Reinforcement Learning for People

Emma Brunskill
Assistant Professor, Computer Science, Stanford
Long History: RL and Decision Making for Societal Benefit

Teaching Machines
Adaptive Clinical Trials, Armitrage 1954
- Simulator of domain
- For Atari / Go huge amount of data in
- Can always try out a new strategy in domain
- No good simulator of human physiology, behavior & learning
- Gathering real data impacts real people
- We have old data about human decisions…but decisions may not be good

VS

- Simulator of domain
- For Atari / Go huge amount of data in
- Can always try out a new strategy in domain
How Could An Agent Learn To Help, As Quickly & Fairly & Accountably as Possible?
Today: Minimize Experience Need to Make Good Decisions
Two Long Known Open Challenges in RL + 1 More

1. **Exploration**: how to quickly gather information to learn to make good decisions
2. Counterfactual
3. Self-assessment
Challenge of Exploration: “Chain”*

- Act for $H$ steps, where number of states = horizon = $N$
- 2 actions (dotted line or solid line)

* [similar to e.g. Strehl & Littman 2008; Osband & Van Roy 2017]
Challenge of Exploration: “Chain”*

- Act for $H$ steps, where number of states = horizon = $N$
- 2 actions (dotted line or solid line)
- Chance reach far left “good” state, $s_N$, under random policy = $(\frac{1}{2})^N$
- Simple exploration strategies like epsilon-greedy:
  - Exponential PAC sample complexity
  - Linear regret

*[similar to e.g. Strehl & Littman 2008; Osband & Van Roy 2017]
Bandits and WWII Allied Scientists

“It proved so intractable that … the problem was proposed to be dropped over Germany so that German scientists could also waste their time on it.” -- Wikipedia
Yet Humans are Very Good at Learning to Make Good Sequences of Decisions

Tsivdis, Pouncy, Xu, Tenenbaum, Gershman. Human Learning in Atari 2017
People are Learning **Orders of Magnitude Faster** than Deep RL Agents at Each Stage of Performance

Tsiivdis, Pouncy, Xu, Tenenbaum, Gershman. Human Learning in Atari 2017
This is True in Multiple Games

Tsivdis, Pouncy, Xu, Tenenbaum, Gershman. Human Learning in Atari 2017
Tabular Episodic Reinforcement Learning

State $\times H$

Reward $\times H$

Action $\times H$

Patient 1
Episode 1

Patient 2
Episode 2

Patient 3
Episode 3

...
Minimax & Problem Dependent Bounds

For episodic RL:

- Tight theoretical bounds for 2 most common analysis frameworks [Dann, Wei, Li, B, to appear ICML 2019]
Regret: How well did the algorithm do compared to how well it could’ve done

Return

Episode \( k \)

Optimal return
Simple Exploration RL Can Fail to Learn to Make Good Decisions

No Intelligent Exploration

$O(T)$
(greedy or epsilon-greedy)

S: #states
A: #actions
T: evaluation time
H: time horizon
Strategic Exploration Can Be Much Better

Efficient Exploration

- $\tilde{O}(\sqrt{HSA{T}})$ (Azar et al. 2017)
- $\tilde{O}(S\sqrt{HAT})$ (Dann, Lattimore, B 2017)
- $\tilde{O}(H\sqrt{SAT})$ (Dann & B 2015)
- $\tilde{O}(HS\sqrt{AT})$ (UCRL2, Jaksch et al. 2010)

No Intelligent Exploration

- $O(T)$ (greedy or epsilon-greedy)

S: #states
A: #actions
T: evaluation time
H: time horizon
We Obtained Minimax Regret Bounds for Episodic Tabular Reinforcement Learning

Lower Bound

\[ \tilde{O}(\sqrt{HSAT}) \]
(Azar et al. 2017)

Efficient Exploration

\[ \tilde{O}(\sqrt{SAT}) \]
(Dann & B 2015)

\[ \tilde{O}(S\sqrt{HAT}) \]
(Dann, Lattimore, B 2017)

\[ \tilde{O}(H\sqrt{SAT}) \]
(UCRL2, Jaksch et al. 2010)

No Intelligent Exploration

\[ O(T) \]
(greedy or epsilon-greedy)

S: #states
A: #actions
T: evaluation time
H: time horizon
But Simple Exploration Sometimes Sufficient to Learn a Good Policy (Bellemare et al. ICML 2017)
Subset of RL Settings Provably Easier to Learn to Achieve High Reward

- Lower regret and/or PAC sample complexity
  - Multi-armed bandits [e.g. Bubeck & Cesa-Bianchi 2012]
  - Deterministic MDPs [Wen and Van Roy 2013]
  - Linear quadratic Gaussian [Dean, Mania, Matni, Recht, Tu. 2017]
Barrier to Democratizing RL

- Some structured settings known to be provably easier
- Most algorithms require knowledge of this structure
  - In other words, e.g. designer needs to know if problem should be modeled as a multi-armed bandit or a MDP...
Towards Problem Agnostic Algorithm with Problem Dependent Bounds

● Want RL algorithms that do not require domain knowledge
● But if the domain is structured, can identify and leverage that structure to speed learning
● To start, focus on tabular episodic stochastic MDPs
Generic RL Algorithm that Leverages Structure When Present in Tabular Episodic MDPs

Problem Dependent Analysis

Lower Bound

Efficient Exploration

No Intelligent Exploration

\( \tilde{O}(\sqrt{Q^*SAT}) \)

\( \tilde{O}(\sqrt{HSAT}) \)

\( \tilde{O}(S\sqrt{HAT}) \)

\( \tilde{O}(H\sqrt{SAT}) \)

\( \tilde{O}(HS\sqrt{AT}) \)

\( O(T) \)

(greedy or epsilon-greedy)

\( Q^* \): problem dependent constant that does not need to be known

S: #states
A: #actions
T: evaluation time
H: time horizon

(Zanette & B 2019)

(Azar et al. 2017)

(Dann & B 2015)

(UCRL2, Jaksch et al. 2010)

(Dann, Wei, Li, B. 2019)
What Helps Characterize “Hardness” Of Quickly Learning to Make Good Decisions?


- Bounds on Achieved Reward in an Episode

\[ Q^* = \max_{s,a} \mathbb{V} \text{ar}_{s' \sim p(\cdot|s,a)} V^*(s') \]

\[ G = \max_{s_0,a_0,\ldots,a_{H-1},s_h} \sum_{i=1}^{H} r_i \]
Main Result

An algorithm with a (high probability) regret bound:

\[ \tilde{O}\left( \min \left[ \sqrt{Q^* SAT}, \sqrt{\frac{G^2}{H}} SAT \right] + S^{1.5} AH^2(H + \sqrt{S}) \right) \]

\[ Q^* = \max_{s,a} Var_{s' \sim p(s'|s,a)} V^*(s') \]

\[ G = \max_{s_0, a_0, \ldots, a_{H-1}, s_H} \sum_{i=1}^{H} r_i \]
Main Result

An algorithm with a (high probability) regret bound:

$$\tilde{O}\left(\min\left[\sqrt{Q^*SAT}, \sqrt{\frac{G^2}{H}SAT}\right] + S^{1.5}AH^2(H + \sqrt{S})\right)$$

Problem dependent constants $G$ & $Q^*$

Algorithm is not given $G$ or $Q^*$

Minimax Optimality in dominant terms

$$Q^* = \max_{s,a} \text{Var}_{s' \sim p(s'|s,a)} V^*(s')$$

$$G = \max_{s_0, a_0, \ldots, a_{H-1}, s_h} \sum_{i=1}^{H} r_i$$
Vs Prior Instance Dependent Bounds

Our result: matches minimax worst case, doesn’t require mixing
**Vs Prior Instance Dependent Bounds**

Worst-case bound not competitive

**Mixing Domains Only**

- (Talebi et al, 2018)
- (Maillard et al, 2014)
- (Ortner, 2018)

Our result: matches minimax worst case, doesn’t require mixing

Our result: doesn’t require us to know span, tighter S dependence

\[ \tilde{O}(HS\sqrt{AT}) \]

[REGAL] (Bartlett et al, 2010)
[SCAL] (Fruit et al, 2018)

Intuition: H upper bounds the value function (it scales the problem)

Idea: if the span is know pass this info to the algorithm

\[ \tilde{O}(\text{span}(V^*)S\sqrt{AT}) \]

Need to know domain properties
Regret: \( \tilde{O} \left( \min \left[ \sqrt{Q^* SAT}, \sqrt{\frac{G^2}{H} SAT} \right] + S^{1.5} AH^2 (H + \sqrt{S}) \right) \)

\[
| Q^*(s, a) - \hat{Q}^*(s, a) | = | p(s, a)^\top V^* - \hat{p}(s, a)^\top \hat{V}^* | \\
(Assuming no reward error)
Regret: \( \tilde{O}\left(\min\left[\sqrt{Q^* SAT}, \sqrt{\frac{G^2}{H} SAT}\right] + S^{1.5} AH^2(H + \sqrt{S})\right) \)

\[ |Q^*(s, a) - \hat{Q}^*(s, a)| = |p(s, a)^T V^* - \hat{p}(s, a)^T \hat{V}^*| \lesssim \frac{H}{\sqrt{n}} \]

(Assuming no reward error) (Hoeffding Inequality)
Regret:

$$\tilde{O} \left( \min \left[ \sqrt{Q^*SAT}, \sqrt{\frac{G^2}{H} SAT} \right] + S^{1.5} AH^2(H + \sqrt{S}) \right)$$

$$|Q^*(s, a) - \hat{Q}^*(s, a)| = |p(s, a)^\top V^* - \hat{p}(s, a)^\top \hat{V}^*| \lesssim \frac{H}{\sqrt{n}}$$  \hspace{1cm} \text{(Hoeffding Inequality)}

(Assuming no reward error)

\[ \lesssim \frac{\sigma_{y^*}^{s,a}}{\sqrt{n}} + \frac{H}{n} \]  \hspace{1cm} \text{(Bernstein Inequality)}

$$\sigma_{s,a}^{V^*} = \text{Var}_{s \sim p(s,a)} V^*$$
Regret: \( \tilde{O} \left( \min \left[ \sqrt{Q^* SAT}, \sqrt{\frac{G^2}{H} SAT} \right] + S^{1.5} AH^2(H + \sqrt{S}) \right) \)

\( |Q^*(s, a) - \hat{Q}^*(s, a)| = |p(s, a)^T V^* - \hat{p}(s, a)^T \hat{V}^*| \lesssim \frac{H}{\sqrt{n}} \)  

(Assuming no reward error)

\( \sigma^V_{s,a} \)  

\( \forall n \frac{\sigma^V_{s,a}}{\sqrt{n}} + \frac{H}{n} \)  

(Bernstein Inequality)

\( \sigma^V_{s,a} = \text{Var}_{s^+ \sim p(s,a)} V^* \)

\( Q^* = \max_{s,a} \text{Var}_{s' \sim p(s'|s,a)} V^*(s') \)
Regret: \( \tilde{O} \left( \min \left[ \sqrt{Q^* SAT}, \sqrt{\frac{G^2}{H} SAT} \right] + S^{1.5} A H^2 (H + \sqrt{S}) \right) \)

\[
| Q^*(s, a) - \hat{Q}^*(s, a) | = | p(s, a)^\top V^* - \hat{p}(s, a)^\top \hat{V}^* | \lesssim \frac{H}{\sqrt{n}}
\]

(Assuming no reward error)

(Hoeffding Inequality)

\[
\mathbb{E}_{s' \sim p(s, a)} V^*(s) + \epsilon \quad 2\epsilon \quad H
\]

(Bernstein Inequality)

\[
\sigma_{s, a}^V = \text{Var}_{s' \sim p(s, a)} V^*
\]

\[
Q^* = \max_{s, a} \text{Var}_{s' \sim p(s' | s, a)} V^*(s')
\]
Regret: \( \tilde{O} \left( \min \left[ \sqrt{Q^* SAT}, \sqrt{\frac{G^2}{H} SAT} \right] + S^{1.5} AH^2 (H + \sqrt{S}) \right) \)

\[
| Q^*(s, a) - \hat{Q}^*(s, a) | = | p(s, a)^T V^* - \hat{p}(s, a)^T \hat{V}^* | \leq \frac{H}{\sqrt{n}}
\]

(Assuming no reward error)

\[
\forall (s, a) \quad \sigma_{s,a}^{V^*} = \sqrt{\text{Var}_{s' \sim p(s, a)} V^*(s)}
\]

(Bernstein Inequality)

If this was known, this would replace H in regret bound
Revisiting Classic “Hard” Exploration Task

\[
r = \frac{1}{4N}
\]

<table>
<thead>
<tr>
<th></th>
<th>Jaksch, Ortner, Auer 2010</th>
<th>Dann, Wei, Li, B 2019; Azar et al. 2017 Minimax</th>
<th>Zanette &amp; B 2019</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generic</td>
<td>(\tilde{O}(HS\sqrt{AT}))</td>
<td>(\tilde{O}(\sqrt{HSAT}))</td>
<td>(\tilde{O}(\sqrt{Q^*SAT}))</td>
</tr>
</tbody>
</table>

*[similar to e.g. Strehl & Littman 2008; Osband & Van Roy 2017]*
Revisiting Classic “Hard” Exploration Task

\[ Q^* = \max_{s,a} \mathbb{E}_{s' \sim p(s'|s,a)} V^*(s') \approx \frac{1}{N} \left(1 - \frac{1}{N}\right) \leq \frac{1}{N} \]

\[ r = \frac{1}{4N} \]

\[ \begin{array}{c}
 s_1 \\
 \downarrow \\
 \frac{1}{N} \\
 \uparrow \\
 s_2 \\
 \downarrow \\
 \frac{1}{N} \\
 \uparrow \\
 s_3 \\
 \downarrow \\
 \frac{1}{N} \\
 \uparrow \\
 \ldots \\
 \downarrow \\
 \frac{1}{N} \\
 \uparrow \\
 s_{N-2} \\
 \downarrow \\
 \frac{1}{N} \\
 \uparrow \\
 s_{N-1} \\
 \downarrow \\
 \frac{1}{N} \\
 \uparrow \\
 s_N \\
 \end{array} \]

\[ r = 1 \]
Corollary I: “Hard’ Exploration RL Problems May Not Always Be As Hard As Thought They Were

\[
\begin{align*}
\hat{O}(HS\sqrt{AT}) & \quad \hat{O}(\sqrt{HSAT}) & \quad \hat{O}(\sqrt{Q^*SAT}) \\
\hat{O}(N^2\sqrt{T}) + […] & \quad \hat{O}(\sqrt{N^2T}) + […] & \quad \hat{O}(\sqrt{T}) + […] \\
\end{align*}
\]

*similar to e.g. Strehl & Littman 2008; Osband & Van Roy 2017*
Summary: Problem Dependent Episodic MDP RL

An algorithm with a (high probability) regret bound:

$$\tilde{O}\left(\min\left[\sqrt{Q^*SAT}, \sqrt{\frac{G^2}{H}SAT}\right] + S^{1.5}AH^2(H + \sqrt{S})\right)$$

Problem dependent constants $G$ & $Q^*$

Algorithm is not given $G$ or $Q^*$

Minimax Optimality in dominant terms

Gives insight into what domains are hard

$$Q^* = \max_{s,a} \mathbb{V}ar_{s' \sim p(s'|s,a)} \mathbb{V}^*(s')$$

$$G = \max_{s_0, a_0, \ldots, a_{H-1}, s_h} \sum_{i=1}^{H} r_i$$
Efficient Reinforcement Learning for Education

Lomas, Forlizzi, Poonwala, Patel, Shodhan, Patel, Koedinger & B, CHI 2016
1. Combine with other forms of structure? Gap dependence on state-action values ala bandits? Even-Dar, Mannor, Mansour 2006 have some results
2. Scaling up to continuous states
   a. Linear function approximation (best results exponential in horizon)
   b. Deep RL?
3. Can these inform new heuristics for exploration in large domains?
   a. Prior examples: count based exploration (Bellemare et al), posterior sampling approximations etc
4. Combining with multi-task / lifelong learning
   a. Sequential multi-task RL can have lower sample complexity by leveraging structure (Li & Brunskill UAI 2013, best paper RLDM 2015)
   b. Lifting to continuous high dimensional states?
Two Long Known Open Challenges in RL + 1 More

1. Exploration: how to quickly gather information to learn to make good decisions
2. Counterfactual: reasoning to best leverage existing data
3. Self-assessment
Classic RL: Tabula Rasa

observation -> The Environment

action

reward
Can’t Always Continually Experiment
But Often Have Prior Data About Decisions & Outcomes
Batch Off Policy Evaluation

Dataset of trajectories of observations, actions, & rewards gathered by $\pi_b$

Policy $\pi_e$: past observations $\rightarrow$ action

Value of Policy $\pi_e$: Expected discounted sum of rewards
Growing Interest in Causal Inference & ML

JUDEA PEARL
WINNER OF THE TURING AWARD
AND DANA MACKENZIE
THE BOOK OF WHY
THE NEW SCIENCE
OF CAUSE AND EFFECT

ESTIMATING CAUSAL EFFECTS OF TREATMENTS IN RANDOMIZED AND NONRANDOMIZED STUDIES
DONALD B. Rubin
Department of Statistics, Princeton University

A discussion of several methods for estimating causal effects, including randomized experiments, non-experimental studies (e.g., regression analysis), and instrumental variables. The methods are useful in a variety of settings, including public policy, economics, and biology.

Elements of Causal Inference
Foundations and Learning Algorithms

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf

Causal inference is a field that aims to establish the relationships between variables, and to understand the mechanisms that underlie these relationships. The goal is to make predictions and decisions based on a deeper understanding of the causal structure of the world.
Substantial Literature Focuses on 1 Binary Decision
Many Cases Making Sequence of Decisions Over Many Possibilities
Reinforcement Learning Designed for Sequential
Reinforcement Learning Struggles with Off Policy Data

- Part of “Deadly triad” [Sutton & Barto]
- Responsible for some classic examples that show where RL with function approximation can be unstable and fail to converge at all or to a good solution
Models: Good for Statistical Efficiency & Can Provide Individual Value Estimates
Models: Good for Statistical Efficiency & Can Provide Individual Value Estimates

\[ E, \sim \{\text{\tiny person}, \text{\tiny baby}, \ldots\} \left( \mathbb{V}\{\text{\tiny person}\} - \mathbb{V}\{\text{\tiny person}\} \right)^2 \]

\[ \text{vs} \]

\[ \left( \mathbb{V}\{\text{\tiny person}\} - \mathbb{V}\{\text{\tiny person}\} \right)^2 \left( \mathbb{V}\{\text{\tiny baby}\} - \mathbb{V}\{\text{\tiny baby}\} \right)^2 \]
Important Issues for Model Fitting for Policy Evaluation

- Data generation process
- Loss function
- Representation

*All the work discussed today operates under assumption of no confounding & coverage*
Data Generation Process:
Different Policies → Different Actions → Different States
Data Generation Process:
Different Policies $\rightarrow$ Different Actions $\rightarrow$ Different States
Important Issues for Model Fitting for Policy Evaluation

- Data generation process
  - State-action distribution differs between behavior decision policy and target policy
- **Loss function**
- Representation

*All the work discussed today operates under assumption of no confounding & coverage.*
Loss Choices For Model Fit

- Ignore covariate shift
  - Minimize mean squared predictive error of model under behavior policy
Loss Choices For Model Fit

- Ignore covariate shift
  - May be highly biased
- Ignore models, use importance sampling to correct for covariate shift?
Importance Sampling to Estimate Evaluation Policy

\[ V^{\pi_e}(s_0) = \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \ p(\tau_n | s_0, \pi_e) \]  

Trajectories sampled from evaluation \( \pi_e \)
Importance Sampling to Estimate Evaluation Policy

\[ V^{\pi_e}(s_0) = \frac{1}{N} \sum_{n=1}^{N} R(\tau_n)p(\tau_n|s_0, \pi_e) \]

Trajectories sampled from evaluation \( \pi_e \)

\[ = \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \frac{p(\tau_n|s_0, \pi_e)}{p(\tau_n|s_0, \pi_b)} \]

Trajectories sampled under behavior \( \pi_b \)
Importance Sampling to Estimate Evaluation Policy

\[ V^{\pi_e}(s_0) = \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) p(\tau_n | s_0, \pi_e) \]

Trajectories sampled from evaluation \( \pi_e \)

\[ = \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \frac{p(\tau_n | s_0, \pi_e)}{p(\tau_n | s_0, \pi_b)} \]

Trajectories sampled under behavior \( \pi_b \)

\[ = \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) p(s_0) \prod_{i=1}^{H-1} \frac{p(s_{n,i+1} | a_{ni}, s_{ni}) p(a_{ni} | \pi_e, s_{ni})}{p(s_{n,i+1} | a_{ni}, s_{ni}) p(a_{ni} | \pi_b, s_{ni})} \]

Don’t need to know true dynamics
But Variance Can Scale Exponentially With Horizon $H$
(e.g. Guo, Thomas, B, NeurIPS 2017)

$$V^{\pi_e}(s_0) = \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) p(\tau_n | s_0, \pi_e)$$

Trajectories sampled from evaluation $\pi_e$

$$= \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \frac{p(\tau_n | s_0, \pi_e)}{p(\tau_n | s_0, \pi_b)}$$

Trajectories sampled under behavior $\pi_b$

$$= \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) p(s_0) \prod_{i=1}^{H-1} \frac{p(s_{n,i+1} | a_{ni}, s_{ni}) p(a_{ni} | \pi_e, s_{ni})}{p(s_{n,i+1} | a_{ni}, s_{ni}) p(a_{ni} | \pi_b, s_{ni})}$$

Don’t need to know true dynamics

& Do need to know behavior $\pi_b$
Loss Choices For Model Fit

- Ignore covariate shift
  - Minimize mean squared predictive error under behavior policy
  - May be highly biased
- Use importance sampling to correct for covariate shift
  - High variance (variance can grow exponentially with horizon (e.g. Guo, Thomas, B 2017))
- Alternatives?
Important Issues for Model Fitting for Policy Evaluation

- Data generation process
- Loss function
- Representation
Fitting Representation Balancing MDP Models for Evaluating 1 RL Policy Using Batch Data

Dynamics, Reward models

Liu, Gottesman, Raghu, Komorowski, Faisal, Doshi-Velez, B NeurIPS 2018
Putting All 3 Together to Fit a Model

- Data generation process
- Loss function
- Representation
Recall Simulation Lemma
[Slight variant of Kearns & Singh 2002]

\[ \mathbb{E}_{s_0} \left[ V^\pi_M(s_0) - V^\pi_M(s_0) \right]^2 \leq 2H \sum_{t=0}^{H-1} \mathbb{E}_{s_t, a_t \sim p_M, \pi} \left[ \bar{l}_r(s_t, a_t, \hat{M}) + \bar{l}_T(s_t, a_t, \hat{M}) \right] \]

Under evaluation $\pi$

Reward error  Dynamics error
Re-Express Using Data from Behavior Policy $\mu$

$$\mathbb{E}_{s_0} \left[ V^\pi_M(s_0) - V^\pi_M(s_0) \right]^2 \leq$$

$$+ \int_S \frac{1}{p_{M,\mu}(a_{0:t} = \pi)} \left( \tilde{\ell}_r(s_t, \pi(s_t), \hat{M}) + \tilde{\ell}_T(s_t, \pi(s_t), \hat{M}) \right) p_{M,\mu}(s_t, a_{0:t} = \pi) ds_t$$

Don’t need to know behavior $\mu$
Re-Express Using Data from Behavior Policy $\mu$ & Learn a Shared Representation

$$E_{s_0} \left[ V^\pi_M(s_0) - V^\pi_M(s_0) \right]^2 \leq 2H \sum_{t=0}^{H-1} \left[ B_{\phi,t}IPM_{G_t} \left( p_{M,\mu}^{\phi,F}(z_t), p_{M,\mu}^{\phi,CF}(z_t) \right) \right]$$

$$+ \int_S \frac{1}{p_{M,\mu}(a_{0:t} = \pi)} \left( \bar{\ell}_r(s_t, \pi(s_t), \widehat{M}) + \bar{\ell}_T(s_t, \pi(s_t), \widehat{M}) \right) p_{M,\mu}(s_t, a_{0:t} = \pi) ds_t$$

Don’t need to know behavior $\mu$
Problem: Very Little Data May Match Target Policy

\[ \mathbb{E}_{s_0} \left[ V_{M}^{\pi}(s_0) - V_{M}^{\pi}(s_0) \right]^2 \leq 2H \sum_{t=0}^{H-1} \left[ B_{\phi,t}^{} IPM_{G_t} \left( p_{M,\mu}^{\phi,F}(z_t), p_{M,\mu}^{\phi,C}(z_t) \right) \right. \\
\left. + \int_{S} \frac{1}{p_{M,\mu}(a_{0:t} = \pi)} \left( \bar{\ell}_r(s_t, \pi(s_t), \hat{M}) + \bar{\ell}_T(s_t, \pi(s_t), \hat{M}) \right) p_{M,\mu}(s_t, a_{0:t} = \pi) ds_t \right] \]

May be high variance estimator
Fine tuning / Regularization

\[ MSE_\pi \leq MSE_\pi + MSE_\mu \]

MSE on original (behavior) data
Representation Balancing MDP: Model Loss

$$\mathcal{L}(\widehat{M}_\phi; \alpha_t) = \widehat{R}_\mu(\widehat{M}_\phi)$$  MSE on original (behavior) data
Representation Balancing MDP: Model Loss

\[ \mathcal{L}(\widehat{M}_\phi; \alpha_t) = \widehat{R}_\mu(\widehat{M}_\phi) \]  

MSE on original (behavior) data

\[ + \int_S \frac{1}{p_{M,\mu}(a_{0:t} = \pi)} \left( \ell_r(s_t, \pi(s_t), \widehat{M}) + \ell_T(s_t, \pi(s_t), \widehat{M}) \right) p_{M,\mu}(s_t, a_{0:t} = \pi) ds_t \]

Upper bound on error of value of desired \( \pi \)

\[ + \sum_{t=0}^{H-1} \alpha_t \text{IPM}_F \left( \widehat{p}_{M,\mu}(z_t), \widehat{p}_{M,\mu}(z_t) \right) \]
Representation Balancing MDP: Model Loss

\[ \mathcal{L}(\widehat{M}_\phi; \alpha_t) = \widehat{R}_\mu(\widehat{M}_\phi) + \text{MSE on original (behavior) data} \]

\[ + \int_S \frac{1}{p_{M,\mu}(a_{0:t} = \pi)} \left( \ell_r(s_t, \pi(s_t), \widehat{M}) + \ell_T(s_t, \pi(s_t), \widehat{M}) \right) p_{M,\mu}(s_t, a_{0:t} = \pi) ds_t \]

\[ + \sum_{t=0}^{H-1} \alpha_t \text{IPM}_F \left( \widehat{P}_{M,\mu}(z_t), \widehat{P}_{M,\mu}^{CF}(z_t) \right) \]

\[ + \frac{\mathcal{R}(\widehat{M}_\phi)}{n^{3/8}} \quad \text{Penalty on complexity of model class} \]
Model Based Batch RL Policy Evaluation

- Data generation process
  - State-action distribution differs between behavior decision policy and target policy

- Loss function
  - Account for mismatch in distributions & regularize with behavior data model

- Representation
  - Learn representation so behavior and target policy state-action distributions similar
Representation Balancing MDP: Theory & Simulator

- Have finite sample bounds on resulting error on estimated evaluation policy value
- Have models can use to simulate policy performance for any initial state
Experiment: Structured Treatment Interruption Simulator for HIV

- Ernst, Stan, Goncalves, and Wehenkel CDC 2006
- Simulator state: Infected CD4+ T-lymphocytes, number of infected macrophages, the number of free virus particles, …
- Treatment: Put on or off
- Reward is a function of cytotoxic T-lymphocytes
  - Immune response of the body to viruses
Better Estimate for HIV Simulator

<table>
<thead>
<tr>
<th></th>
<th>RepBM</th>
<th>AM</th>
<th>IS</th>
<th>PSIS</th>
<th>WPSIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td><strong>0.062</strong></td>
<td>0.067</td>
<td>0.95</td>
<td>0.273</td>
<td>0.146</td>
</tr>
</tbody>
</table>

- RepBM: Our model estimator. DR: doubly robust with RepBM
- AM: approximate model fit using max likelihood, DR(AM): doubly robust with AM
- IS: Importance sampling
- PSIS: Per state importance sample
- WPSIS: weighted PSIS
Representation Balancing Model is More Accurate
Model Based Batch RL Policy Evaluation Summary

- **Data generation process**
  - State-action distribution differs between behavior decision policy and target policy

- **Loss function**
  - Account for mismatch in distributions & regularize with behavior data model

- **Representation**
  - Learn representation so behavior and target policy state-action distributions similar

⇒ Much more accurate, individual state policy value estimates
Counterfactual / Batch Reinforcement Learning

Dataset $\sim \pi_b$ → Policy $\pi_e$ → Value of Policy $\pi_e$ → Dataset $\sim \pi_b$ → “Good” Policy $\pi_o$
Counterfactual / Batch Reinforcement Learning

Dataset ~ $\pi_b$ → Policy $\pi_e$ → Value of Policy $\pi_e$ → "Good" Policy $\pi_o$


AAAI 2015, AAAI 2016, L@S 2017, UAI 2017 (best paper award), arxiv 2019
Counterfactual / Batch Reinforcement Learning


AAAI 2015, AAAI 2016, L@S 2017, UAI 2017 (best paper award), arxiv 2019

- Applicable to educational technology (AAMAS 2014, L@S 2017), patient treatment support (NeurIPS 2018, ICML 2019), news recommendation systems (IJCAI 2016), e-commerce and many more
Used to Increase Student Persistence by +30% in Online Fractions Game with 2000 Learners
Batch Policy Optimization with Models, Values & Policies: Many Open Questions

- Data generation process
- Loss function
- Representation
Batch Policy Optimization with Models, Values & Policies: Many Open Questions

- Data generation process
- Loss function
- Representation

One of most exciting frontiers of beautiful math & real world impact

Nascent work on exploration + counterfactual reasoning/accountability
[Dimakopoulou et al. AAAI 2019, Dann, Wei, Li, B ICML 2019]
Two Long Known Open Challenges in RL + 1 More

1. Exploration: how to quickly gather information to learn to make good decisions
2. Counterfactual: reasoning to best leverage existing data
3. **Self-assessment**: quickly diagnosing when the system is insufficient
When comparing fractions we need to determine if they are equal to, greater than, or less than each other. Let’s use a procedure called cross multiplication!

1. $4 \times 3 = 12$

2. Because 10 is less than 12 we can conclude that $2/3$ is $< \frac{4}{5}$.

3. $\frac{2}{3} = \frac{10}{15}$ and $\frac{4}{5} = \frac{12}{15}$.

How does this make sense?

1. Correctly complete the sentences to the right by dragging and dropping the phrases below.

   - less than
   - numerator
   - denominators
   - 10
   - equivalent to
   - 15
   - greater than

   4/5 is _______ $2/3$ because when cross multiplying, the product on the left is _______ the product on the right. This makes sense because cross multiplying finds the _______ for the fractions that are equivalent to our first fractions and share a common denominator of _______.

Submit
Can you order the following fractions, smallest to largest?
Drag and drop fractions to reorder them.

1. \( \frac{5}{17} < \frac{5}{12} < \frac{5}{6} \)  
   - Smallest: \( \frac{5}{17} \)  
   - Middle: \( \frac{5}{12} \)  
   - Largest: \( \frac{5}{6} \)

2. \( \frac{1}{4} < \frac{1}{3} < \frac{1}{2} \)  
   - Smallest: \( \frac{1}{4} \)  
   - Middle: \( \frac{1}{3} \)  
   - Largest: \( \frac{1}{2} \)

3. \( \frac{2}{21} < \frac{3}{5} < \frac{3}{4} \)  
   - Smallest: \( \frac{2}{21} \)  
   - Middle: \( \frac{3}{5} \)  
   - Largest: \( \frac{3}{4} \)

4. \( \frac{1}{12} < \frac{1}{2} < \frac{1}{8} \)  
   - Smallest: \( \frac{1}{12} \)  
   - Middle: \( \frac{1}{2} \)  
   - Largest: \( \frac{1}{8} \)

1. Compare these fractions using the cross-multiplication strategy.

   \[ \frac{4}{5} \times \frac{10}{10} = \frac{40}{50} \]
   \[ \frac{9}{10} \times \frac{5}{5} = \frac{45}{50} \]

   - Check: \( 40 < 45 \)

2. Since 40 is less than 45, the fraction on the left is less than the one on the right. So, we know:

   \[ \frac{4}{5} \]
Can you order the following fractions, smallest to largest?
Drag and drop fractions to reorder them.

1. \( \frac{5}{17} < \frac{5}{12} < \frac{5}{6} \)
   - Check

2. \( \frac{1}{4} < \frac{1}{3} < \frac{1}{2} \)
   - Check

3. \( \frac{2}{21} < \frac{3}{5} < \frac{3}{4} \)
   - Check

4. \( \frac{1}{12} < \frac{1}{2} < \frac{1}{8} \)
   - Check

1. Compare these fractions using the cross-multiplication strategy.
   - \( \frac{4 \times 10}{5 \times 9} = \frac{40}{45} \)
   - Since 40 is less than 45, the fraction on the left is less than the one on the right. So, we know:
   - \( \frac{4}{5} < \frac{9}{10} \)

<table>
<thead>
<tr>
<th>Decision Policies</th>
<th>Expert Human Ordering 1</th>
<th>Expert Human Ordering 2</th>
<th>Automated Mastery Based</th>
<th>ML Method 1</th>
<th>ML Method 2</th>
</tr>
</thead>
</table>

+1000 4th/5th Graders, Multi-School Study And...

<table>
<thead>
<tr>
<th>Expert Human Ordering 1</th>
<th>Expert Human Ordering 2</th>
<th>Automated Mastery Based</th>
<th>ML Method 1</th>
<th>ML Method 2</th>
</tr>
</thead>
</table>

A) Can you order the following fractions, smallest to largest? Drag and drop fractions to reorder them.

1.  \[
\frac{4}{5} < \frac{5}{12} < \frac{5}{9} \]
2.  \[
\frac{1}{4} < \frac{1}{3} < \frac{1}{2} \]
3.  \[
\frac{2}{21} < \frac{3}{5} < \frac{3}{4} \]
4.  \[
\frac{1}{12} < \frac{1}{2} < \frac{1}{8} \]

B) Compare these fractions using the cross-multiplication strategy.

1. \[
\frac{4}{5} \times 10 = 40 \quad \frac{9}{10} \times 5 = 45
\]
2. Since 40 is less than 45, the fraction on the left is less than the one on the right. So, we know:

\[
\frac{4}{5} < \frac{9}{10}
\]
+1000 4th/5th Graders, Multi-School Study And...

<table>
<thead>
<tr>
<th></th>
<th>Expert Human Ordering 1</th>
<th>Expert Human Ordering 2</th>
<th>Automated Mastery Based</th>
<th>ML Method 1</th>
<th>ML Method 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Score</td>
<td>5.5 +/- 2.6</td>
<td>5.1 +/- 3.2</td>
<td>5.5 +/- 3.0</td>
<td>5.6 +/- 3.2</td>
<td>4.9 +/- 1.8</td>
</tr>
</tbody>
</table>
No Significant Difference Between Any Decision Policy For Selecting Activities, & Learning Gains Pretty Limited

<table>
<thead>
<tr>
<th></th>
<th>Expert Human Ordering 1</th>
<th>Expert Human Ordering 2</th>
<th>Automated Mastery Based</th>
<th>ML Method 1</th>
<th>ML Method 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Score</td>
<td>5.5 +/- 2.6</td>
<td>5.1 +/- 3.2</td>
<td>5.5 +/- 3.0</td>
<td>5.6 +/- 3.2</td>
<td>4.9 +/- 1.8</td>
</tr>
</tbody>
</table>
How Quickly Can We Tell if Any Decision Policy is Good Enough?

- In scientific experiments often test is A better than B
  - Power analysis for how many samples need
  - Hypothesis test
- Related but distinct question here:
- Does there exist a decision policy such that its value $\geq$ threshold $b$?
How Quickly Can We Tell if Any Decision Policy is Good Enough?

- In scientific experiments often test is A better than B
  - Power analysis for how many samples need
  - Hypothesis test
- Related but distinct question here:
- Does there exist a decision policy such that its value $\geq$ threshold $b$?
- Idea 1: Learn optimal policy, estimate its value, then answer question
- Idea 2: Maintain upper and lower bounds on optimal policy, halt if lower bound $\geq b$ or upper bound $\leq b$ [Dann, Wei, Li, B ICML 2019 could be used]
- Can we do better?
How Quickly Can We Tell if Any Decision Policy is Good Enough?

- Does there exist a decision policy $\pi$ such that its value $\geq$ threshold $b$?
- Can we identify if $\pi$ exists with less samples than need to provide such a $\pi$?
How Quickly Can We Tell if Any Decision Policy is Good Enough?

- Does there exist a decision policy $\pi$ such that its value $\geq$ threshold b?
- Can we identify if $\pi$ exists with less samples than need to provide such a $\pi$?
- Kong and Valiant (2018):
  - For classifiers, yes
How Quickly Can We Tell if Any Decision Policy is Good Enough?

- Does there exist a decision policy $\pi$ such that its value $\geq$ threshold $b$?
- Can we identify if $\pi$ exists with less samples than need to provide such a $\pi$?
- Kong and Valiant (2018):
  - For classifiers, yes
- Can we do this for K-armed bandits with d-dimensional state context?
  - Can we identify with $< O(d)$ samples, if there exists a linear threshold policy with expected value over a threshold $b$?
- Joint ongoing work with Kong, Zanette & Valiant, but preliminary results suggest yes
How Quickly Can We Tell if Any Decision Policy is Good Enough?

- Does there exist a decision policy $\pi$ such that its value $\geq$ threshold $b$?
- Can we identify if $\pi$ exists with less samples than need to provide such a $\pi$?
- Can we do this for general contextual bandits?
- Can we do this for Markov decision processes?
How Quickly Can We Tell if Any Decision Policy is Good Enough?

- This is important, because in the real world, we will not always start with an action space or a state space that is sufficient to desired result,
- But once we can diagnose this, can create better policy space
Towards Quickly Learning to Make Good Decisions

1. Exploration
   - Maybe it’s (provably) not so hard, if we use tighter optimism

2. Counterfactual
   - Choosing the right loss function to account for state-action distribution mismatch

3. Self-assessment
   - Can we diagnose no policy is good faster than we can learn a good policy
Creating AI That Helps Students

Learn 1.3x Faster

Persist +30% Longer

Learn 1.3x More
Towards Quickly Learning to Make Good Decisions

1. Exploration
   - Maybe it’s (provably) not so hard, if we use tighter optimism

2. Counterfactual
   - Choosing the right loss function to account for state-action distribution mismatch

3. Self-assessment
   - Can we diagnose no policy is good faster than we can learn a good policy

⇒ All towards continually improving agents that help people accomplish their goals