CS292F StatRL Lecture 11 Exploration in Linear MDP & Introduction to offline RL

Instructor: Yu-Xiang Wang

Spring 2021

UC Santa Barbara

Logistics

- Project midterm milestone due
  - Important as I need to allocate space for student presentation
- For those who haven’t submitted HW1
  - You don’t have to solve everything, just submit what you have
  - HW1 is long I am thinking of adjusting grading criteria
- HW2 is not as long
  - Don’t wait

Recap: Lecture 10

- Exploration in Linear MDPs
- Properties of Linear MDPs
- Algorithm: UCB-VI for Linear MDPs
- Regret analysis

Recap: Impossibility results

$s,a \to \phi(s,a)$

- What are the assumptions to make?
  - **$Q^*(s,a)$ approximately linear?**

    $\exists\, \theta$ s.t. $|\theta^T \phi(s,a) - Q^*(s,a)| \le \varepsilon$
  - **$Q^\pi(s,a)$ is approximately linear for all π?**

    for each π $\exists\, \theta^\pi$ $Q^\pi(s,a) = (\theta^\pi)^T \phi(s,a)$
  - ~~$Q^*(s,a)$ is exactly linear?~~

    Weisz et al (ALT-2020): http://proceedings.mlr.press/v132/weisz21a.html
  - $Q^\pi(s,a)$ is exactly linear for all π?

    open problem

Exponential sample complexity / regret lower bounds for the approximate case…

(Du, Kakade, Wang, Yang, 2019) Is a good representation sufficient for sample efficient reinforcement learning?

Recap: Linear MDPs

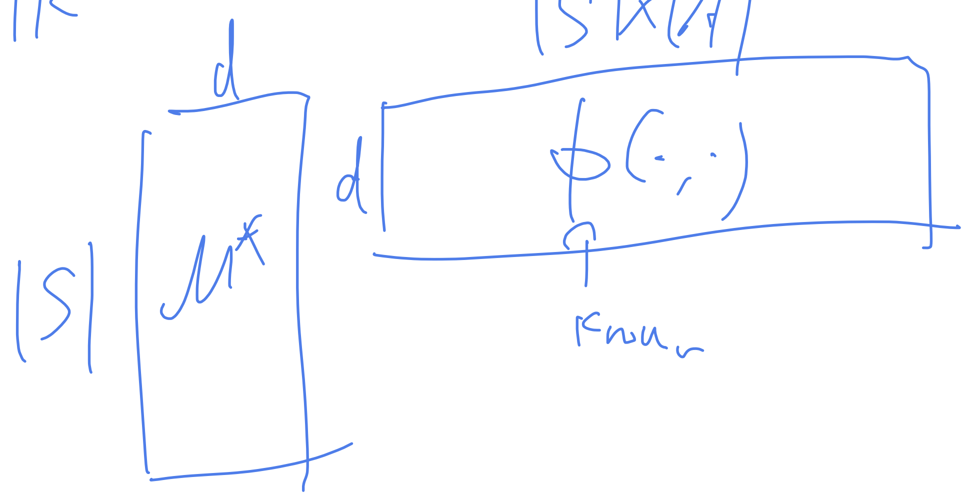
$|S|$ is exponentially large

- Exists feature map $\phi : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}^d$
  - Such that:

$$r_h(s,a) = \theta_h^\star \cdot \phi(s,a), \quad P_h(\cdot|s,a) = \mu_h^\star \phi(s,a), \forall h$$

$$P_h(\cdot|\cdot,\cdot) \in \mathbb{R}^{|S| \times |S||A|}$$

$=$



(Jin et al., 2020) Provably efficient reinforcement learning with linear function approximation

Recap: UCB-VI for Linear MDPs

- In every round:
  1. Run Ridge regression for estimating the model

$$\widehat{\mu}_h^n = \mathrm{argmin}_{\mu \in \mathbb{R}^{|S| \times d}} \sum_{i=0}^{n-1} \left\| \mu \phi(s_h^i, a_h^i) - \delta(s_{h+1}^i) \right\|_2^2 + \lambda \|\mu\|_F^2.$$

$$\widehat{\mu}_h^n = \sum_{i=0}^{n-1} \delta(s_{h+1}^i) \phi(s_h^i, a_h^i)^\top (\Lambda_h^n)^{-1}$$

  2. Construct the exploration bonuses

$$b_h^n(s, a) = \beta \sqrt{\phi(s, a)^\top (\Lambda_h^n)^{-1} \phi(s, a)},$$

  3. Run optimistic value iterations, and update greedy policy

Recap: Regret bound

- Choose

$$\beta = Hd\boxed{\left(\sqrt{\ln \frac{H}{\delta}} + \sqrt{\ln(W+H)} + \sqrt{\ln B} + \sqrt{\ln d} + \sqrt{\ln N}\right)}$$

$$\lambda = 1$$

$$\Lambda_h^n = \sum_i \phi(s_h^i, a_h^i)\,\phi(s_h^i, a_h^i)^\top + \lambda I \in \mathbb{R}^{d\times d}$$

$$b_h^n(s,a) = \beta \underbrace{\sqrt{\phi(s,a)^\top (\Lambda_h^n)^{-1} \phi(s,a)}}_{\|\phi(s,a)\|_{(\Lambda_h^{n})^{-1}}},$$

- Regret

$$\tilde{O}\left(H^2\sqrt{d^3 N}\right)$$

# of episodes

Recap: Regret analysis

total regret

$$= \sum_{n=0}^{N-1} \beta \cdot b_h^n(S_h^{(i)}, A_h^{(i)})$$

$$\le \beta \sqrt{N \sum (b_h^n(S_h, A_h))^2}$$

- Regret of episode t n

Per-episode Regret:

$$V^* - V^{\pi_n} = V_0^*(s_0) - V_0^{\pi_n}(s_0) \overset{\text{optimism}}{\le} \hat{V}_0^{\pi_n}(s_0) - V_0^{\pi_n}(s_0) \le \sum_{h=0}^{H-1} E^{\pi_n} b_h^n(S_h, A_h)$$

Simulation Lemma $\rightarrow$ $\le \sum_{h=0}^{H-1} E^{\pi_n}\left[b_h^n(S_h, A_h) + \left(\hat{P}_h^n(\cdot|S_h,A_h) - P_h(\cdot|S_h,A_h)\right) \cdot \hat{V}_{h+1} \right]$

data dependent

  - Optimism / simulation lemma
- Sum them up to get total regret

Lemma (Information Gain bound)

$\forall S_h^n, A_h^n$ sequence $\sum_{n=0}^{N-1} \phi(S_h^n, A_h^n)^T (\Lambda_h^n)^{-1} \phi(S_h^n, A_h^n) = \tilde{O}(d \log N)$

  - Same information-gain bound from linear bandits

$$-P(\cdot|s,a) + \hat{P}(\cdot|s,a) = \hat{\mu}\cdot\phi(s,a) - \mu^{*}\cdot\phi(s,a)$$

Recap: It remains to prove

$$f : S \to R$$

$$\left(\hat{P}(\cdot|s,a) - P(\cdot|s,a)\right)^{\top} f(\cdot)$$

$$= \sum_{s'} \left(\hat{P}(s'|s,a) - P(s'|s,a)\right) f(s')$$

- 1. Uniform convergence bound

$$\boxed{\sup_{f(\cdot)} \left| \left(\hat{P}(\cdot|s,a) - P(\cdot|s,a)\right)\cdot f \right| \le \epsilon}$$

- 2. “Optimism”

The same induction argument as in the UCB-VI for tabular MDP
(Read Lemma 7.10 in AJKS)

- 3. “Information gain” bound

The same argument as in the Linear Bandits case.
(Read Lemma 7.12 in AJKS)

Recap: Bound for a fixed V

$\epsilon_h^i = \delta_{s_{h+1}^i} - E[\delta_{s_{h+1}^i} \mid s_h^i, a_h^i]$

- Lemma 7.3 AJKS

$$\widehat{\mu}_h^n - \mu_h^\star = -\lambda \mu_h^\star \left(\Lambda_h^n\right)^{-1} + \sum_{i=1}^{n-1} \epsilon_h^i \phi(s_h^i, a_h^i)^\top \left(\Lambda_h^n\right)^{-1}.$$

$\Lambda_0 = \lambda I$

- The quantity of interest is a inner product with this:

$$(*) = \left[(\hat{\mu}_h^n - \mu_h^n)\cdot\phi(s,a)\right]^T V = \phi(s,a)^T (\hat{\mu}_h^n - \mu_h^n)^T \cdot V$$

$$= \phi(s,a)^T (\Lambda_h^n)^{-1}\left[-\lambda \mu_h^* + \sum_i \phi(s_h^i, a_h^i)\, \epsilon_i^T\right] V$$

d, |S|, |S|

$$|(*)| \le \|\phi(s,a)\|_{(\Lambda_h^n)^{-1}} \|-\lambda \mu_h^* \cdot V\|_{(\Lambda_h^n)^{-1}} + \|\phi(s,a)\|_{(\Lambda_h^n)^{-1}} \left\|\sum_{i=1}^{n-1} \phi(s_h^i, a_h^i)\cdot \epsilon_i^T V\right\|_{(\Lambda_h^n)^{-1}}$$

bias

Λ^0

$\|V\|_\infty \le H$

$\lambda \cdot \frac{1}{\lambda}\ \|\mu_h^* V\|_2 \le \sqrt{d}\cdot H$

$\le 3H\sqrt{\log \det(\Lambda_h^{n-1}) / \det(\Lambda_0)}$

$\le 3H\sqrt{d \log N}$

Challenge: we cannot use union bound because we have an infinite number of value functions

$\sup_{f \in \mathcal{F}} \left\| \sum \phi(s_h^i, a_h^i) \cdot \epsilon_i^T f \right\|_{(\Lambda_h^{t})^{-1}}$

- A covering number argument.

$\forall f \in \mathcal{F}$ $\exists \tilde{f} \in \mathcal{F}_\epsilon$ s.t. $\|f - \tilde{f}\|_\infty \le \epsilon$

$\sup_x |f(x) - \tilde{f}(x)| \le \epsilon$

- Covering number: the number of balls with radius ε that is needed to cover all points in a set.

N_ϵ = # of balls needed to cover $\mathcal{F}$ s.t. $|f - \tilde{f}| \le \epsilon$ $\forall f \ \exists \tilde{f} \in \mathcal{F}_\epsilon$

Family of value functions we consider

$\hat{V}_h = \max_a \left(b_h(s,a) + \phi(s,a)^\top \hat{\mu}^{*\top} \hat{V}_{h+1} \right) \in \mathbb{R}^d$

$$f_{w,\beta,\Lambda}(s) = \min\left\{ \max_a \left(w^\top \phi(s,a) + \underbrace{\beta \sqrt{\phi(s,a)^\top \Lambda^{-1} \phi(s,a)}}_{b_h} \right),\ H \right\}, \forall s \in \mathcal{S}.$$

$$\mathcal{F} = \{ f_{w,\beta,\Lambda} : \|w\|_2 \le L, \beta \in [0, B], \sigma_{\min}(\Lambda) \ge \lambda \}.$$

$\hat{V} \in \mathcal{F}$

What is a finite set to cover this class such that for every f in this set, there is a function in the finite set, such that they are ε-close in sup-norm?

$\boxed{\text{Lemma: } N_\varepsilon\left(\{ x \in \mathbb{R}^d \mid \|x\|_2 \le B \}\right) = O\left(\left(\frac{B}{\varepsilon}\right)^d\right)}$

$\hat{\mathcal{F}}_\varepsilon = W_{\frac{\varepsilon}{3}} \times B_{\frac{\varepsilon}{3}} \times \Lambda^{-1}_{\frac{\varepsilon}{3}}$ $\quad \|\phi_{s,a}\|_2 \le 1$

Covering number calculations

$f \in \mathcal{F}(w, \beta, \Lambda) \qquad \hat{f} \in \hat{\mathcal{F}}_\varepsilon \qquad (\hat{w}, \hat{\beta}, \hat{\Lambda})$

$$|f(s) - \hat{f}(s)| \le \left| \max_a \left(w^T \phi(s,a) + \beta \sqrt{\phi_{s,a}^T \Lambda^{-1} \phi_{s,a}} \right) - \max_a \left(\hat{w} \phi(s,a) + \hat{\beta} \sqrt{\phi(s,a)^T \hat{\Lambda}^{-1} \phi(s,a)} \right) \right|$$

$$\le \max_a \left| (w - \hat{w})^T \phi(s,a) \right| + \max_a \left| (\beta - \hat{\beta}) \sqrt{\phi_{s,a}^T \Lambda^{-1} \phi_{s,a}} \right| + \max_a \left| \hat{\beta} \left(\sqrt{\phi(s,a)^T \Lambda^{-1} \phi(s,a)} - \sqrt{\phi_{s,a}^T \hat{\Lambda}^{-1} \phi_{s,a}} \right) \right|$$

$$\le \underbrace{\|w - \hat{w}\|_2}_{\le \frac{\varepsilon}{3}} + \underbrace{\frac{|\beta - \hat{\beta}|}{\sqrt{\lambda}}}_{\le \frac{\varepsilon}{3}} + \underbrace{\beta \sqrt{\|\Lambda^{-1} - \hat{\Lambda}^{-1}\|_F}}_{\frac{\varepsilon}{3}}$$

$$\le \varepsilon$$

$N_\varepsilon = |\hat{\mathcal{F}}_\varepsilon| = |W_{\frac{\varepsilon}{3}}| \cdot |B_{\frac{\varepsilon}{3}}| \cdot |\Lambda^{-1}_{\frac{\varepsilon}{3}}|$

$N_\varepsilon(\mathcal{F}) \le \left(\frac{1}{\varepsilon}\right)^{3d}$

$\hat{\mathcal{F}}_\varepsilon = \{ f_{w, \beta, \Lambda} \text{ s.t. } | w \in W_{\frac{\varepsilon}{3}}, \beta \in B_{\frac{\varepsilon}{3}}, \Lambda^{-1} \in \Lambda^{-1}_{\frac{\varepsilon}{3}} \}$

$x^T A x = \mathrm{tr}(x^T A x) = \mathrm{tr}(A x x^T) \le \langle A, x x^T \rangle \le \|A\|_F \|x x^T\|_F$

From covering number to a uniform convergence bound

$$\sup_{f\in\mathcal{F}} \left\|\sum \phi(s_i,a_i)\cdot \varepsilon_i^T f\right\|_{(\Lambda_n^n)^{-1}} \le \sup_{f\in\mathcal{F}} \left\|\sum \phi_{s_i,a_i}\varepsilon_i^T(f-\check f_f+\check f_f)\right\|_{(\Lambda_n^n)^{-1}}$$

$$\le \sup_f \left\|\sum \phi_{s_i a_i}\underline{\varepsilon_i^T(f-\check f_f)}\right\|_{\Lambda^{-1}} + \underline{\sup_f \left\|\sum \phi_{s_i a_i}\varepsilon_i^T \check f_f\right\|_{\Lambda^{-1}}}$$

$$\le \underbrace{2\varepsilon\cdot\left\|\sum \phi_i\right\|_{\Lambda^{-1}}}_{\le 2\varepsilon N} + \sup_{\check f}\left\|\sum \phi_{s_i a_i}\varepsilon_i^T \check f\right\|_{\Lambda^{-1}}$$

$$\left\|\sum_i \phi_i\right\|_{\Lambda^{-1}} \le \sum_i \sqrt{\phi_i^T\Lambda^{-1}\phi_i} \le \sqrt{N\sum_i \boxed{\phi_i^T\Lambda^{-1}\phi_i}} \le N$$

$$\Lambda_i = \sum_{j=1}^N \phi_{ij}\phi_{ij}^T + \lambda I$$

$$\boxed{\begin{aligned} &\varepsilon_i = \delta_i - \mathbb{E}\delta_i \\ &\|\varepsilon_i\|_1 \le 2 \\ &\|f-\check f\|_\infty \le \varepsilon \end{aligned}}$$

1. apply pointwise result for fixed $\check f$
2. apply union bound

$$\tilde{O}\left(\sqrt{d\log N + \log\frac{N_\varepsilon}{\delta}}\right)$$

Final notes about linear MDPs

- A semi-parametric model
  - The number of parameters to describe the model can be exponentially large: d S describe μ^*
  - Efficient algorithm with regret independent to S

- Still very strong assumption on the feature map
  - Interesting open problems:
    - Representation learning ϕ is unknow
    - Nonlinear parametric models
    - Suboptimal rates when naively applying to the tabular case

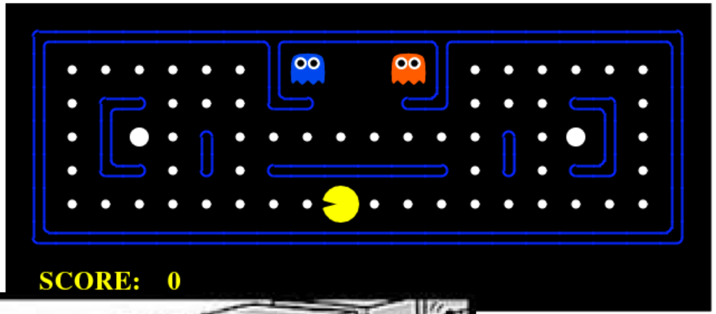
$d = S$ $\quad d\sqrt{T}$ but $\sqrt{ST}$

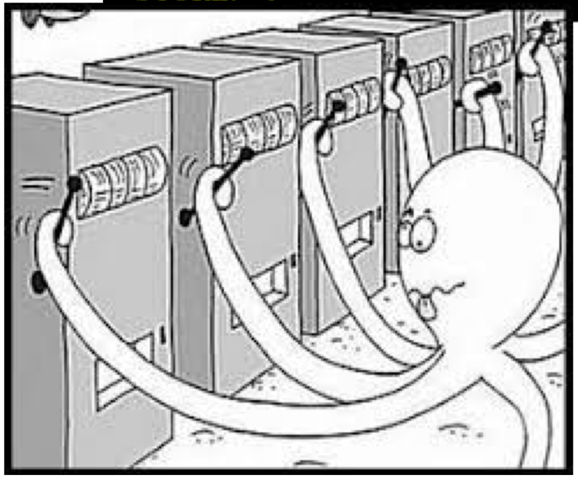
Remainder of the lecture

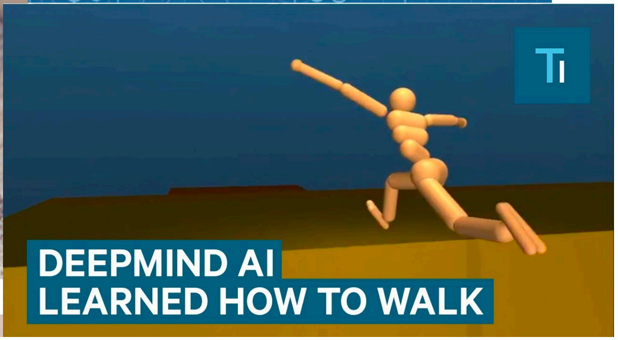
- Introduction to offline reinforcement learning

- Off-policy evaluation in contextual bandits

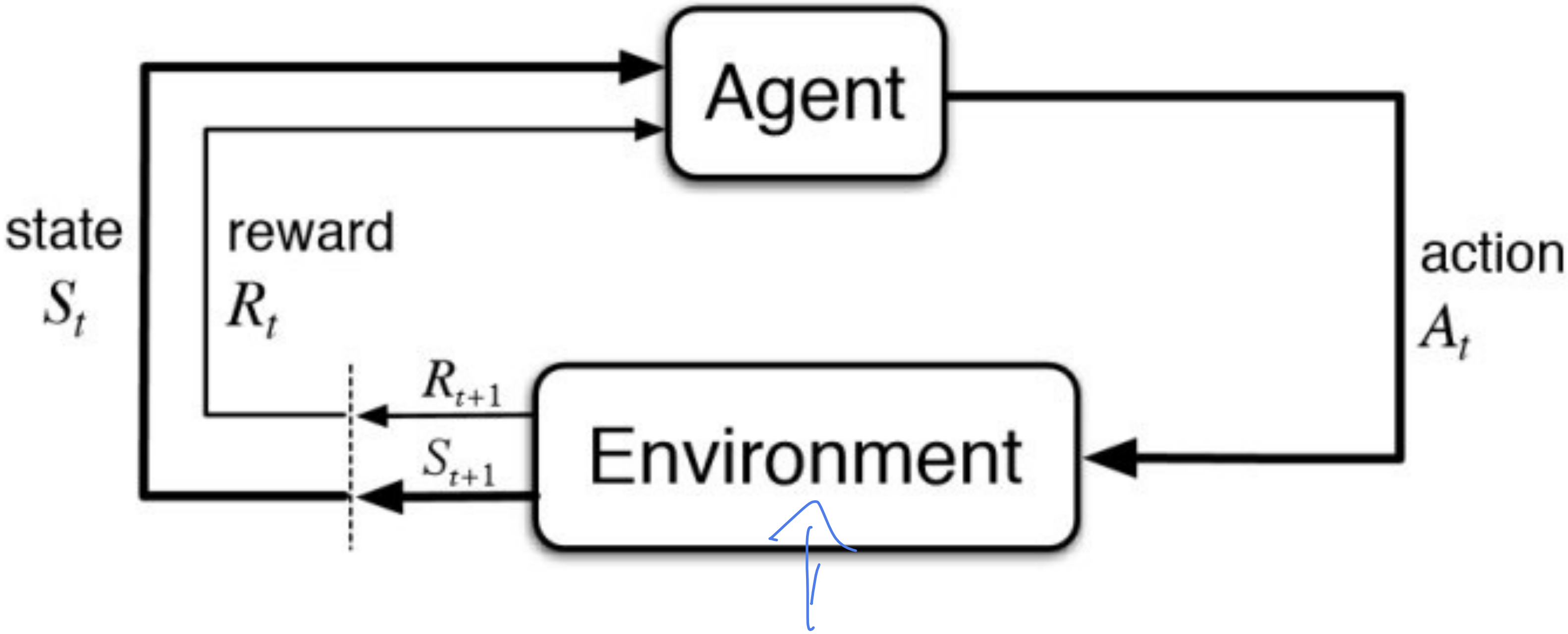
Recap: RL is among the hottest area of research in ML!







An RL agent learns interactively through the feedbacks of an environment.



- Learning how the world works (dynamics) and how to maximize the long-term reward (control) at the same time.

Applications of RL in the real life

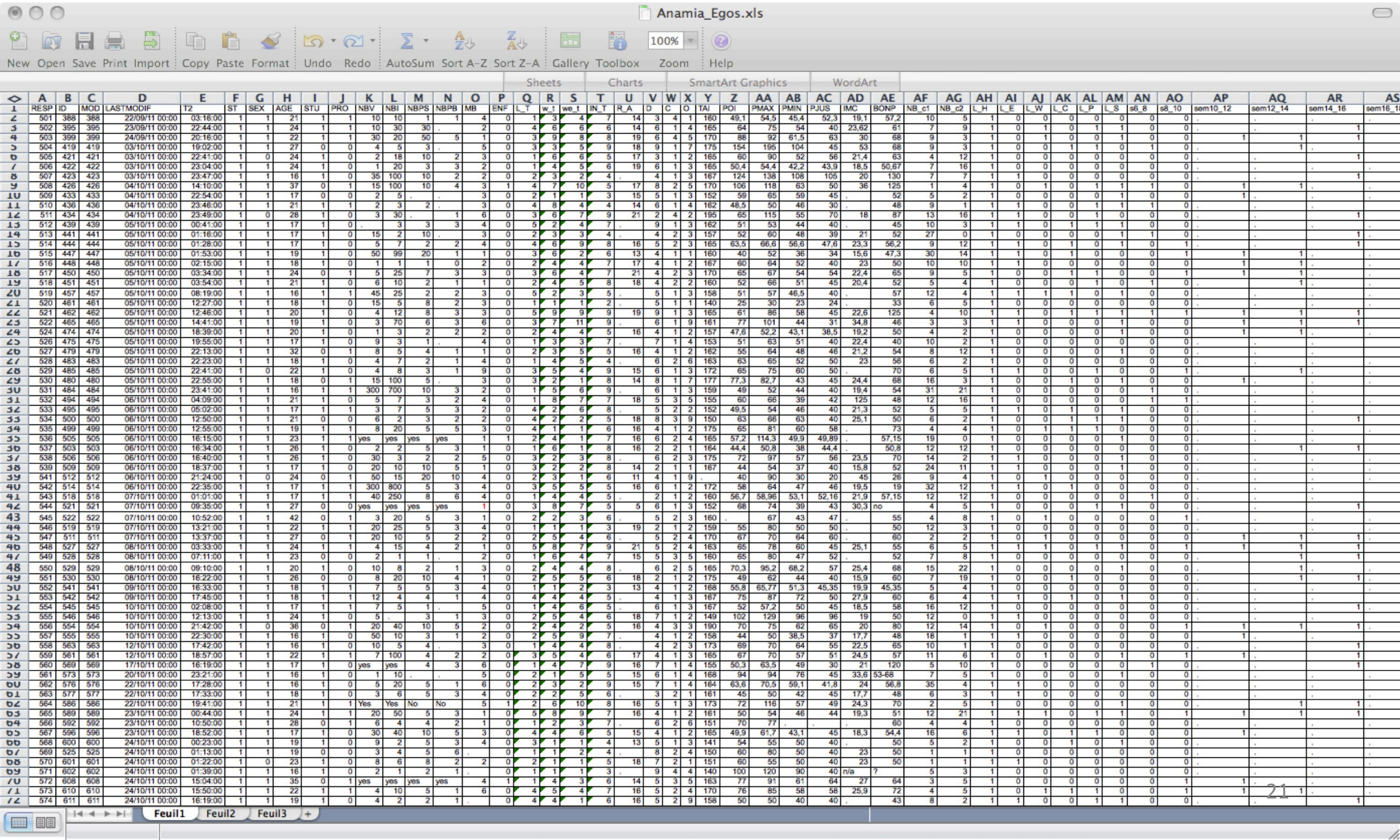
- RL for robotics.
- RL for dialogue systems.
- RL for personalized medicine.
- RL for self-driving cars.
- RL for new material discovery.
- RL for sustainable energy.
- RL for feature-based dynamic pricing.
- RL for maximizing user satisfaction.
- RL for QoE optimization in networking
- …

Challenges of Reinforcement in the real life

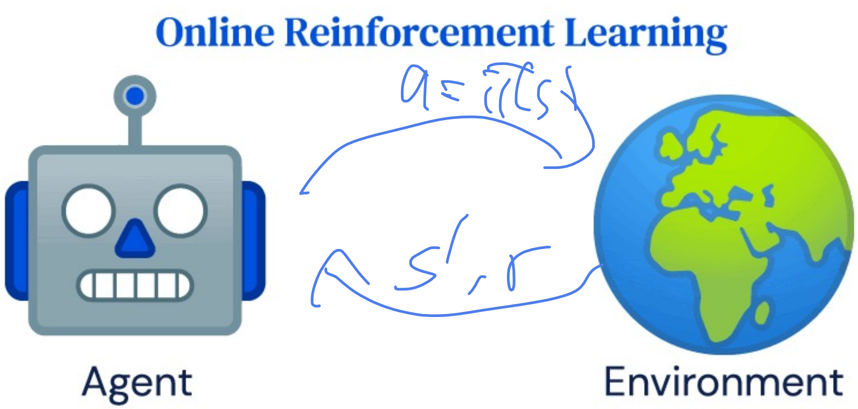
learning

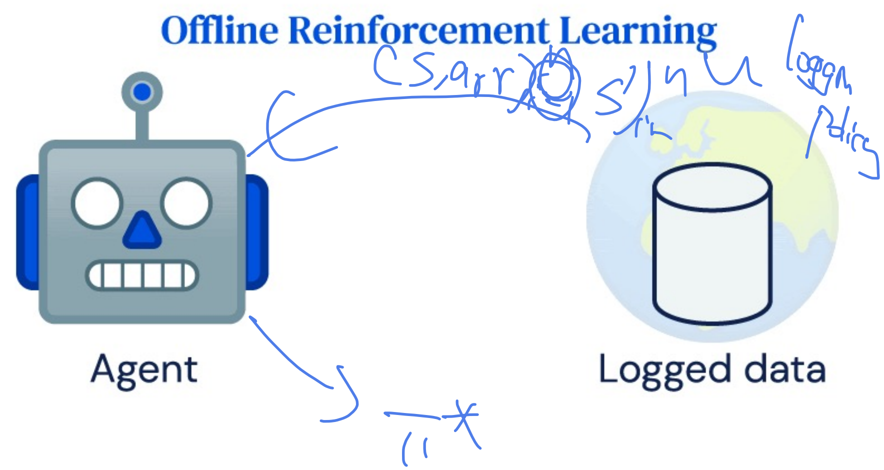
- No access to a simulator
- Every data point is costly.
- Legal, safety issues associated with exploration.
- Large / complex state-space, action space.
- Long horizon
- Limited adaptivity (cannot run too many iterations)

From an Applied ML Scientist point of view, the starting point of a project is often:



Online RL vs Offline RL





Exploration is often **expensive**, **unsafe**, **unethical** or **illegal** in practice, e.g., in self-driving cars, or in medical applications.

Can we learn a policy from already **logged interaction data**?

Off-Policy learning: an example

amazon

Recommendations

buy or not buy

- How to evaluate a new algorithm without actually running it live?
- How to learn a better system than the one that is deployed.

Offline Reinforcement Learning, aka. Batch RL

- Task 1: Offline Policy Evaluation. (OPE)

Offline Trajectory data D Collected by running $\boldsymbol{\mu}$

Task: design OPE methods

Evaluate fixed Target Policy $\boldsymbol{\pi}$

Via Uniform OPE

- Task 2: Offline Policy Learning. (OPL)

Offline Trajectory data D Collected by running $\boldsymbol{\mu}$

Task: design OPO methods

Find near optimal Policy $\widehat{\boldsymbol{\pi}}^*$

Contextual bandits model

- Contexts: /state
  - $x_1, \ldots, x_n \sim \lambda$ drawn iid, possibly infinite domain $\mathbb{R}^d$
- Actions:
  - $a_i \sim \mu(a|x_i)$ Taken by a randomized "Logging" policy
- Reward:
  - $r_i \sim D(r|x_i, a_i)$ Revealed only for the action taken
- Value:
  - $v^{\boxed{\mu}} = \mathbb{E}_{x\sim\lambda}\mathbb{E}_{\boxed{a\sim\mu(\cdot|x)}}\mathbb{E}_D[r|x,a]$
- We collect data $(x_i, a_i, r_i)_{i=1}^n$ by the above processes.

Off-policy Evaluation and Learning

Off-policy evaluation

Estimate the value of a fixed target policy π

$$v_\pi := \mathbb{E}_\pi[\text{Reward}]$$

Off-policy learning

find $\pi \in \Pi$

that maximizes v_π

- Using data $(x_i, a_i, r_i)_{i=1}^n$

$(x_i, a_i, r_i, \mu_i)_{i=1}^n$ $\mu(a_i|x)$

- often the policy μ or logged propensities $(\mu_i)_{i=1}^n$

ATE estimation is a special case of off-policy evaluation

- a: Action ⇔ T: Treatment {0,1}
- r: Reward ⇔ Y: Response variable
- x: Contexts ⇔ X: covariates

Direct Method / Regression-estimator

- Fit a regression model of the reward

$$\hat{r}(x,a) \approx \mathbb{E}(r|x,a) \quad \text{using the data}$$

- Then for any target policy

$$\hat{v}^{\pi}_{\mathrm{DM}} = \frac{1}{n}\sum_{i=1}^{n}\sum_{a\in\mathcal{A}} \boxed{\hat{r}(x_i,a)}\,\pi(a|x_i)$$

Pros:

- Low-variance.
- Can evaluate on unseen contexts

Cons:

- Often high bias
- The model can be wrong/hard to learn

Inverse propensity score / Importance sampling

(Horvitz & Thompson, 1952)

$$\mathbb{E}[\hat{V}^{\pi}_{\text{IPS}}] = \cancel{\frac{1}{n}\sum_i} E\left(\frac{\pi(a_i|x_i)}{\mu(a_i|x_i)} r_i\right) = E_{x_i}\left[\sum_a \mu(a|x_i)\frac{\pi(a|x_i)}{\mu(a|x_i)} E[r_i|\ldots]\right] = V^{\pi}$$

Importance weights

$$\hat{v}^{\pi}_{\text{IPS}} = \frac{1}{n}\sum_{i=1}^{n} \boxed{\frac{\pi(a_i|x_i)}{\mu(a_i|x_i)}} r_i$$

$$\frac{\pi(a_i|x_i)}{\mu(a_i|x_i)} =: \rho_i$$

Pros:

- No assumption on rewards
- Unbiased
- Computationally efficient

Cons:

- High variance when the weight is large

Analyzing the performance of importance sampling estimator

Importance Sampling and Direct Method are surprisingly similar in some cases

- Consider the MAB case

Next lecture: OPE for reinforcement learning

- Importance sampling

- Marginalized importance sampling