Model-based Offline RL with Partial Coverage

Wen Sun

CS 6789: Foundations of Reinforcement Learning

Potential Applications of Offline RL

Robotics manipulation:

[Kalashnikov et.al, 18]

Potential Applications of Offline RL

Robotics manipulation:

[Kalashnikov et.al, 18]

Autonomous driving:

[Codevilla et.al, 18]

Potential Applications of Offline RL

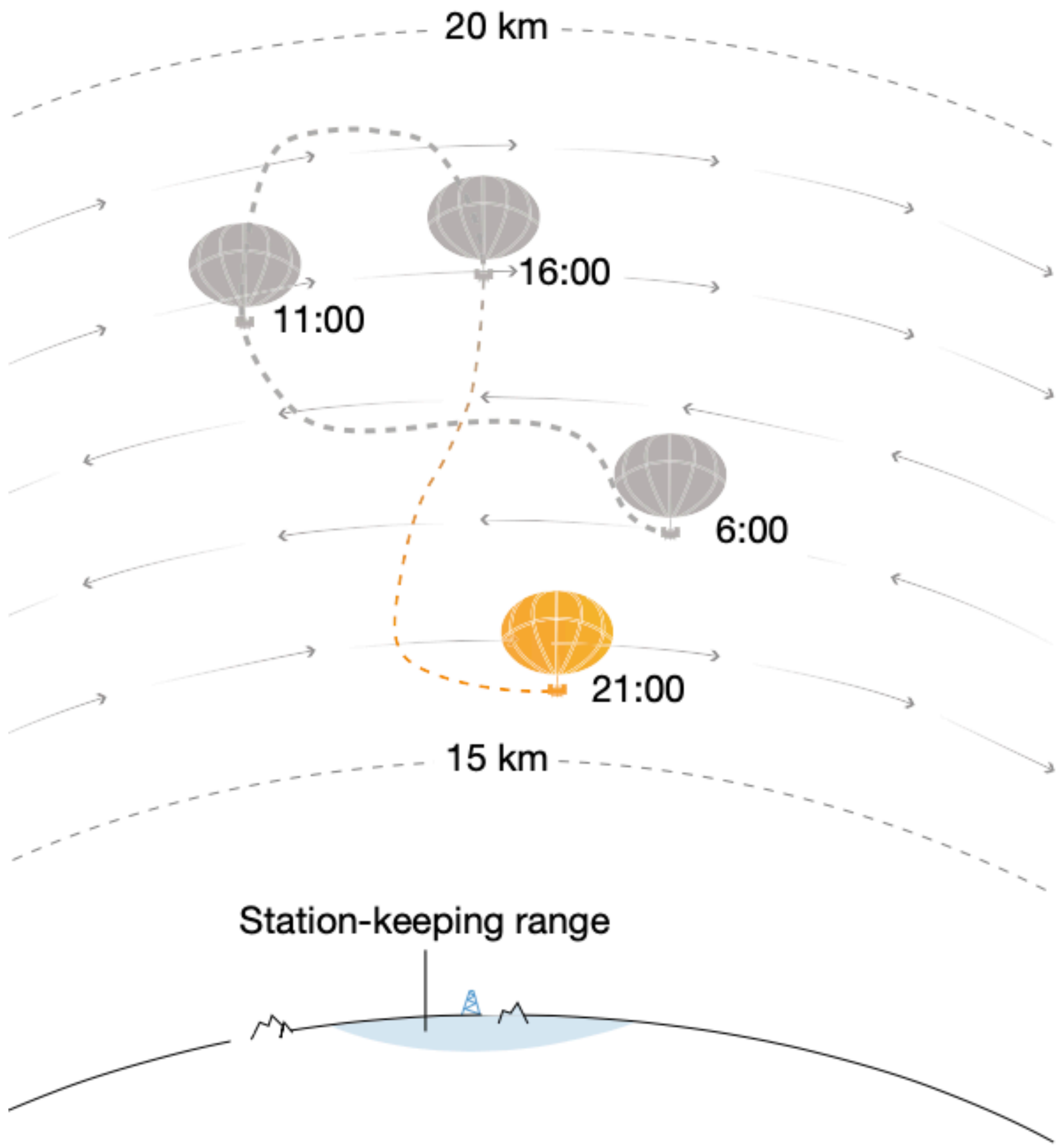
Robotics manipulation:

[Kalashnikov et.al, 18]

Autonomous driving:

[Codevilla et.al, 18]

Stratospheric balloon navigation



[Bellemare et.al,21, Nature]

Potential Applications of Offline RL

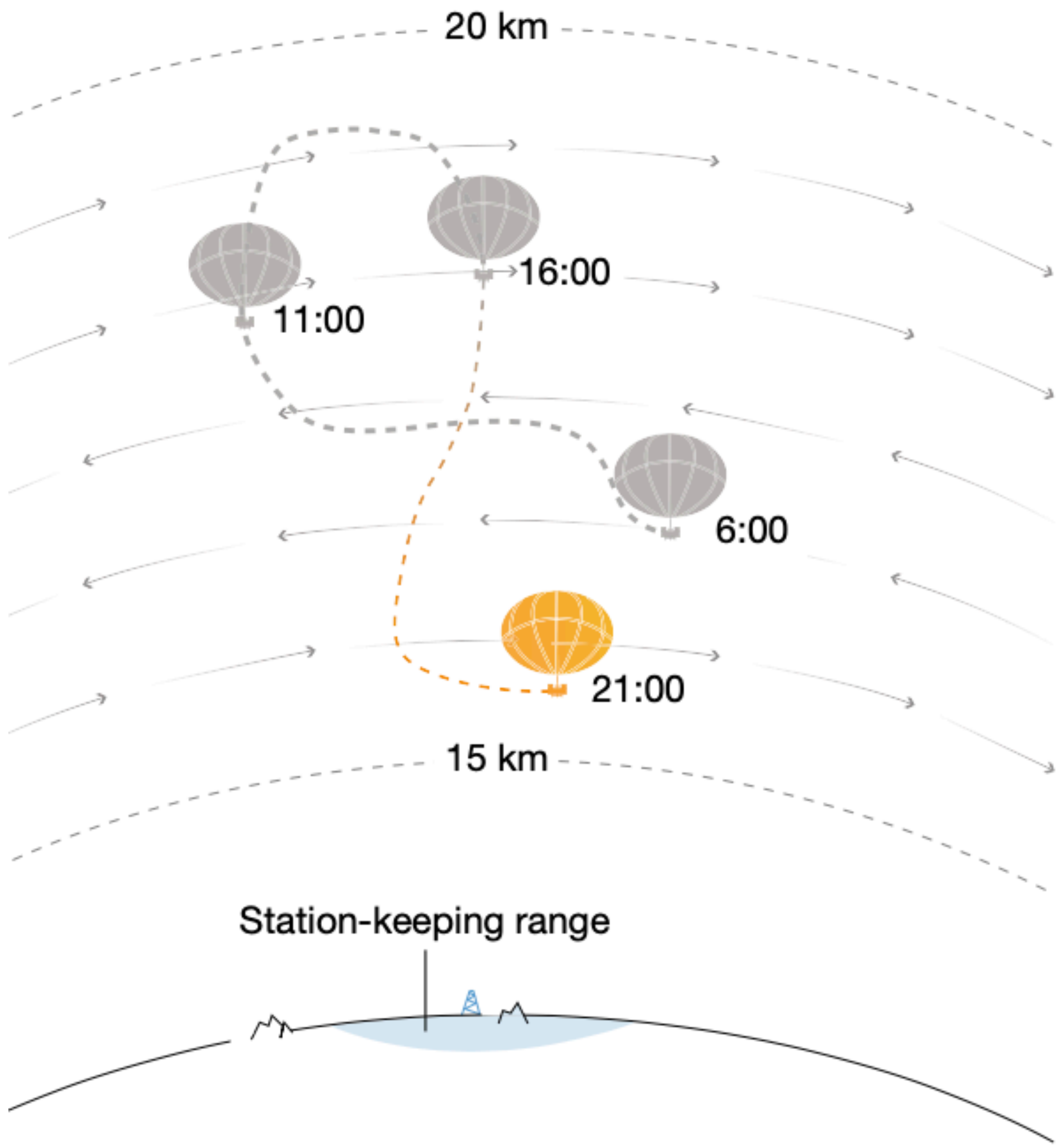
Robotics manipulation:

[Kalashnikov et.al, 18]

Autonomous driving:

[Codevilla et.al, 18]

Stratospheric balloon navigation



[Bellemare et.al,21, Nature]

AND healthcare applications!

(See Levine et.al for a list of applications and existing works)

Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems

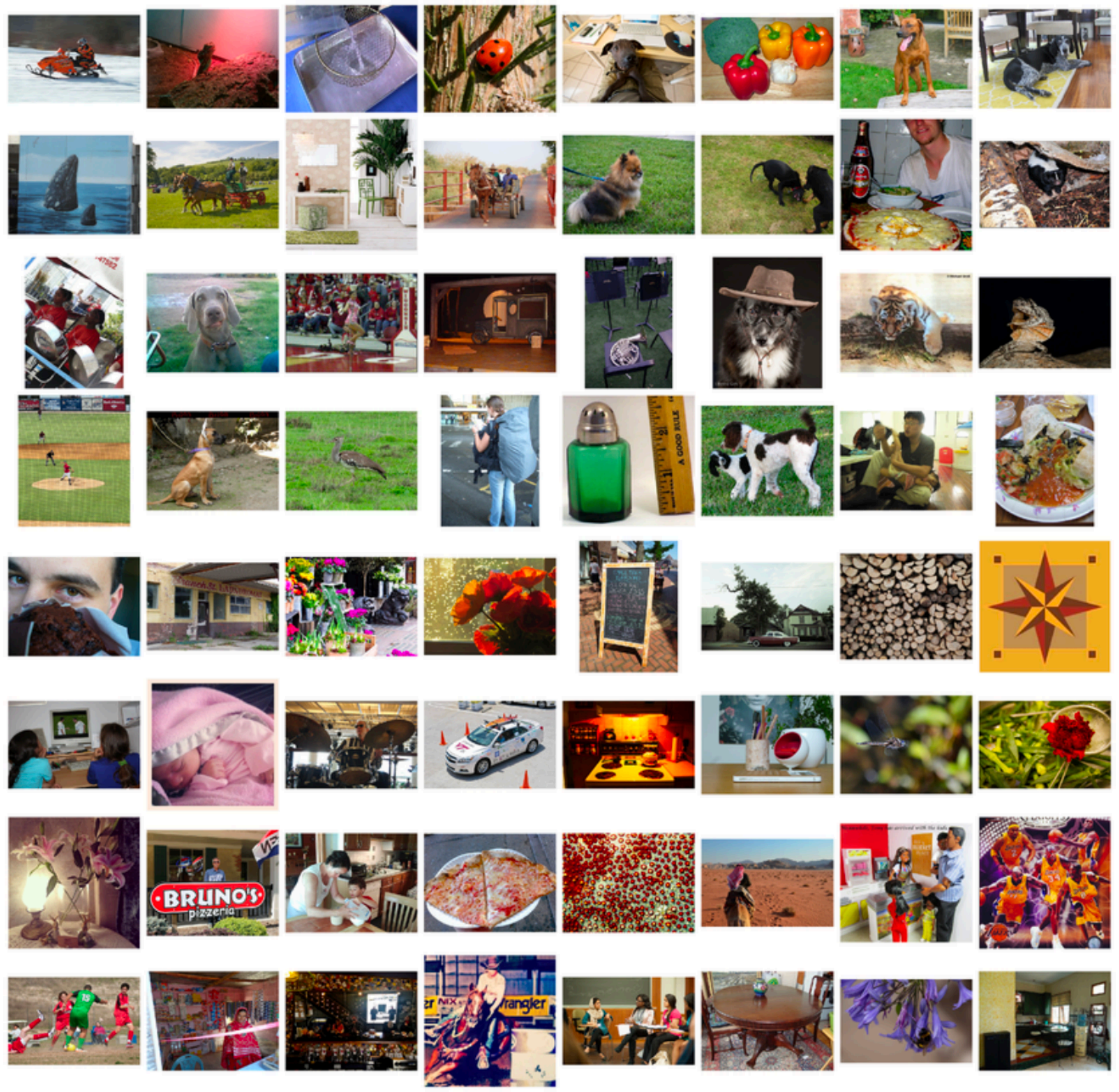
Why offline RL is challenging?

Let's contrast it to traditional Supervised Learning

Why offline RL is challenging?

Let's contrast it to traditional Supervised Learning

e.g., image classification

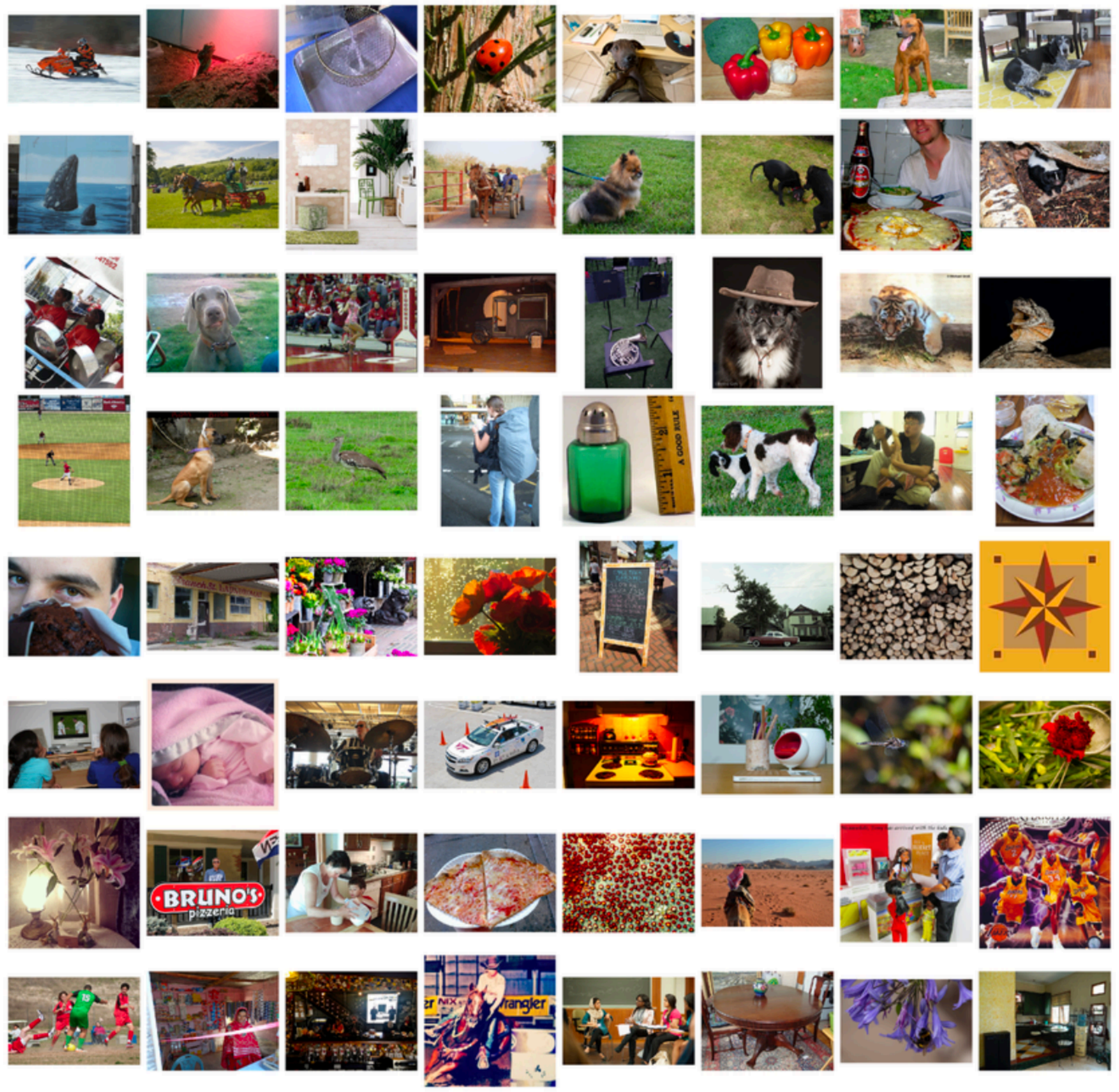


[ImageNet]

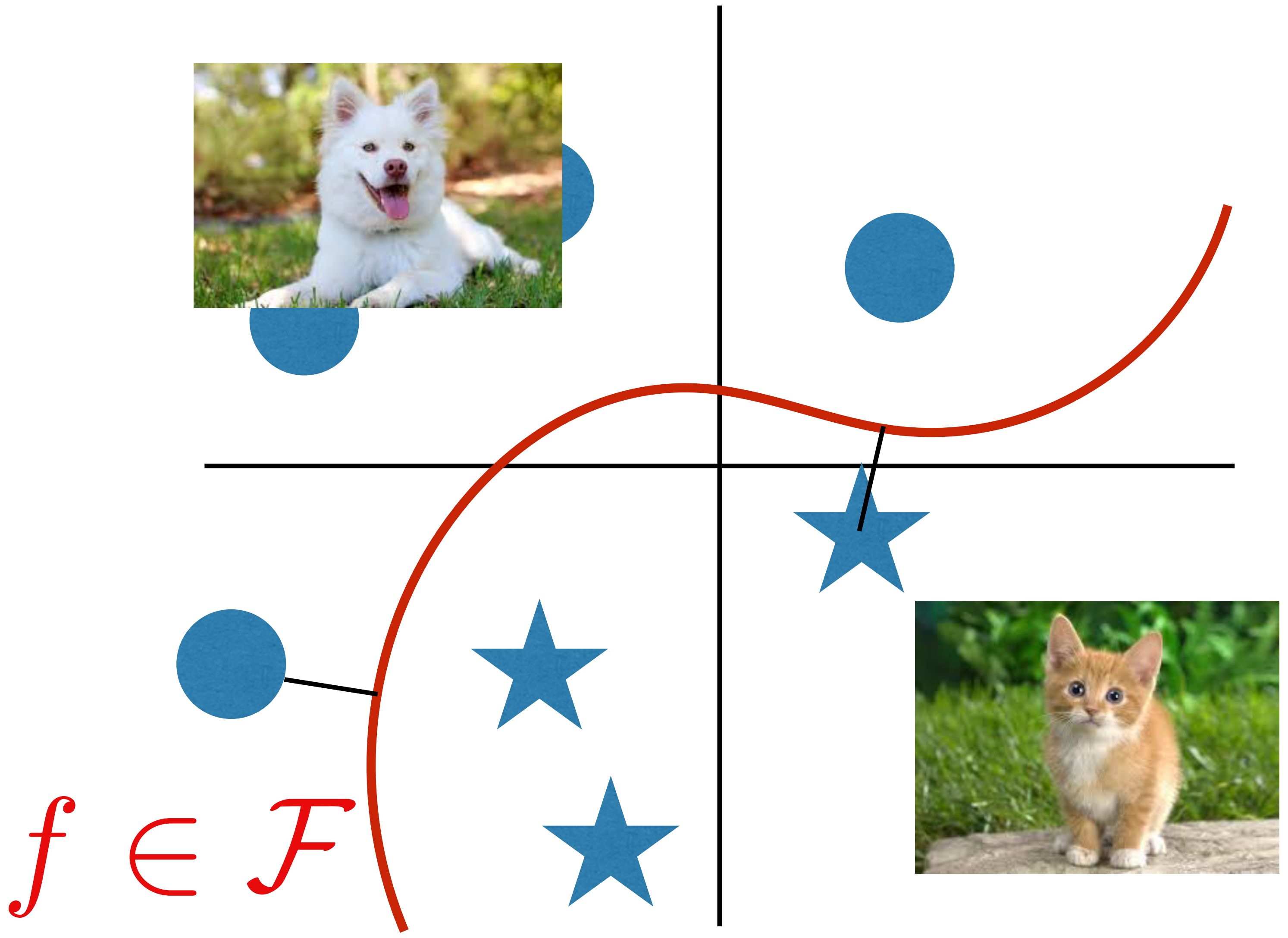
Why offline RL is challenging?

Let's contrast it to traditional Supervised Learning

e.g., image classification



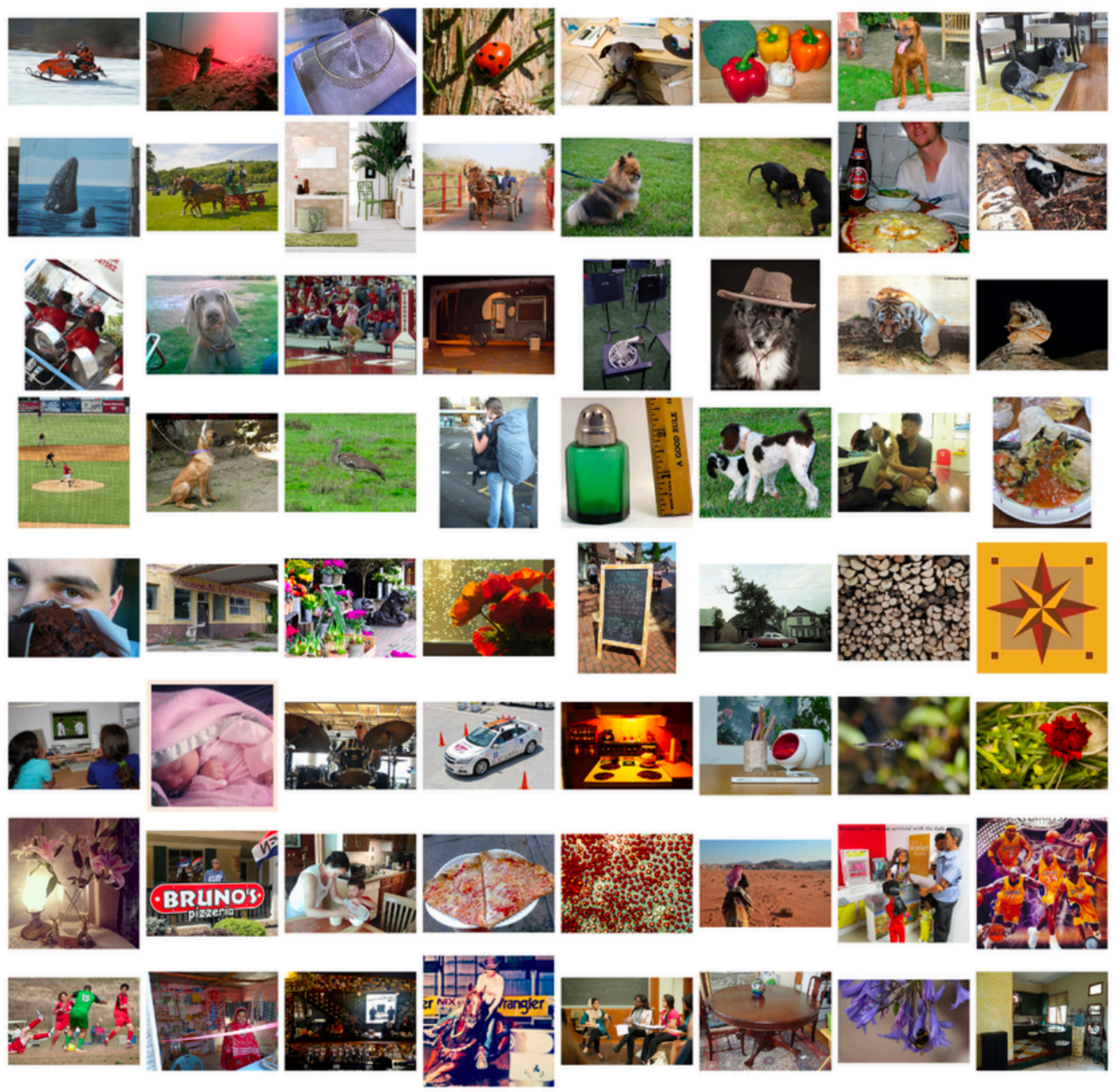
[ImageNet]



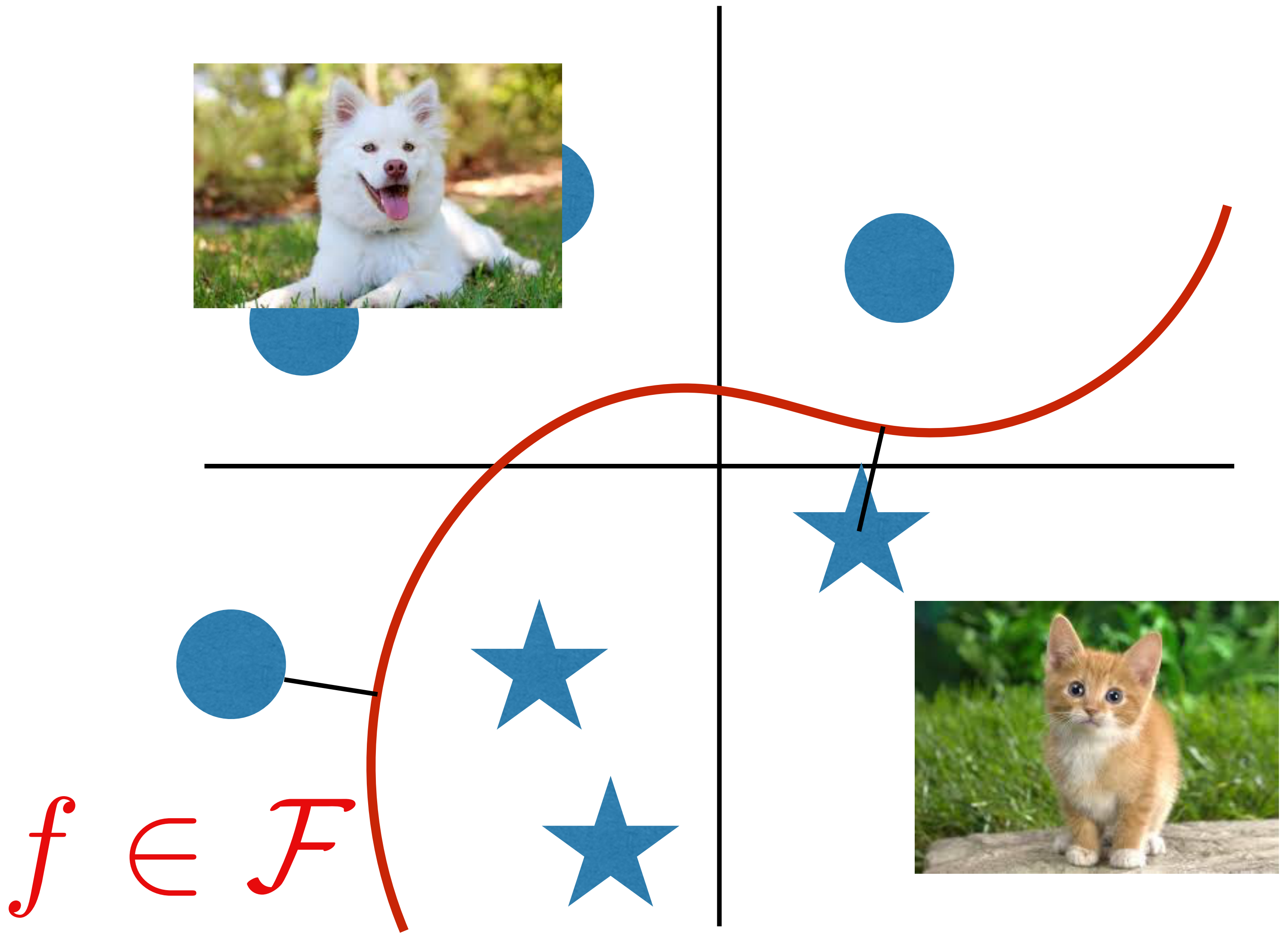
Why offline RL is challenging?

Let's contrast it to traditional Supervised Learning

e.g., image classification



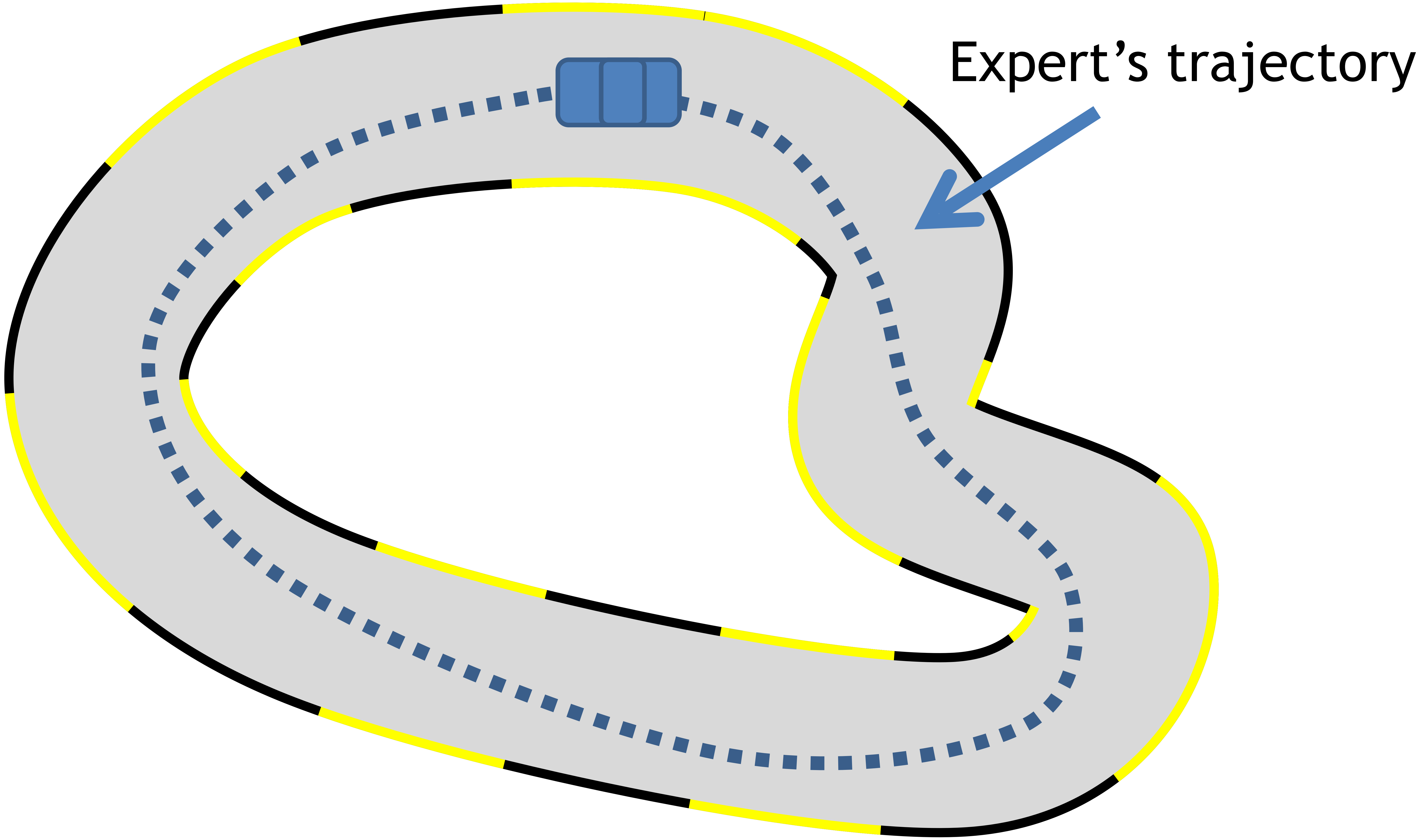
[ImageNet]



Training distribution $=$ Testing distribution

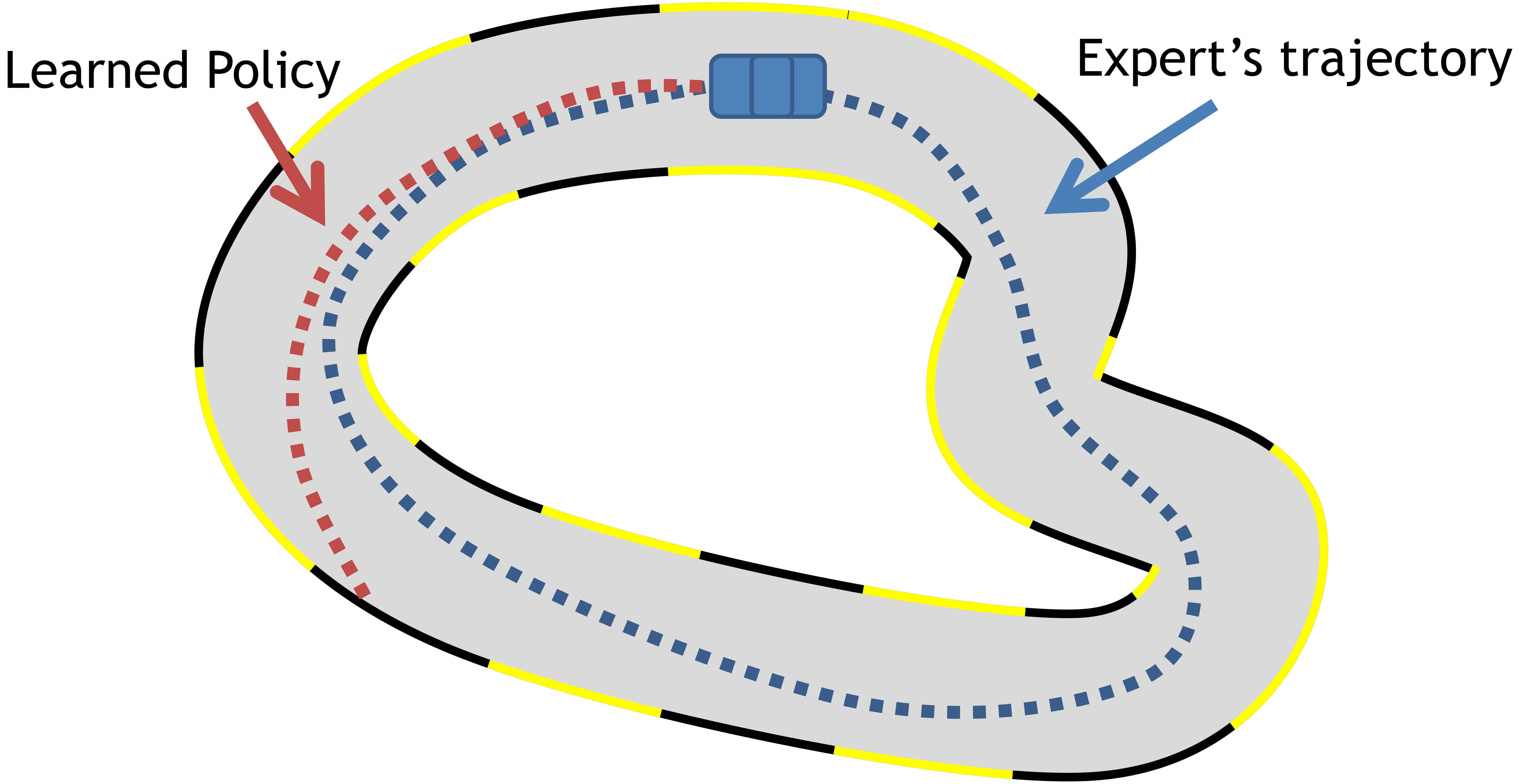
Why offline RL is challenging?

In RL, we are facing **distribution shift!**



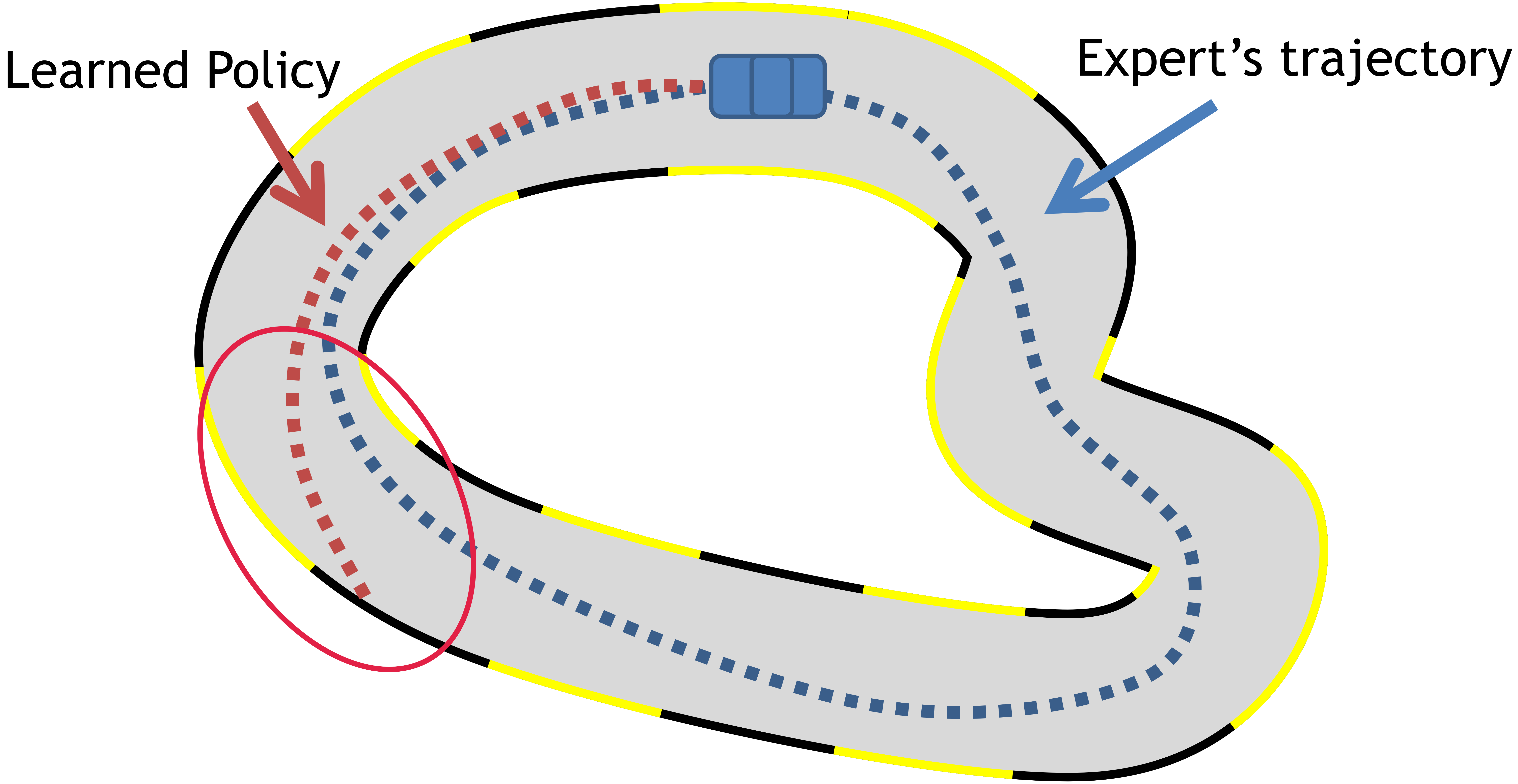
Why offline RL is challenging?

In RL, we are facing **distribution shift!**



Why offline RL is challenging?

In RL, we are facing **distribution shift!**



Today:

How should we tackle offline RL (e.g., settings, goals)

when offline data has insufficient coverage?

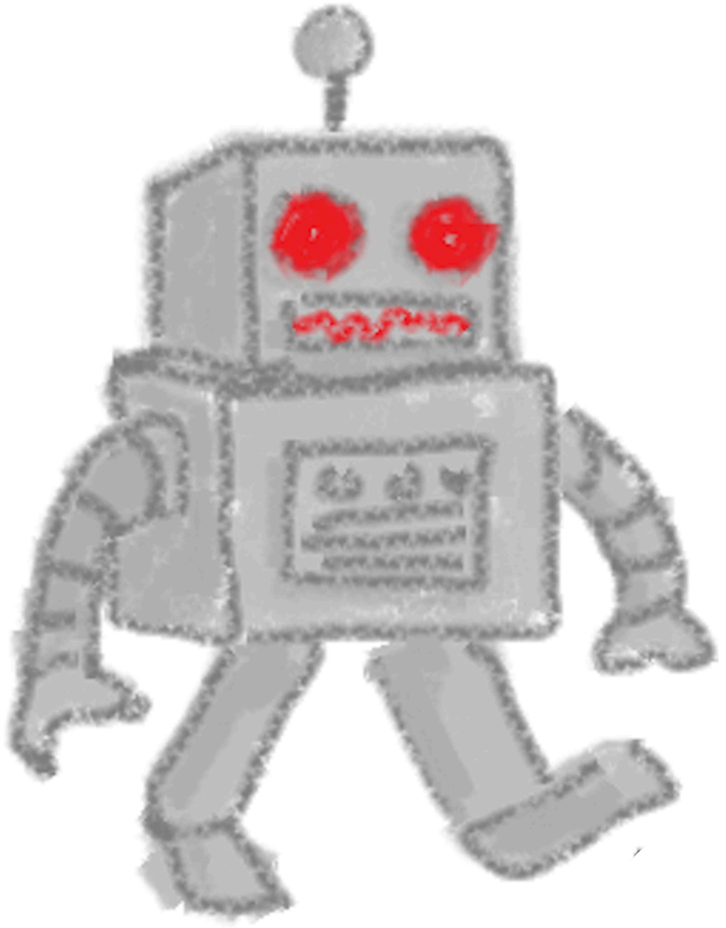
Outline:

1. Rethinking the goal in offline RL: Robustness

2. Can we achieve the goal?

Finite Horizon MDPs

agent



Policy: state to action

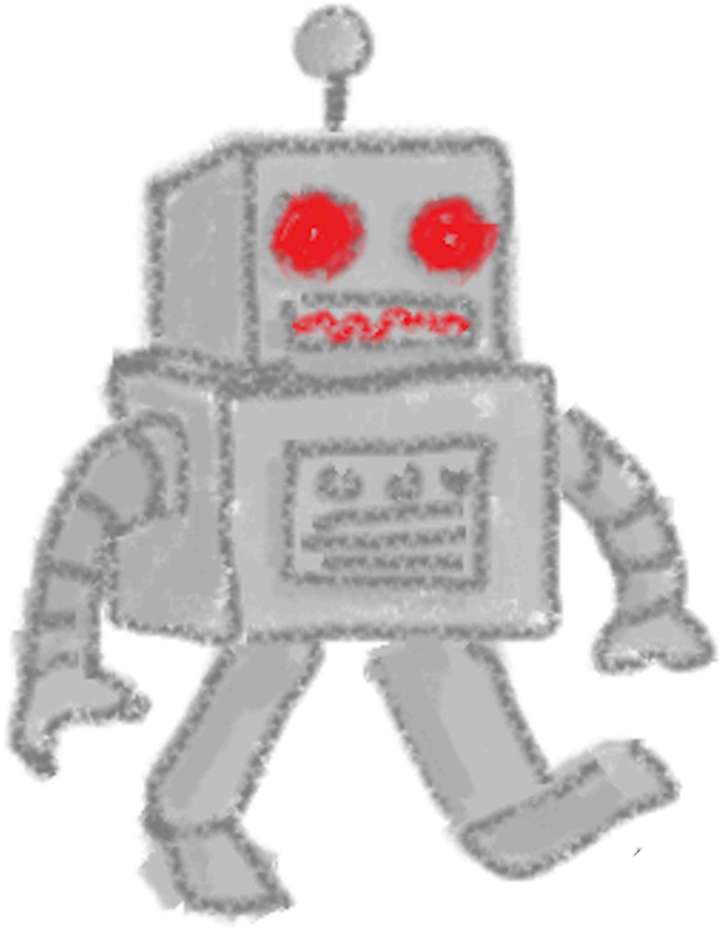
$$\pi(s) \rightarrow a$$

Reward & Next State

$$r(s,a), s' \sim P^\star(\cdot \mid s,a)$$

Finite Horizon MDPs

agent



Policy: state to action

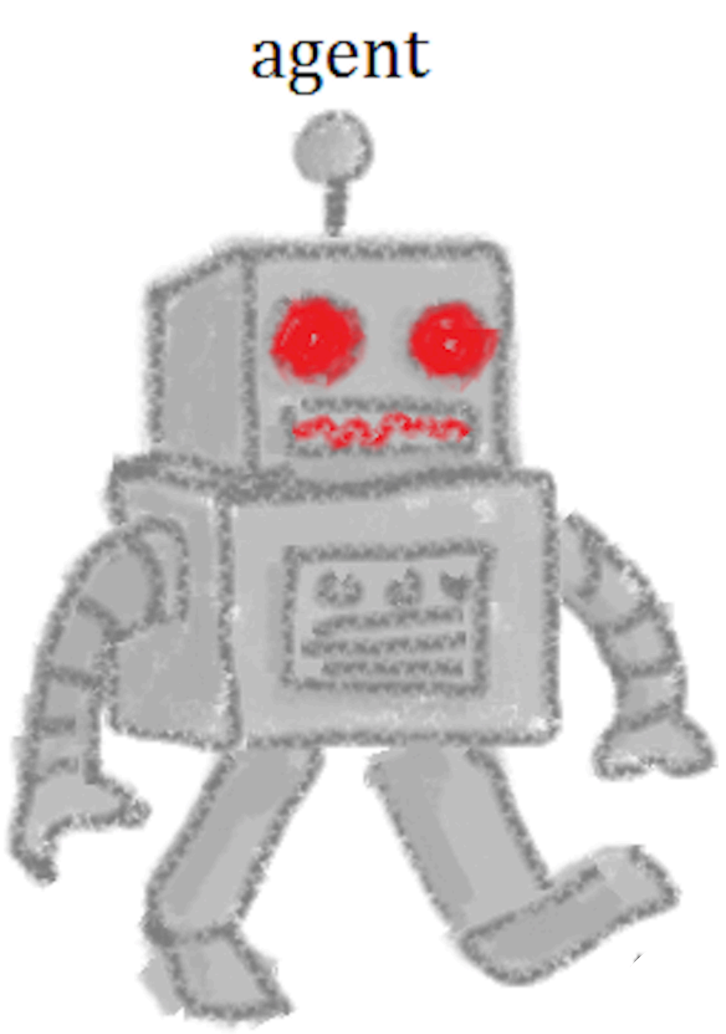
$\pi(s) \rightarrow a$

H steps

Reward & Next State

$r(s, a), s' \sim P^{\star}(\cdot \mid s, a)$

Finite Horizon MDPs



Policy: state to action

$$\pi(s) \rightarrow a$$

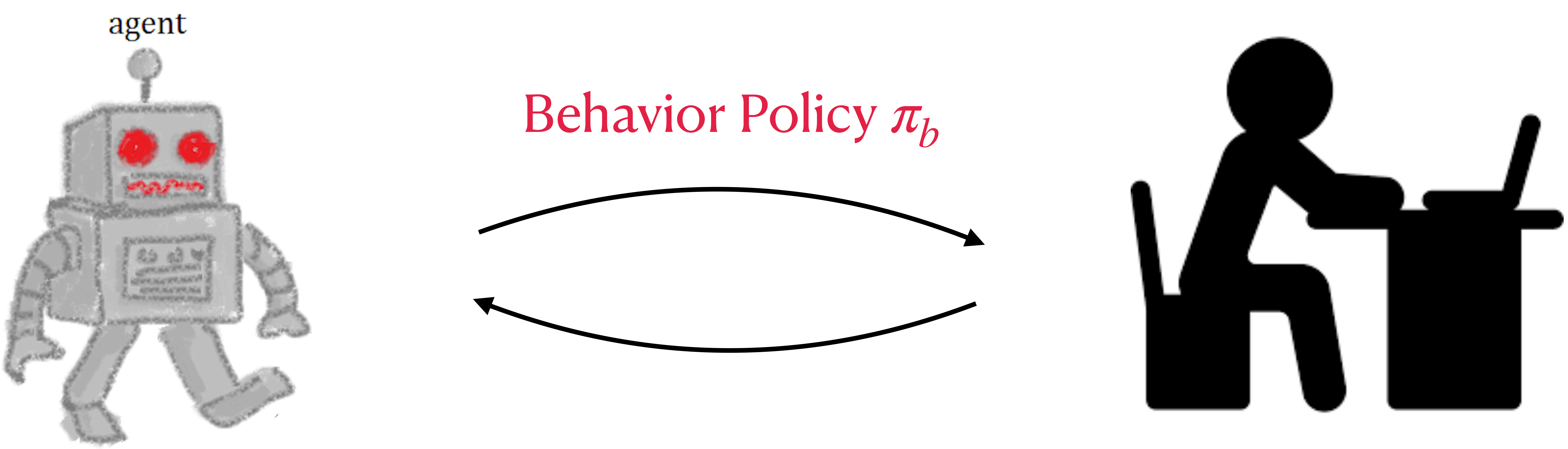
H steps

Reward & Next State

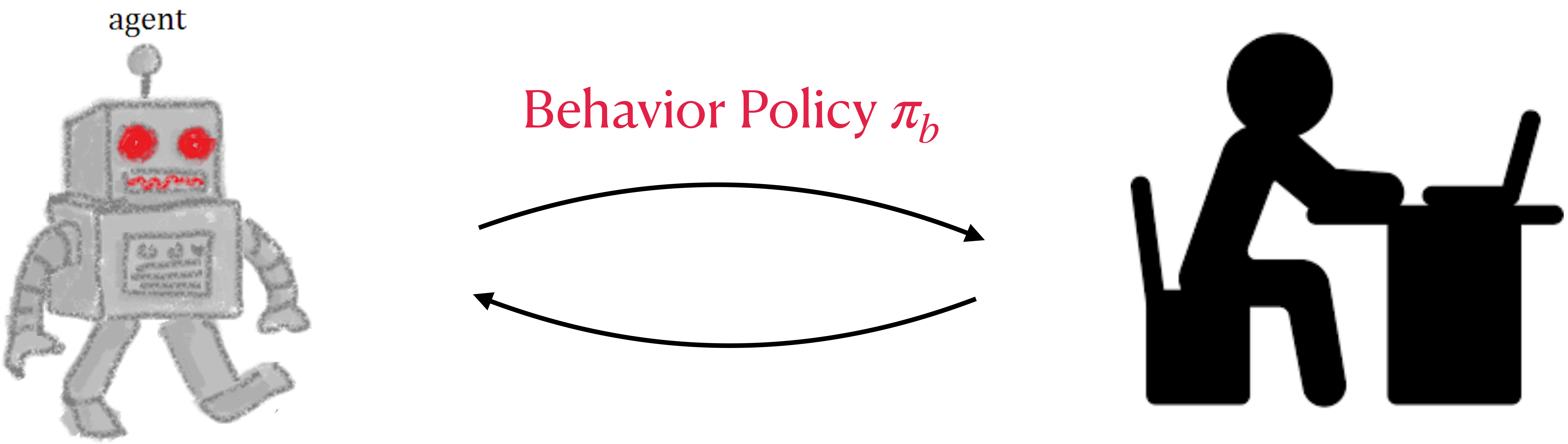
$$r(s, a), s' \sim P^\star(\,\cdot\,|\, s, a)$$

Objective: $\max_\pi J(\pi; P^\star, r)$, where $J(\pi; P^\star, r) := \mathbb{E}\left[\sum_{h=0}^{H-1} r(s_h, a_h) \,|\, a \sim \pi, P^\star\right]$

Offline Data Collection



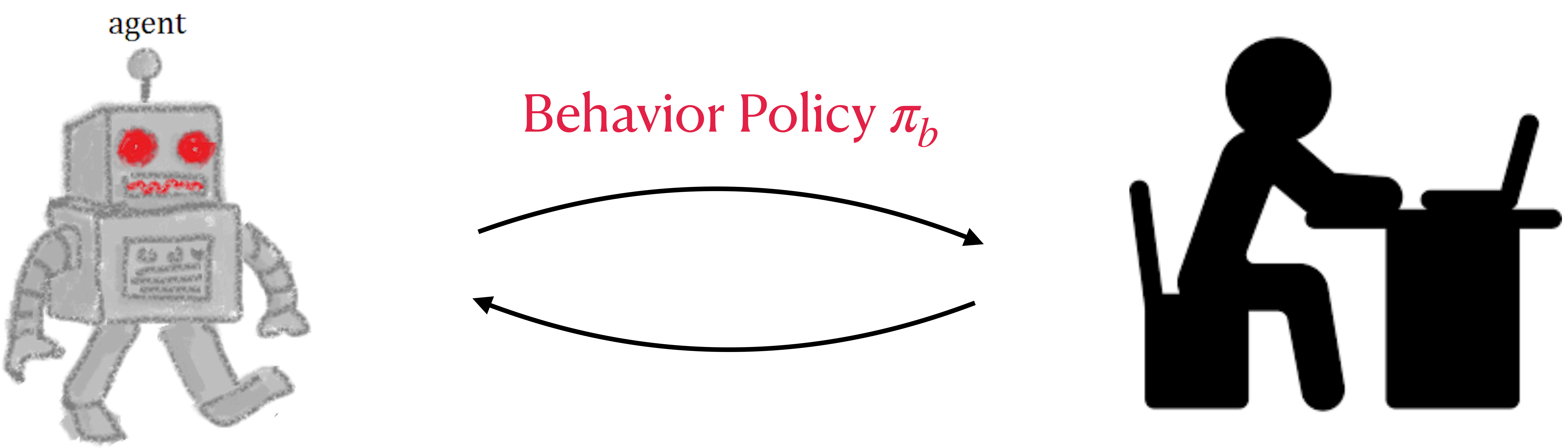
Offline Data Collection



Induced state-action distribution of π_b:

$$d^{\pi_b} \in \Delta(S \times A)$$

Offline Data Collection

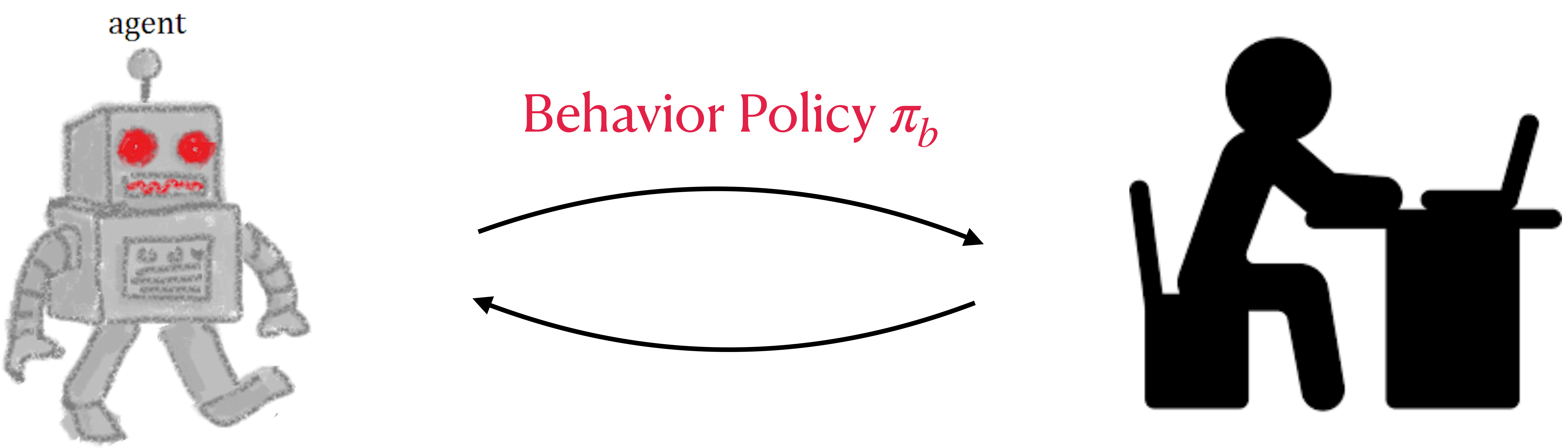


Induced state-action distribution of π_b:

$$d^{\pi_b} \in \Delta(S \times A)$$

$$\mathcal{D} = \{s, a, s'\}, \text{ where } s, a \sim d^{\pi_b}, s' \sim P^\star(\cdot \mid s, a)$$

Offline Data Collection



Induced state-action distribution of π_b:

$$d^{\pi_b} \in \Delta(S \times A)$$

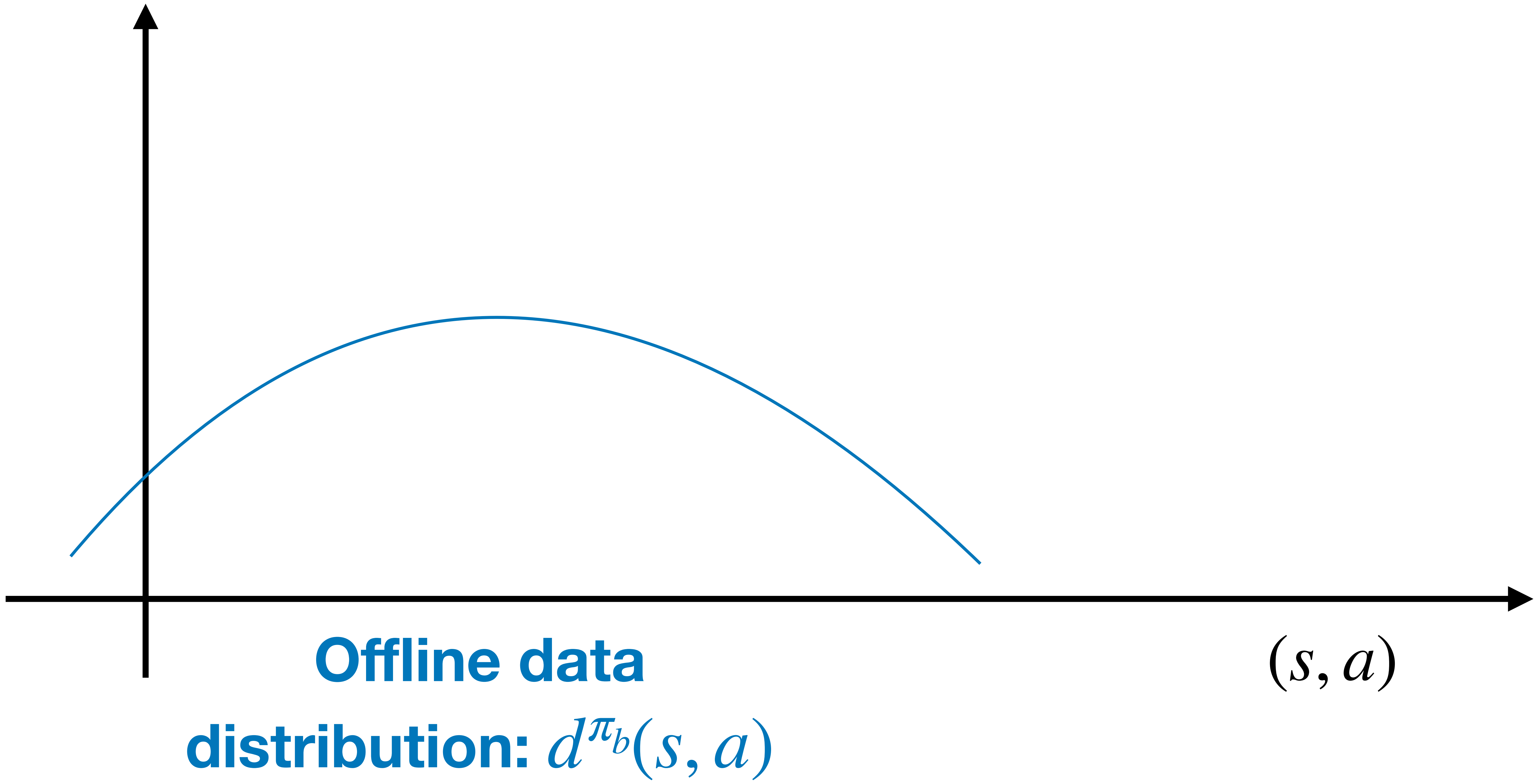
$$\mathcal{D} = \{s, a, s'\}, \text{ where } s, a \sim d^{\pi_b}, s' \sim P^\star(\cdot \mid s, a)$$

i.i.d triples

Offline Data Coverage

$$d^{\pi_b} \in \Delta(S \times A)$$

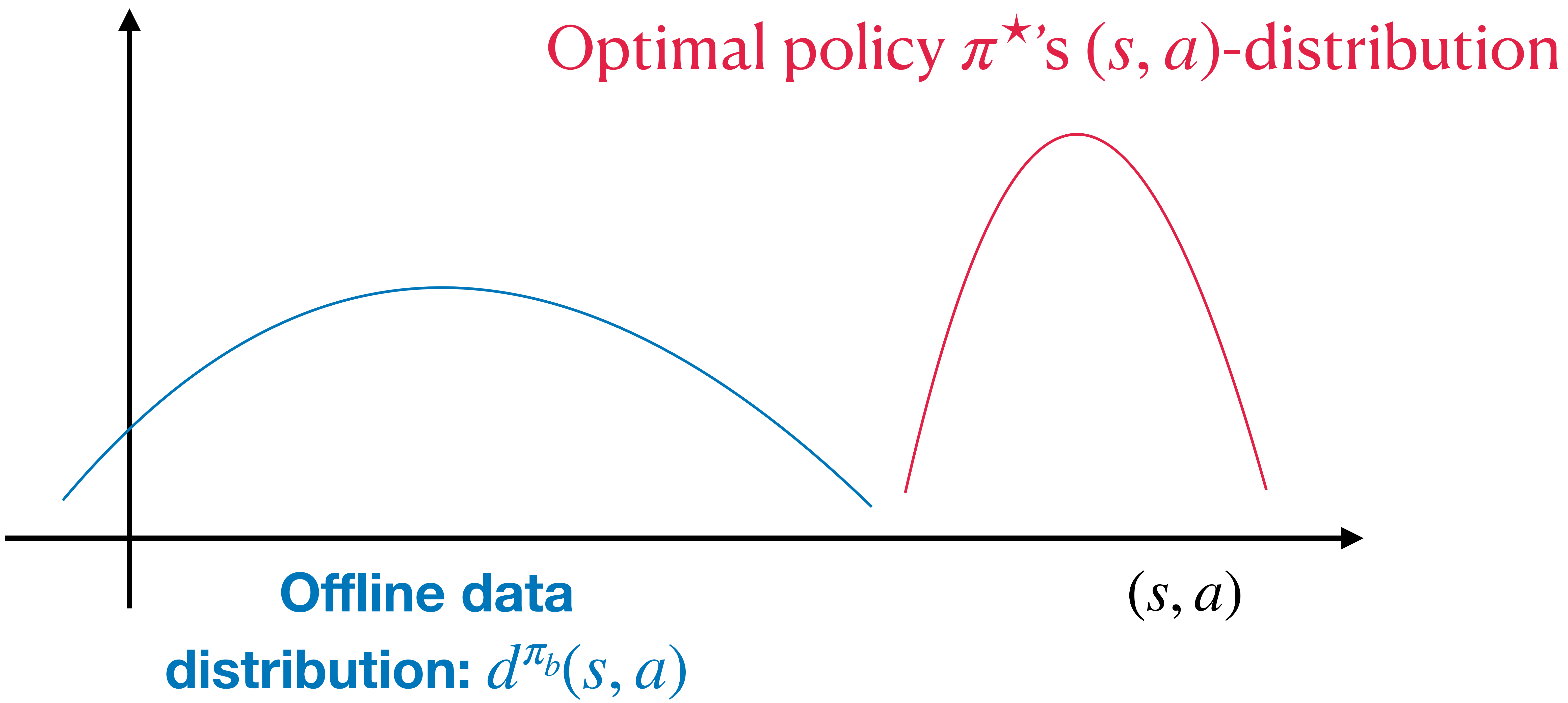
$$\mathcal{D} = \{s, a, s'\}, \text{ where } s, a \sim d^{\pi_b}, s' \sim P(\cdot \mid s, a)$$



Offline Data Coverage

$$d^{\pi_b} \in \Delta(S \times A)$$

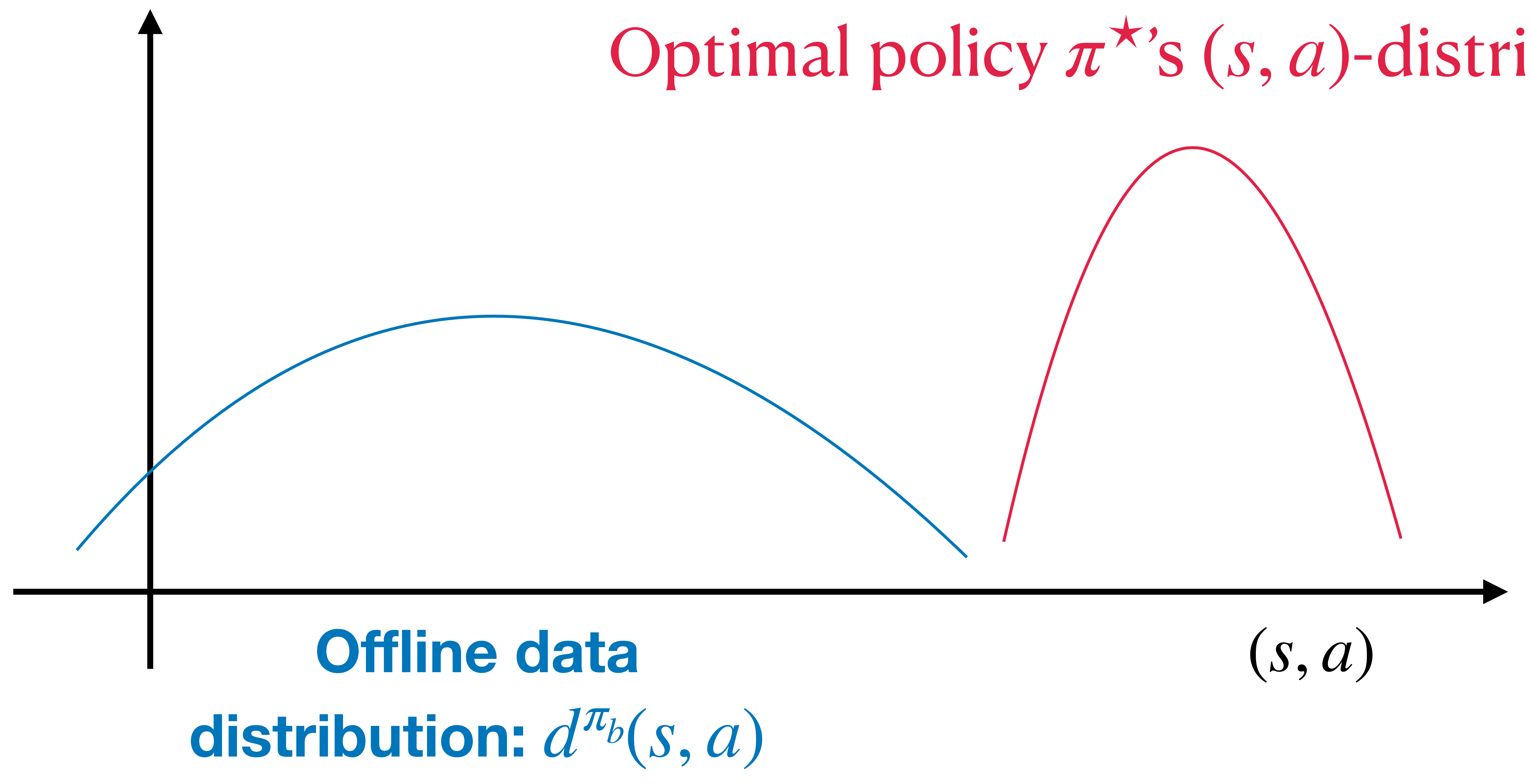
$$\mathscr{D} = \{s, a, s'\}, \text{ where } s, a \sim d^{\pi_b}, s' \sim P(\cdot \mid s, a)$$



Offline Data Coverage

$$d^{\pi_b} \in \Delta(S \times A)$$

$$\mathcal{D} = \{s, a, s'\}, \text{ where } s, a \sim d^{\pi_b}, s' \sim P(\cdot \mid s, a)$$

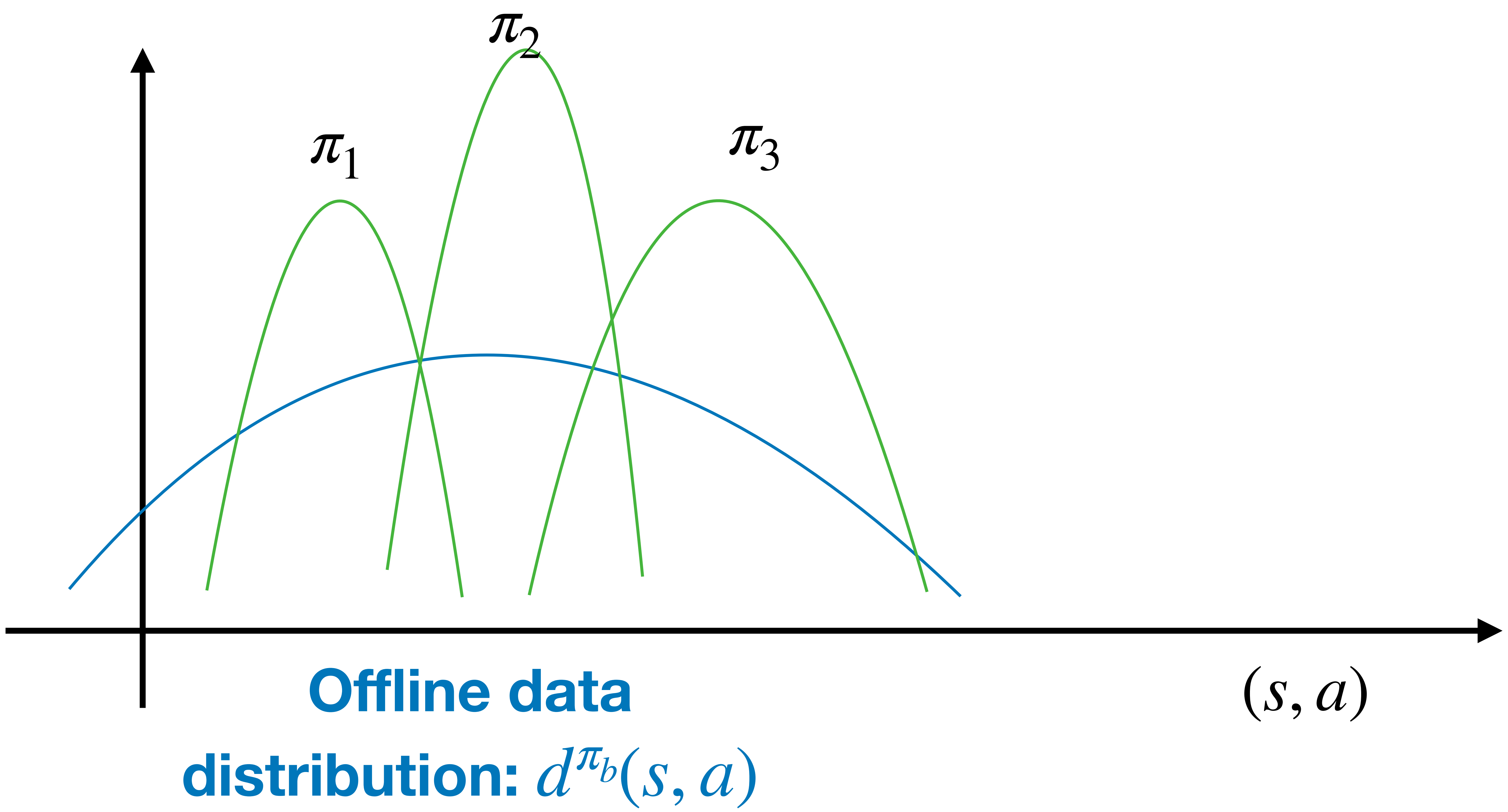


Finding $\pi^\star$ seems hopeless!

Offline Data Coverage

$$d^{\pi_b} \in \Delta(S \times A)$$

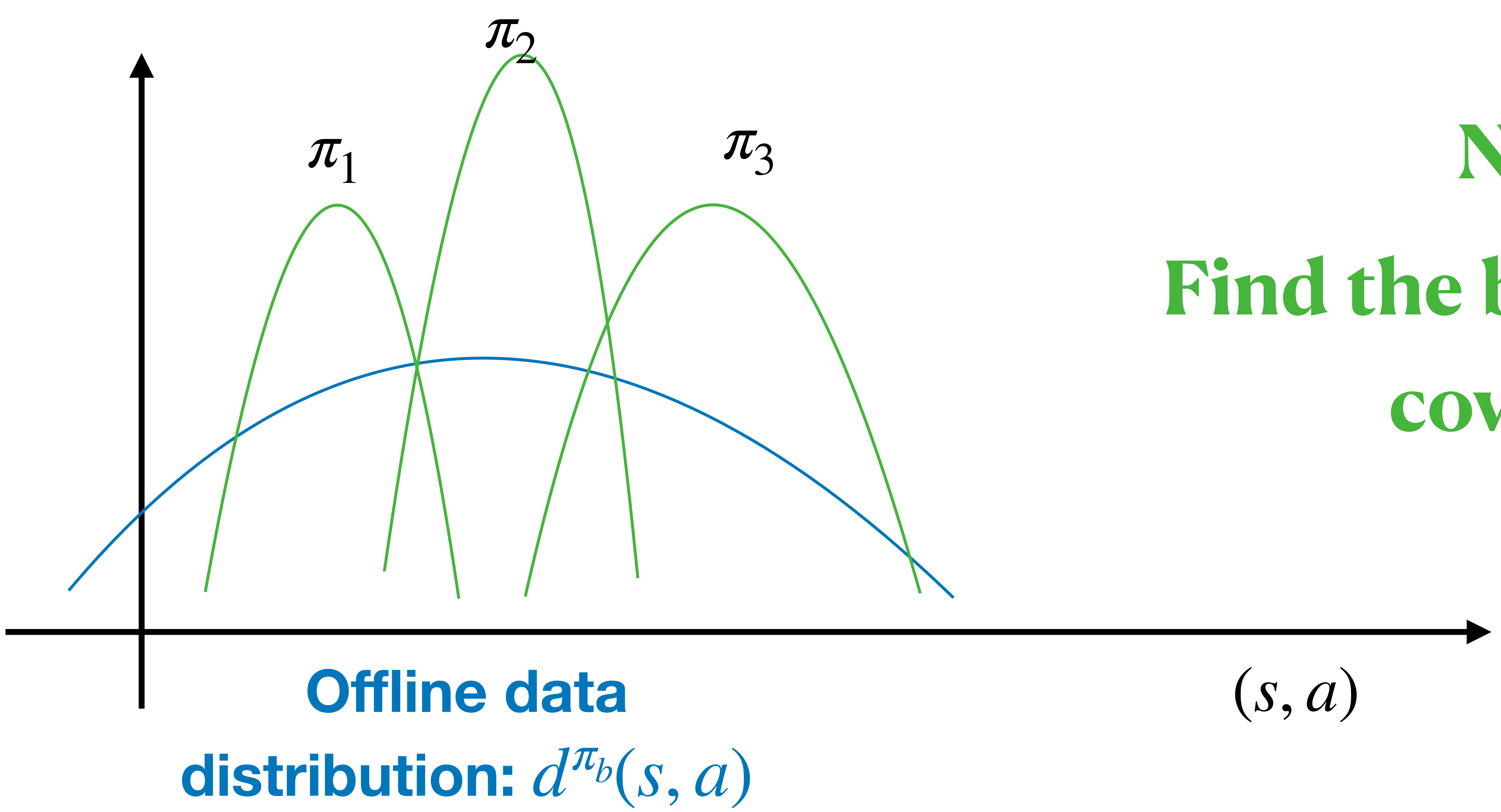
$$\mathcal{D} = \{s, a, s'\}, \text{ where } s, a \sim d^{\pi_b}, s' \sim P(\cdot \mid s, a)$$



Offline Data Coverage

$$d^{\pi_b} \in \Delta(S \times A)$$

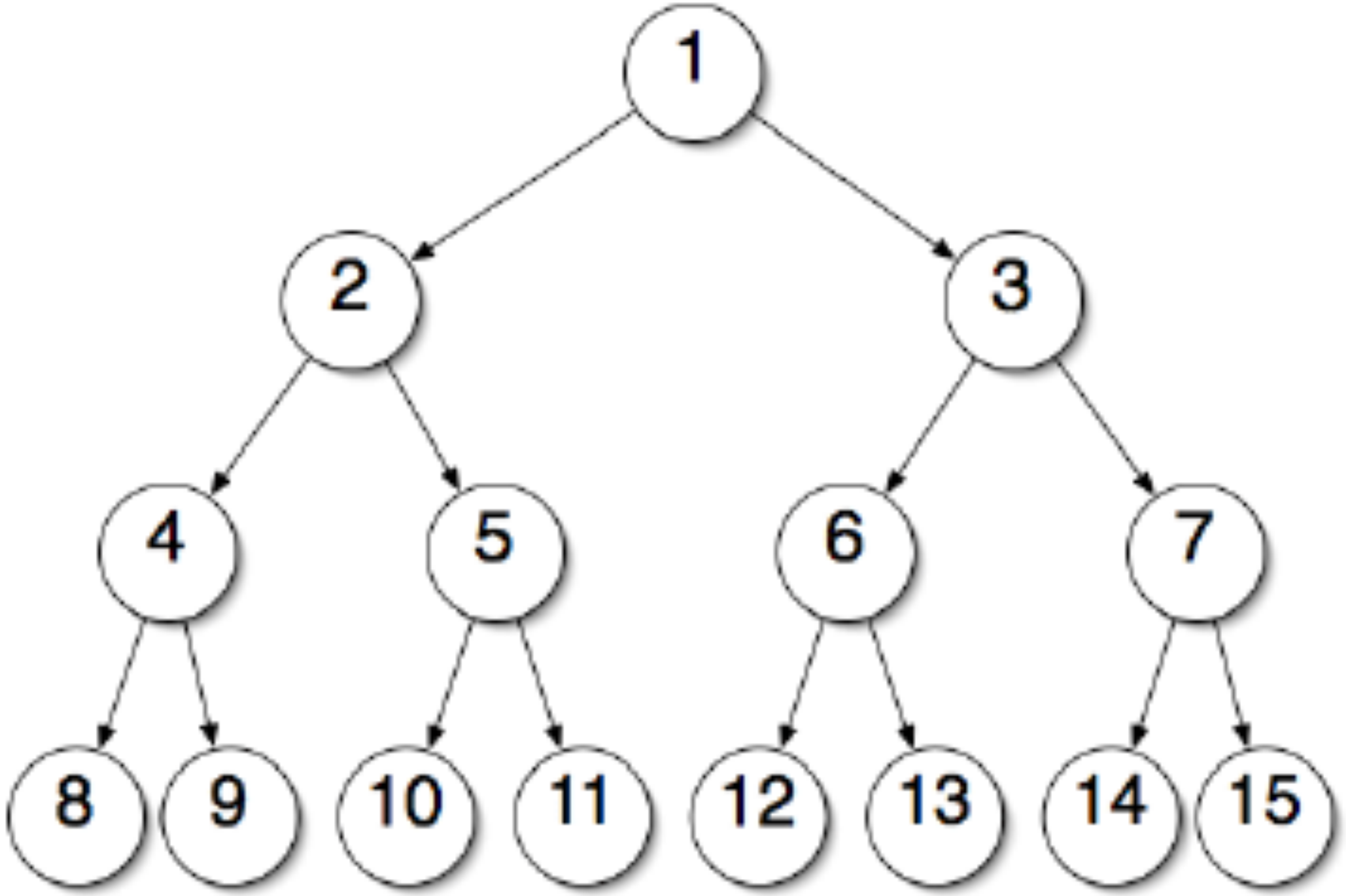
$$\mathcal{D} = \{s, a, s'\}, \text{ where } s, a \sim d^{\pi_b}, s' \sim P(\cdot \mid s, a)$$



New Goal:

Find the best among those covered by d^{π_b}

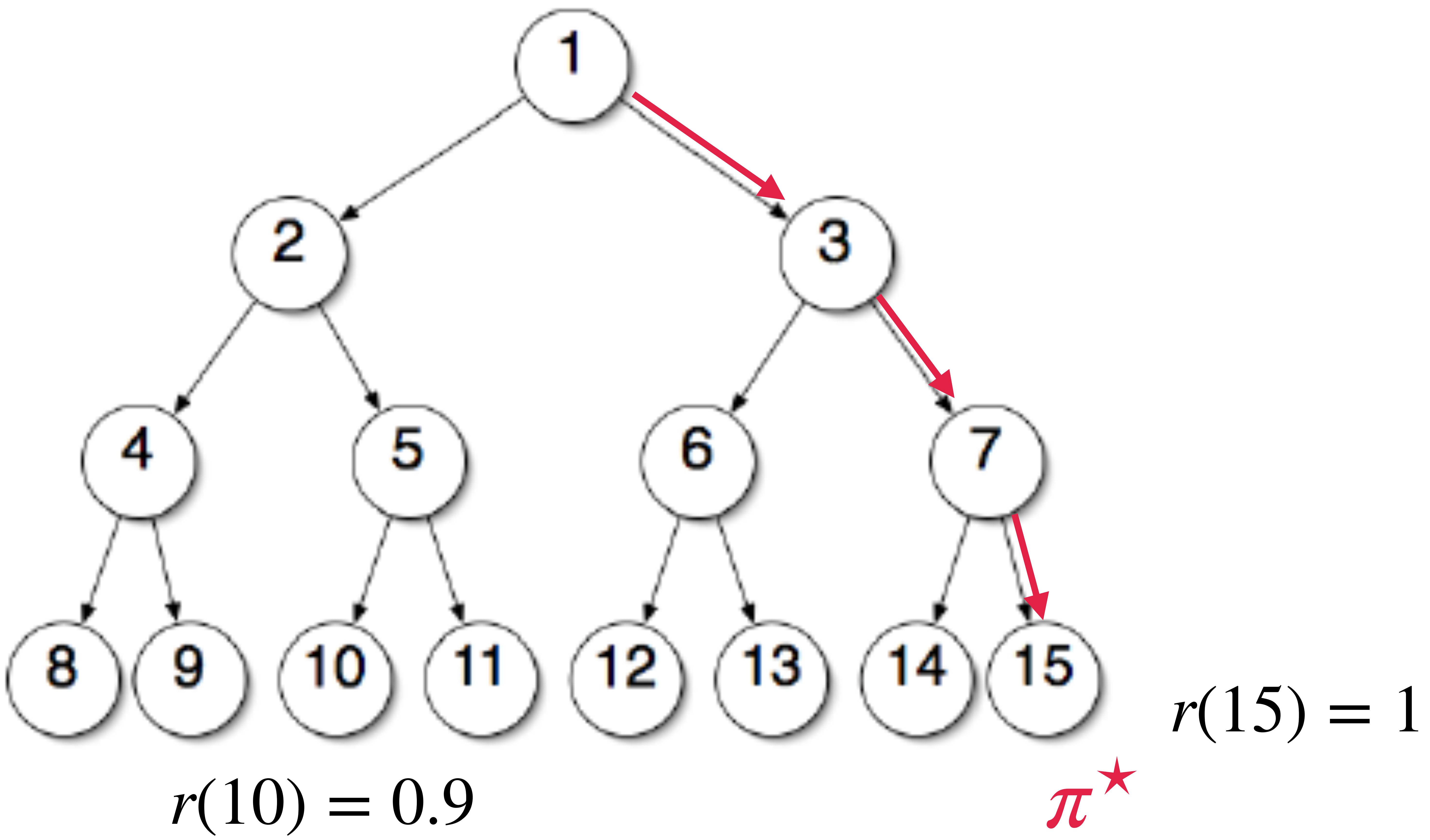
Learning goal in Offline RL: Robustness



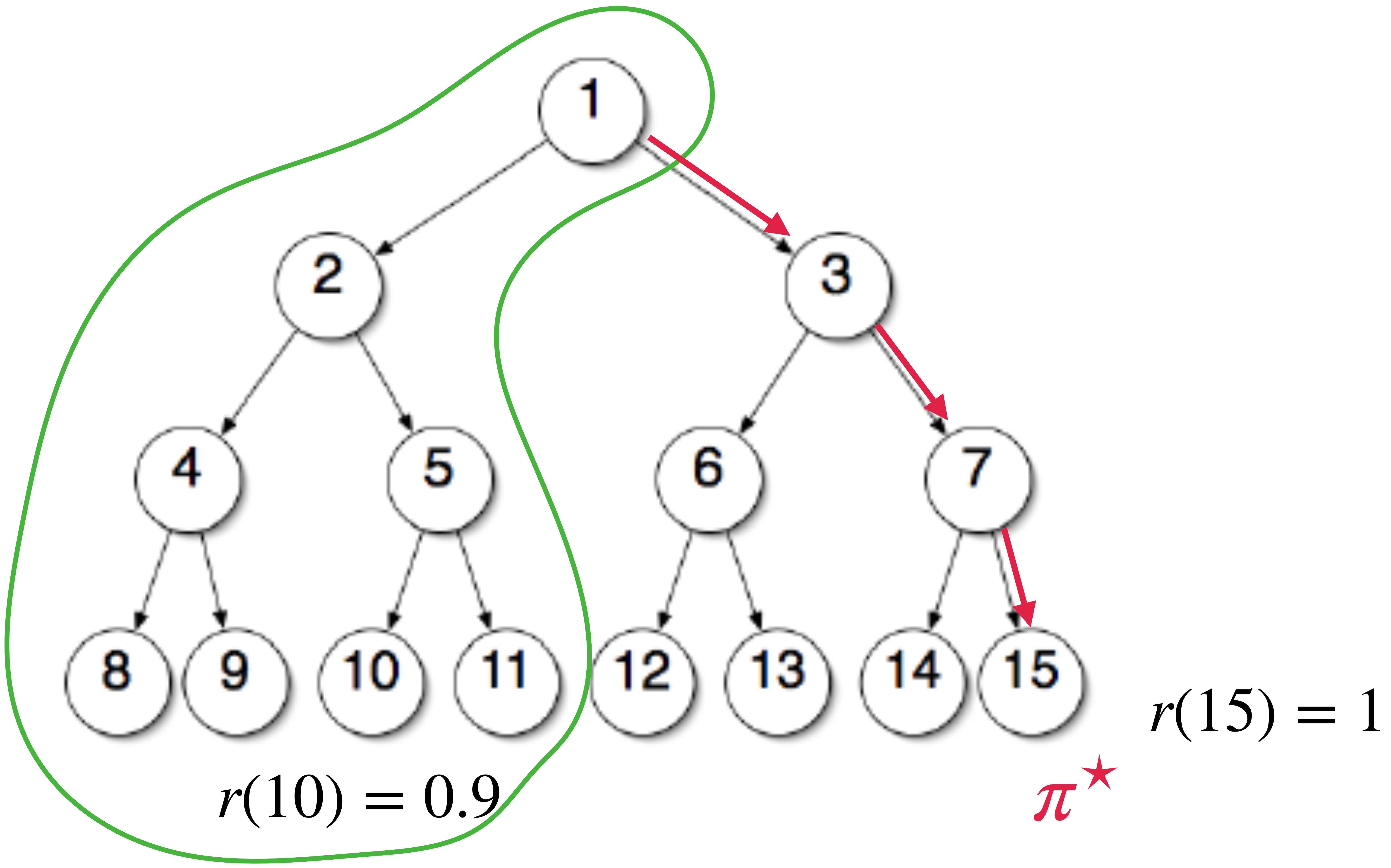
$r(15) = 1$

$r(10) = 0.9$

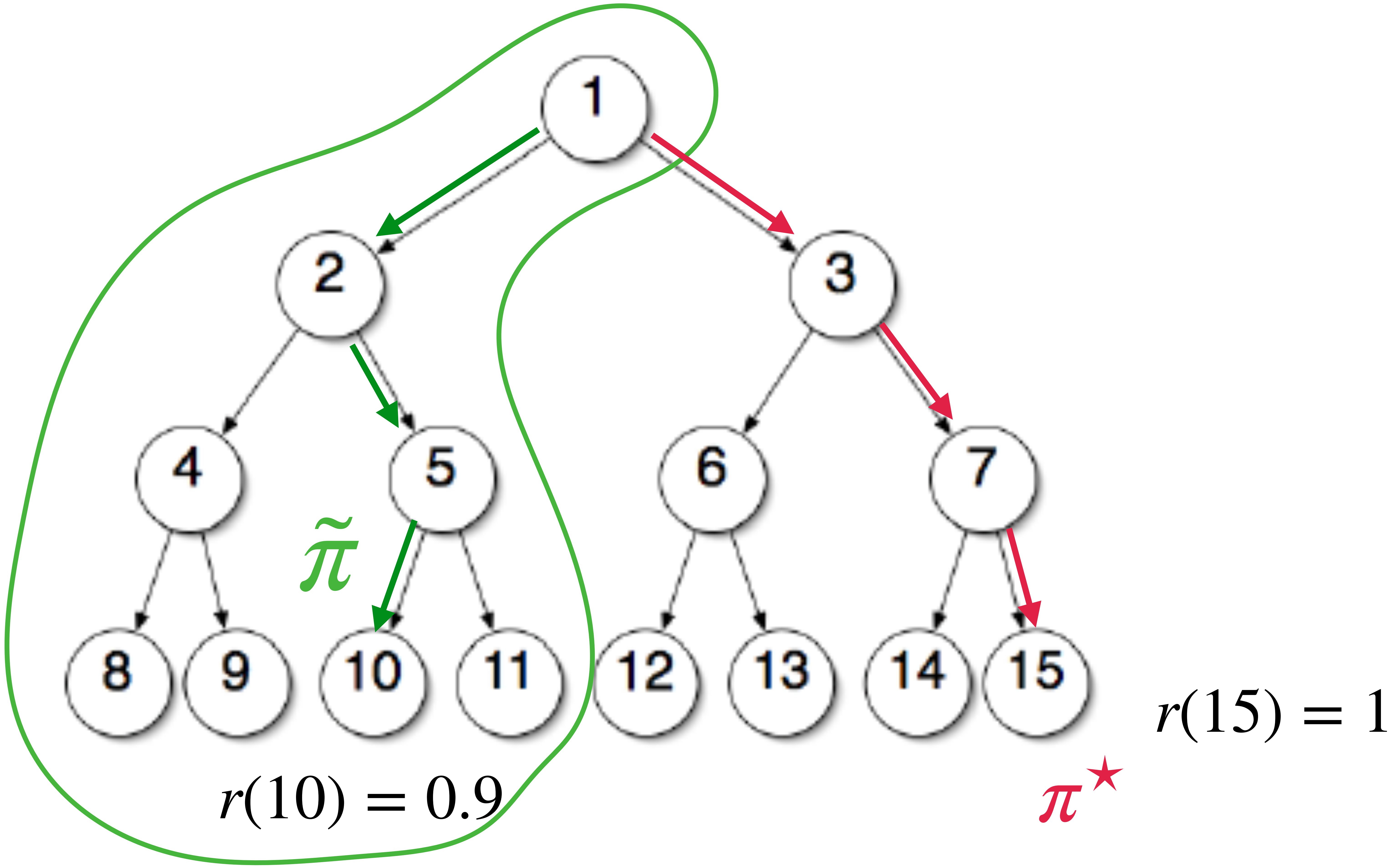
Learning goal in Offline RL: Robustness



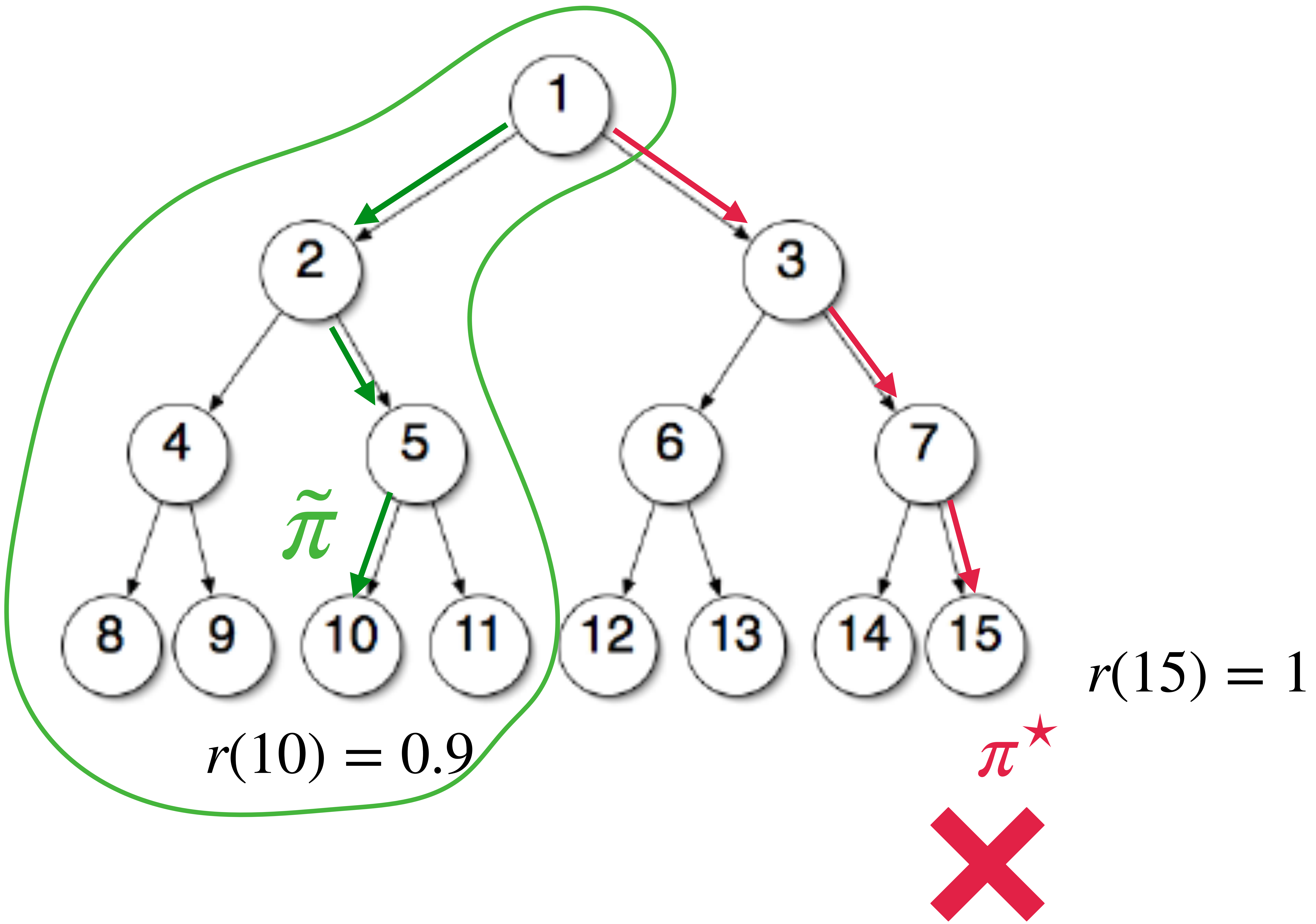
Learning goal in Offline RL: Robustness



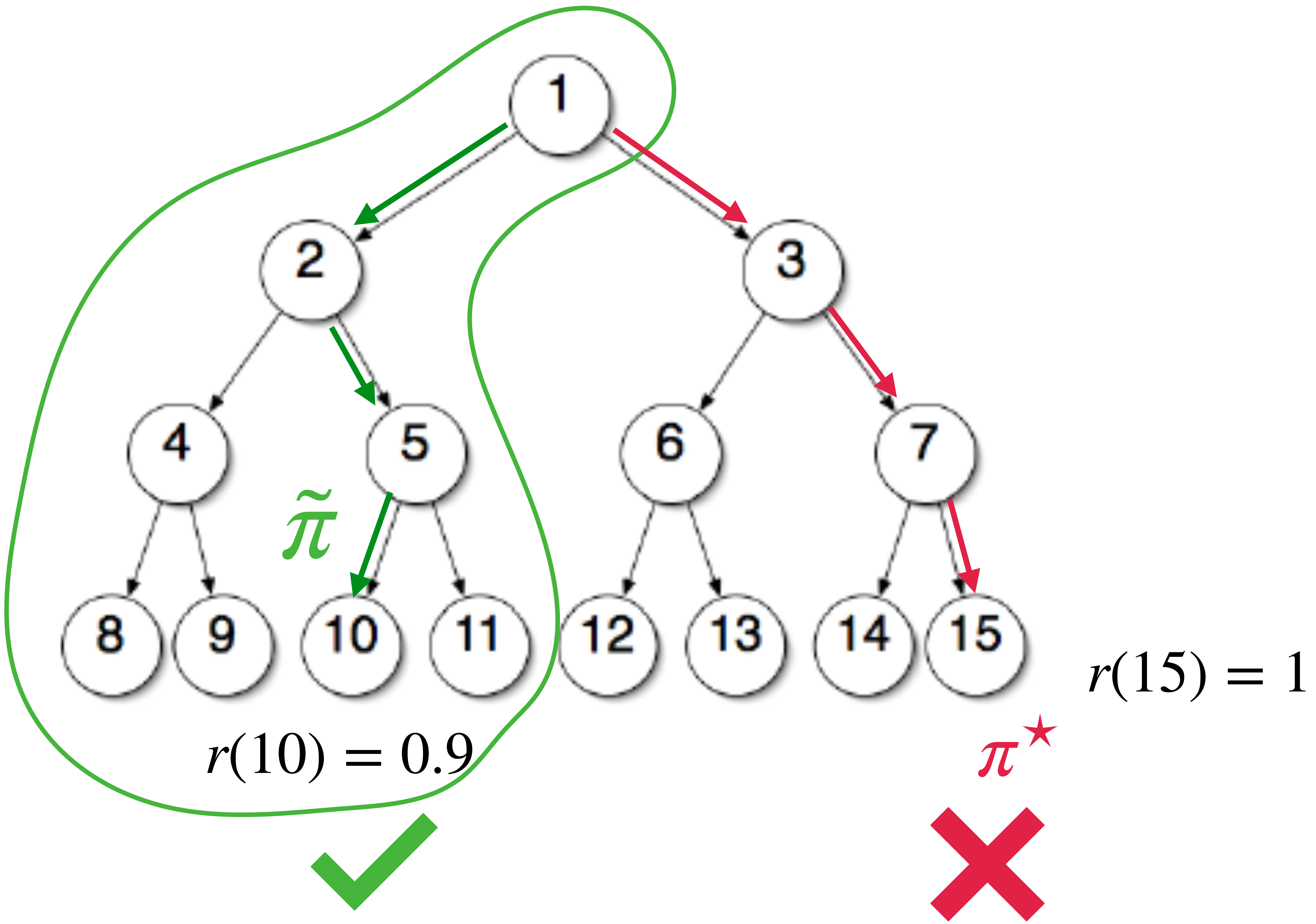
Learning goal in Offline RL: Robustness



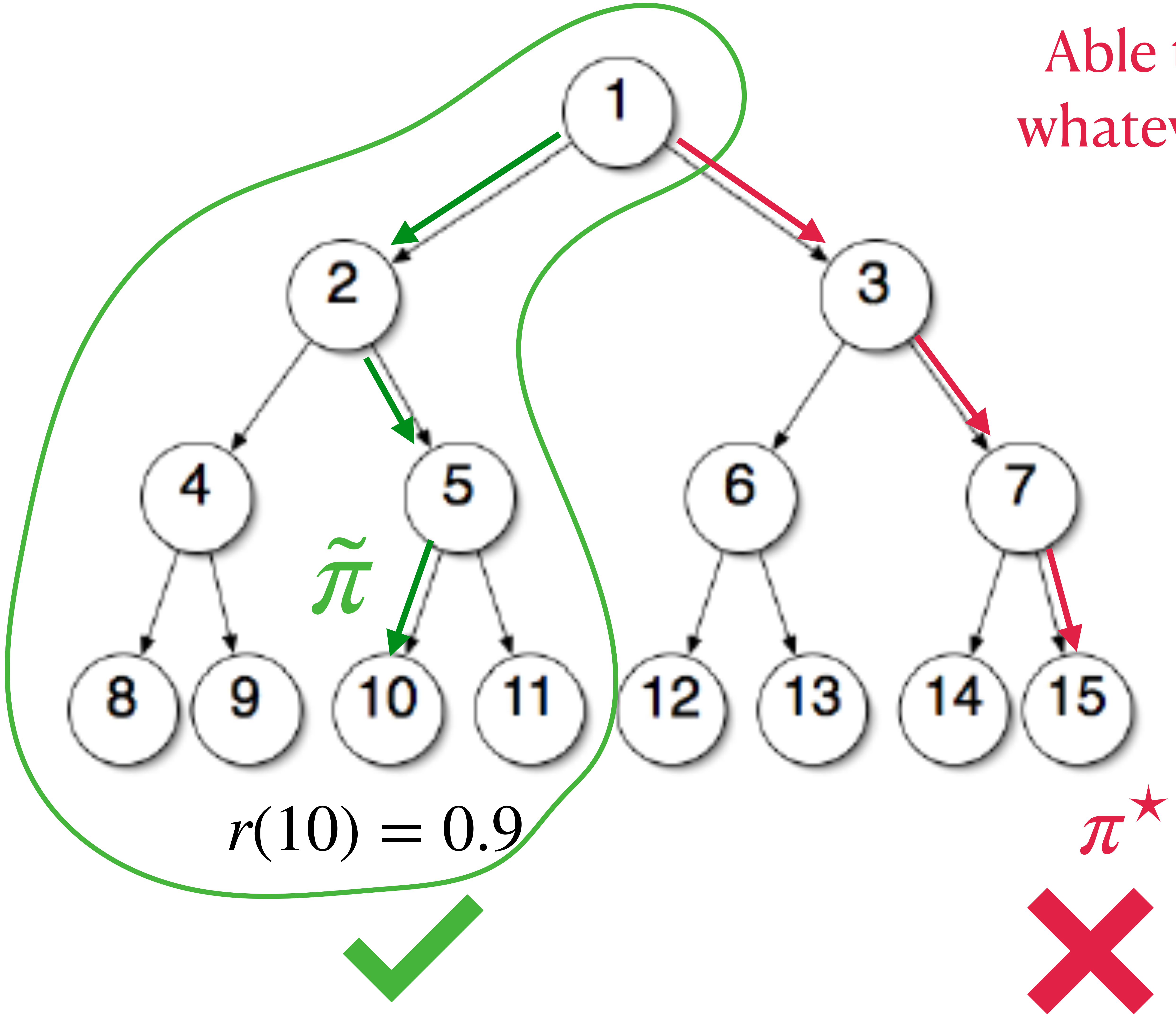
Learning goal in Offline RL: Robustness



Learning goal in Offline RL: Robustness



Learning goal in Offline RL: Robustness



Able to compete against whatever is covered by the offline data

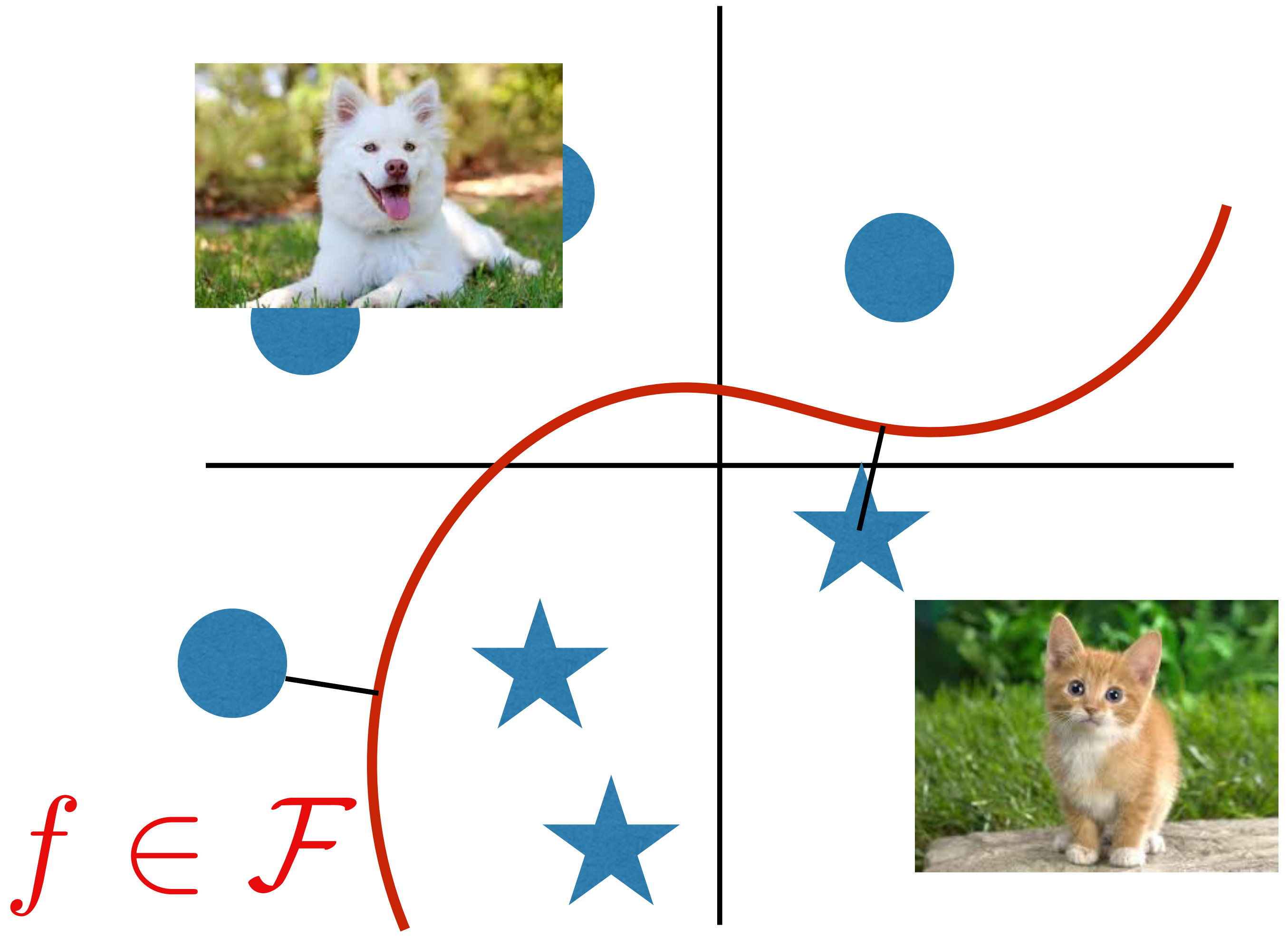
Learning goal in Offline RL: Generalization

Supervised Learning:

Offline RL:

Learning goal in Offline RL: Generalization

Supervised Learning:

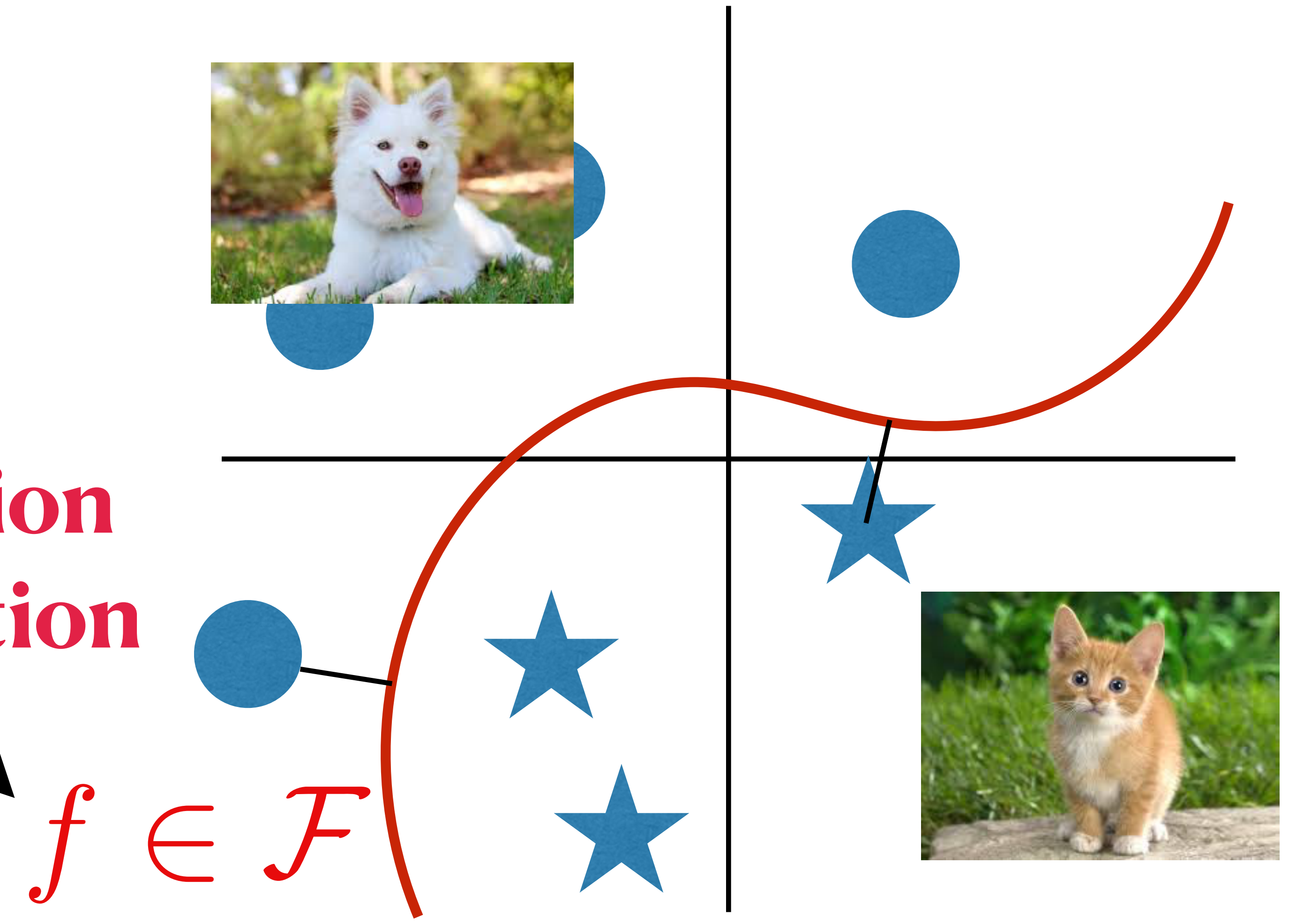


Offline RL:

Learning goal in Offline RL: Generalization

Supervised Learning:

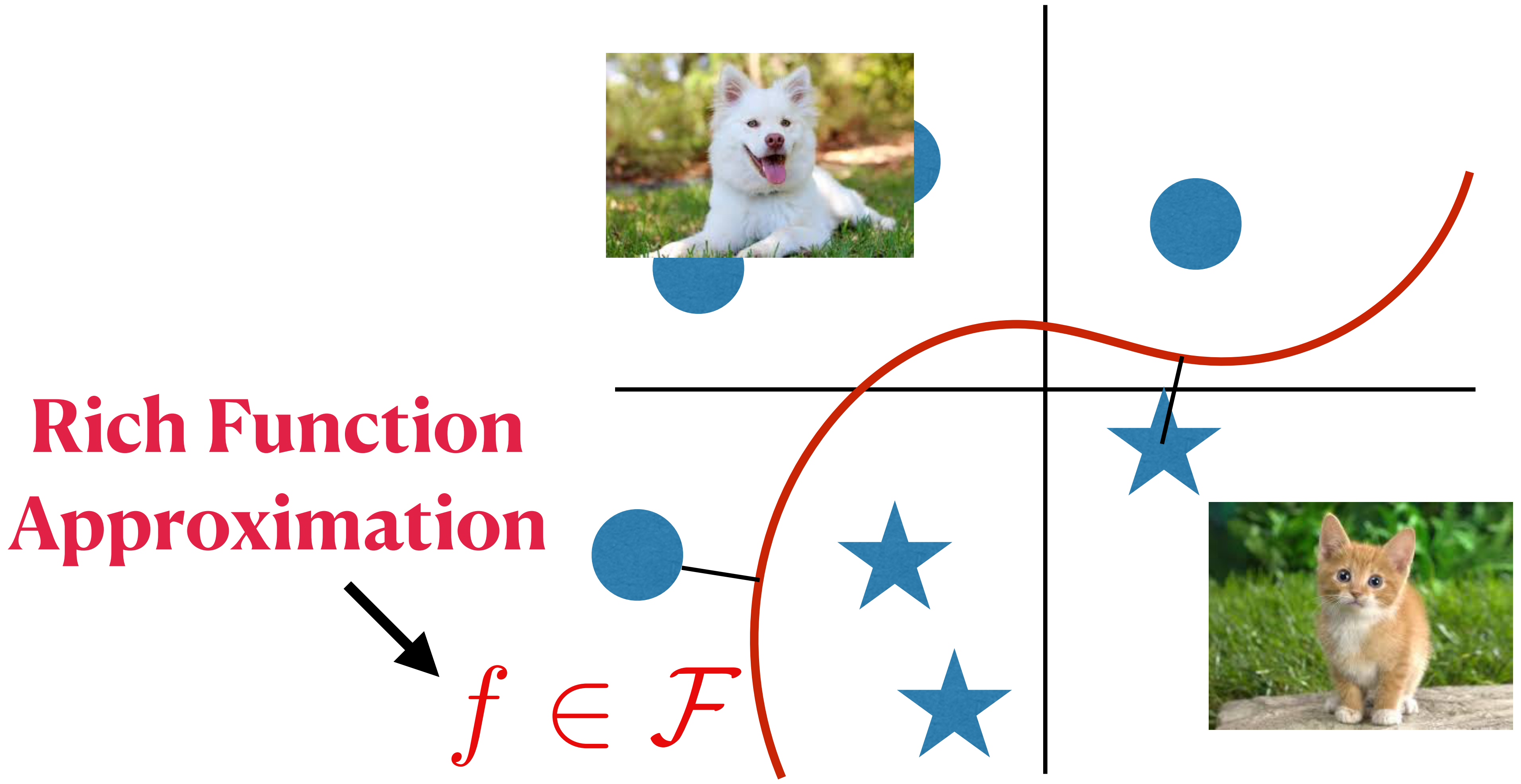
Rich Function Approximation



Offline RL:

Learning goal in Offline RL: Generalization

Supervised Learning:

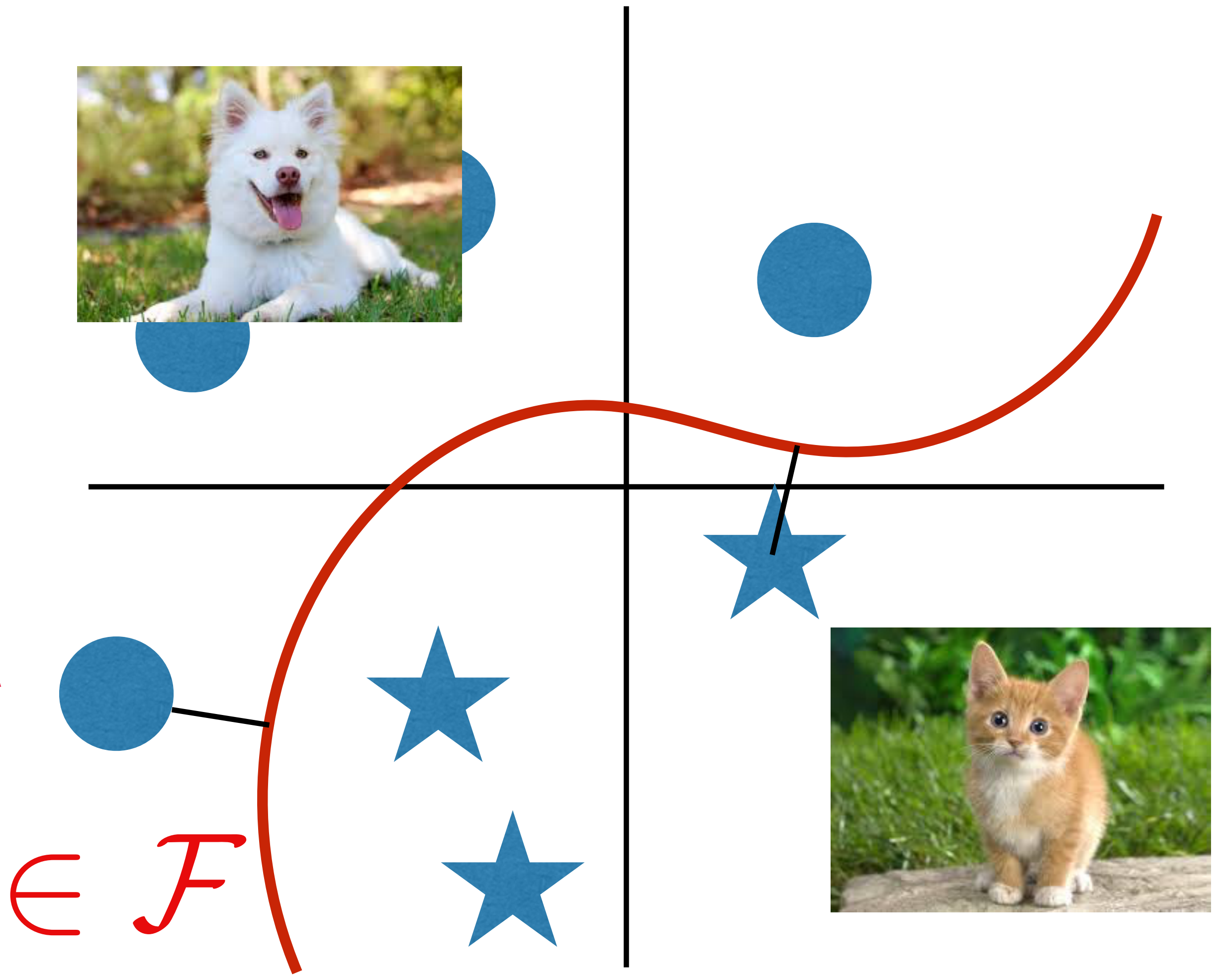


Sample complexity depends on **complexity of $\mathscr{F}$** (e.g., VC-dim, Rademacher, covering dim)

Offline RL:

Learning goal in Offline RL: Generalization

Supervised Learning:



Rich Function Approximation

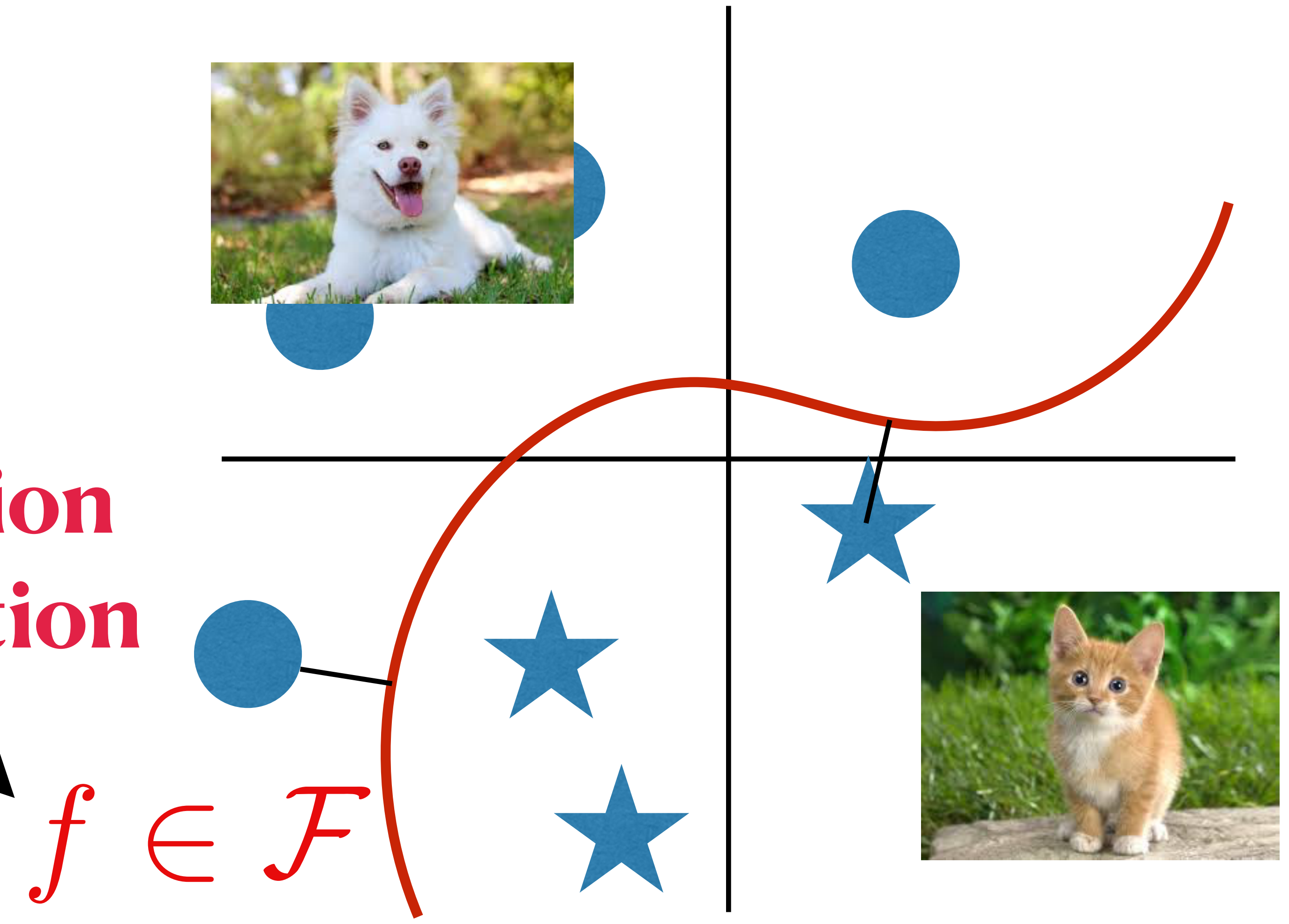
$f \in \mathcal{F}$

Offline RL:

Learning goal in Offline RL: Generalization

Supervised Learning:

Rich Function Approximation



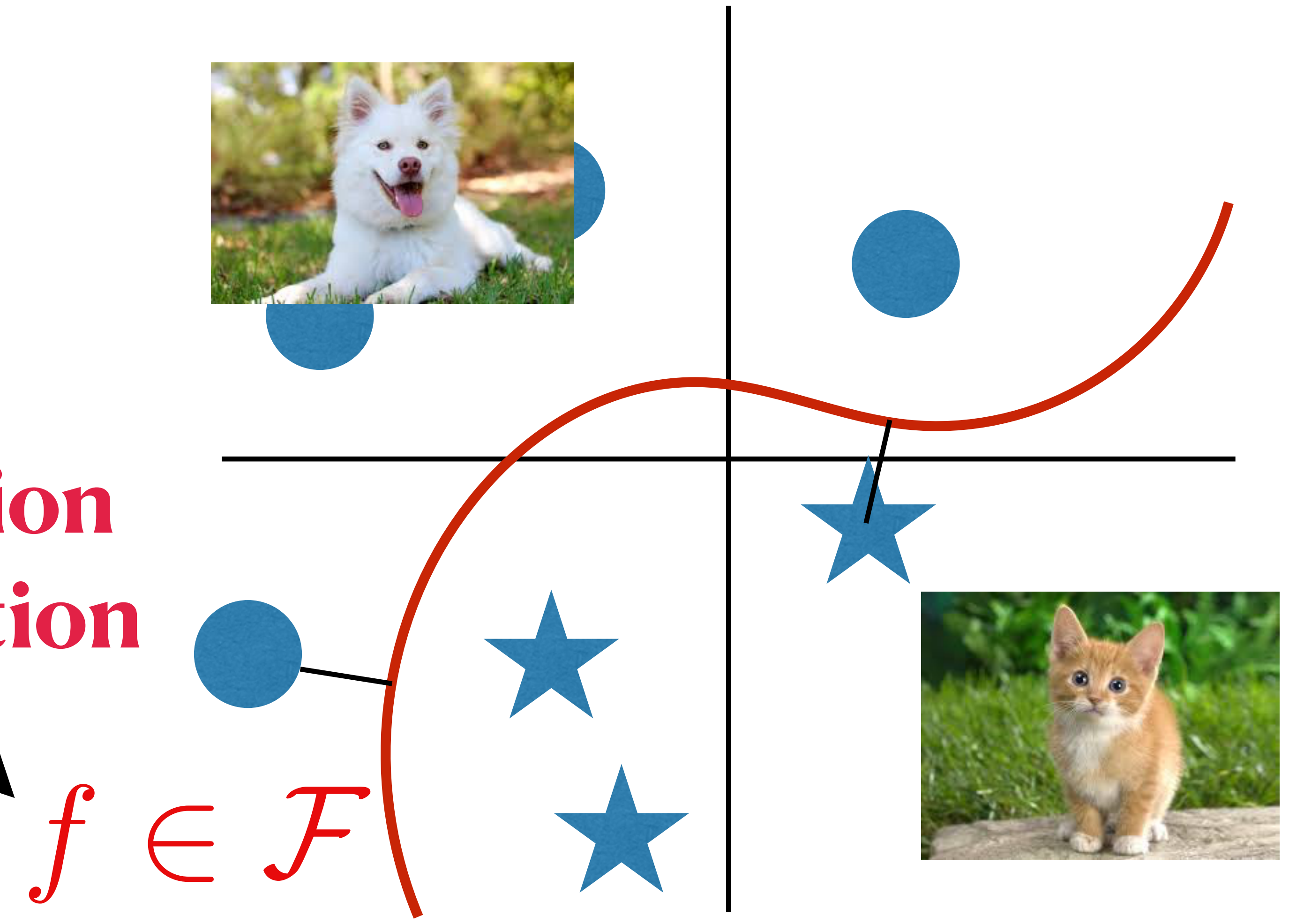
Polynomial Dependency of #
of unique images

Offline RL:

Learning goal in Offline RL: Generalization

Supervised Learning:

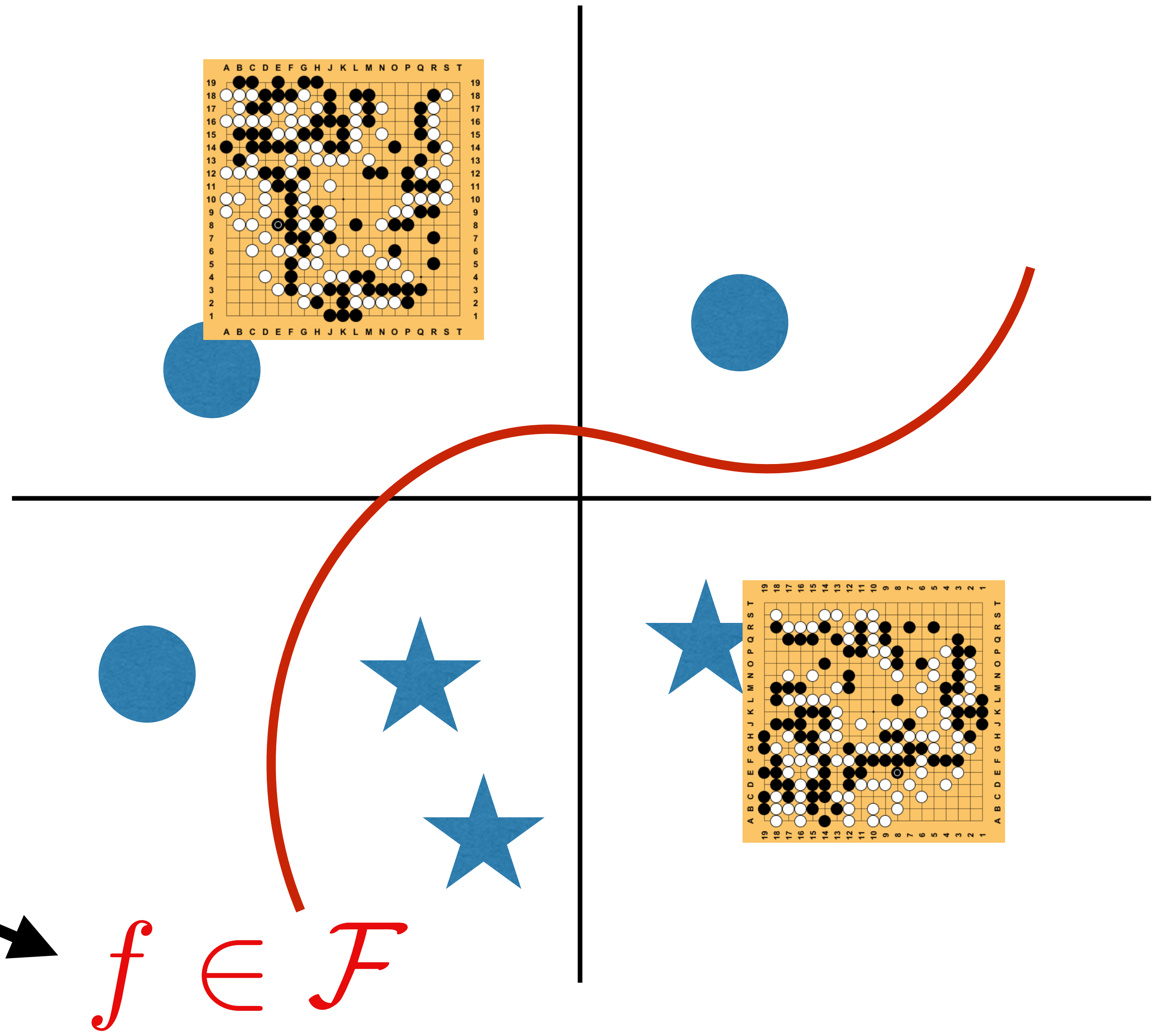
Rich Function Approximation



Polynomial Dependency of #
of unique images

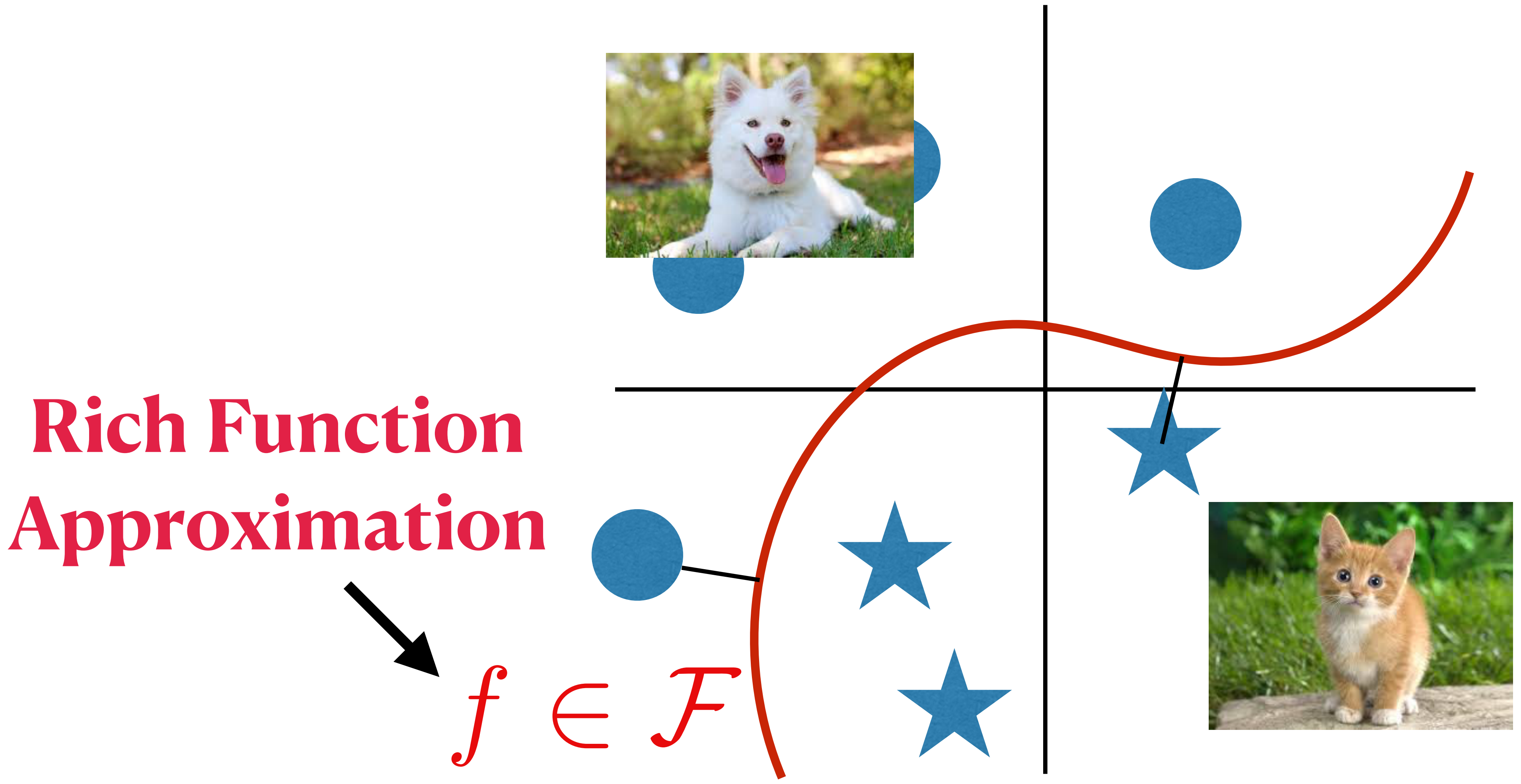
Offline RL:

Rich Function Approximation



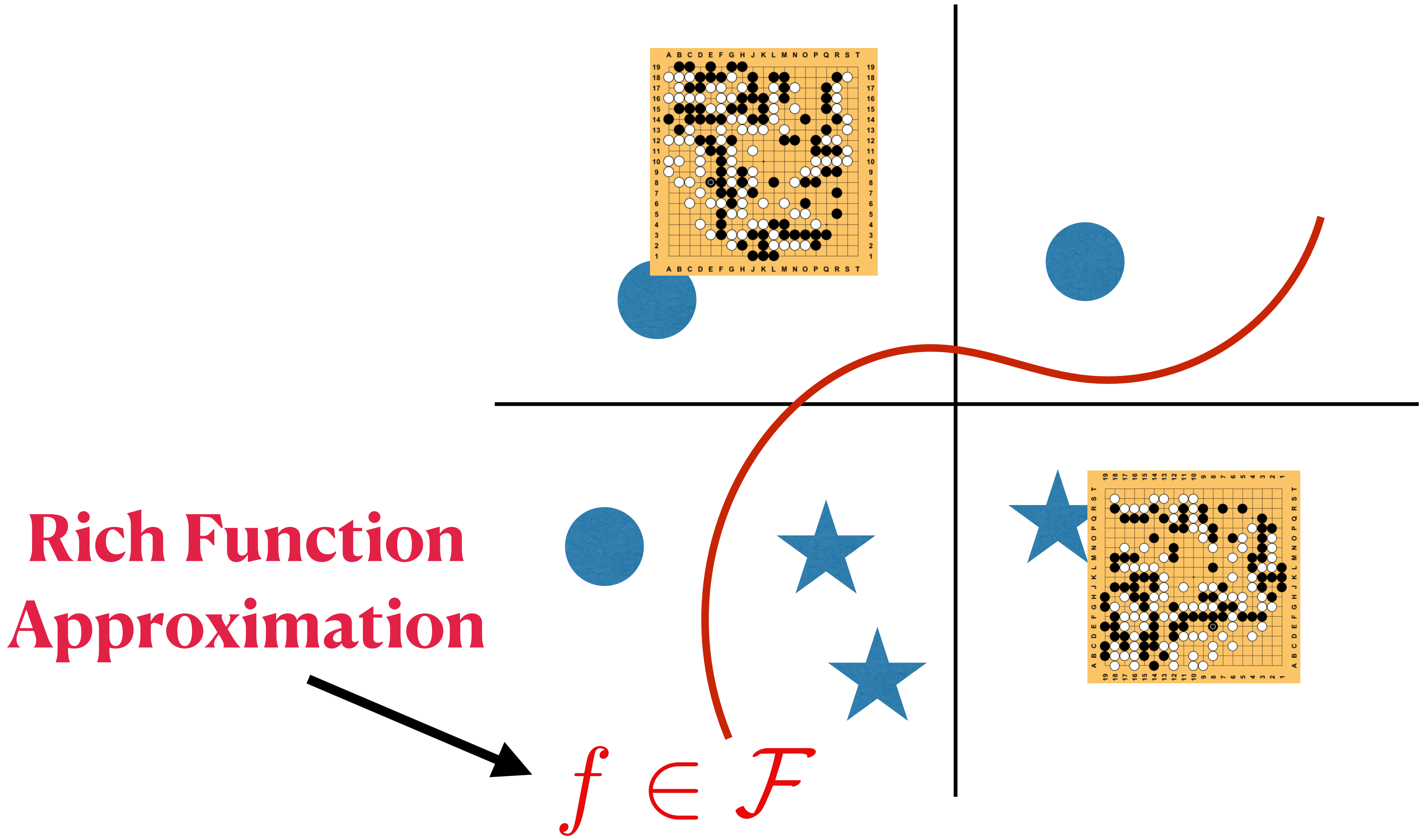
Learning goal in Offline RL: Generalization

Supervised Learning:



Polynomial Dependency of # **of unique images**

Offline RL:



Identify a high quality policy w/ # of offline samples scaling wrt complexity of $\mathscr{F}$

Learning goal in Offline RL: Robustness & Generalization

Can we

(a) compete against the best policy among those covered by d^{π_b},

(b) w/ # of offline samples scaling polynomially wrt the complexity of $\mathscr{F}$?

Outline:

1. Rethinking the goal in offline RL: Robustness + Generalization

2. Can we achieve the goal ?

A Naive Model-based Approach

Certainty Equivalence:

A Naive Model-based Approach

Certainty Equivalence:

1. Fit model by MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

A Naive Model-based Approach

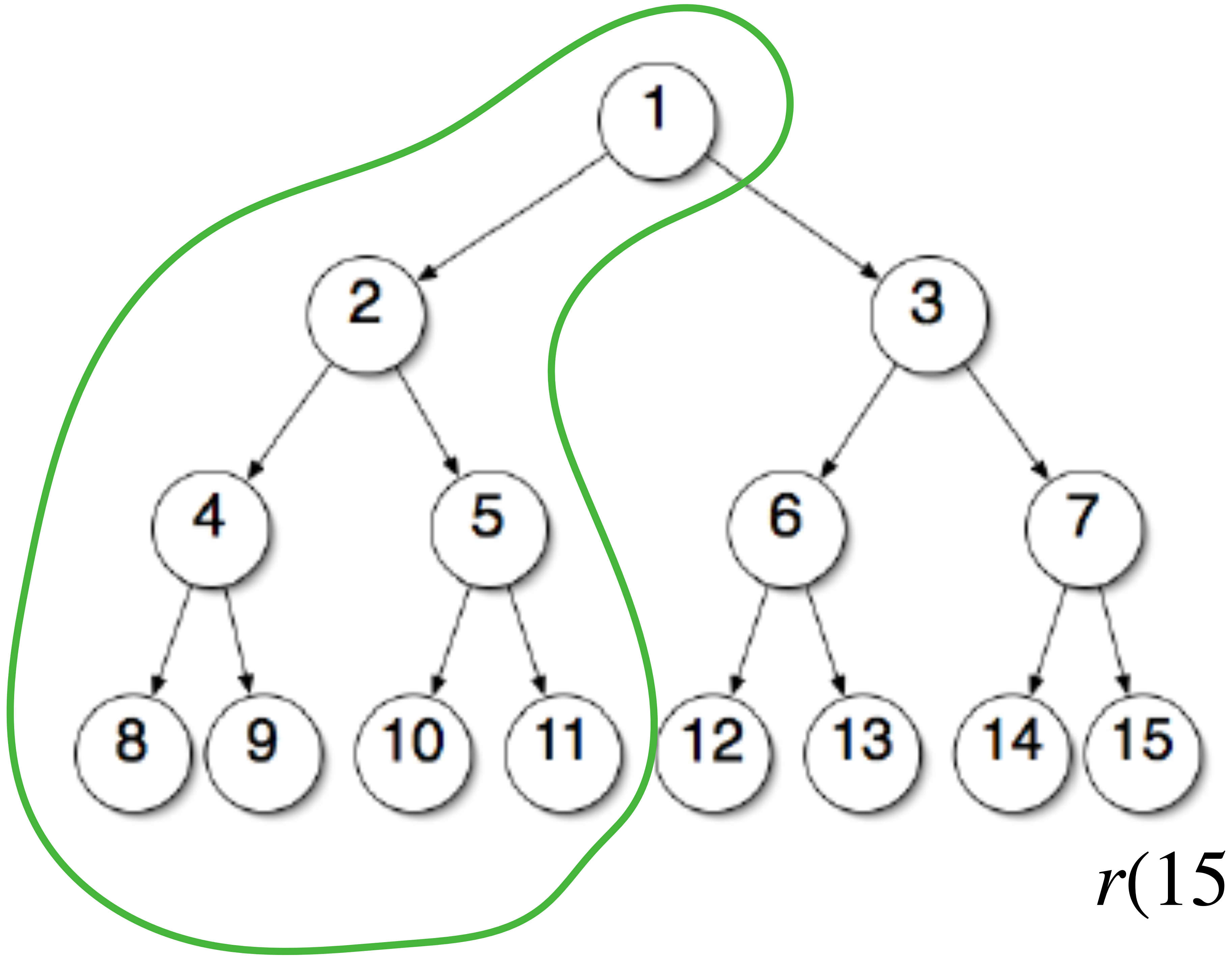
Certainty Equivalence:

1. Fit model by MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$
2. Plan inside $\hat{P}$: $\hat{\pi} = \mathrm{OP}(\hat{P}, r)$

A Naive Model-based Approach

Certainty Equivalence:

1. Fit model by MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$
2. Plan inside $\hat{P}$: $\hat{\pi} = \mathrm{OP}(\hat{P}, r)$

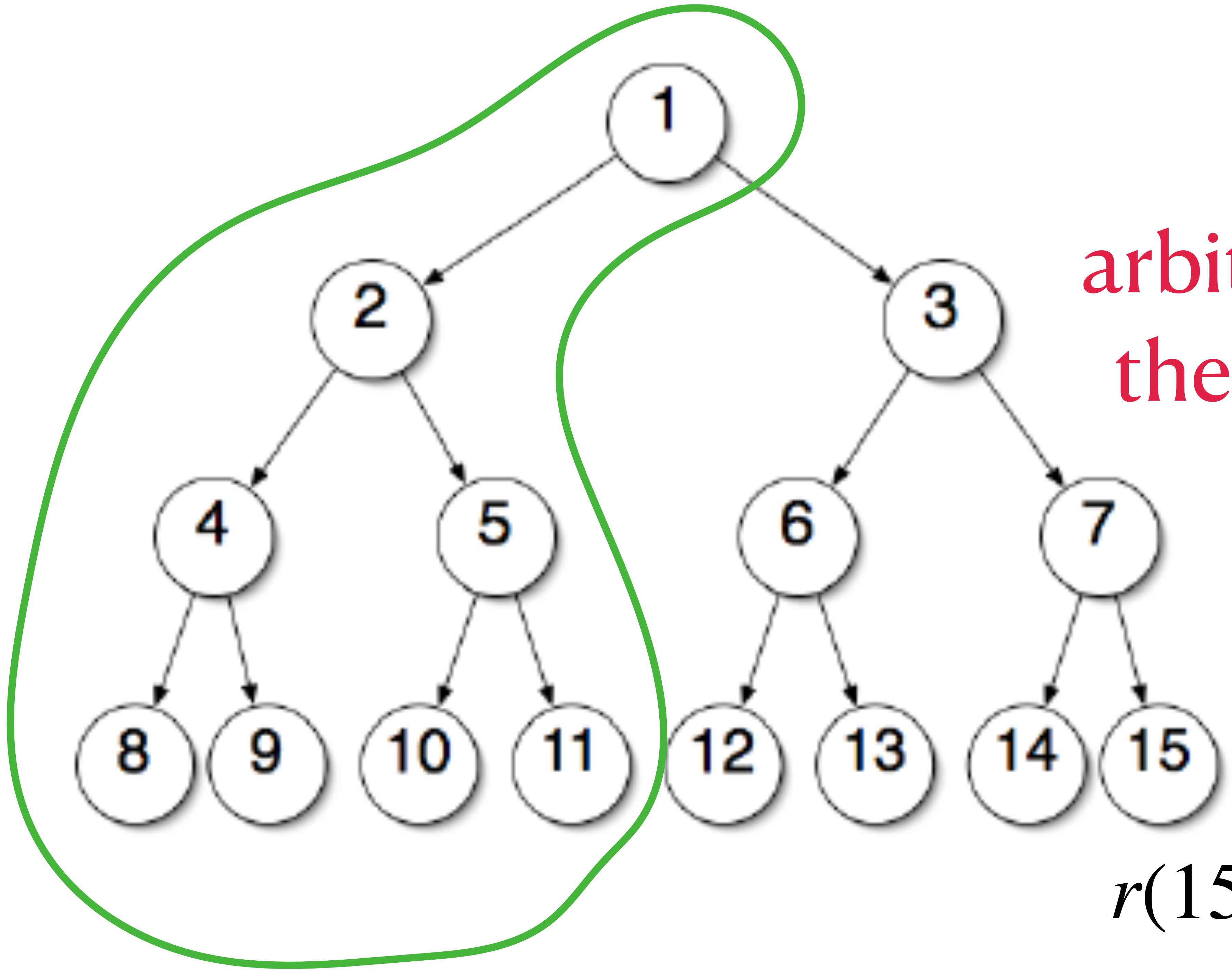


$r(15) = 1$

A Naive Model-based Approach

Certainty Equivalence:

1. Fit model by MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$
2. Plan inside $\hat{P}$: $\hat{\pi} = \mathrm{OP}(\hat{P}, r)$



$\hat{P}$ could be arbitrarily wrong in the right sub-tree

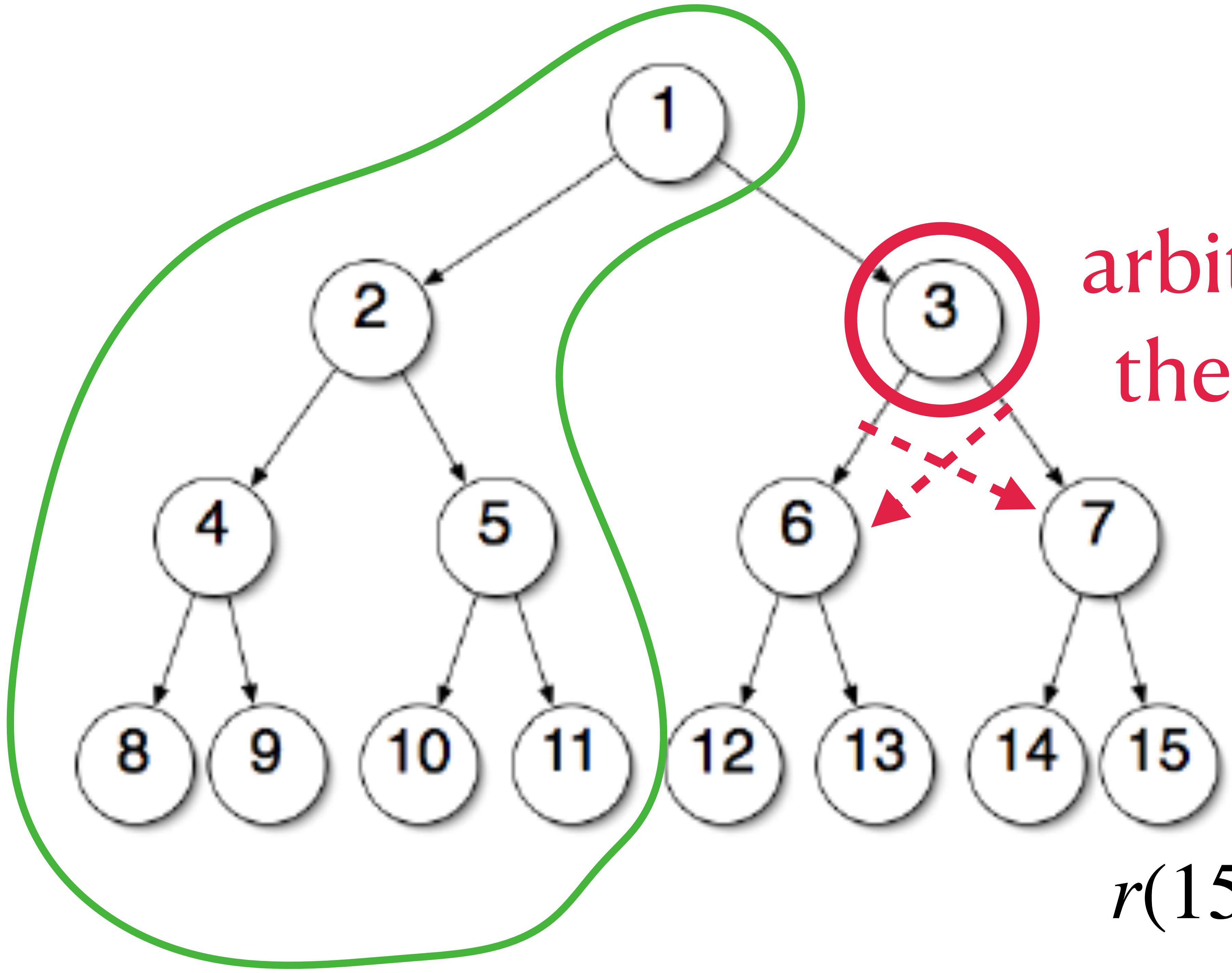
$r(15) = 1$

A Naive Model-based Approach

Certainty Equivalence:

1. Fit model by MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

2. Plan inside $\hat{P}$: $\hat{\pi} = \mathrm{OP}(\hat{P}, r)$



$\hat{P}$ could be arbitrarily wrong in the right sub-tree

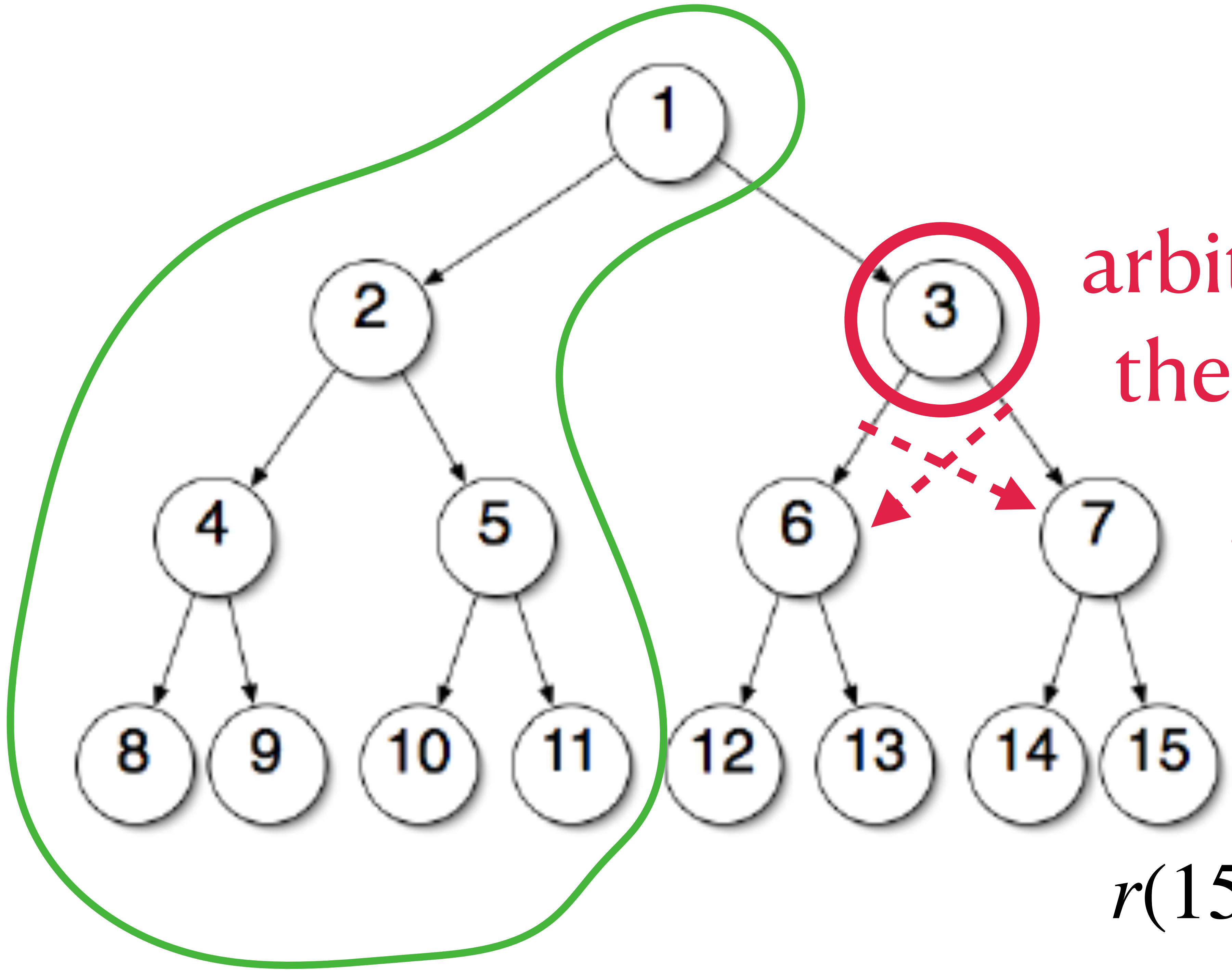
$r(15) = 1$

A Naive Model-based Approach

Certainty Equivalence:

1. Fit model by MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

2. Plan inside $\hat{P}$: $\hat{\pi} = \mathrm{OP}(\hat{P}, r)$



$\hat{P}$ could be arbitrarily wrong in the right sub-tree

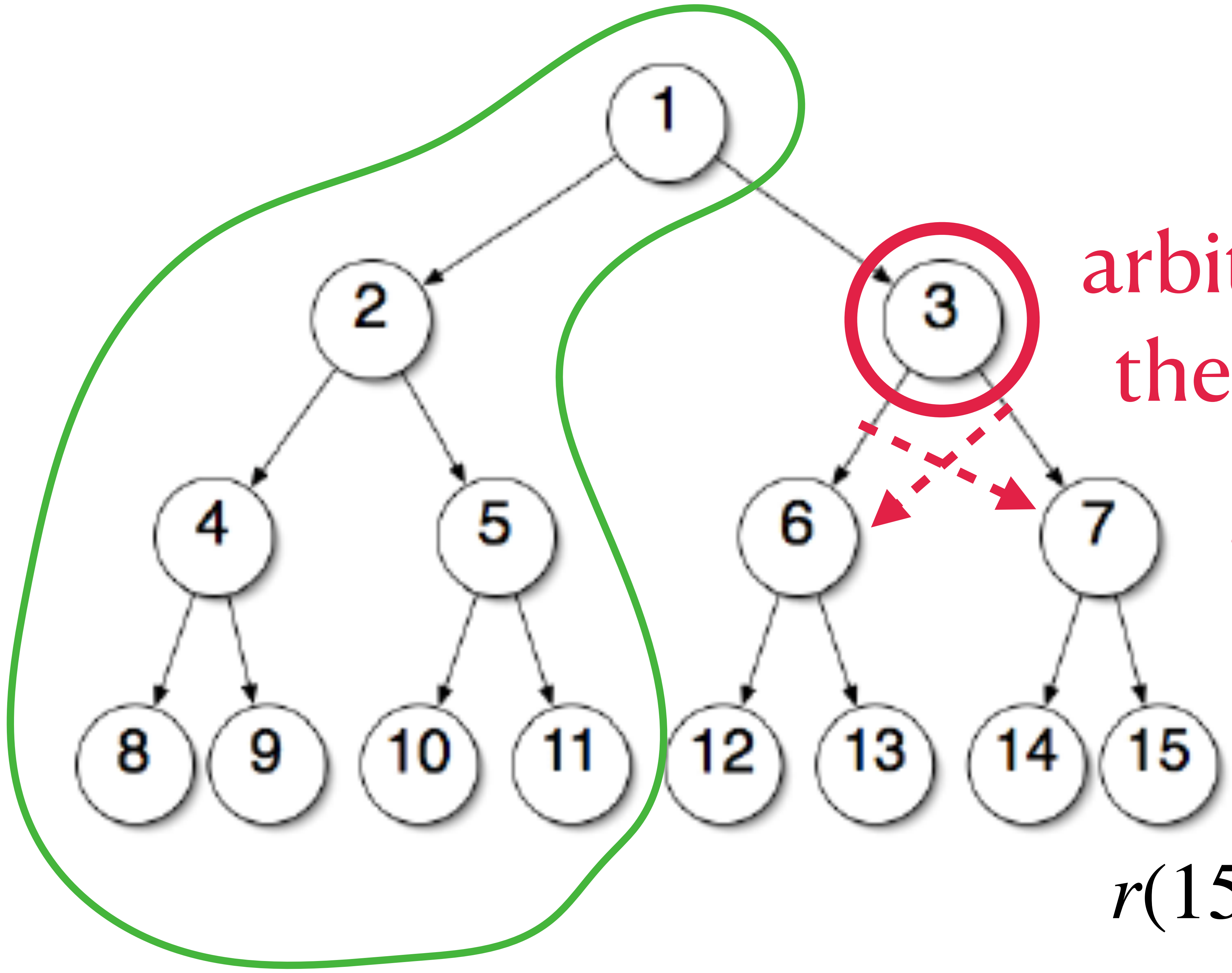
$\hat{\pi} = \{\text{right, left, right}\}$

$r(15) = 1$

A Naive Model-based Approach

Certainty Equivalence:

1. Fit model by MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$
2. Plan inside $\hat{P}$: $\hat{\pi} = \mathrm{OP}(\hat{P}, r)$



$r(15) = 1$

$\hat{P}$ could be arbitrarily wrong in the right sub-tree

$\hat{\pi} = \{\text{right, left, right}\}$

In real $P^\star$, $\hat{\pi}$ not only miss $r(15)$, also miss **good policies inside the green!**

Constrained Pessimistic Policy Optimization (CPPO)

1. MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

2. Constrained Pessimistic Policy Optimization

Constrained Pessimistic Policy Optimization (CPPO)

1. MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

2. Constrained Pessimistic Policy Optimization

$$\max_{\pi} \min_{P \in \mathscr{P}} J(\pi; P)$$

$$\text{s.t.}, \frac{1}{|\mathscr{D}|} \sum_{s,a \in \mathscr{D}} \left\| P(\cdot \mid s, a) - \hat{P}(\cdot \mid s, a) \right\|_1 \leq \delta$$

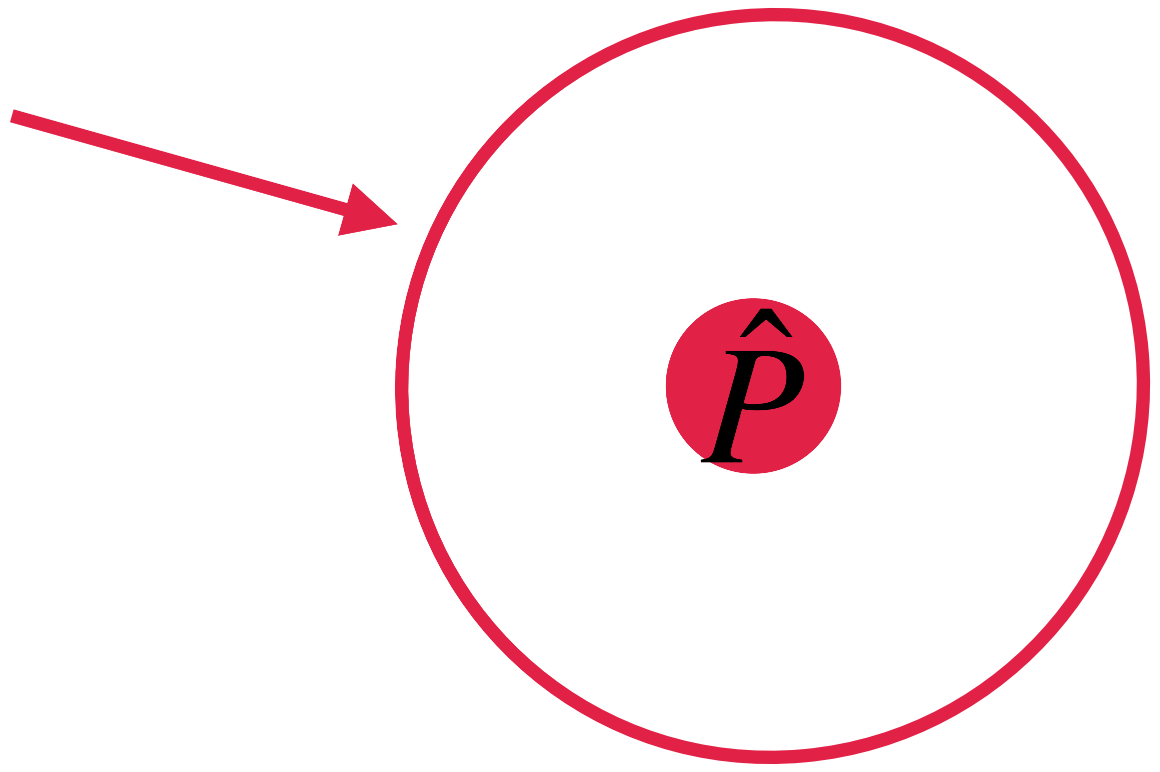
Constrained Pessimistic Policy Optimization (CPPO)

1. MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

2. Constrained Pessimistic Policy Optimization

$$\max_{\pi} \min_{P \in \mathscr{P}} J(\pi; P)$$

$$\text{s.t.,} \frac{1}{|\mathscr{D}|} \sum_{s,a \in \mathscr{D}} \left\| P(\cdot \mid s, a) - \hat{P}(\cdot \mid s, a) \right\|_1 \leq \delta$$



Constrained Pessimistic Policy Optimization (CPPO)

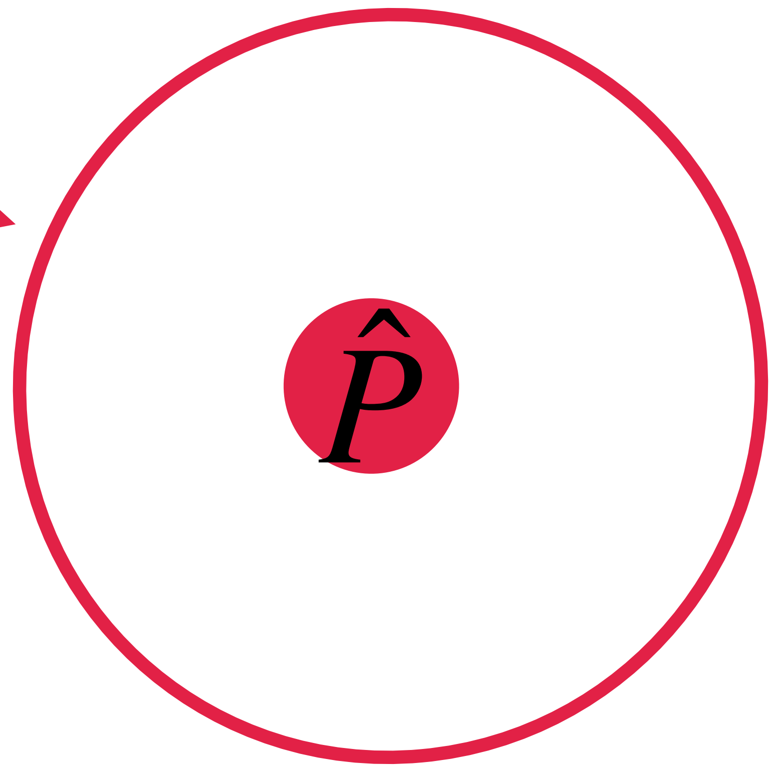
1. MLE: $\hat{P} = \max_{P \in \mathcal{P}} \sum_{s,a,s' \in \mathcal{D}} \ln P(s' \mid s, a)$

2. Constrained Pessimistic Policy Optimization

$$\max_{\pi} \min_{P \in \mathcal{P}} J(\pi; P)$$

$$\text{s.t.,} \frac{1}{|\mathcal{D}|} \sum_{s,a \in \mathcal{D}} \left\| P(\cdot \mid s, a) - \hat{P}(\cdot \mid s, a) \right\|_1 \leq \delta$$

$$\left(\text{or } \frac{1}{|\mathcal{D}|} \sum_{s,a,s' \in \mathcal{D}} \ln P(s' \mid s, a) \geq \frac{1}{|\mathcal{D}|} \sum_{s,a,s' \in \mathcal{D}} \ln \hat{P}(s' \mid s, a) - \delta \right)$$



Constrained Pessimistic Policy Optimization (CPPO)

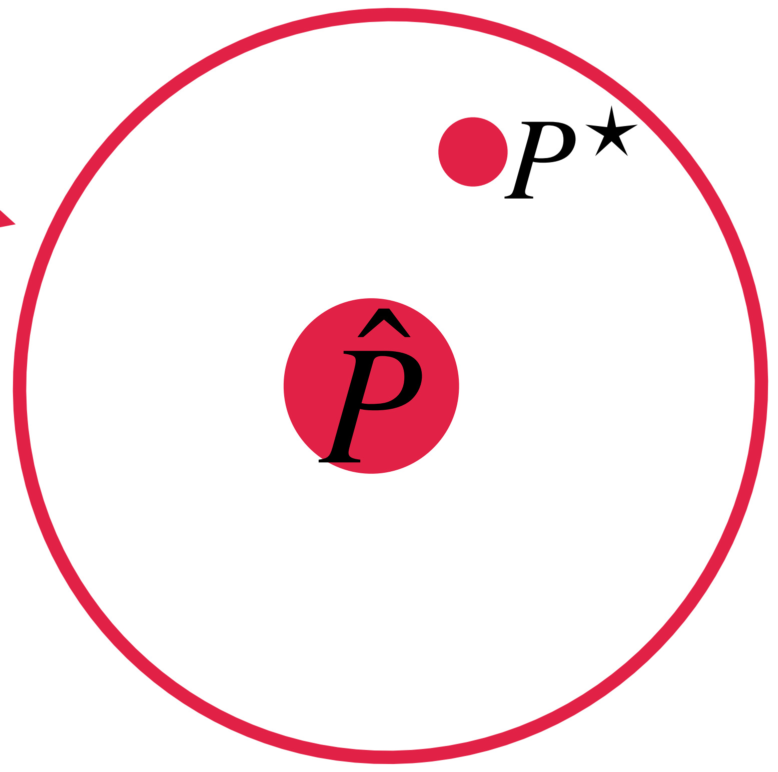
1. MLE: $\hat{P} = \max_{P \in \mathcal{P}} \sum_{s,a,s' \in \mathcal{D}} \ln P(s' \mid s, a)$

2. Constrained Pessimistic Policy Optimization

$$\max_{\pi} \min_{P \in \mathcal{P}} J(\pi; P)$$

$$\text{s.t.,} \frac{1}{|\mathcal{D}|} \sum_{s,a \in \mathcal{D}} \left\| P(\cdot \mid s, a) - \hat{P}(\cdot \mid s, a) \right\|_1 \leq \delta$$

$$\left(\text{or } \frac{1}{|\mathcal{D}|} \sum_{s,a,s' \in \mathcal{D}} \ln P(s' \mid s, a) \geq \frac{1}{|\mathcal{D}|} \sum_{s,a,s' \in \mathcal{D}} \ln \hat{P}(s' \mid s, a) - \delta \right)$$



Constrained Pessimistic Policy Optimization (CPPO)

1. MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

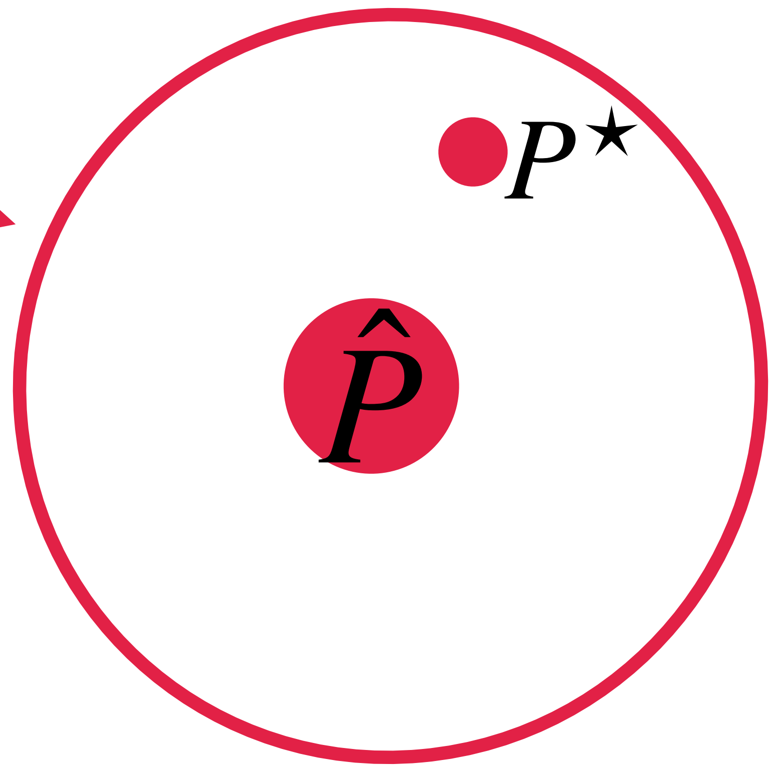
2. Constrained Pessimistic Policy Optimization

$$\max_{\pi} \min_{P \in \mathscr{P}} J(\pi; P)$$

$$\text{s.t.,} \frac{1}{|\mathscr{D}|} \sum_{s,a \in \mathscr{D}} \left\| P(\cdot \mid s, a) - \hat{P}(\cdot \mid s, a) \right\|_1 \leq \delta$$

$$\left(\text{or } \frac{1}{|\mathscr{D}|} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a) \geq \frac{1}{|\mathscr{D}|} \sum_{s,a,s' \in \mathscr{D}} \ln \hat{P}(s' \mid s, a) - \delta \right)$$

Select the least favorable model!



Formal Theoretical Guarantee for CPPO

1. Definition of offline data coverage

Given a policy π, define:

$$C_\pi^\dagger = \sup_{P' \in \mathscr{P}} \frac{\mathbb{E}_{(s,a)\sim d^\pi}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}{\mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}$$

Formal Theoretical Guarantee for CPPO

1. Definition of offline data coverage

π's state-action distribution

Given a policy π, define:

$$C_\pi^\dagger = \sup_{P' \in \mathscr{P}} \frac{\mathbb{E}_{(s,a)\sim d^\pi}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}{\mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}$$

Formal Theoretical Guarantee for CPPO

1. Definition of offline data coverage

Given a policy π, define:

π's state-action distribution

$$C_\pi^\dagger = \sup_{P' \in \mathscr{P}} \frac{\mathbb{E}_{(s,a)\sim d^\pi}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}{\mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}$$

offline state-action distribution

Formal Theoretical Guarantee for CPPO

1. Definition of offline data coverage

Given a policy π, define:

π's state-action distribution

$$C_\pi^\dagger = \sup_{P' \in \mathscr{P}} \frac{\mathbb{E}_{(s,a)\sim d^\pi}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}{\mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}$$

offline state-action distribution

Remark 1: $C_\pi^\dagger \leq \sup_{s,a} \frac{d^\pi(s,a)}{d^{\pi_b}(s,a)}$

Formal Theoretical Guarantee for CPPO

1. Definition of offline data coverage

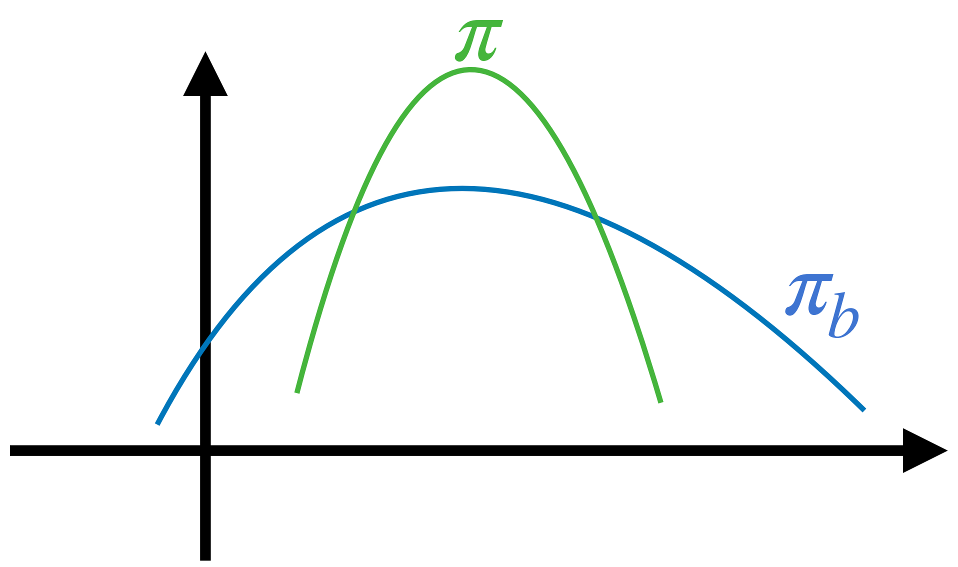
Given a policy π, define:

π's state-action distribution

$$C_\pi^\dagger = \sup_{P' \in \mathscr{P}} \frac{\mathbb{E}_{(s,a)\sim d^\pi}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}{\mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}$$

offline state-action distribution

Remark 1: $C_\pi^\dagger \le \sup_{s,a} \frac{d^\pi(s,a)}{d^{\pi_b}(s,a)}$



Formal Theoretical Guarantee for CPPO

1. Definition of offline data coverage

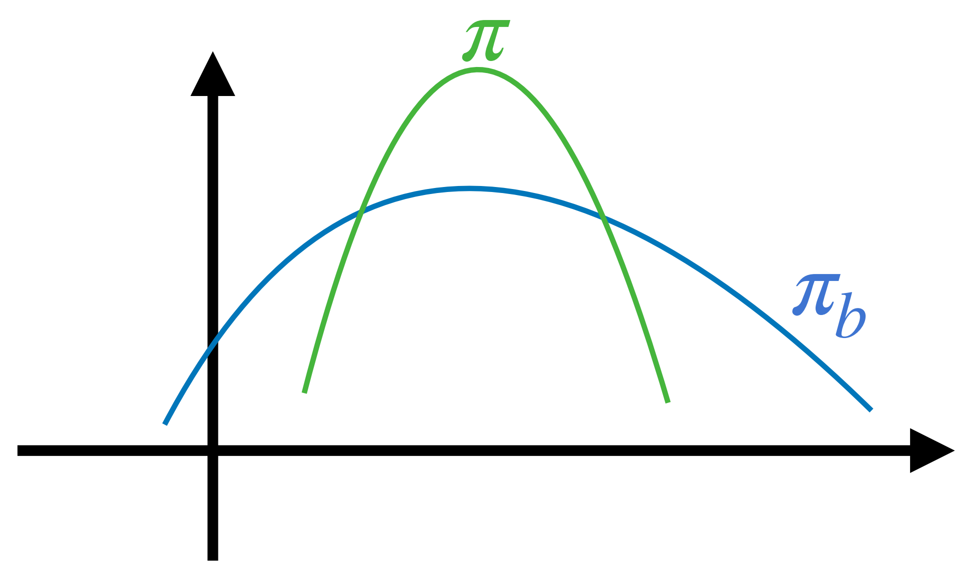
Given a policy π, define:

π's state-action distribution

$$C_\pi^\dagger = \sup_{P' \in \mathscr{P}} \frac{\mathbb{E}_{(s,a)\sim d^\pi}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}{\mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\|P'(\cdot \mid s,a) - P^\star(\cdot \mid s,a)\|_1^2\right]}$$

offline state-action distribution

Remark 1: $C_\pi^\dagger \leq \sup_{s,a} \frac{d^\pi(s,a)}{d^{\pi_b}(s,a)}$



Remark 2: when $P = P^\star, \forall P \in \mathscr{P}$, we have $C_\pi^\dagger = 1$

Formal Theoretical Guarantee for CPPO

2. CPPO's Sample Complexity:

Given n (i.i.d) offline data points, with high probability:

$$\forall \pi^*; V_{P^\star}^{\pi^*} - V_{P^\star}^{\hat{\pi}} = O\left(H^2 \sqrt{\frac{C_{\pi^*}^{\dagger} \ln(|\mathscr{P}|/\delta)}{n}} \right)$$

Formal Theoretical Guarantee for CPPO

2. CPPO's Sample Complexity:

Given n (i.i.d) offline data points, with high probability:

$$\forall \pi^*; V_{P^\star}^{\pi^*} - V_{P^\star}^{\hat{\pi}} = O\left(H^2 \sqrt{\frac{C_{\pi^*}^{\dagger} \ln(|\mathscr{P}|/\delta)}{n}} \right)$$

The cost we pay if want to compete w/ less covered policy π^*

Formal Theoretical Guarantee for CPPO

2. CPPO's Sample Complexity:

Given n (i.i.d) offline data points, with high probability:

$$\forall \pi^*; V_{P^\star}^{\pi^*} - V_{P^\star}^{\hat{\pi}} = O\left(H^2 \sqrt{\frac{C_{\pi^*}^{\dagger} \ln(|\mathscr{P}|/\delta)}{n}} \right)$$

The cost we pay if want to compete w/ less covered policy π^*

Robustness!

Formal Theoretical Guarantee for CPPO

2. CPPO's Sample Complexity:

Given n (i.i.d) offline data points, with high probability:

$$\forall \pi^*; V_{P^\star}^{\pi^*} - V_{P^\star}^{\hat{\pi}} = O\left(H^2 \sqrt{\frac{C_{\pi^*}^{\dagger} \ln(|\mathscr{P}|/\delta)}{n}} \right)$$

The cost we pay if want to compete w/ less covered policy π^*

Statistical complexity of $\mathscr{P}$; no poly dependence on $|S|, |A|$

Robustness!

Formal Theoretical Guarantee for CPPO

2. CPPO's Sample Complexity:

Given n (i.i.d) offline data points, with high probability:

$$\forall \pi^*; V_{P^\star}^{\pi^*} - V_{P^\star}^{\hat{\pi}} = O\left(H^2 \sqrt{\frac{C_{\pi^*}^{\dagger} \ln(|\mathscr{P}|/\delta)}{n}}\right)$$

The cost we pay if want to compete w/ less covered policy π^*

Robustness!

Statistical complexity of $\mathscr{P}$; no poly dependence on $|S|, |A|$

SL-style Generalization!

CPPO and its general bound applies to many examples...

The non-parametric kernel model

$$s_{t+1} = f^\star(s_t, a_t) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2 I),$$

where $f^\star$ is from some RKHS w/ kernel $k([s, a], [s', a']) = \langle \phi(s, a), \phi(s', a') \rangle$

CPPO and its general bound applies to many examples...

The non-parametric kernel model

$$s_{t+1} = f^\star(s_t, a_t) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2 I),$$

where $f^\star$ is from some RKHS w/ kernel $k([s,a],[s',a']) = \langle \phi(s,a), \phi(s',a') \rangle$

Coverage def is reduced to a relative condition number:

$$C_\pi^\dagger = \max_x \frac{x^\top \mathbb{E}_{s,a\sim d^\pi} \phi(s,a)\phi(s,a)^\top x}{x^\top \mathbb{E}_{s,a\sim d^{\pi_b}} \phi(s,a)\phi(s,a)^\top x}$$

CPPO and its general bound applies to many examples...

The non-parametric kernel model

$$s_{t+1} = f^\star(s_t, a_t) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2 I),$$

where $f^\star$ is from some RKHS w/ kernel $k([s,a],[s',a']) = \langle \phi(s,a), \phi(s',a') \rangle$

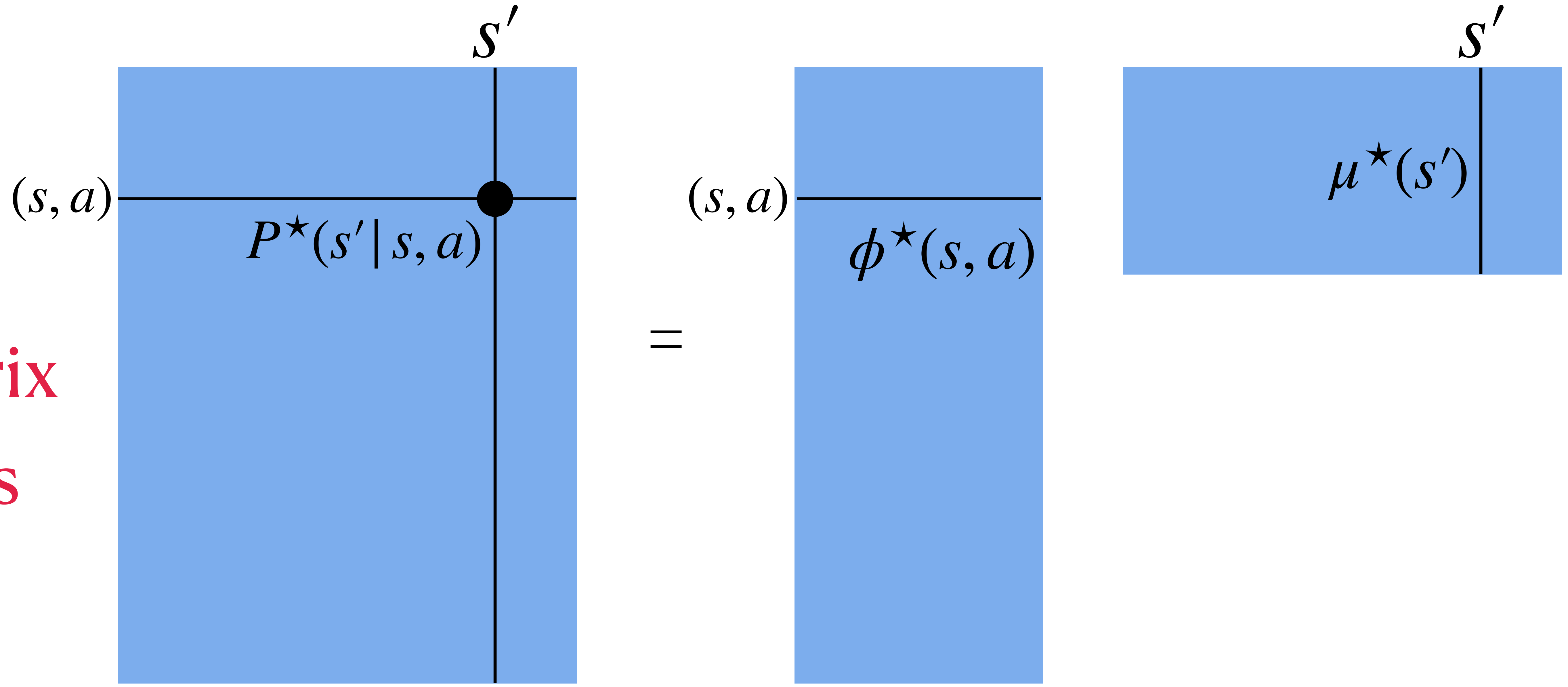
Coverage def is reduced to a relative condition number:

$$C_\pi^\dagger = \max_x \frac{x^\top \mathbb{E}_{s,a\sim d^\pi} \phi(s,a)\phi(s,a)^\top x}{x^\top \mathbb{E}_{s,a\sim d^{\pi_b}} \phi(s,a)\phi(s,a)^\top x}$$

i.e., measuring coverage using subspace..

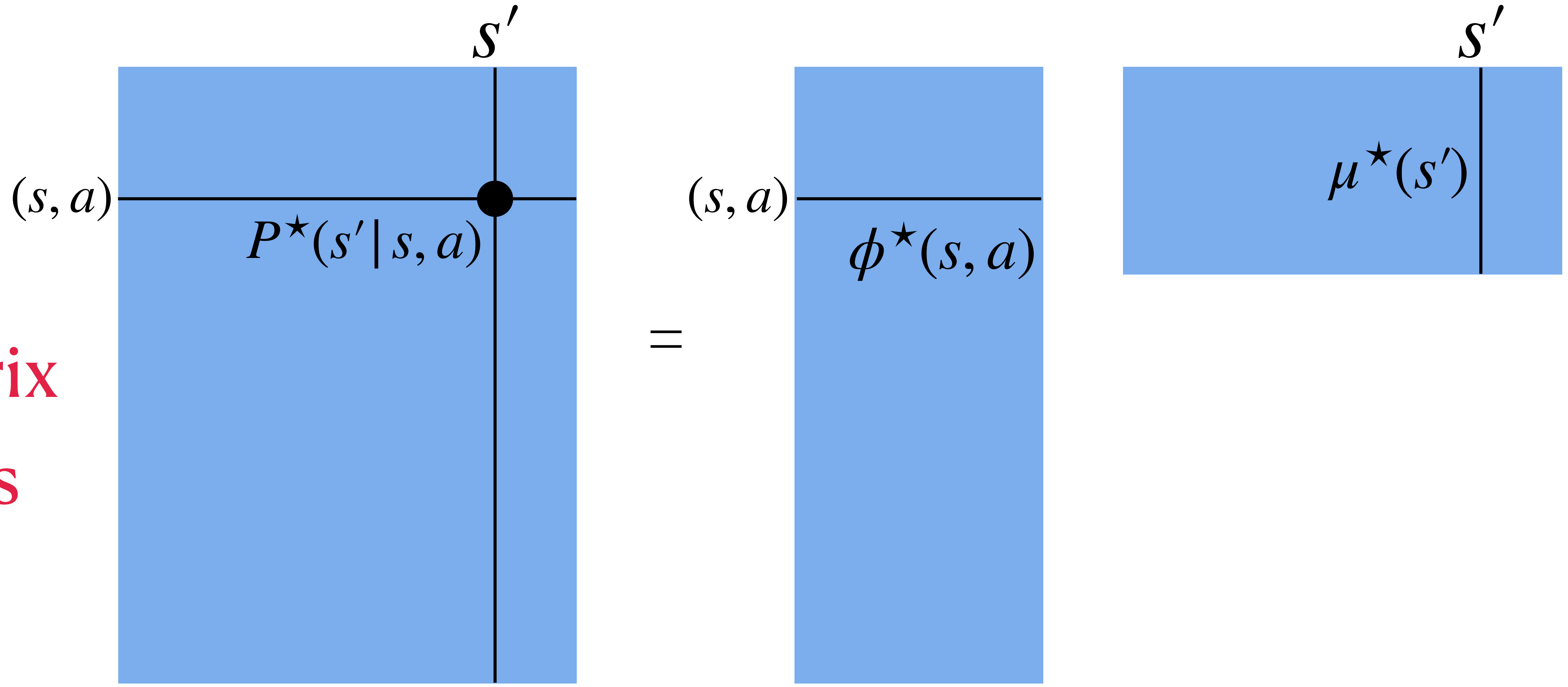
CPPO for Low-rank MDPs

Transition matrix
$P \in \mathbb{R}^{SA \times S}$ has
rank d



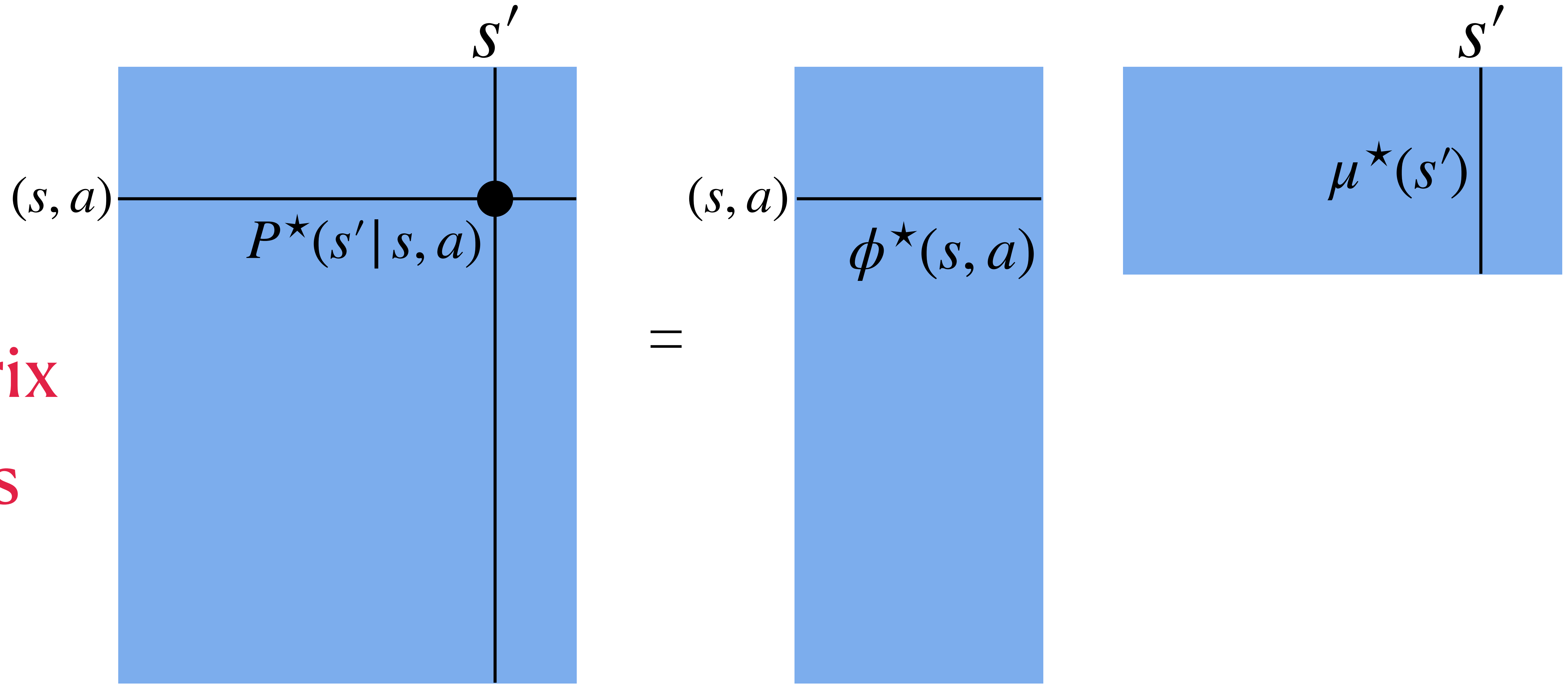
CPPO for Low-rank MDPs

Transition matrix $P \in \mathbb{R}^{SA \times S}$ has rank d



$$\exists \mu^\star, \phi^\star : \quad \forall s, a, s', P^\star(s' \mid s, a) = \mu^\star(s')^\top \phi^\star(s, a)$$

CPPO for Low-rank MDPs



Transition matrix $P \in \mathbb{R}^{SA \times S}$ has rank d

$$\exists \mu^\star, \phi^\star : \quad \forall s, a, s', P^\star(s' \mid s, a) = \mu^\star(s')^\top \phi^\star(s, a)$$

In low-rank MDP, neither $\mu^\star$ nor $\phi^\star$ is known

CPPO for Low-rank MDPs

Two Function classes Γ & Φ

Realizability: $\mu^\star \in \Gamma$, $\phi^\star \in \Phi$

CPPO for Low-rank MDPs

Two Function classes Γ & Φ

Realizability: $\mu^\star \in \Gamma$, $\phi^\star \in \Phi$

Coverage def: relative condition number under ground truth (unknown) $\phi^\star$

$$C_\pi^\dagger = \max_x \frac{x^\top \mathbb{E}_{s,a\sim d^\pi} \phi^\star(s,a)\phi^\star(s,a)^\top x}{x^\top \mathbb{E}_{s,a\sim d^{\pi_b}} \phi^\star(s,a)\phi^\star(s,a)^\top x}$$

CPPO for Low-rank MDPs

Two Function classes Γ & Φ

Realizability: $\mu^\star \in \Gamma$, $\phi^\star \in \Phi$

Coverage def: relative condition number under ground truth (unknown) $\phi^\star$

$$C_\pi^\dagger = \max_x \frac{x^\top \mathbb{E}_{s,a\sim d^\pi} \phi^\star(s,a)\phi^\star(s,a)^\top x}{x^\top \mathbb{E}_{s,a\sim d^{\pi_b}} \phi^\star(s,a)\phi^\star(s,a)^\top x}$$

$$\forall \pi^*; V_{P^\star}^{\pi^*} - V_{P^\star}^{\hat{\pi}} = O\left(H^2 \sqrt{\frac{d C_{\pi^*}^\dagger \ln(|\Gamma||\Phi|/\delta)}{n}} \right)$$

CPPO for Low-rank MDPs

Two Function classes Γ & Φ

Realizability: $\mu^\star \in \Gamma$, $\phi^\star \in \Phi$

Coverage def: relative condition number under ground truth (unknown) $\phi^\star$

$$C_\pi^\dagger = \max_x \frac{x^\top \mathbb{E}_{s,a\sim d^\pi} \phi^\star(s,a)\phi^\star(s,a)^\top x}{x^\top \mathbb{E}_{s,a\sim d^{\pi_b}} \phi^\star(s,a)\phi^\star(s,a)^\top x}$$

$$\forall \pi^*; V_{P^\star}^{\pi^*} - V_{P^\star}^{\hat{\pi}} = O\left(H^2 \sqrt{\frac{d C_{\pi^*}^\dagger \ln(|\Gamma||\Phi|/\delta)}{n}} \right)$$

(Many more interesting examples: linear mixture MDP, factored MDP, etc)

Implementation

1. MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

2: Treat constraint as a penalty w/ Lagrangian multiplier:

Implementation

1. MLE: $\hat{P} = \max_{P \in \mathscr{P}} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a)$

2: Treat constraint as a penalty w/ Lagrangian multiplier:

$$\max_{\pi} \min_{P} J(\pi; P) + \max_{\lambda \leq 0} \lambda \left(\frac{1}{|\mathscr{D}|} \sum_{s,a,s' \in \mathscr{D}} \ln P(s' \mid s, a) - \frac{1}{|\mathscr{D}|} \sum_{s,a,s' \in \mathscr{D}} \ln \hat{P}(s' \mid s, a) + \delta \right)$$

Practical version of CPPO (Rigter et al. Neurips22)

“Uehara et al. (2021) provides the theoretical motivation for solving Problem 1. In this work, we focus on developing a practical approach to solving Problem 1.”

CPPO

Practical version of CPPO

		Ours	Model-based baselines				Model-free baselines		
		RAMBO	**RepB-SDE**	**COMBO**	**MOPO**	**MOReL**	**CQL**	**IQL**	**TD3+BC**
Random	HalfCheetah	40.0 ± 2.3	32.9	38.8	35.4	25.6	19.6	-	11.0
	Hopper	21.6 ± 8.0	8.6	17.9	4.1	53.6	6.7	-	8.5
	Walker2D	11.5 ± 10.5	21.1	7.0	4.2	37.3	2.4	-	1.6
Medium	HalfCheetah	77.6 ± 1.5	49.1	54.2	69.5	42.1	49.0	47.4	48.3
	Hopper	92.8 ± 6.0	34.0	94.9	48.0	95.4	66.6	66.3	59.3
	Walker2D	86.9 ± 2.7	72.1	75.5	-0.2	77.8	83.8	78.3	83.7
Medium Replay	HalfCheetah	68.9 ± 2.3	57.5	55.1	68.2	40.2	47.1	44.2	44.6
	Hopper	96.6 ± 7.0	62.2	73.1	39.1	93.6	97.0	94.7	60.9
	Walker2D	85.0 ± 15.0	49.8	56.0	69.4	49.8	88.2	73.9	81.8
Medium Expert	HalfCheetah	93.7 ± 10.5	55.4	90.0	72.7	53.3	90.8	86.7	90.7
	Hopper	83.3 ± 9.1	82.6	111.1	3.3	108.7	106.8	91.5	98.0
	Walker2D	68.3 ± 20.6	88.8	96.1	-0.3	95.6	109.4	109.6	110.1
MuJoCo-v2 Total:		826.2 ± 33.8	614.1	769.7	413.4	773.0	767.4	692.6*	698.5

SOTA

Summary

Rethinking offline RL's learning objective:

Generalization & Robustness

Summary

Rethinking offline RL's learning objective:

Generalization & Robustness

Like SL, learning via function approximation, i.e., generalization rather than memorization / numeration

Summary

Rethinking offline RL's learning objective:

Generalization & Robustness

Like SL, learning via function approximation, i.e., generalization rather than memorization / numeration

- Expecting offline data has global coverage is too much;
- Learning to compete the best among those covered