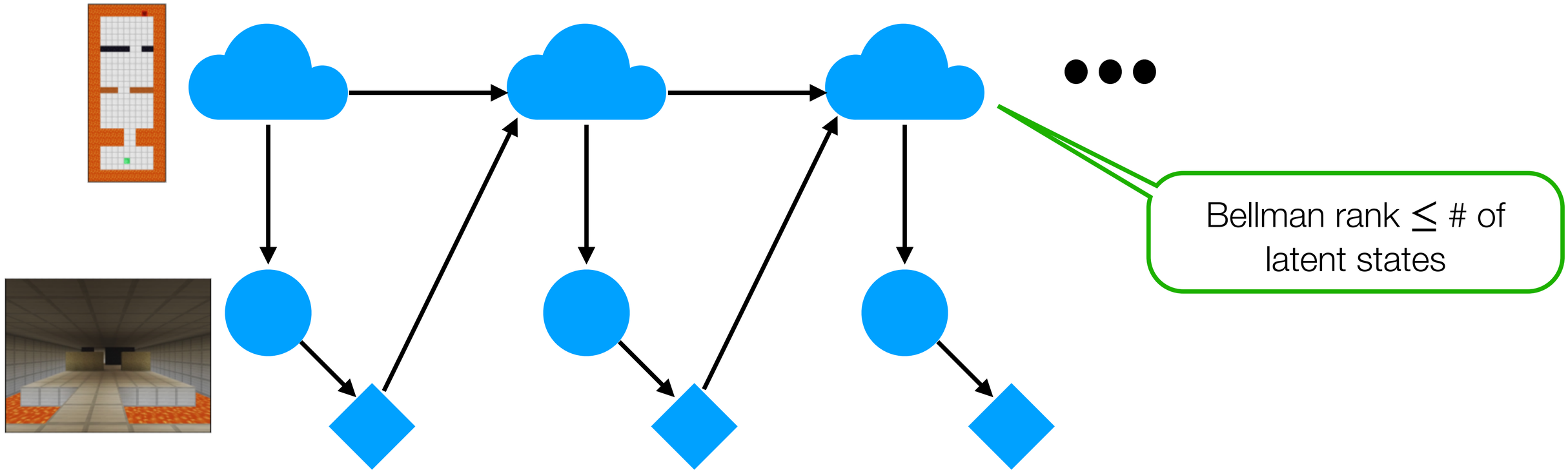
$$= \mathbb{E}_{\pi_f}[\langle \theta_g - w_g, \phi(x, a) \rangle] = \langle \theta_g - w_g, \mathbb{E}_{\pi_f}[\phi(x, a)] \rangle$$

- Using $\pi_{\mathrm{est}(f)} = \pi_f$ avoids dependence on A
- Still a very strong assumption! Can break when adding features

1. [Zanette-Lazaric-Kochenderfer-Brunskill-20]
2. [Antos-Munos-Szepesvari-08]

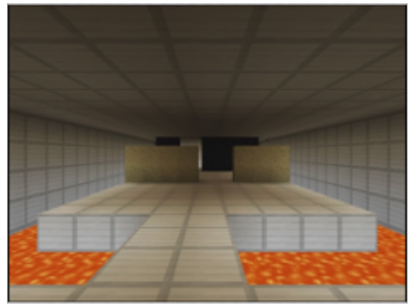
Part 3B: Algorithms

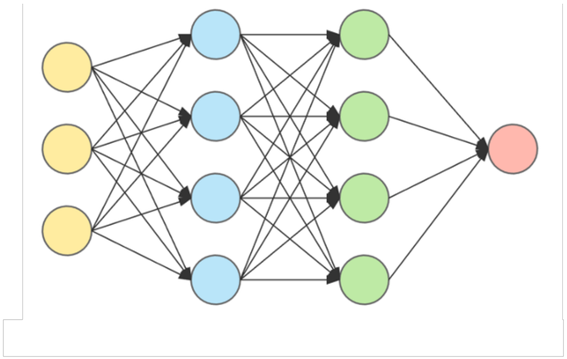
Block MDPs

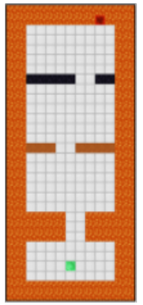


Rich observation problem with discrete latent state space
Agent operates on rich observations
Latent states are decodable from observations, so no partial observability

Approach: Representation learning + reductions







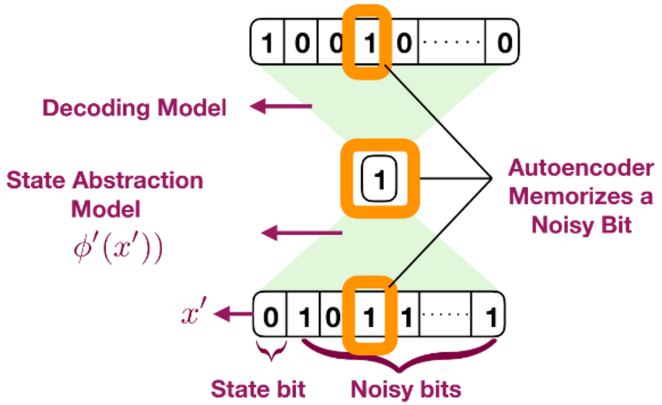
Idea: If we knew latent state, could run tabular algorithm

Algorithm: Use function class to “decode” latent state, then run tabular algorithm

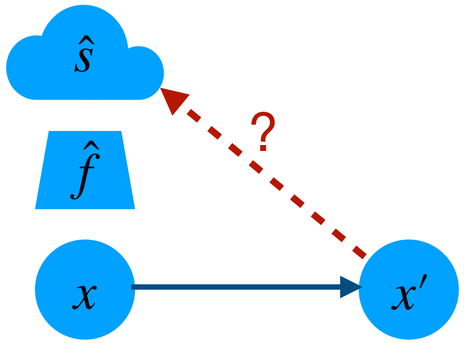
Reductions: Assume we can solve optimization problems over function class efficiently

Representation learning in Block MDPs

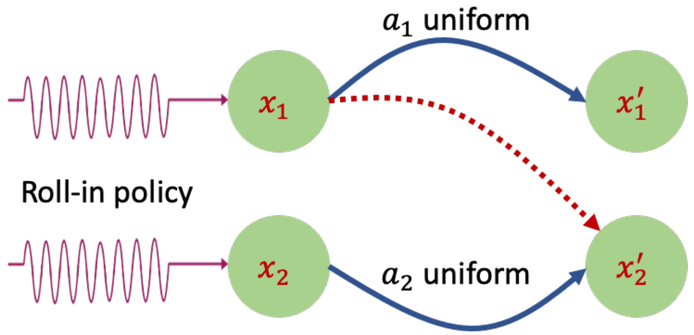
Autoencoding



Backward prediction



Contrastive Learning



Reason about MDP structure uncovered by Bayes optimal predictor

A guarantee

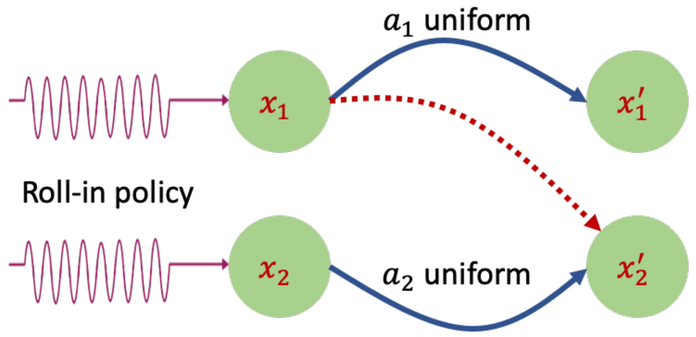
Theorem (informal) [Misra-Henaff-Krishnamurthy-Langford-20]:

If latent states are η-reachable and we have "realizability" can learn policy cover Ψ s.t.,

$$\forall \pi, s : P^{\pi}[s] \leq (2S) \cdot P^{\Psi}[s]$$

In poly$(S, A, H, 1/\eta, \text{comp}(\mathcal{F}))$ samples and in oracle computational model

- Use contrastive learning to learn state representation
- "Intrinsic" rewards encourage visiting learned states
- Policy optimization to maximize intrinsic rewards



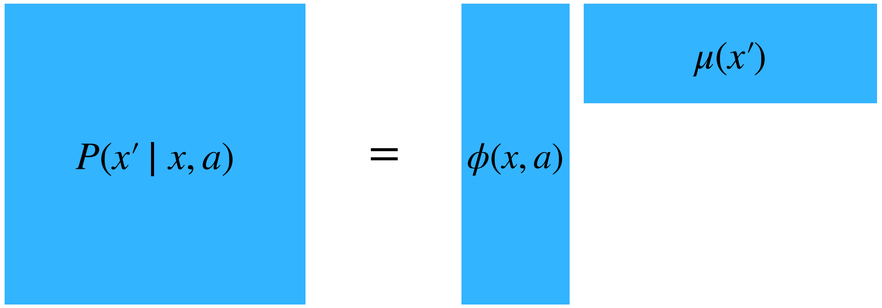
Implementable and effective on hard exploration problems!

Representation learning in low rank MDPs

Natural assumption: function class Φ containing true features $\phi \in \Phi$

Stronger: $\phi \in \Phi, \mu \in \Upsilon$

Model-based realizability



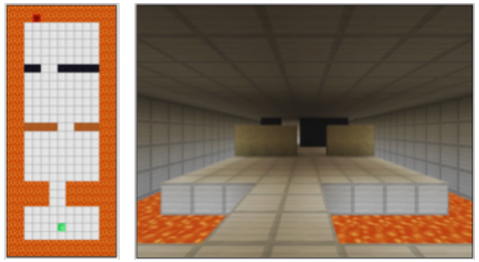
Theorem [Agarwal-Kakade-Krishnamurthy-Sun-20]: Can learn model $(\hat{\phi}, \hat{\mu})$ such that

$$\forall \pi : \mathbb{E}_{\pi} \| \langle \hat{\phi}(x, a), \hat{\mu}(\cdot) \rangle - P(\cdot \mid x, a) \|_{\mathrm{TV}} \leq \epsilon$$

in poly$(d, A, H, 1/\epsilon, \log |\Phi| |\Upsilon|)$ samples and in oracle model

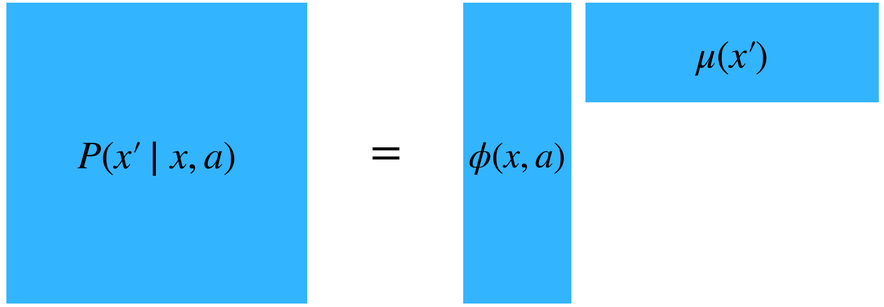
- Fit model using maximum likelihood estimation
- Plan to visit all directions in learned feature map (elliptical bonus + LSVI-UCB)
- Elliptical potential: coverage in true feature map

Other representation learning results



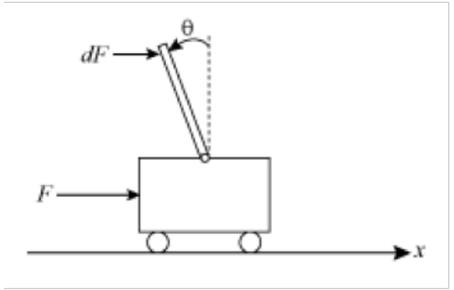
Block MDP

[Du et al., 19, Feng et al., 20, Foster et al., 20, Wu et al., 21]



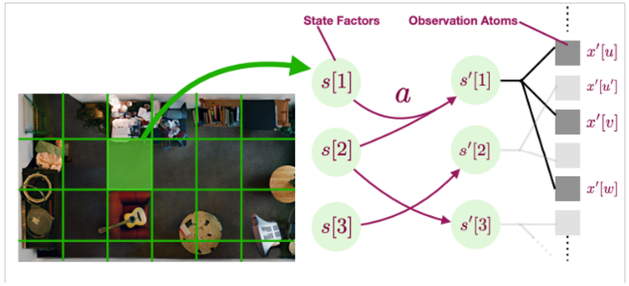
Low rank MDP

[Modi et al., 21]



Linear quadratic control

[Dean-Recht 20, Mhammedi et al., 20]



Factored MDP

[Misra et al., 20]

Exogenous MDP

[Efroni et al., 21]

Discussion

- RL is not like supervised learning: strong assumptions!
- Info theory: Bilinear classes framework is quite comprehensive
- Algorithms: Stronger assumptions than required, worse guarantees
- Huge theory-practice gap!
- Many vibrant sub-topics that we did not cover today!

Many unresolved issues and much work to do. Come join the fun!