

# Batch / Offline RL

Emma Brunskill  
CS234  
Winter 2021

Thanks to Phil Thomas for some figures

# Refresh Your Understanding: Fast RL

Select all that are true:

- In Thompson sampling for MDPs, the posterior over the dynamics can be updated after each transition
- When using a Beta prior for a Bernoulli reward parameter for an  $(s,a)$  pair, the posterior after  $N$  samples of that pair time steps can be the same as after  $N+2$  samples
- The optimism bonuses discussed for MBIE-EB depend on the maximum reward but not on the maximum value function
- In class we discussed adding a bonus term to an update for a  $(s,a,r,s')$  tuple using Q-learning with function approximation. Adding this bonus term will ensure all Q estimates used to make decisions online using DQN are optimistic with respect to  $Q^*$
- Not sure

# Refresh Your Understanding: Fast RL Solutions

Select all that are true:

- In Thompson sampling for MDPs, the posterior over the dynamics can be updated after each transition (True)
- When using a Beta prior for a Bernoulli reward parameter for an (s,a) pair, the posterior after N samples of that pair time steps can be the same as after N+2 samples (False)
  - Beta(alpha,beta) could be Beta(alpha+2,beta), Beta(alpha+1,beta+1), Beta(alpha,beta+2)
- The optimism bonuses discussed for MBIE-EB depend on the maximum reward but not on the maximum value function (False)
  - **The optimism bonuses depend on the max value**
- In class we discussed adding a bonus term to an update for a (s,a,r,s') tuple using Q-learning with function approximation. Adding this bonus term will ensure all Q estimates used to make decisions online using DQN are optimistic with respect to  $Q^*$  (False)
  - **Function approximation may mean that the resulting estimate is not always optimistic**

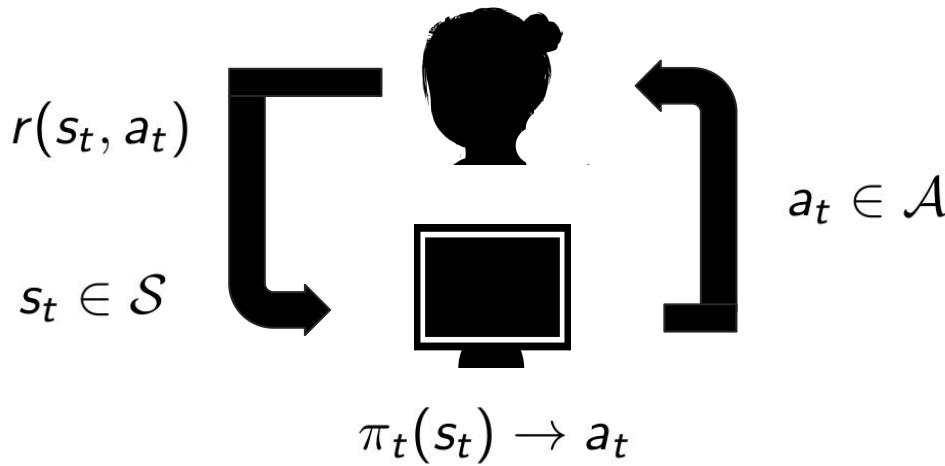
# Outline for Today

1. Introduction and Setting
2. Offline batch evaluation using models
3. Offline batch evaluation using Q functions
4. Offline batch evaluation using importance sampling

# Class Progress

- Last time: Fast RL III
- This time: Batch RL
- Next time: Guest lecture with Professor Doshi-Velez

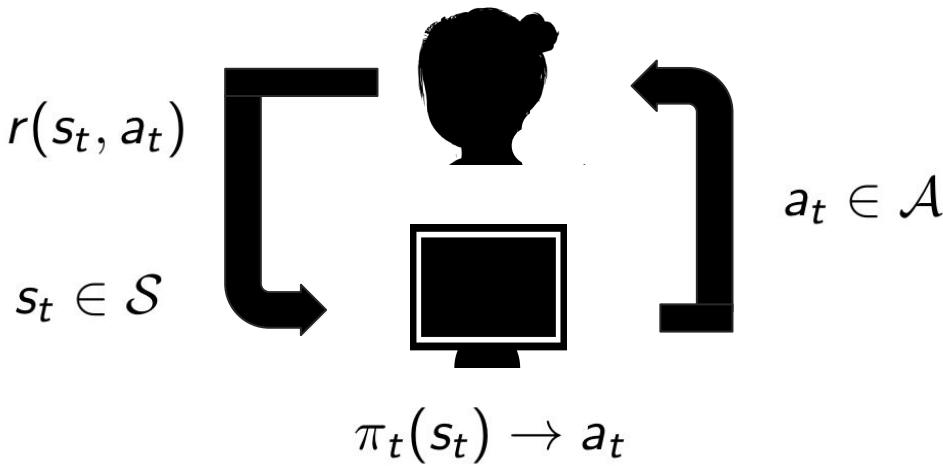
# Reinforcement Learning



$$\underbrace{V^\pi(s)}_{\text{Value func.}} = \underbrace{r(s, \pi(s))}_{\text{Reward}} + \gamma \sum_{s'} \underbrace{p(s'|s, a)}_{\text{Dynamics}} V^\pi(s')$$

Only observed through samples (experience)

# Today: Counterfactual / Batch RL

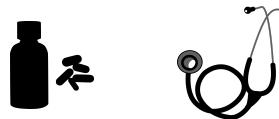


$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

# Outline for Today

1. Introduction and Setting
2. Offline batch evaluation using models
3. Offline batch evaluation using Q functions
4. Offline batch evaluation using importance sampling

Patient group 1 →



Outcome: 92

Patient group 2 →



Outcome: 91

Patient group 1 →



Outcome: 92

Patient group 2 →



Outcome: 91



?

# “What If?” Reasoning Given Past Data

Patient group 1 →   → Outcome: 92

Patient group 2 →   → Outcome: 91



?

# Data Is Censored in that Only Observe Outcomes for Decisions Made

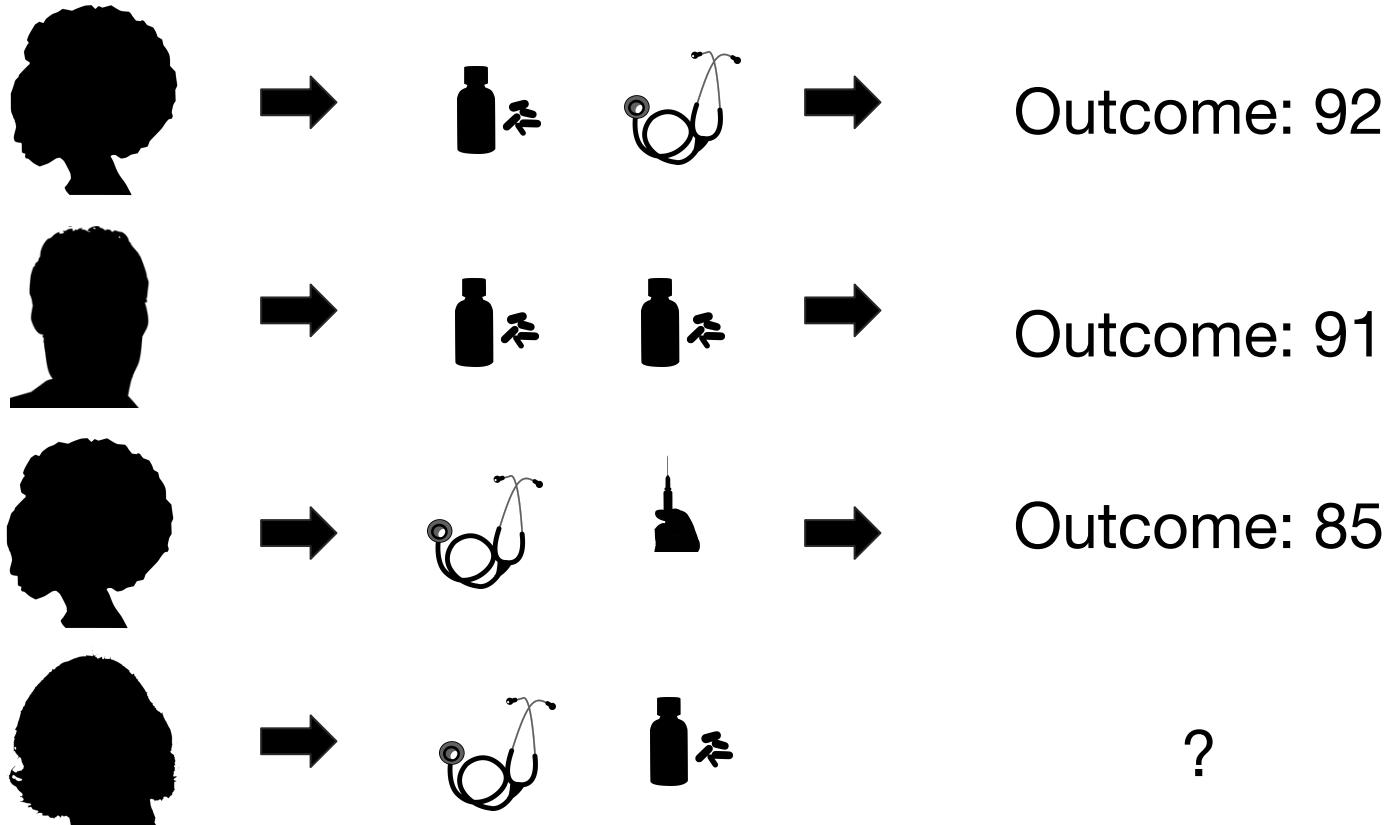
Patient group 1 →   → Outcome: 92

Patient group 2 →   → Outcome: 91



?

# Need for Generalization



# Off Policy Reinforcement Learning

Watkins 1989

Watkins and Dayan 1992

Precup et al. 2000

Lagoudakis and Parr 2002

Murphy 2005

Sutton, Szepesvari and Maei 2009

Shortreed, Laber, Lizotte, Stroup, Pineau, & Murphy 2011

Degirs, White, and Sutton 2012

Mnih et al. 2015

Mahmood et al. 2014

Jiang & Li 2016

Hallak, Tamar and Mannor 2015

Munos, Stepleton, Harutyunyan and Bellemare 2016

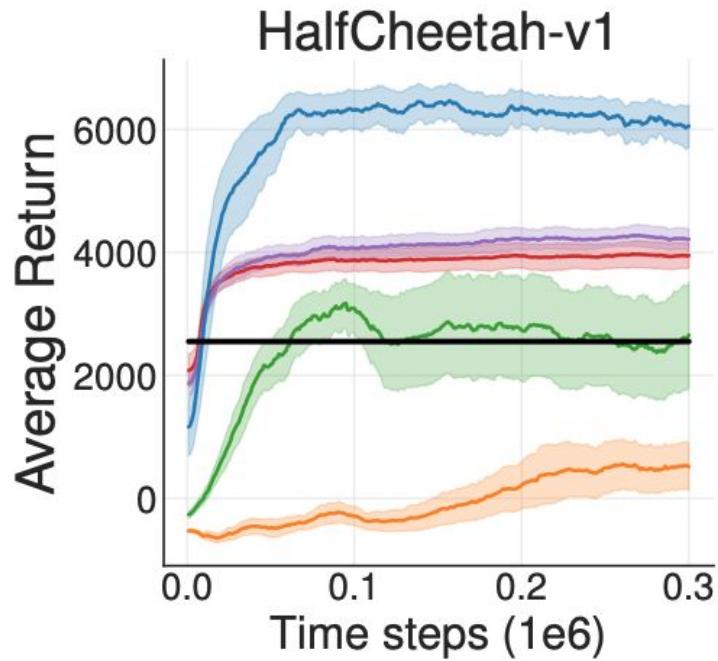
Sutton, Mahmood and White 2016

Du, Chen, Li, Ziao, and Zhou 2016 ...

# Why Can't We Just Use Q-Learning?

- Q-learning is an off policy RL algorithm
  - Can be used with data different than the state--action pairs would visit under the optimal Q state action values
- But deadly triad of bootstrapping, function approximation and off policy, and can fail

# Important in Practice



BCQ figure from Fujimoto,  
Meger, Precup ICML 2019

BCQ

DDPG

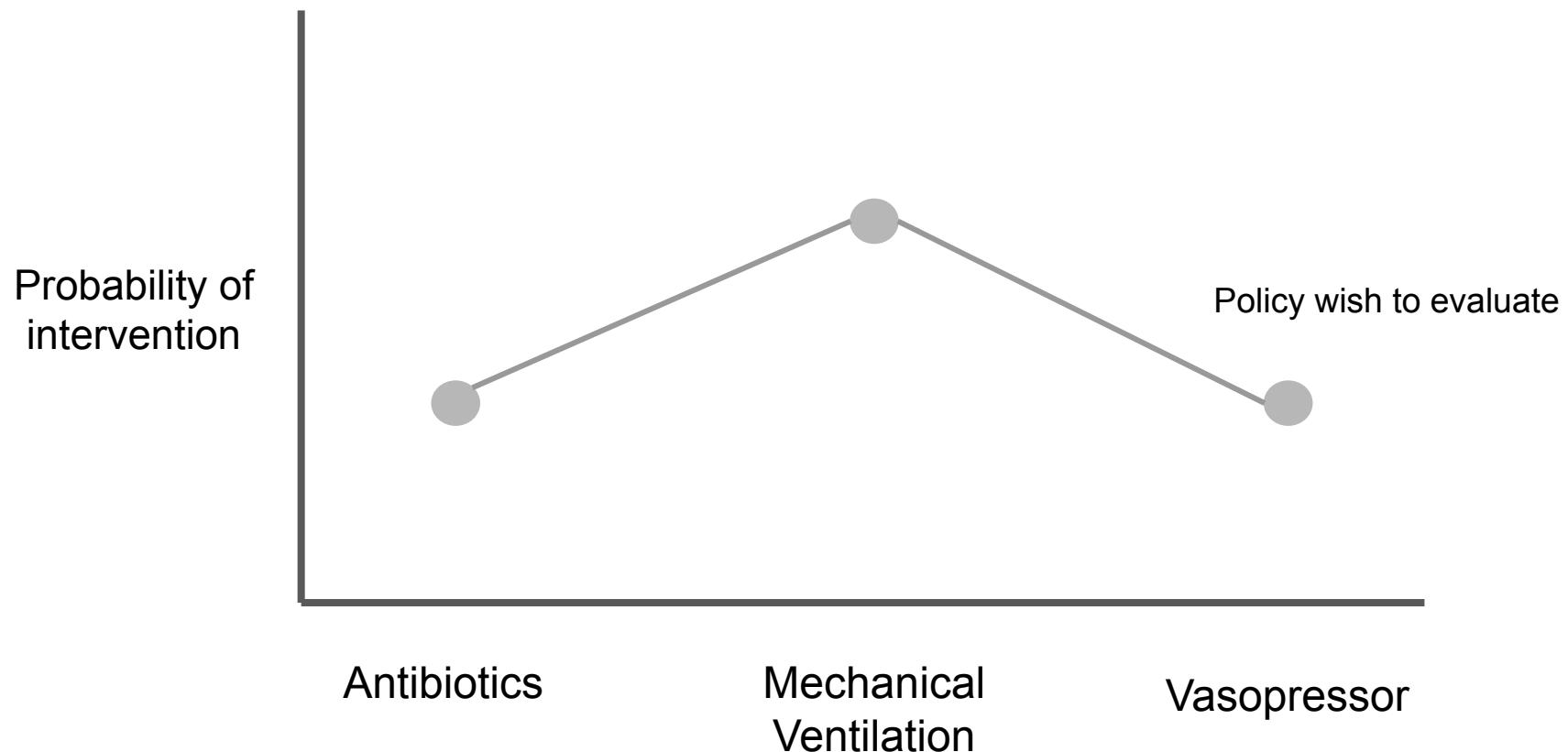
DQN

BC

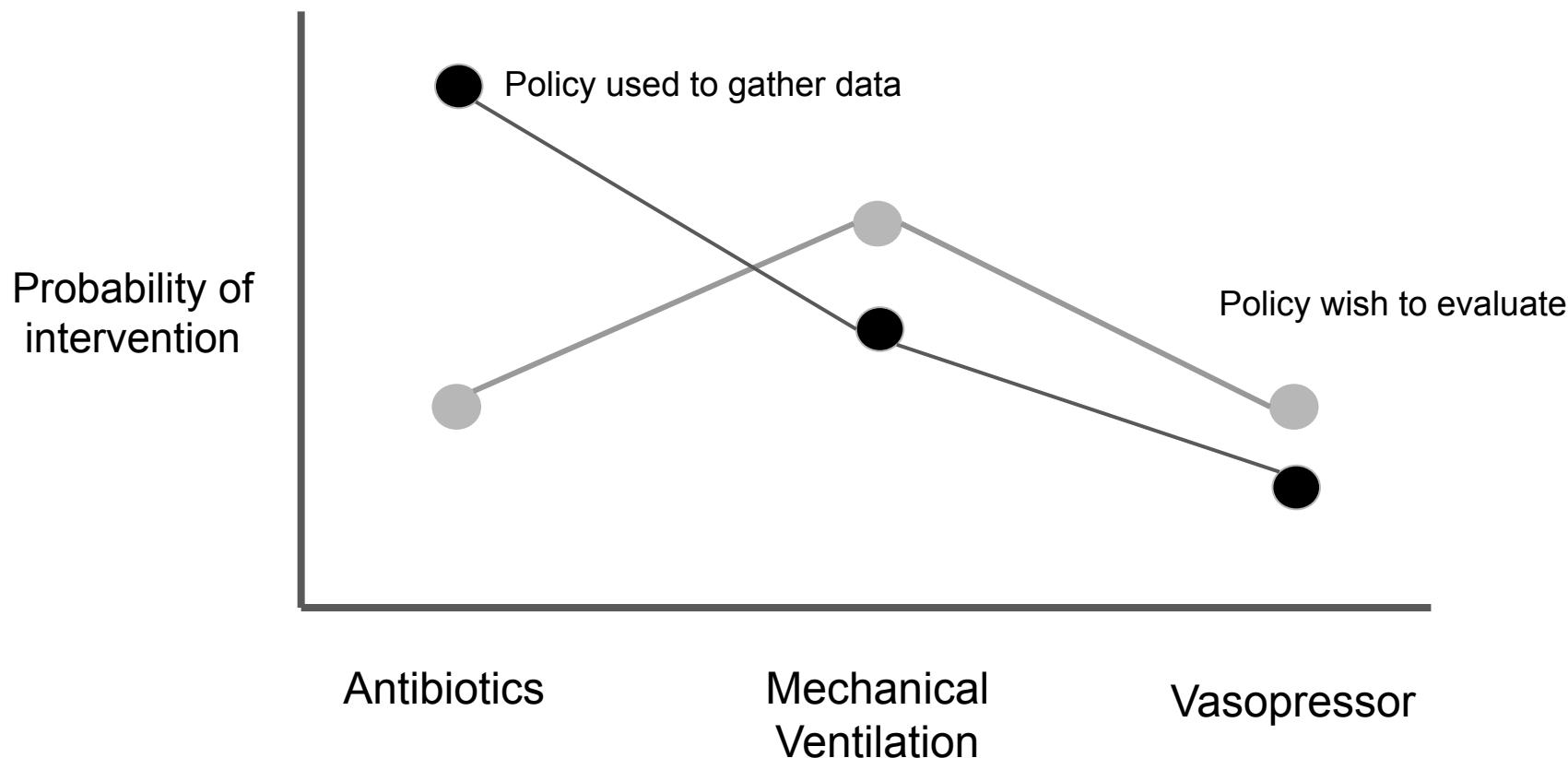
VAE-BC

Behavioral

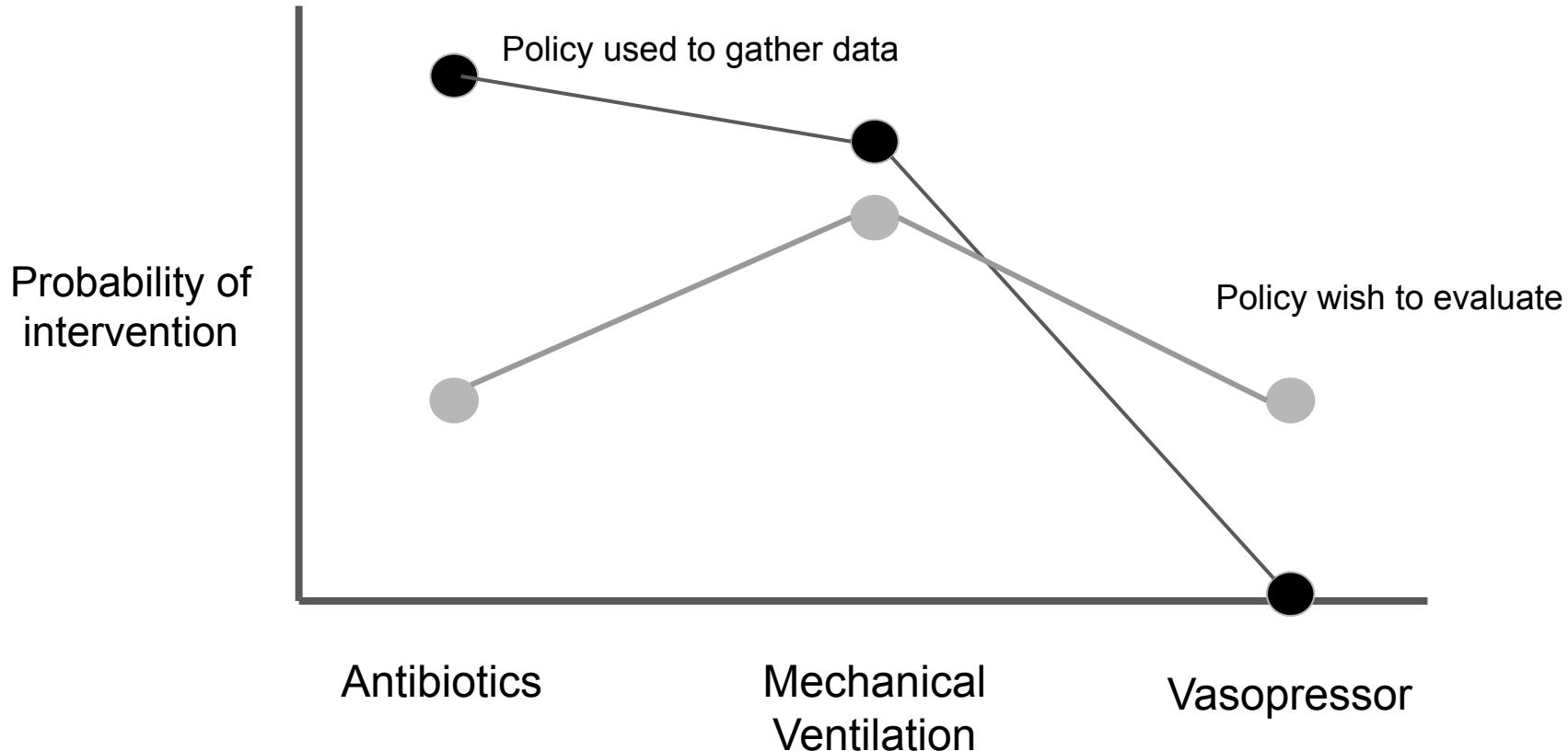
# Challenge: Overlap Requirement



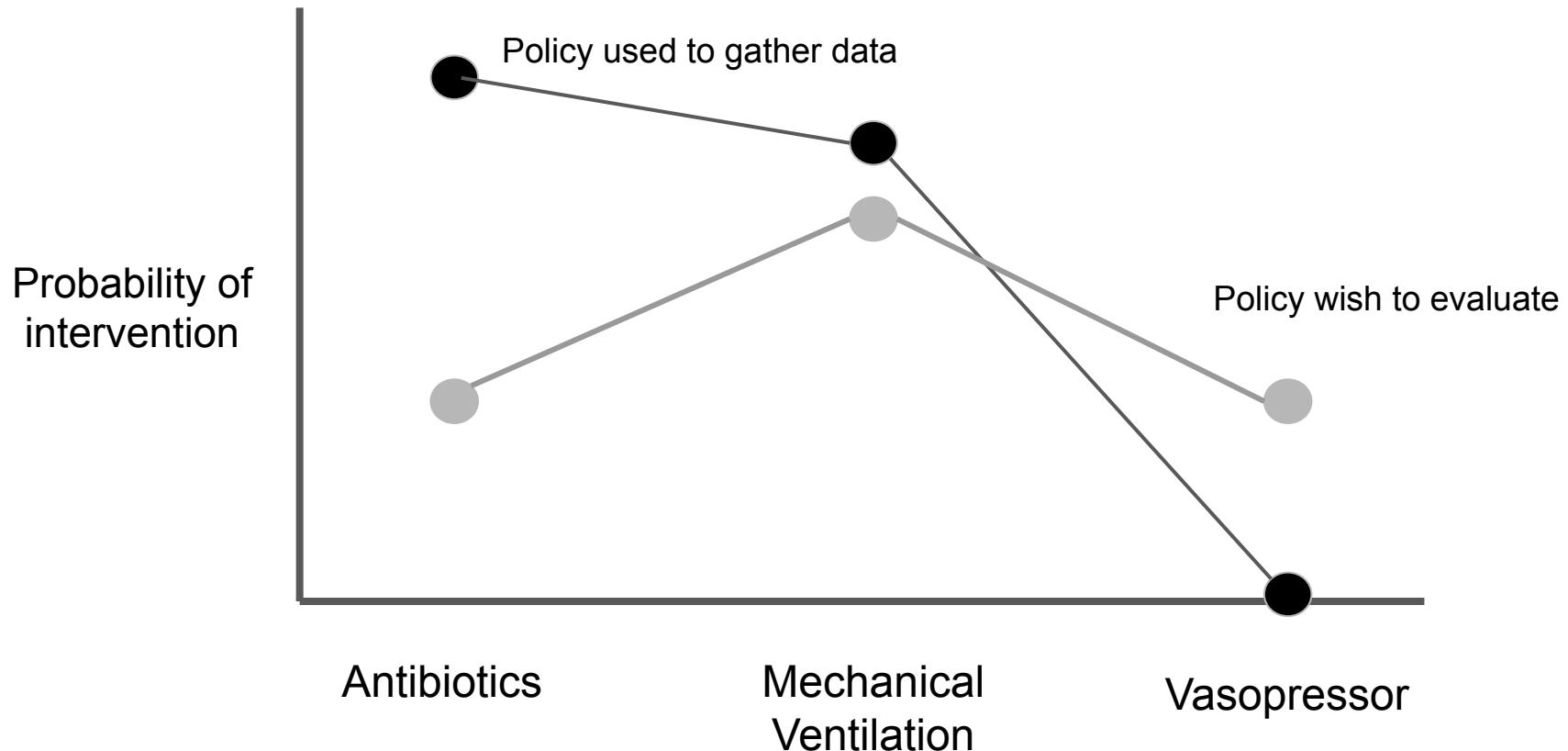
# Overlap Requirement: Data Must Support Policy Wish to Evaluate



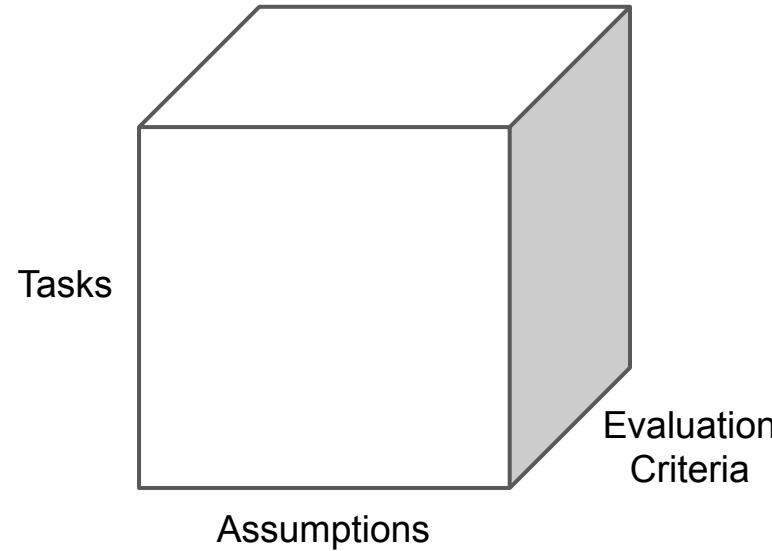
# No Overlap for Vasopressor $\Rightarrow$ Can't Do Off Policy Estimation for Desired Policy



# How to Evaluate Sufficient Overlap in Real Data?



# Offline / Batch Reinforcement Learning



$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

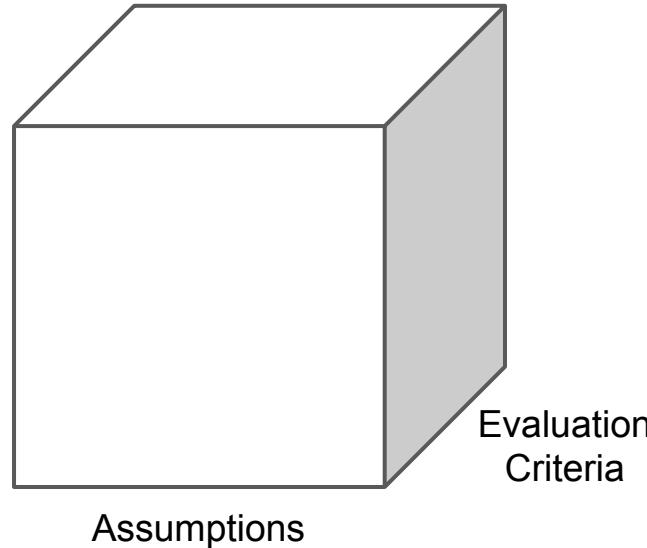
$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Common Tasks: Off Policy Evaluation & Optimization

Tasks

$$\int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds$$

$$\arg \max_{\pi \in \mathcal{H}_i} \int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds$$



$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Common Assumptions

- Stationary process: Policy will be evaluated in or deployed in the same stationary decision process as the behavior policy operated in to gather data
- **Markov**
- Sequential ignorability (no confounding)

$$\{Y(A_{1:(t-1)}, a_{t:T}), S_{t'}(A_{1:(t-1)}, a_{t:(t'-1)})\}_{t'=t+1}^T \perp\!\!\!\perp A_t \mid \mathcal{F}_t$$

- Overlap

$$\forall(s, a) \mu_e(s, a) > 0 \rightarrow \mu_b(s, a) > 0$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

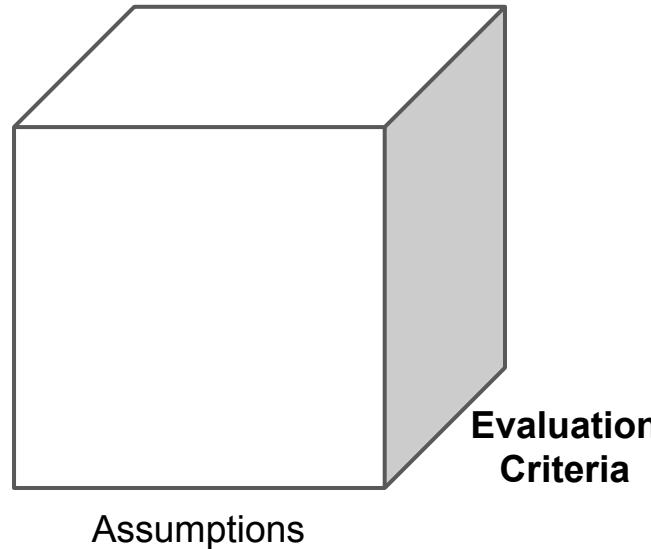
$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Common Tasks: Off Policy Evaluation & Optimization

Tasks

$$\int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds$$

$$\arg \max_{\pi \in \mathcal{H}_i} \int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds$$



$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Off Policy Reinforcement Learning

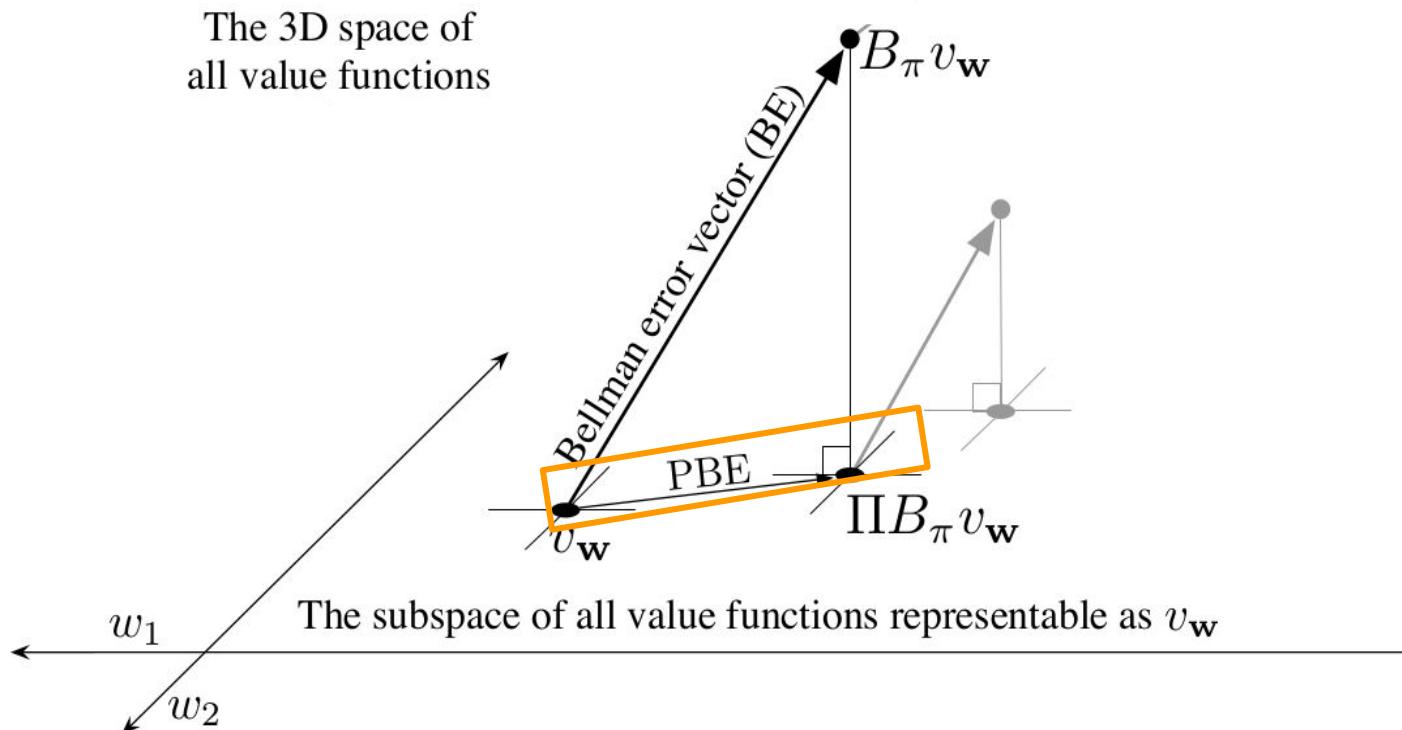


Figure from Sutton & Barto 2018

# Off Policy Reinforcement Learning

The 3D space of all value functions

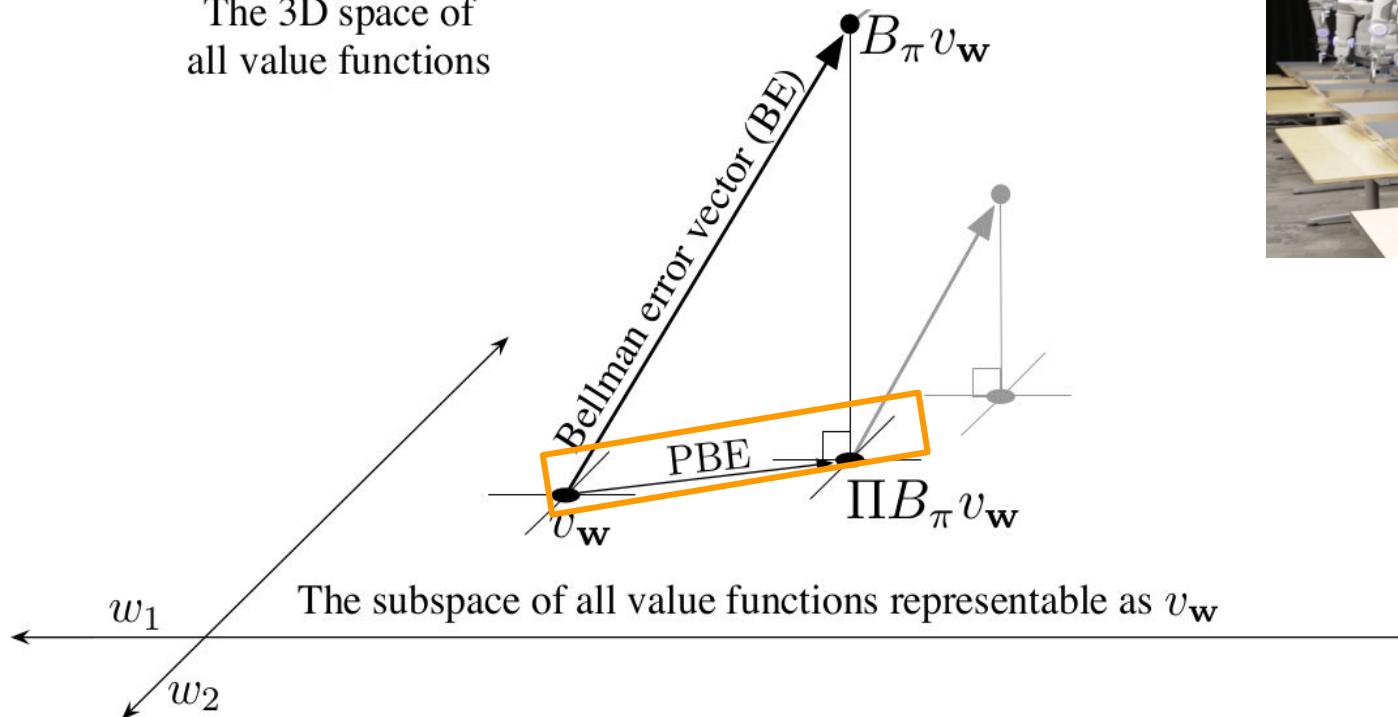


Figure from Sutton & Barto 2018

# Batch Off Policy Reinforcement Learning

The 3D space of all value functions

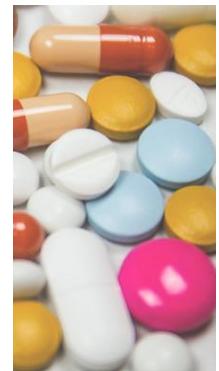
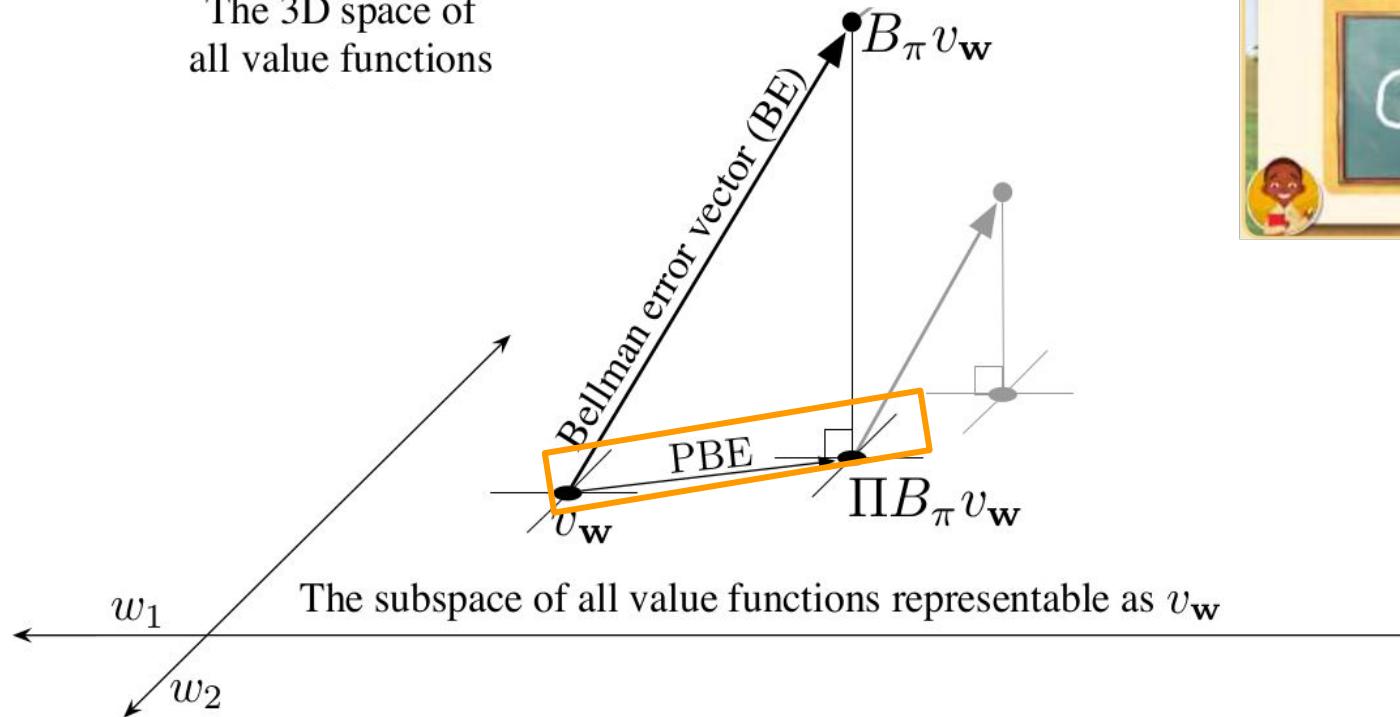


Figure from Sutton & Barto 2018

# Batch Off Policy Reinforcement Learning

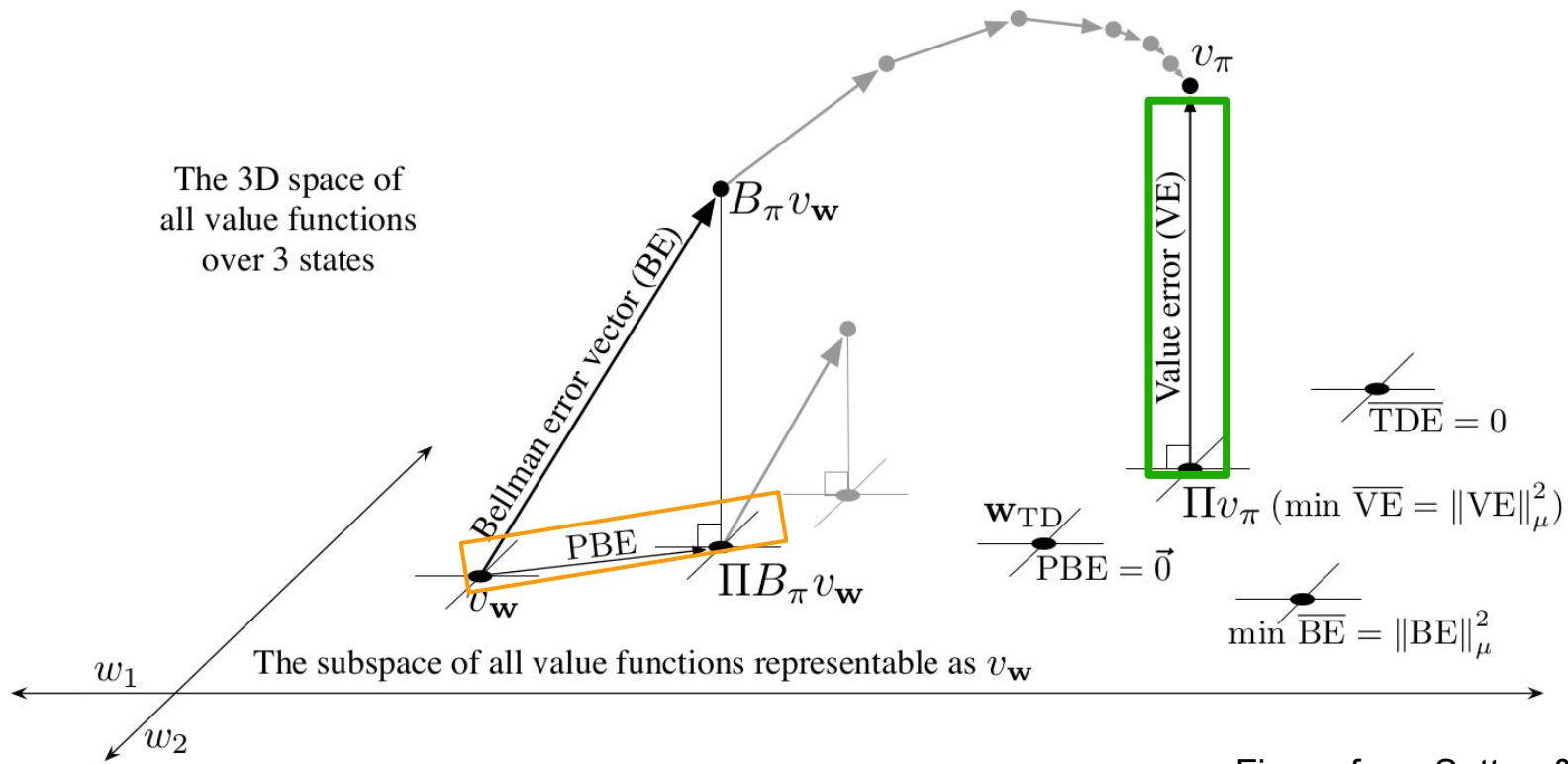


Figure from Sutton & Barto 2018

# Common Evaluation Criteria for Off Policy Evaluation

- Computational efficiency
- Performance accuracy

$$\forall \mathcal{D}_i \in \{\mathcal{D}_1 \sim \mathcal{M}_1, \mathcal{D}_2 \sim \mathcal{M}_2, \dots, \mathcal{D}_K \sim \mathcal{M}_K\} \quad \frac{1}{|\rho|} \sum_{s_0 \in \rho} (\hat{V}_{\mathcal{M}_i}^{\pi}(s_0, \mathcal{D}_i) - V_{\mathcal{M}_i}^{\pi}(s_0))^2$$

$$\lim_{|\mathcal{D}| \rightarrow \infty} \frac{1}{|\rho|} \sum_{s_0 \in \rho} \hat{V}^{\pi}(s_0, \mathcal{D}) \rightarrow \frac{1}{|\rho|} \sum_{s_0 \in \rho} V^{\pi}(s_0)$$

$$\frac{1}{|\rho|} \sum_{s_0 \in \rho} \hat{V}^{\pi}(s_0, \mathcal{D}) \leq \frac{1}{|\rho|} \sum_{s_0 \in \rho} V^{\pi}(s_0) - f(n, \dots)$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^{\pi}(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Offline / Batch Reinforcement Learning

Tasks

$$\int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds$$

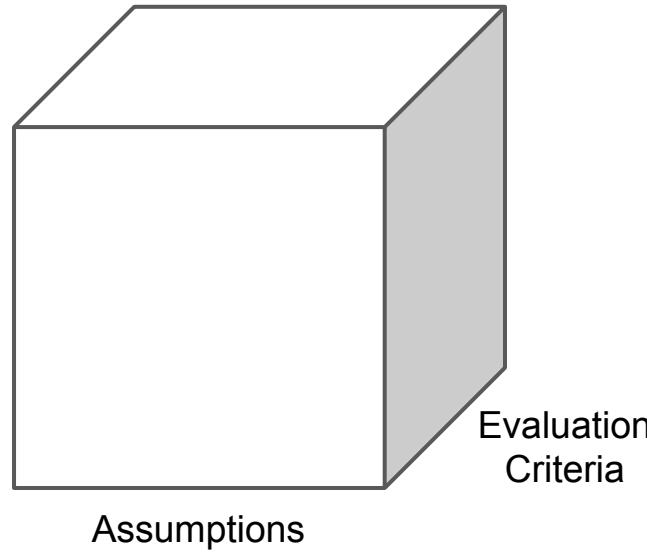
$$\arg \max_{\pi \in \mathcal{H}_i} \int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$



- Empirical accuracy
- Consistency
- Robustness
- Asymptotic efficiency
- Finite sample bounds
- Computational cost

- Markov?
- Overlap?
- Sequential ignorability?

# Batch Policy Optimization: Find a Good Policy That Will Perform Well in the Future

$$\underbrace{\arg \max_{\pi \in \mathcal{H}_i} \max_{\mathcal{H}_i \in \{\mathcal{H}_1, \mathcal{H}_2, \dots\}}}_{\text{Policy Optimization}} \quad \underbrace{\int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds}_{\text{Policy Evaluation}}$$

$$\mathcal{H} = \mathcal{M}, \mathcal{V}, \Pi ?$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$   
 $\pi$ : Policy mapping  $s \rightarrow a$   
 $S_0$ : Set of initial states  
 $\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Batch Policy Evaluation: Estimate the Performance of a Particular Decision Policy

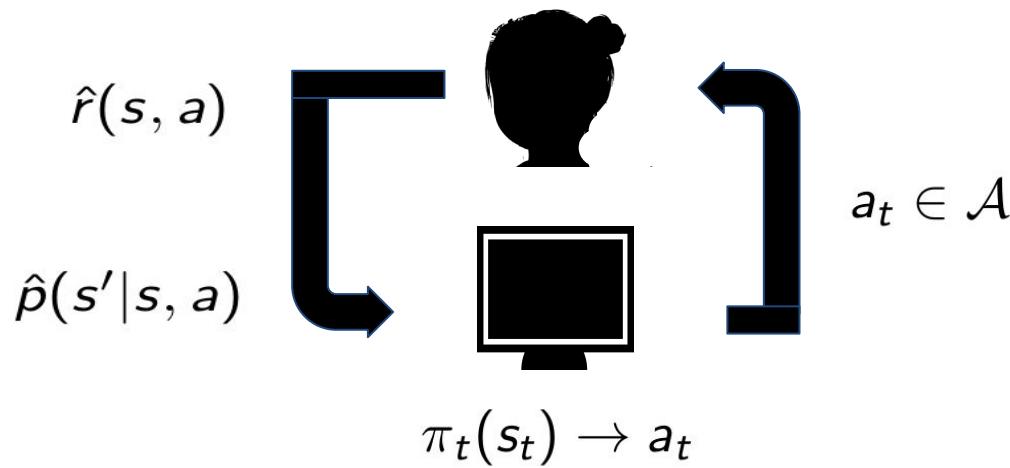
$$\underbrace{\arg \max_{\pi \in \mathcal{H}_i} \max_{\mathcal{H}_i \in \{\mathcal{H}_1, \mathcal{H}_2, \dots\}}}_{\text{Policy Optimization}}$$
$$\underbrace{\int_{s \in S_0} \hat{V}^\pi(s, \mathcal{D}) ds}_{\text{Policy Evaluation}}$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$   
 $\pi$ : Policy mapping  $s \rightarrow a$   
 $S_0$ : Set of initial states  
 $\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Outline

1. Introduction and Setting
2. **Offline batch evaluation using models**
3. Offline batch evaluation using Q functions
4. Offline batch evaluation using importance sampling
5. Safe batch RL

# Learn Dynamics and Reward Models from Data, Evaluate Policy



$$V^\pi \approx (I - \gamma \hat{P}^\pi)^{-1} \hat{R}^\pi$$

$$P^\pi(s'|s) = p(s'|s, \pi(s))$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

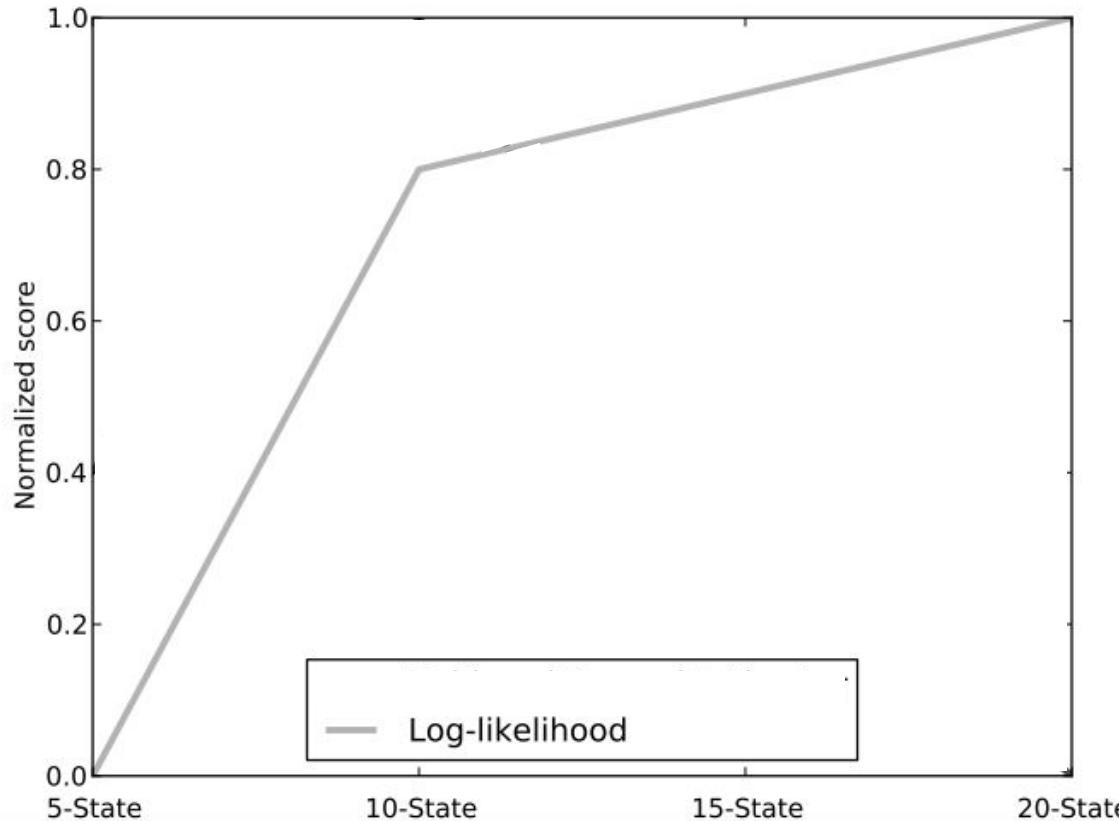
$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

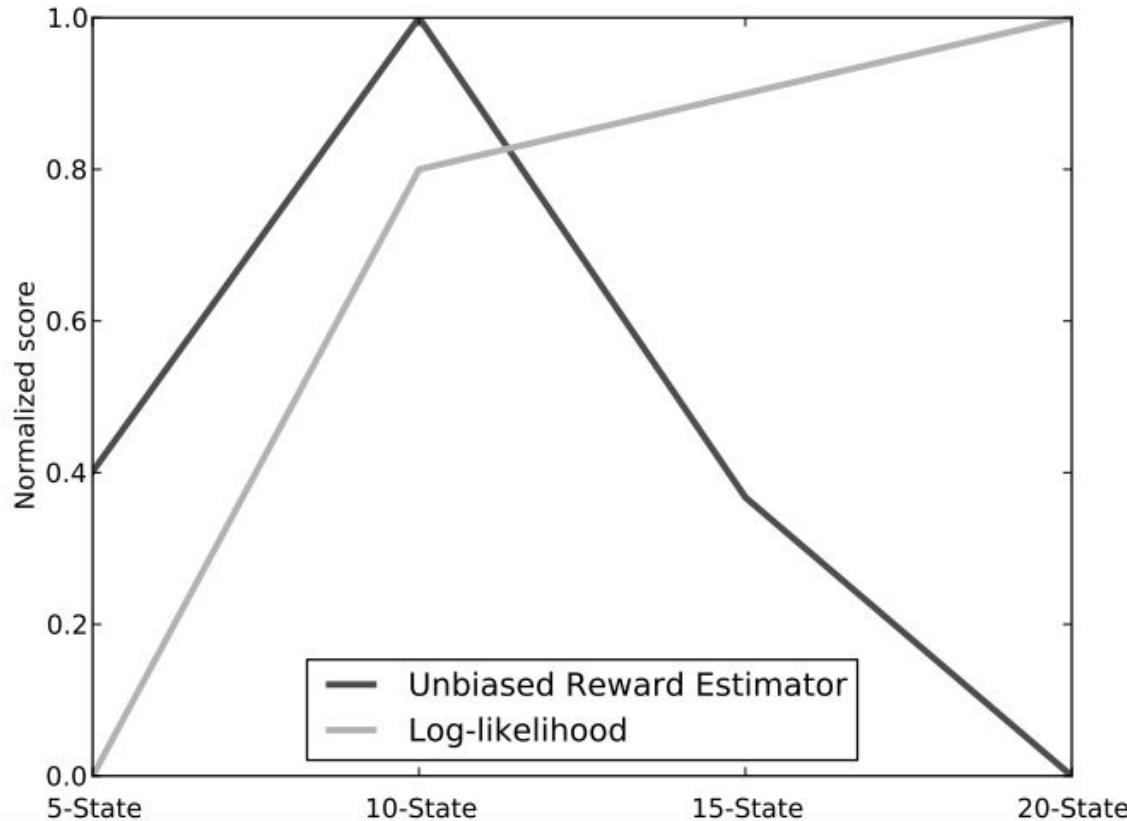
$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

- Mannor, Simster, Sun, Tsitsiklis 2007

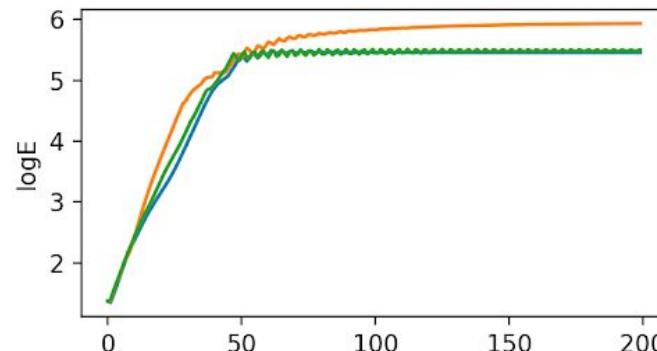
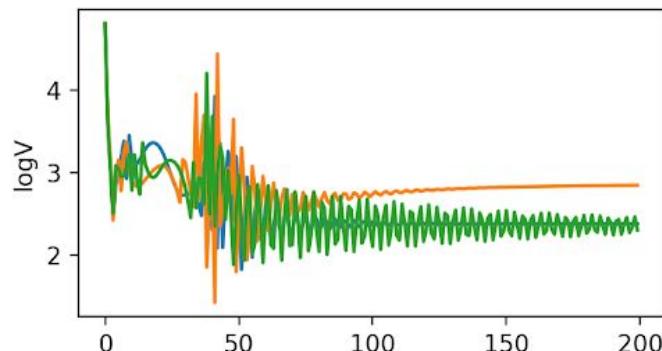
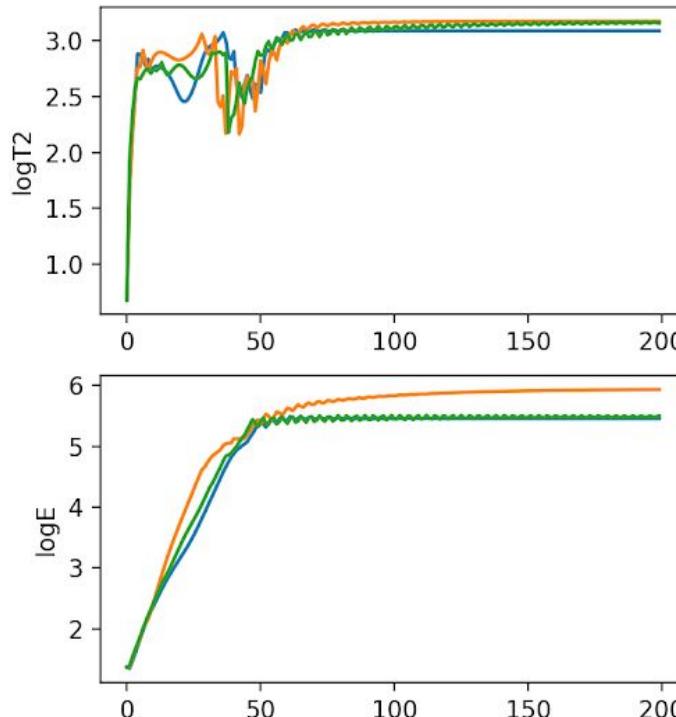
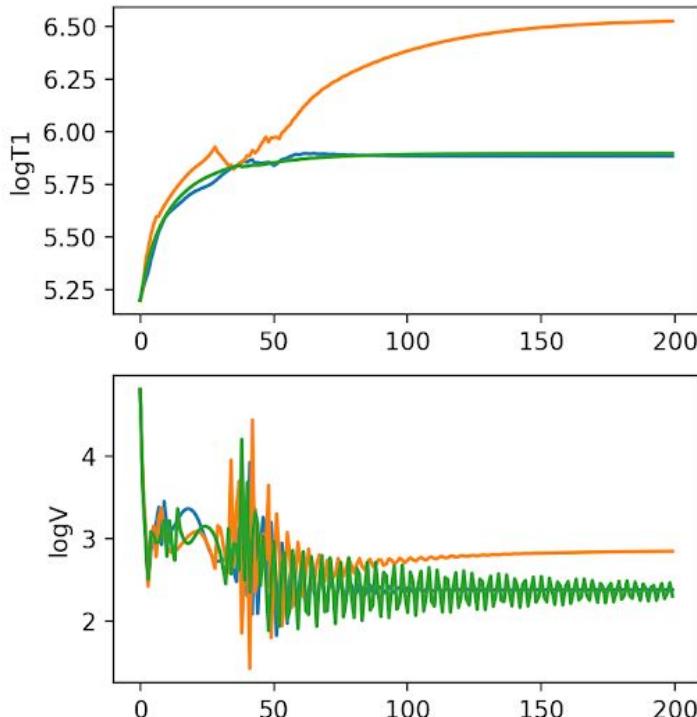
# Better Dynamics/Reward Models for Existing Data (Improve likelihood)



# Better Dynamics/Reward Models for Existing Data, May **Not** Lead to Better Policies for Future Use → Bias due to Model Misspecification



# Models Fit for Off Policy Evaluation Can Result in Better Estimates When Trained Under a **Different Loss Function**



— RepBM — MLE Model — Ground truth

# Outline

1. Introduction and Setting
2. Offline batch evaluation using models
3. **Offline batch evaluation using Q functions**
4. Offline batch evaluation using importance sampling

# Model Free Value Function Approximation

$$\mathcal{D} = (s_i, a_i, r_i, s_{i+1}) \quad \forall i$$

$$\tilde{Q}^\pi(s_i, a_i) = r_i + \gamma V_\theta^\pi(s_{i+1})$$

$$\arg \min_{\theta} \sum_i (Q_\theta^\pi(s_i, a_i) - \tilde{Q}^\pi(s_i, a_i))^2$$

- Fitted Q evaluation, LSTD, ...

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Example Fitted Q Evaluation Guarantees

$$d_F^\pi = \sup_{g \in F} \inf_{f \in F} \|f - B^\pi g\|_\pi$$

**Theorem 4.2** (Generalization error of FQE). *Under Assumption 1, for  $\epsilon > 0$  &  $\delta \in (0, 1)$ , after  $K$  iterations of Fitted Q Evaluation (Algorithm 3), for  $n = O\left(\frac{\bar{C}^4}{\epsilon^2} (\log \frac{K}{\delta} + \dim_F \log \frac{\bar{C}^2}{\epsilon^2} + \log \dim_F)\right)$ , we have with probability  $1 - \delta$ :*

$$\left| \int_{s_0 \in \rho} \hat{V}^\pi(s_0) - V^\pi(s_0) \right| \leq \frac{\gamma^5}{(1 - \gamma)^{1.5}} \left( \sqrt{\beta_u} (2d_F^\pi + \epsilon) + \frac{2\gamma^{K/2}\bar{C}}{(1 - \gamma)^{.5}} \right)$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Model Free Policy Evaluation

- Challenge: still relies on Markov assumption
- Challenge: still relies on models being well specified or have no computable guarantees if there is misspecification

# Outline

1. Introduction and Setting
2. Offline batch evaluation using models
3. Offline batch evaluation using Q functions
4. **Offline batch evaluation using importance sampling**

# Off Policy Evaluation With Minimal Assumptions

- Would like a method that doesn't rely on models being correct or Markov assumption
- Monte Carlo methods did this for online policy evaluation
- We would like to do something similar
- Challenge: data distribution mismatch

# Computing Expected Return Under a Distribution

$$\mathbb{E}_p[r] = \sum_x p(x)r(x)$$

# Computing Expected Return Under a Alternate Distribution: Simple Idea

$$\mathbb{E}_p[r] = \sum_x p(x)r(x)$$

# Computing Expected Return Under a Alternate Distribution: Simple Idea, Worked Example

	Arm 1	Arm 2	Arm 3	Arm 4
Gaussian mean	10	1	0	0.5
Behavior policy q	0.2	0.5	0.15	0.15
Evaluation policy p	0.8	0.2	0	0
Num samples from behavior q	20	50	15	15

# Why Did This Fail?

	Arm 1	Arm 2	Arm 3	Arm 4
Gaussian mean	10	1	0	0.5
Behavior policy q	0.2	0.5	0.15	0.15
Evaluation policy p	0.8	0.2	0	0
Num samples from behavior q	20	50	15	15

# Importance Sampling

$$\mathbb{E}_p[r] = \sum_x p(x)r(x)$$

# Importance Sampling: Can Compute Expected Value Under An Alternate Distribution!

$$\begin{aligned}\mathbb{E}_p[r] &= \sum_x p(x)r(x) \\ &= \sum_x \frac{p(x)q(x)}{q(x)}r(x) \\ &\approx \frac{1}{N} \sum_{i=1, x \sim q}^N \frac{p(x_i)}{q(x_i)}r(x_i)\end{aligned}$$

Importance Sampling is an Unbiased Estimator of True Expectation Under Desired Distribution If

$$\begin{aligned}\mathbb{E}_p[r] &= \sum_x p(x)r(x) \\ &= \sum_x \frac{p(x)q(x)}{q(x)}r(x) \\ &\approx \frac{1}{N} \sum_{i=1, x \sim q}^N \frac{p(x_i)}{q(x_i)}r(x_i)\end{aligned}$$

- The sampling distribution  $q(x) > 0$  for all  $x$  s.t.  $p(x) > 0$  (Coverage / overlap)
- No hidden confounding

# Importance Sampling (IS) Example

$$\begin{aligned}\mathbb{E}_p[r] &= \sum_x p(x)r(x) \\ &= \sum_x \frac{p(x)q(x)}{q(x)} r(x) \\ &\approx \frac{1}{N} \sum_{i=1, x \sim q}^N \frac{p(x_i)}{q(x_i)} r(x_i)\end{aligned}$$

	Arm 1	Arm 2	Arm 3	Arm 4
Gaussian mean	10	1	0	0.5
Behavior policy q	0.2	0.5	0.15	0.15
Evaluation policy p	0.8	0.2	0	0
Num samples from behavior q	20	50	15	15

# Importance Sampling (IS) Example

$$\begin{aligned}\mathbb{E}_p[r] &= \sum_x p(x)r(x) \\ &= \sum_x \frac{p(x)q(x)}{q(x)} r(x) \\ &\approx \frac{1}{N} \sum_{i=1, x \sim q}^N \frac{p(x_i)}{q(x_i)} r(x_i)\end{aligned}$$

	Arm 1	Arm 2	Arm 3	Arm 4
Gaussian mean	10	1	0	0.5
Behavior policy q	0.2	0.5	0.15	0.15
Evaluation policy p	0.8	0.2	0	0
Num samples from behavior q	20	50	15	15

X = arms

Expected reward for following behavior policy?  $0.2*10 + 0.5*1 + 0*0.15 + 0.15*.5$

Expected reward for target policy p?  $0.8*10 + 0.2*1 = 8.2$

Computing expected reward for p using IS:  $(20/100) * (.8/.2) * 10 + (50/100)*(2/.5)*1 = 8.2$

# Check Your Understanding: Importance Sampling

We can use importance sampling to do batch bandit policy evaluation. Consider we have a dataset for pulls from 3 arms. Consider that arm 1 is a Bernoulli where with probability .98 we get 0 and with probability 0.02 we get 100. Arm 2 is a Bernoulli where with probability 0.55 the reward is 2 else the reward is 0. Arm 3 has a probability of yielding a reward of 1 with probability 0.5 else it gets 0. Select all that are true.

- Data is sampled from  $\pi_1$  where with probability 0.8 it pulls arm 3 else it pulls arm 2. The policy we wish to evaluate,  $\pi_2$ , pulls arm 2 with probability 0.5 else it pulls arm 1.  $\pi_2$  has higher true reward than  $\pi_1$ .
- We cannot use  $\pi_1$  to get an unbiased estimate of the average reward  $\pi_2$  using importance sampling.
- If rewards can be positive or negative, we can still get a lower bound on  $\pi_2$  using data from  $\pi_1$  using importance sampling
- Now assume  $\pi_1$  selects arm1 with probability 0.2 and arm2 with probability 0.8. We can use importance sampling to get an unbiased estimate of  $\pi_2$  using data from  $\pi_1$ .
- Still with the same  $\pi_1$ , it is likely with  $N=20$  pulls that the estimate using IS for  $\pi_2$  will be higher than the empirical value of  $\pi_1$ .
- Not Sure

# Check Your Understanding: Importance Sampling Answers

We can use importance sampling to do batch bandit policy evaluation. Consider we have a dataset for pulls from 3 arms. Consider that arm 1 is a Bernoulli where with probability .98 we get 0 and with probability 0.02 we get 100. Arm 2 is a Bernoulli where with probability 0.55 the reward is 2 else the reward is 0. Arm 3 has a probability of yielding a reward of 1 with probability 0.5 else it gets 0. Select all that are true.

- Data is sampled from  $\pi_1$  where with probability 0.8 it pulls arm 3 else it pulls arm 2. The policy we wish to evaluate,  $\pi_2$ , pulls arm 2 with probability 0.5 else it pulls arm 1.  $\pi_2$  has higher true reward than  $\pi_1$ .  
**(True)**
- We cannot use  $\pi_1$  to get an unbiased estimate of the average reward  $\pi_2$  using importance sampling.  
**(True,  $\pi_1$  never pulls arm 1 which is taken by  $\pi_2$ )**
- If rewards can be positive or negative, we can still get a lower bound on  $\pi_2$  using data from  $\pi_1$  using importance sampling  
**(False, only if rewards are positive)**
- Now assume  $\pi_1$  selects arm1 with probability 0.2 and arm2 with probability 0.8. We can use importance sampling to get an unbiased estimate of  $\pi_2$  using data from  $\pi_1$ . (True)
- Still with the same  $\pi_1$ , it is likely with  $N=20$  pulls that the estimate using IS for  $\pi_2$  will be higher than the empirical value of  $\pi_1$ . (False)

# Importance Sampling for RL Policy Evaluation

$$V^\pi(s) = \sum_{\tau} p(\tau|\pi, s) R(\tau)$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Importance Sampling for RL Policy Evaluation

$$\begin{aligned} V^\pi(s) &= \sum_{\tau} p(\tau|\pi, s) R(\tau) \\ &= \sum_{\tau} p(\tau|\pi_b, s) \frac{p(\tau|\pi, s)}{p(\tau|\pi_b, s)} R_\tau \\ &\approx \sum_{i=1, \tau_i \sim \pi_b}^N \frac{p(\tau_i|\pi, s)}{p(\tau_i|\pi_b, s)} R_{\tau_i} \end{aligned}$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Importance Sampling for RL Policy Evaluation

$$\begin{aligned} V^\pi(s) &= \sum_{\tau} p(\tau|\pi, s) R(\tau) \\ &= \sum_{\tau} p(\tau|\pi_b, s) \frac{p(\tau|\pi, s)}{p(\tau|\pi_b, s)} R_\tau \\ &\approx \sum_{i=1, \tau_i \sim \pi_b}^N \frac{p(\tau_i|\pi, s)}{p(\tau_i|\pi_b, s)} R_{\tau_i} \\ &= \sum_{i=1, \tau_i \sim \pi_b}^N R_{\tau_i} \prod_{t=1}^{H_i} \frac{p(s_{i,t+1}|s_{it}, a_{it}) p(a_{it}|\pi, s_{it})}{p(s_{i,t+1}|s_{it}, a_{it}) p(a_{it}|\pi_b, s_{it})} \\ &= \sum_{i=1, \tau_i \sim \pi_b}^N R_{\tau_i} \prod_{t=1}^{H_i} \frac{p(a_{it}|\pi, s_{it})}{p(a_{it}|\pi_b, s_{it})} \end{aligned}$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

# Importance Sampling for RL Policy Evaluation: Don't Need to Know Dynamics Model!

$$\begin{aligned}
 V^\pi(s) &= \sum_{\tau} p(\tau|\pi, s) R(\tau) \\
 &= \sum_{\tau} p(\tau|\pi_b, s) \frac{p(\tau|\pi, s)}{p(\tau|\pi_b, s)} R_\tau \\
 &\approx \sum_{i=1, \tau_i \sim \pi_b}^N \frac{p(\tau_i|\pi, s)}{p(\tau_i|\pi_b, s)} R_{\tau_i} \\
 &= \sum_{i=1, \tau_i \sim \pi_b}^N R_{\tau_i} \prod_{t=1}^{H_i} \frac{p(s_{i,t+1}|s_{it}, a_{it}) p(a_{it}|\pi, s_{it})}{p(s_{i,t+1}|s_{it}, a_{it}) p(a_{it}|\pi_b, s_{it})} \\
 &= \sum_{i=1, \tau_i \sim \pi_b}^N R_{\tau_i} \prod_{t=1}^{H_i} \frac{p(a_{it}|\pi, s_{it})}{p(a_{it}|\pi_b, s_{it})}
 \end{aligned}$$

$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

$\pi$ : Policy mapping  $s \rightarrow a$

$S_0$ : Set of initial states

$\hat{V}^\pi(s, \mathcal{D})$ : Estimate  $V(s)$  w/dataset  $\mathcal{D}$

- First used for RL by Precup, Sutton & Singh 2000. Recent work includes: Thomas, Theocharous, Ghavamzadeh 2015; Thomas and Brunskill 2016; Guo, Thomas, Brunskill 2017; Hanna, Niekum, Stone 2019

# Importance Sampling

- Does not rely on Markov assumption
- Requires minimal assumptions
- Provides unbiased estimator
- Similar to Monte Carlo estimator but corrects for distribution mismatch

## Check Your Understanding: Importance Sampling 2

Select all that you'd guess might be true about importance sampling

- It requires the behavior policy to visit all the state--action pairs that would be visited under the evaluation policy in order to get an unbiased estimator
- It is likely to be high variance
- Not Sure

# Check Your Understanding: Importance Sampling 2 Answers

Select all that you'd guess might be true about importance sampling

- It requires the behavior policy to visit all the state--action pairs that would be visited under the evaluation policy in order to get an unbiased estimator (True)
- It is likely to be high variance (True)
- Not Sure

# Per Decision Importance Sampling (PDIS)

- Leverage temporal structure of the domain (similar to policy gradient)

$$IS(D) = \frac{1}{n} \sum_{i=1}^n \left( \prod_{t=1}^L \frac{\pi_e(a_t | s_t)}{\pi_b(a_t | s_t)} \right) \left( \sum_{t=1}^L \gamma^t R_t^i \right)$$

$$PSID(D) = \sum_{t=1}^L \gamma^t \frac{1}{n} \sum_{i=1}^n \left( \prod_{\tau=1}^t \frac{\pi_e(a_\tau | s_\tau)}{\pi_b(a_\tau | s_\tau)} \right) R_t^i$$

# Importance Sampling Variance

- Importance sampling, like Monte Carlo estimation, is generally high variance
- Importance sampling is particularly high variance for estimating the return of a policy in a sequential decision process

$$= \sum_{i=1, \tau_i \sim \pi_b}^N R_{\tau_i} \prod_{t=1}^{H_i} \frac{p(a_{it} | \pi, s_{it})}{p(a_{it} | \pi_b, s_{it})}$$

- Variance can generally scale exponentially with the horizon
  - a. Concentration inequalities like Hoeffding scale with the largest range of the variable
  - b. The largest range of the variable depends on the product of importance weights
  - c. Check your understanding: for a  $H$  step horizon with a maximum reward in a single trajectory of 1, and if  $p(a|s, \pi_b) = .1$  and  $p(a|s, \pi) = 1$  for each time step, what is the maximum importance-weighted return for a single trajectory?

$$R_{\tau_i} \prod_{t=1}^{H_i} \frac{p(a_{it} | \pi, s_{it})}{p(a_{it} | \pi_b, s_{it})}$$

# Outline

1. Introduction and Setting
2. Offline batch evaluation using models
3. Offline batch evaluation using Q functions
4. Offline batch evaluation using importance sampling
5. **Example application & What You Should Know**



COMPUTER SCIENCE

## Preventing undesirable behavior of intelligent machines

Philip S. Thomas<sup>1\*</sup>, Bruno Castro da Silva<sup>2</sup>, Andrew G. Barto<sup>1</sup>, Stephen Giguere<sup>1</sup>, Yuriv Brun<sup>1</sup>, Emma Brunskill<sup>3</sup>

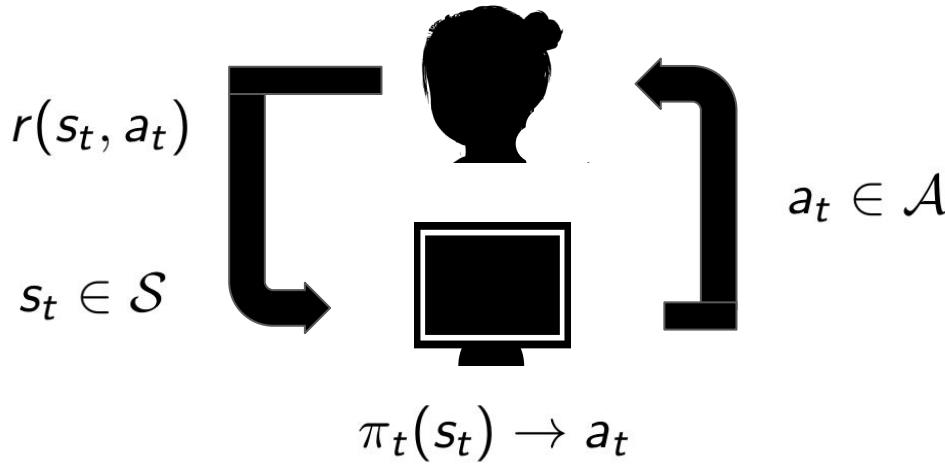
Science November 2019

# Optimizing while Ensuring Solution Won't, in the Future, Exhibit Undesirable Behavior

$$\begin{aligned} & \arg \max_{a \in \mathcal{A}} f(a) \\ \text{s.t. } & \text{s.t. } \forall i \in \{1, \dots, n\}, \Pr\left(g_i(a(D)) \leq 0\right) \geq 1 - \delta_i \end{aligned}$$

↓  
Constraints

# Counterfactual RL with Constraints on Future Performance of Policy



$\mathcal{D}$ : Dataset of  $n$  traj.s  $\tau$ ,  $\tau \sim \pi_b$

# Related Work in Decision Making

$$\arg \max_{a \in \mathcal{A}} f(a)$$

$$\text{s.t. } \forall i \in \{1, \dots, n\}, \Pr(g_i(a(D)) \leq 0) \geq 1 - \delta_i$$

- Chance constraints, data driven robust optimization have similar aims
- Most of this work has focused on ensuring computational efficiency for  $f$  and/or constraints  $g$  with certain structure (e.g. convex)
- Also need to be able to capture broader set of aims & constraints

# Batch RL with Safety Constraints

$$g(\theta) = \mathbf{E}[r'(H)|\theta_0] - \mathbf{E}[r'(H)|\theta]$$


↑  
Default policy      Potential policy

- $r'(H)$  is a function of the trajectory  $H$

# 1 Algorithm for Batch RL with Safety Constraints

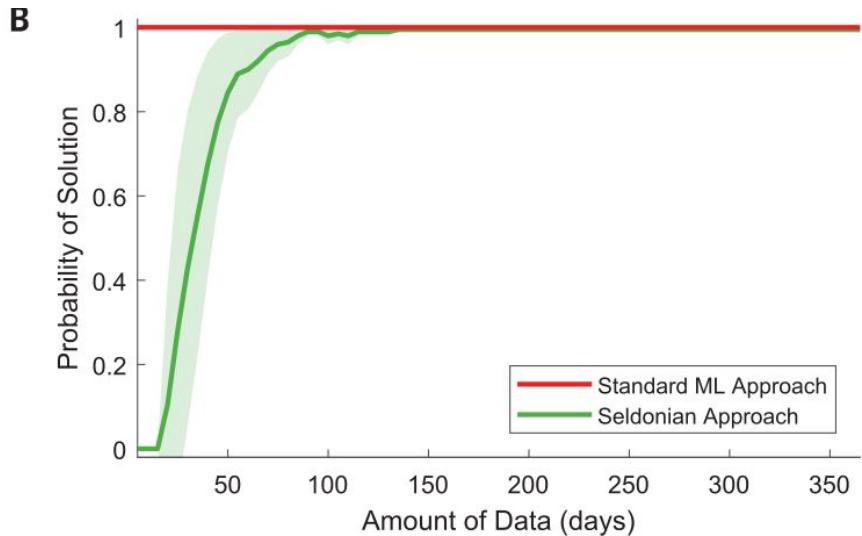
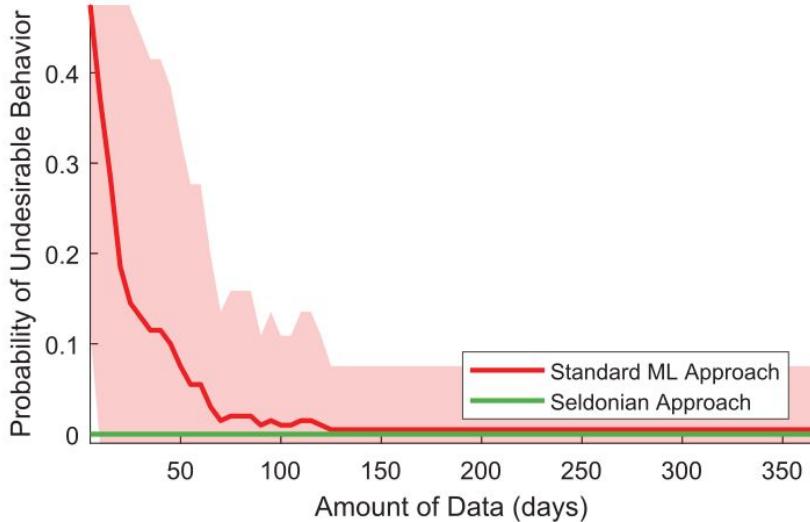
- Take in desired behavior constraints  $g$  and confidence level & data
- Given a finite set of decision policies, for each policy  $i$ 
  - Compute generalization bound for each constraint
  - If passes all with desired confidence\*,  $\text{Safe}(i) = \text{true}$
- Estimate performance  $f$  of all policies that are safe
- Return best policy that is safe, or no solution if safe set is empty

# Diabetes Insulin Management



- Blood glucose control
- Action: insulin dosage
- Search over policies
- Constraint:  
hypoglycemia
- **Very accurate  
simulator: approved by  
FDA to replace early  
stage animal trials**

# Personalized Insulin Dosage: Safe Batch Policy Improvement



# What You Should Know

- Be able to define and apply importance sampling for off policy policy evaluation
- Define some limitations of IS (variance)
- Define why we might want to do batch offline RL

# Class Progress

- Last time: Fast Reinforcement Learning
- This time: Batch RL
- Next time: Guest Lecture by Professor Doshi-Velez