Batch Value-Function Tournament

Nan Jiang

ML Pipelines

	Training	Validation	Testing (Evaluation)
Supervised Learning	difficult (optimization)	easy: cross/holdout validation	easy: just… test it
Offline RL	more difficult (hyperparam sensitivity)	even more difficult	most difficult (validation reduces to evaluation)

Reduction to OPE?

- Training algorithms produce $\pi_1, \pi_2, \pi_3, \ldots$. Choose (apprx) best one on validation data
- Natural solution: use OPE (off-policy evaluation) to estimate $J(\pi_i)$
- OPE approaches
  - Importance sampling [Precup et al'00, Jiang & Li'16, etc]: exponential variance
  - ADP (e.g., Fitted-Q [Paine et al'20]) / ALP [Liu et al'18, Nachum et al'19, Uehara et al'20, etc]: require additional function approximation
- **Elephant in the room**: to tune hyperparameters you need to tune hyperparameters!

- Analog of SL holdout-validation? i.e., hyperparameter-free?

Reformulation: Value-function Selection

Training algs often produce **more than policies**… so, select value functions?

Simple(?) Problem

- Run your fav training alg with different neural architectures
- Get candidate value functions f_1, f_2, …
- Select the best approx of Q^* using a "small" holdout dataset?
    - "small" = no |S| or exponential-in-horizon
    - & no further function approximation!

What was known

- nothing: can't even handle 2 functions
    - hardness conjecture [Chen & Jiang, ICML-19]
- Our solution: BVFT [Xie & Jiang, ICML-21] with deep RL implementation [Zhang & Jiang, NeurIPS-21]

Markov Decision Process (MDP)

- For $t = 0, 1, 2, \ldots$, the agent
  - observes state $s_t \in S$ (very large)
  - chooses action $a_t \in A$ (finite)
  - receives reward $r_t = R(s_t, a_t)$

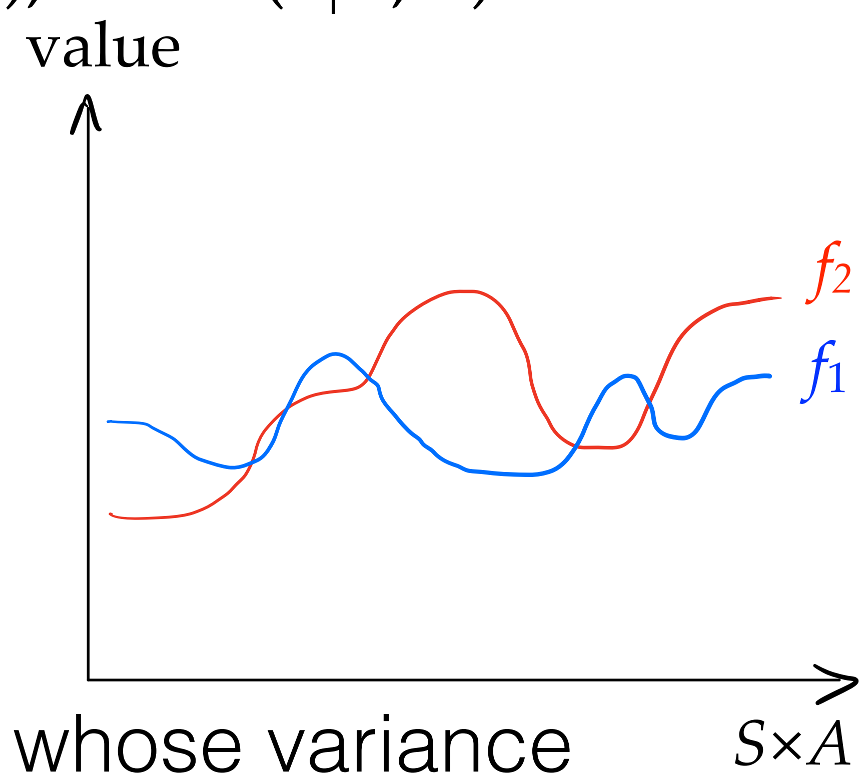
transition dynamics
$P: S \times A \to \Delta(S)$

reward function
$R: S \times A \to [0,1]$

- Policy $\pi: S \to A$
- Expected return $J(\pi) := (1-\gamma)\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t | s_0; \pi]$
  - assume initial state s_0 wlog
- Key solution concepts
  - Bellman eq: $Q^\star = \mathcal{T} Q^\star$, $Q^\pi = \mathcal{T}^\pi Q^\pi$
    where $(\mathcal{T} f)(s,a) = R(s,a) + \gamma \mathbb{E}_{s' \sim P(s,a)}[\max_{a'} f(s', a')]$
  - Occupancy: $d^\pi(s,a) = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \mathbb{P}[s_t = s, a_t = a \mid \pi]$

Value-function selection in large MDPs

- Dataset $D = \{(s, a, r, s')\}$
  - $(s, a) \sim d^D$ ("data distribution"), $r = R(s, a)$, $s' \sim P(\cdot \mid s, a)$
- Candidate functions: f_1, f_2
- Suppose one of them is Q^*… how to identify it?
- Minimal requirement on the algorithm
  - Consistent (∞ data => Q^* identified)
  - On finite data, never estimate anything whose variance grows w/ $|S|$ or $\exp(H)$ (H is effective horizon $1/(1-\gamma)$)
    - can have $poly(1/\varepsilon)$ dependence
- Hardness results [Wang et al'20, Zanette'21, Foster et al'21]



Challenge in value-function selection

- Seems possible to verify $Q^\star = \mathcal{T} Q^\star$ on data?
- Problem: $f - \mathcal{T} f$ is unlearnable [Sutton & Barto'18]
- Naive "1-sample" estimator is biased

$$\mathbb{E}_{d^D}\left[\left(f(s,a) - r - \gamma \max_{a'} f(s',a')\right)^2\right]$$

$$= \mathbb{E}_{d^D}\left[(f - \mathcal{T} f)^2\right] + \mathbb{E}_{d^D}\left[\mathbb{V}_{s'|s,a}[r + \gamma \max_{a'}(s',a')]\right]$$

$:= \|f - \mathcal{T} f\|_{2,d^D}^2$, what we want

Bayes-error-like term depending on f

- unbiased estimation requires "*double sampling*" [Baird'95] or helper class $\mathcal{G} \ni \mathcal{T} f$ [Antos'08] ("*Bellman-completeness*")

Seemingly Impossible?

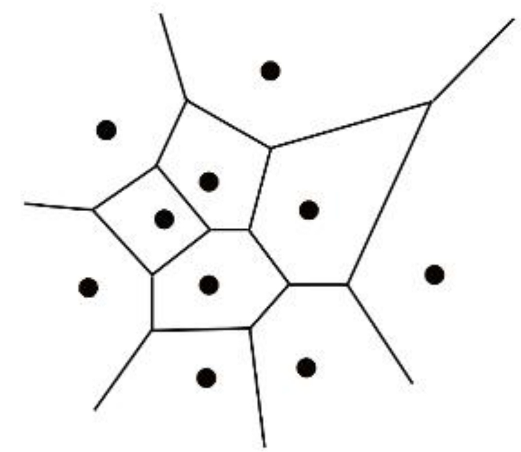
- Validation is just training w/o optimization difficulties!
- Open problem in offline RL (now resolved)

> **Is poly-sample learning possible w/**
> - Exploratory data
> - F s.t. $Q^* \in F$ (*realizability*)

- All existing algorithms require stronger assumptions on (e.g., Bellman-completeness)
- Is a positive result possible?

Projected Bellman error $\|f - \Pi_{\mathcal{G}}\mathcal{T}f\|_{2,d^D}$

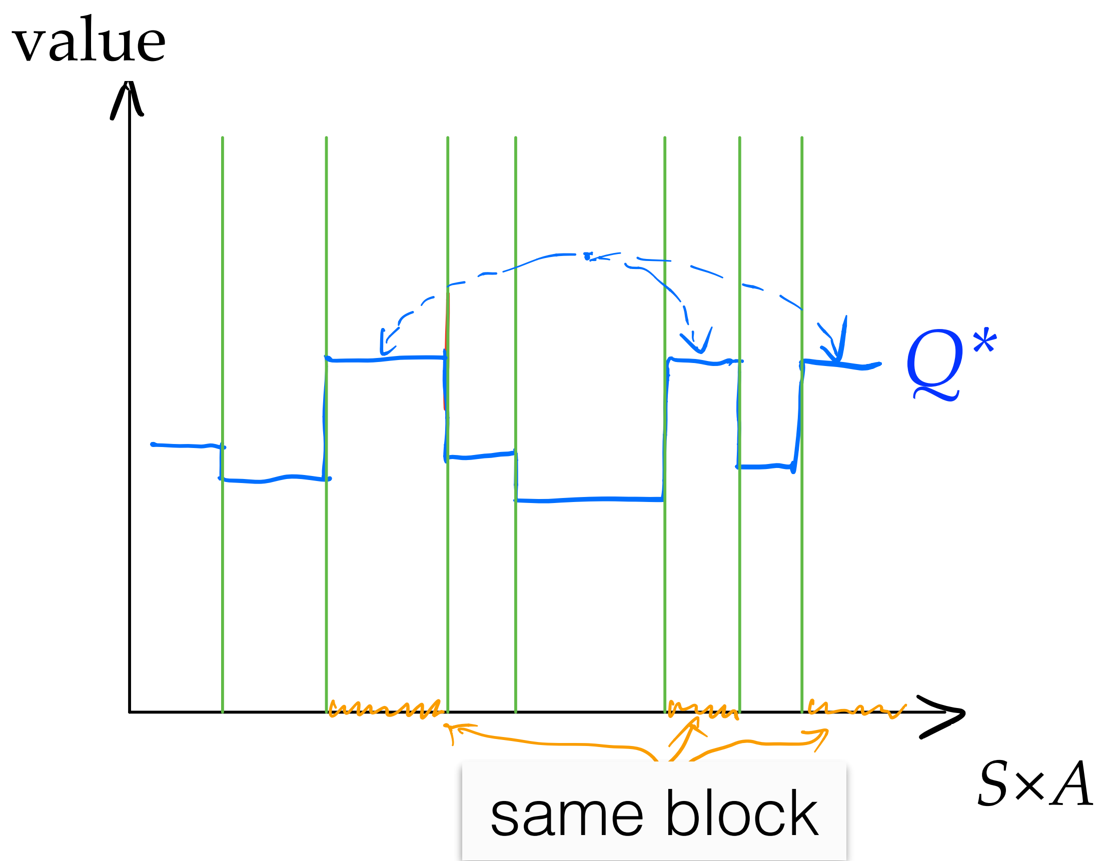
- Estimation: $\Pi_{\mathcal{G}}\mathcal{T}f \approx$ ERM of $\{(s,a) \mapsto r + \gamma \max_{a'} f(s',a')\}$ in G
  - G needs to have bounded complexity
- Consistent, i.e., $\|f - \Pi_{\mathcal{G}}\mathcal{T}f\|_{2,d^D} = 0 \Leftrightarrow f = Q^\star$, if
  - $Q^\star \underset{\sim}{\in} \mathcal{G}$
  - G is piecewise constant (induced by some partitioning) [Gordon'95]
  - Reason: $\Pi_{\mathcal{G}}\mathcal{T}$ is contraction for piecewise-constant G
  - Related to "Q^*-irrelevant abstractions" [Li et al'06]



- Where to find such a magical G?
  - create it "out of nothing"!

The ideal choice of G

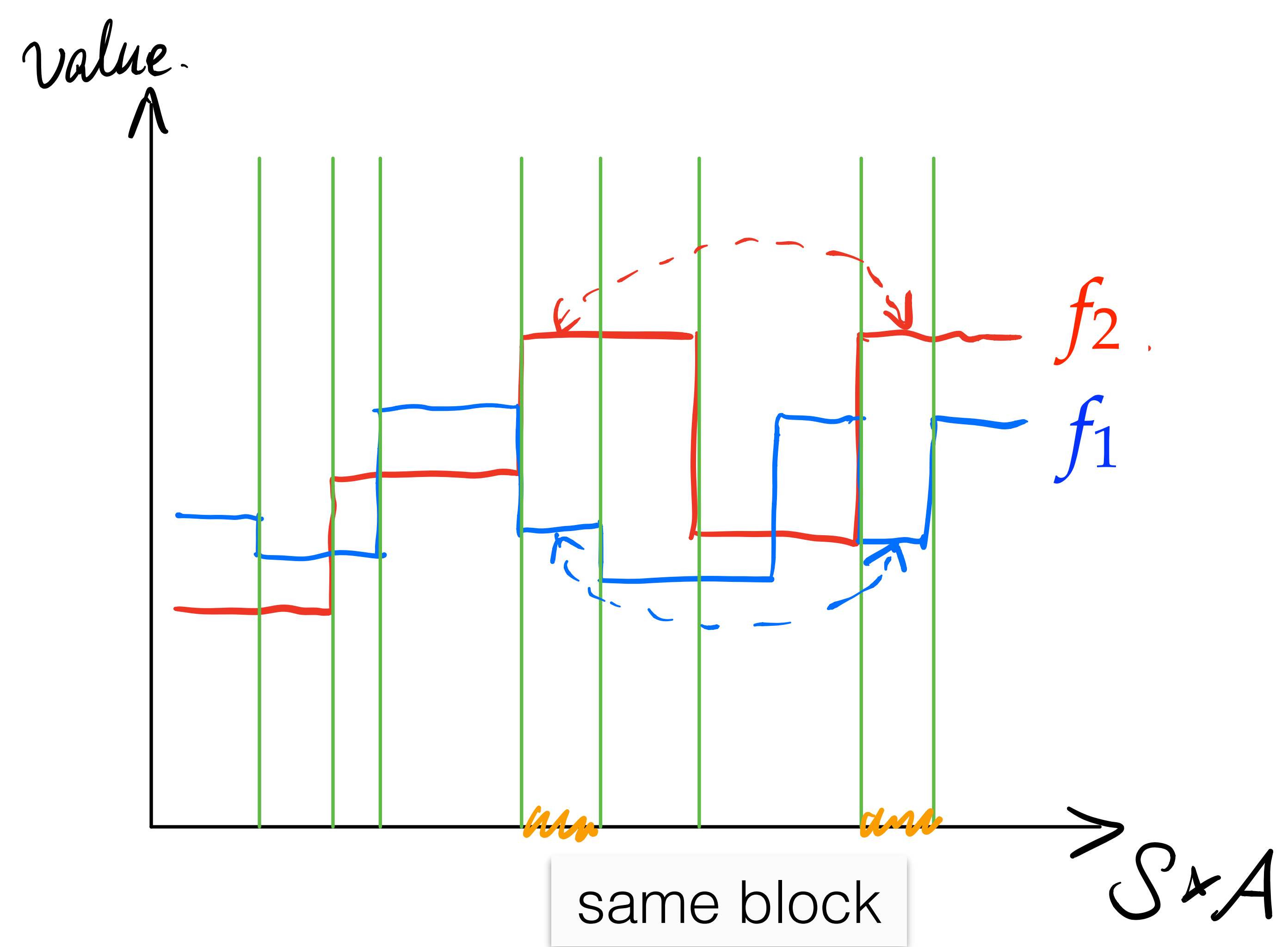
- Does a low-complexity G always exist?
- YES! Just partition SxA according to Q^*
  - (SxA).groupBy { (s, a) => round($Q^*(s, a)$ / ε) }
  - #partitions: $O(1/\varepsilon)$ (ε is discretization error)



- Chicken-and-egg: only if I knew Q^*…

Pairwise Comparison

- Recall that problem is still nontrivial even when $|F|$=2!
  - One f_1, f_2 of is Q^*: how to find out from data?
- Partition SxA according to both functions in F simultaneously!
  - size of ϕ: $O(1/\varepsilon^2)$ — affordable!!!
- Fixed point of $\widehat{\mathcal{T}}_\phi^\mu$ will be close to Q^* => choose the one w/ lower $\|f - \widehat{\mathcal{T}}_\phi^\mu f\|$
- Extend to large F?
  - Naive: generate partition of size $O(1/\varepsilon^{|F|})$ X



Batch Value-Function Tournament [Xie & Jiang'20b]

- Algorithm: $\arg\min_{f\in\mathcal{F}} \max_{f'\in\mathcal{F}} \|f - \widehat{\mathcal{T}}_{\phi_{f,f'}} f\|_{2,D}$ ← partition created out of f and f'
  - Inspired by Scheffé tournament & tournament algorithms for model selection in RL [Hallak et al'13, Jiang et al'15]
- Concern: not every ϕ is "good" (i.e., Q^*-irrelevant)
  - For $f = Q^*$: always tested on good ϕ => small error for all f'
  - For bad f : tested on a good ϕ when $f' = Q^*$ => large max error

Theorem: when F is realizable, the sample complexity of BVFT for obtaining an ε-optimal policy is $\tilde{O}\left(\frac{C^2 \ln \frac{|\mathcal{F}|}{\delta}}{\epsilon^4(1-\gamma)^8}\right)$, where C is a constant that characterizes the exploratoriness of the dataset.

Finite-sample analysis

- Previous reasoning builds on consistency of Q*-irrelevant abstractions
- Finite-sample guarantee additionally requires:

1. Concentration bounds: $\|f - \widehat{\mathcal{T}}_\phi^\mu f\|_{2,D} \approx \|f - \mathcal{T}_\phi^\mu f\|_{2,\mu}$
   - Part of it is to show $\widehat{\mathcal{T}}_\phi^\mu f \approx \mathcal{T}_\phi^\mu f$, i.e., ERM close to population minimizer for non-realizable least-square!
   - Proof idea: all regression problems are *effectively* realizable in the eyes of histogram regressor
   - The other part: $\|\cdot\|_{2,D} \approx \|\cdot\|_{2,\mu}$ with $1/\sqrt{n}$ rate
2. Error-propagation: how $\|f - \mathcal{T}_\phi^\mu f\|_{2,\mu}$ controls $\|f - Q^\star\|_{2,\mu}$
   - In BRM: $f - Q^\star = (f - \mathcal{T} f) + (\mathcal{T} f - \mathcal{T} Q^\star)$
   - In BVFT: $f - Q^\star = (f - \mathcal{T}_\phi^\mu f) + (\mathcal{T}_\phi^\mu f - \mathcal{T}_\phi^\mu Q^\star)$

controlled by alg | determines error prop

Error propagation

How $\|f - \mathcal{T}_\phi^\mu f\|_{2,\mu}$ controls $\|f - Q^\star\|_{2,\mu}$

- Standard assumption: μ puts enough prob in each "block" of ϕ
- Corresponds to well-conditioned design matrix for linear class
- Problem: our ϕ is quite arbitrary
- Any assumption that is independent of ϕ?

Assumption 1. We assume that $\mu(s,a) > 0 \ \forall s,a$. We further assume that
(1) There exists constant $1 \le C_\mathcal{A} < \infty$ such that for any $s \in \mathcal{S}, a \in \mathcal{A}$, $\mu(a|s) \ge 1/C_\mathcal{A}$.
(2) There exists constant $1 \le C_\mathcal{S} < \infty$ such that for any $s \in \mathcal{S}, a \in \mathcal{A}, s' \in \mathcal{S}$, $P(s'|s,a)/\mu(s') \le C_\mathcal{S}$. Also $d_0(s)/\mu(s) \le C_\mathcal{S}$.
It will be convenient to define $C = C_\mathcal{S} C_\mathcal{A}$.

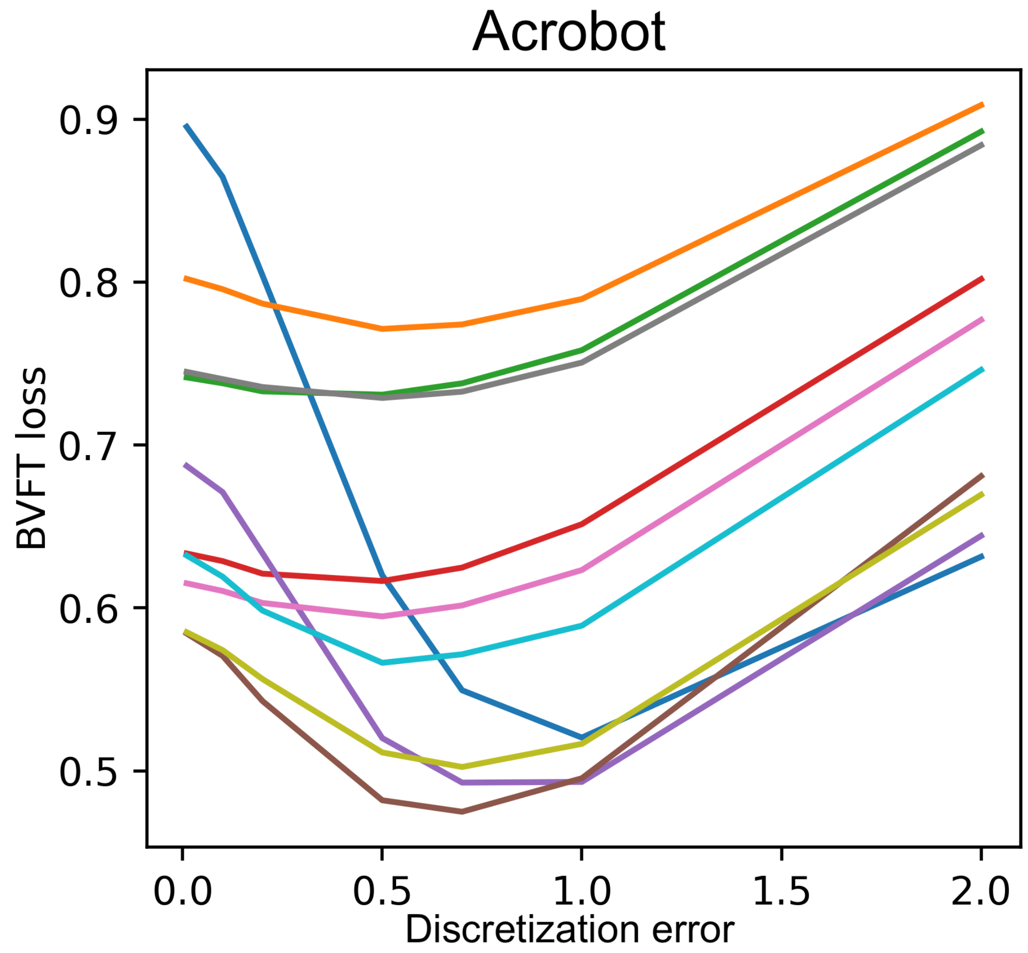
- Key part: $P(s'|s,a)/\mu(s') \le C_\mathcal{S}$ [Munos'03]
- Satisfiable in MDPs whose transition matrix admits low-rank stochastic factorization

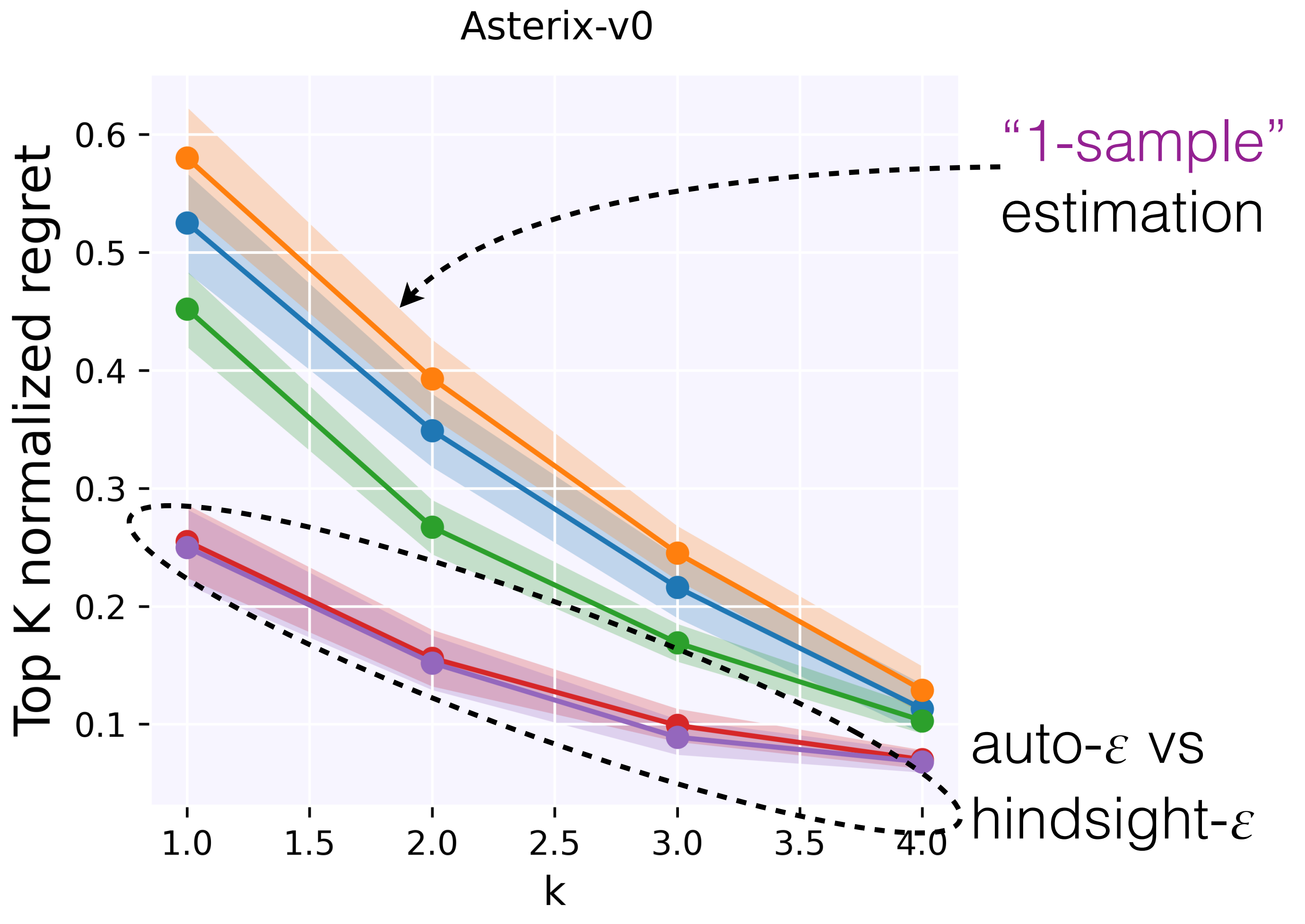
sample complexity:

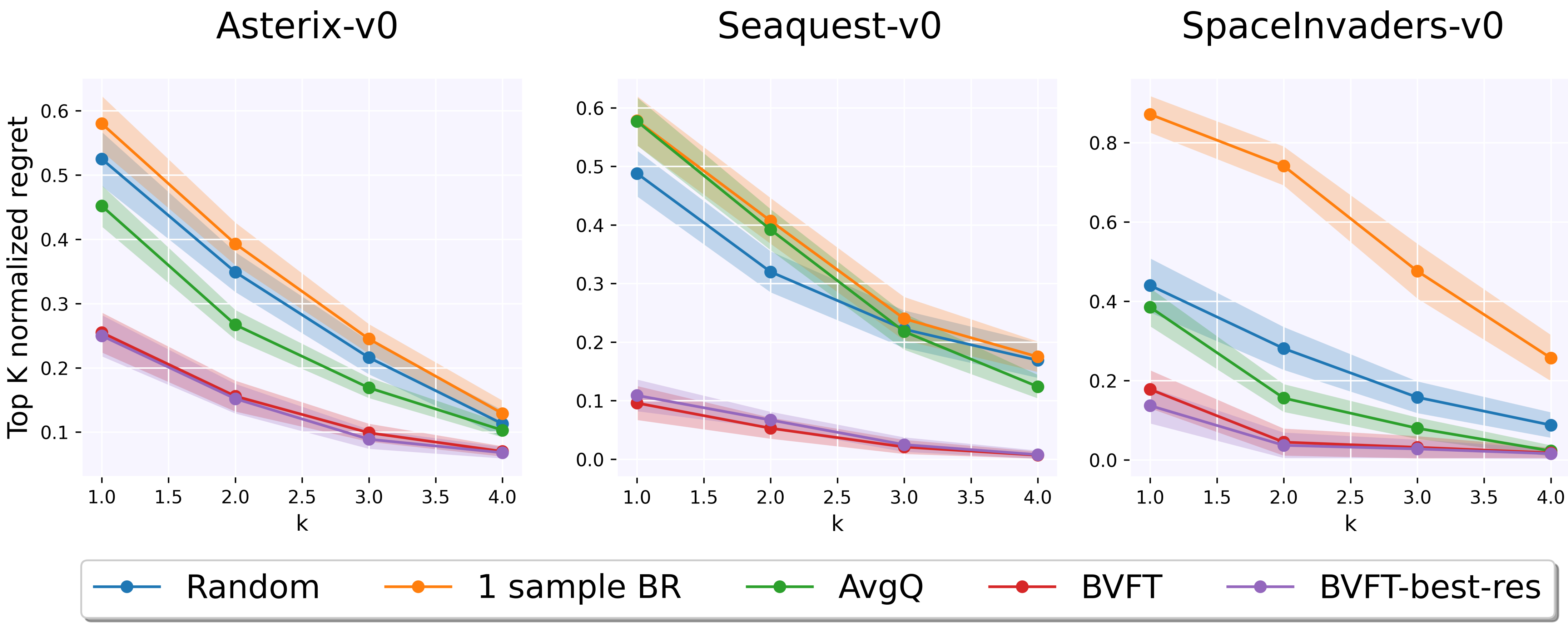
$$\tilde{O}\left(\frac{C^2 \ln \frac{|\mathcal{F}|}{\delta}}{\epsilon^4 (1-\gamma)^8}\right)$$

Practical Implementation of BVFT

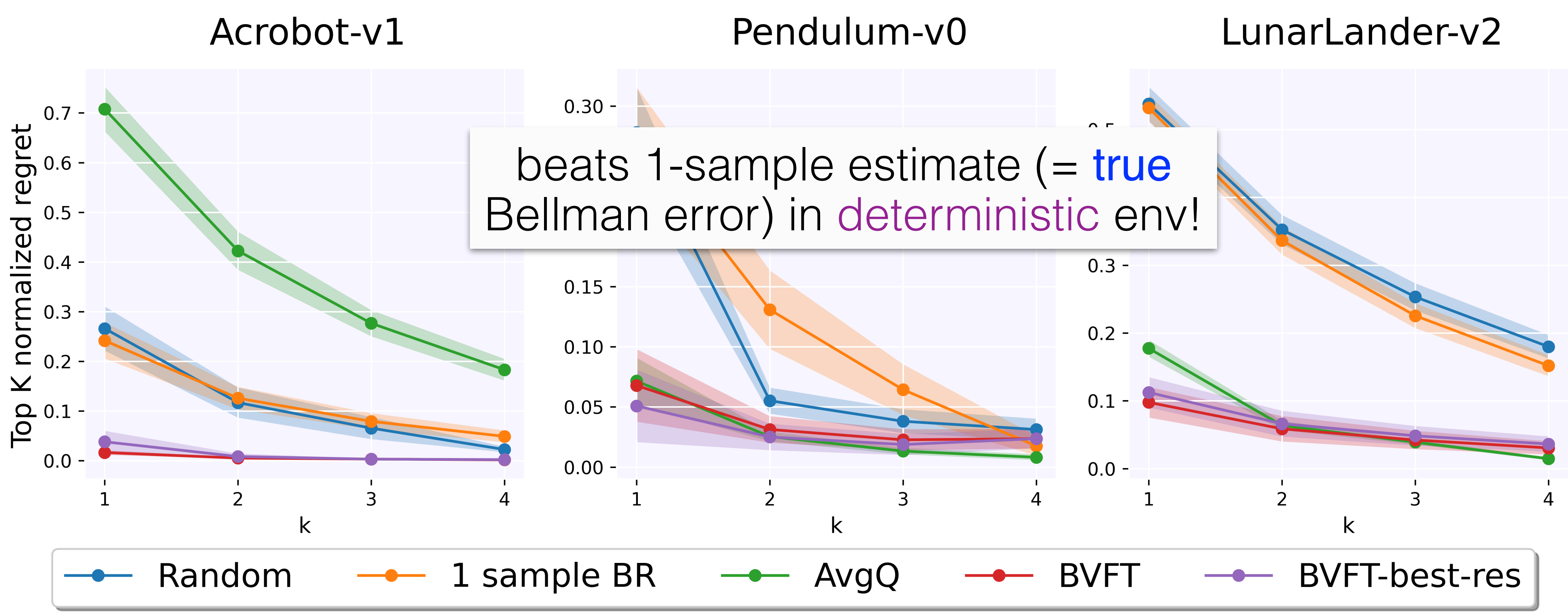
- Challenge: how to set the discretization-level ε
- Observation: degrades to "1-sample" estimation when ε=0
  $\left(f(s,a) - (r + \gamma \max_{a'} f(s',a'))\right)^2$ => positively biased
- Prediction: loss should be U-shaped in ε
- Choice of ε: minimize loss





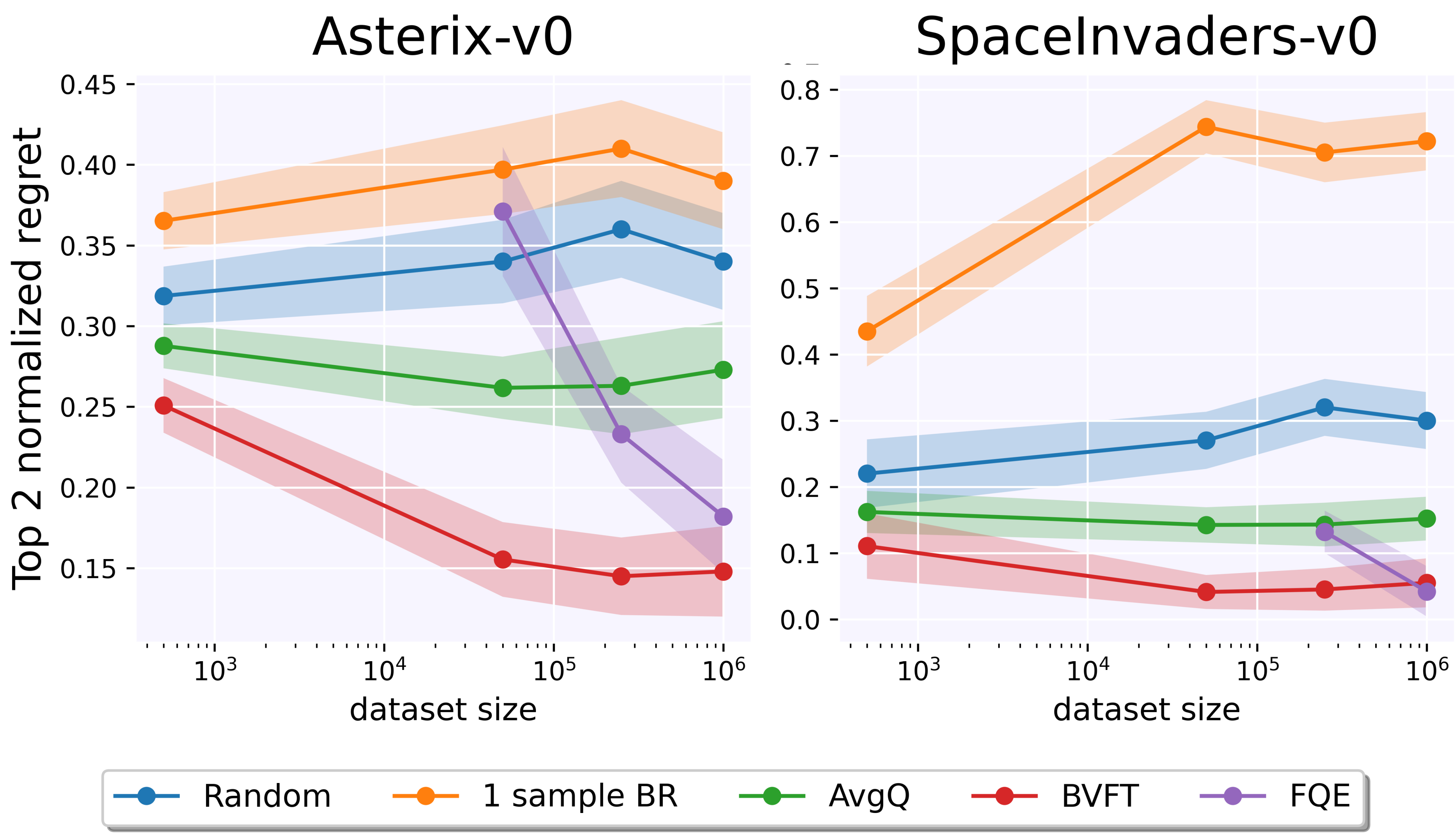



Siyuan Zhang



beats 1-sample estimate (= true Bellman error) in deterministic env!

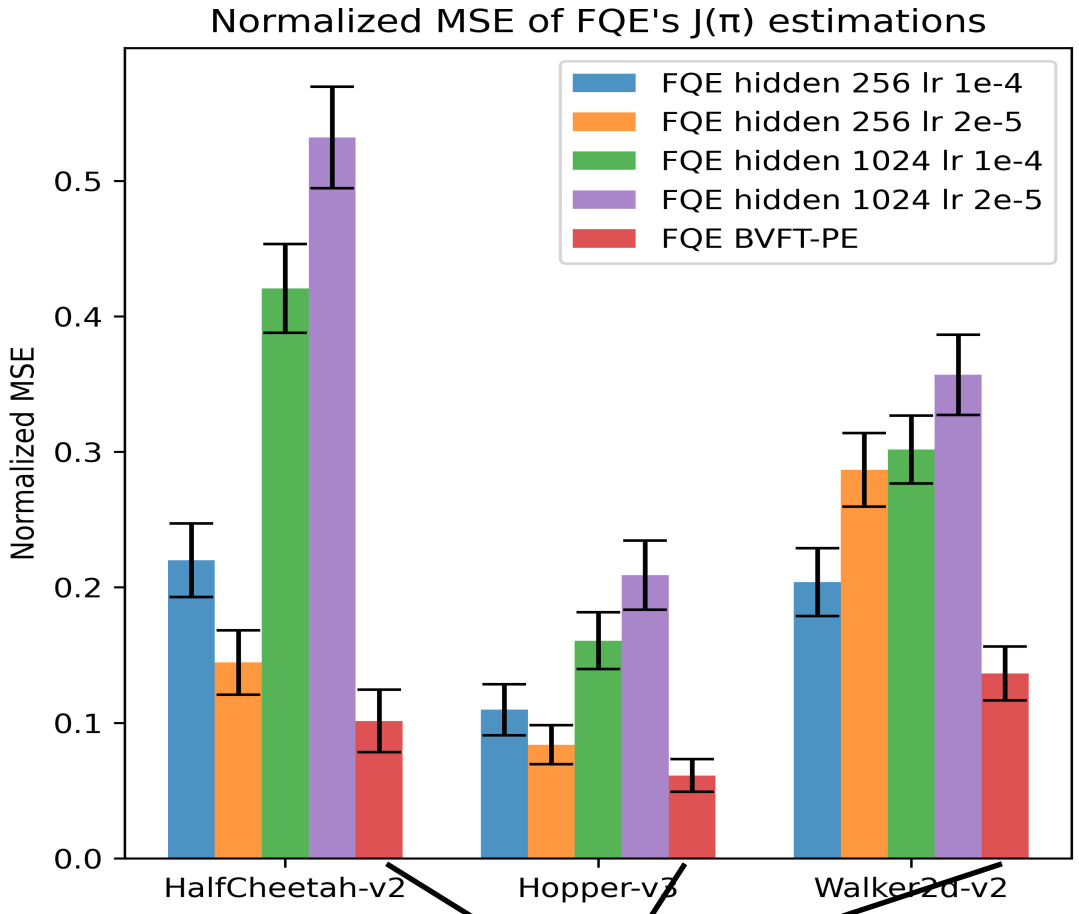
Comparison to FQE (estimating Q^{π} via Fitted-Q)



- Open question: how to tune FQE's neural architecture
- We cheated using training architecture that produces the best policy in Asterix
- FQE needs to handle pixel input and hence sample-inefficient
- BVFT does not care about complexity of state-action space

Hyperparameter tuning for OPE

- Actor-critic algorithms can produce poor critics
  - i.e., all candidates are bad
- Only hope: OPE, but don't know how to tune hyperparams…
- BVFT-PE: can identify Q^{π} from candidate q's



BVFT-PE outperforms best fixed architecture