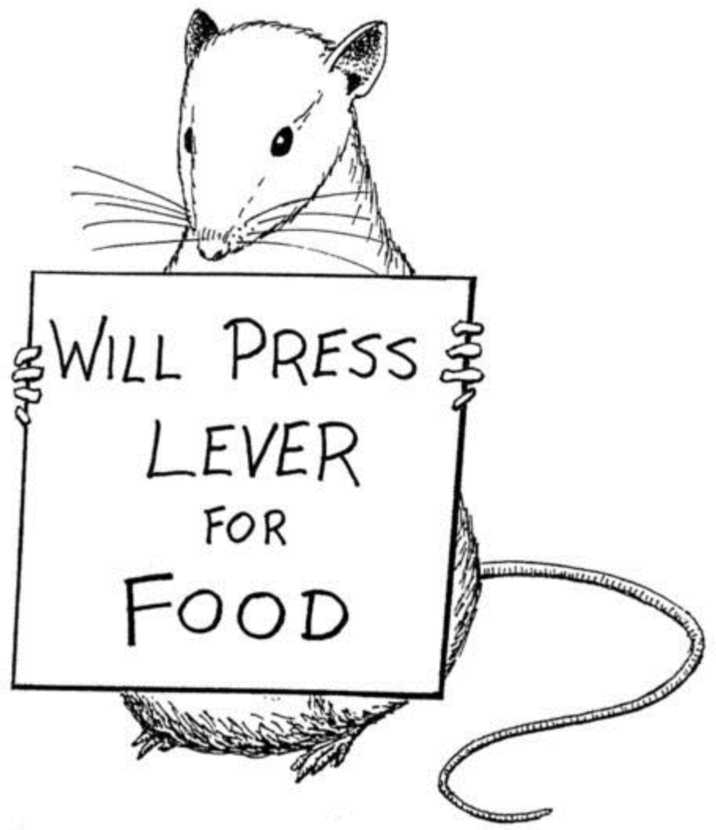
CS 277: Control and Reinforcement Learning

Winter 2024

Lecture 11: Exploration

Roy Fox
Department of Computer Science
School of Information and Computer Sciences
University of California, Irvine



Logistics

assignments

- Exercise 3 due next Monday
- Quiz 6 published shortly, due next Monday
- Quiz schedule (hopefully) finalized

lectures

- After this week: advanced topics
  - Including some important algorithms
- Next Tuesday: guest lecture on RLHF

Today's lecture

Multi-Armed Bandits

Exploration in Deep RL

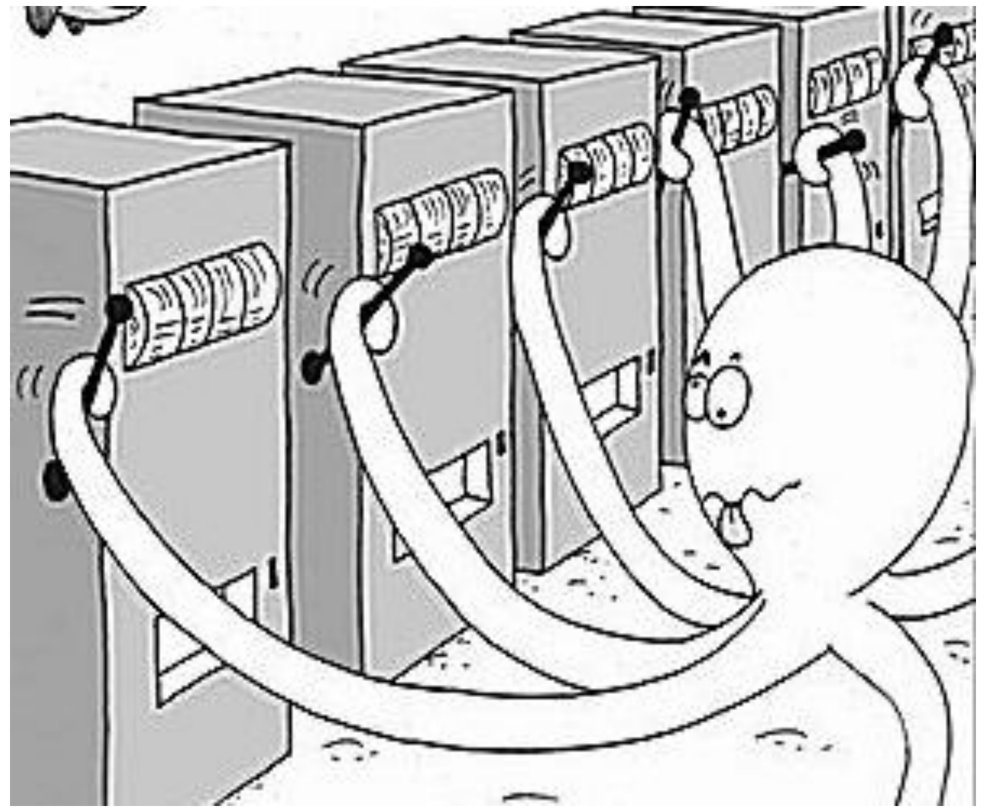
Sparse rewards

Multi-Armed Bandits (MABs)

- Basic setting: single instance x, multiple actions $a_1, \ldots, a_k$
  - Each time we take action a_i we see a noisy reward $r_t \sim p_i$
- Can we maximize the expected reward $\max_i \mathbb{E}_{r \sim p_i}[r]$?
  - We can use the mean as an estimate $\mu_i = \mathbb{E}_{r \sim p_i}[r] \approx \frac{1}{n(i)} \sum_{t \in \mathcal{T}_i} r_t$
- Challenge: is the best mean so far the best action?
  - Or is there another that's better than it appeared so far?

One-armed bandit

Multi-armed bandit



Exploration vs. exploitation

- Exploitation = choose actions that seems good (so far)
- Exploration = see if we're missing out on even better ones
- Naïve solution: learn r by trying every action enough times
  - Suppose we can't wait that long: we care about rewards while we learn
- Regret = how much worse our return is than an optimal action

$$\rho(T) = T\mu_{a^*} - \sum_{t=0}^{T-1} r_t$$

  - Can we get the regret to grow sub-linearly with T? $\implies$ average goes to 0: $\frac{\rho(T)}{T} \to 0$

Let's play!

- http://iosband.github.io/2015/07/28/Beat-the-bandit.html

Simple exploration: ϵ-greedy

- With probability ϵ:
  - Select action uniformly at random
- Otherwise (w.p. $1 - \epsilon$):
  - Select best (on average) action so far
- Problem 1: all non-greedy actions selected with same probability
- Problem 2: must have $\epsilon \to 0$, or we keep accumulating regret
  - But at what rate should ϵ vanish?

Boltzmann exploration

- Keep an average of past rewards $\hat{\mu}_i = \frac{1}{n(i)} \sum_{t \in \mathcal{T}_i} r_t$
- Boltzmann (softmax) exploration: $\pi(a_i) = \mathrm{softmax}_\beta \hat{\mu}_i = \frac{\exp(\beta \hat{\mu}_i)}{\sum_j \exp(\beta \hat{\mu}_j)}$
- Obviously bad actions $\hat{\mu}_i \ll \max_j \hat{\mu}_j$ are unlikely to be used (but can!)
  - Problem: still must have $\beta \to \infty$, or we keep accumulating regret
  - Some evidence that β should increase linearly

Optimism under uncertainty

- Tradeoff: explore less used actions, but don't be late to start exploiting what's known
  - Principle: optimism under uncertainty = explore to the extent you're uncertain, otherwise exploit
- By the central limit theorem, the mean reward $\hat{\mu}_i$ of arm i quickly $\to \mathcal{N}\left(\mu_i, O\left(\frac{1}{n(i)}\right)\right)$
- Be optimistic by slowly-growing number of standard deviations:

$$a = \arg\max_i \hat{\mu}_i + \sqrt{\frac{2 \ln T}{n(i)}}$$

  - Upper confidence bound (UCB): likely $\mu_i \leq \hat{\mu}_i + c\sigma_i$; unknown variance $\implies$ let c grow
  - But not too fast, or we fail to exploit what we do know
- Regret: $\rho(T) = O(\log T)$, provably optimal

Thompson sampling

- Consider a model of the reward distribution $p_{\theta_i}(r \,|\, a_i)$
- Suppose we start with some prior $q(\theta)$
  - Taking action a_t, see reward $r_t \implies$ update posterior $q(\theta \,|\, \{(a_{\leq t}, r_{\leq t})\})$
- Thompson sampling:
  - Sample $\theta \sim q$ from the posterior
  - Take the optimal action $a^* = \max_i \mathbb{E}_{r \sim p_{\theta i}}[r]$
  - Update the belief (different methods for doing this)
  - Repeat

Other online learning settings

- What is the reward for action a_i?
  - MAB: random variable with distribution $p_i(r)$
  - Adversarial bandits: adversary selects r_i for every action
    - The adversary knows our algorithm! And past action selection! But not future actions
      - Learner must be stochastic (= unpredictable), but we can still have guarantees
  - Dueling bandits: just 1 bit of feedback, is a_i better or a_j?
- Contextual bandits: we also get instance $x \sim p$, make decision $\pi(a \mid x)$
  - Can we generalize to unseen instances?

Today's lecture

Multi-Armed Bandits

Exploration in Deep RL

Sparse rewards

Learning with sparse rewards

- Montezuma's Revenge
  - Key = 100 points
  - Door = 500 points
  - Skull = 0 points
    - Is it good? Bad? Affects something off-screen? Opens up an easter egg?
  - Humans have a head start with transfer from known objects

- Exploration before learning:
  - Random walk until you get some points — could take a while!

RL exploration is more complicated...

- Need to consider states and dynamics
- Need coordinated behavior to get *anywhere*
  - E.g., cross a bridge to get the game started...
  - Random exploration will kill us with high probability
    - Structured exploration: noise over time has joint distribution, temporal structure
- How to define regret?
  - With respect to constant action? We can outperform it
  - With respect to optimal policy? May be too hard to learn $\Longrightarrow$ linear regret
  - Most approaches are heuristic, no regret guarantees; often train-time rewards don't matter

Count-based exploration

- Generalizing UCB exploration $a = \arg\max_i \hat{\mu}_i + \sqrt{\frac{2 \ln T}{n(i)}}$ from MAB to RL

- Count visitations to each state $n(s)$ (or state-action $n(s, a)$)

- Optimism under uncertainty: add exploration bonus to scarcely-visited states

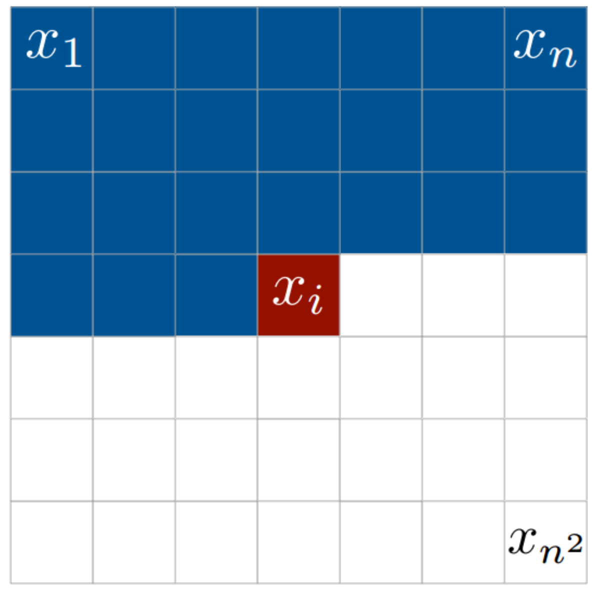
$$\tilde{r} = r + r_e(n(s))$$

  - r_e should be monotonic decreasing in $n(s)$

  - Need to tune its weight

Density model for count-based exploration

- How to represent "counts" in large state spaces?
  - We may never see the same state twice
  - If a state is very similar to ones we've seen often, is it new?
- Train a density model $p_\phi(s)$ over past experience
- Unlike generative models, we care about getting the density correctly
  - But we don't care about the quality of samples



- Density models for images:
  - CTS, PixelRNN, PixelCNN, etc.

Pseudo-counts

- How to infer pseudo-counts from a density model?

$$p_\phi(s) = \frac{n(s)}{N}$$

- After another visit:

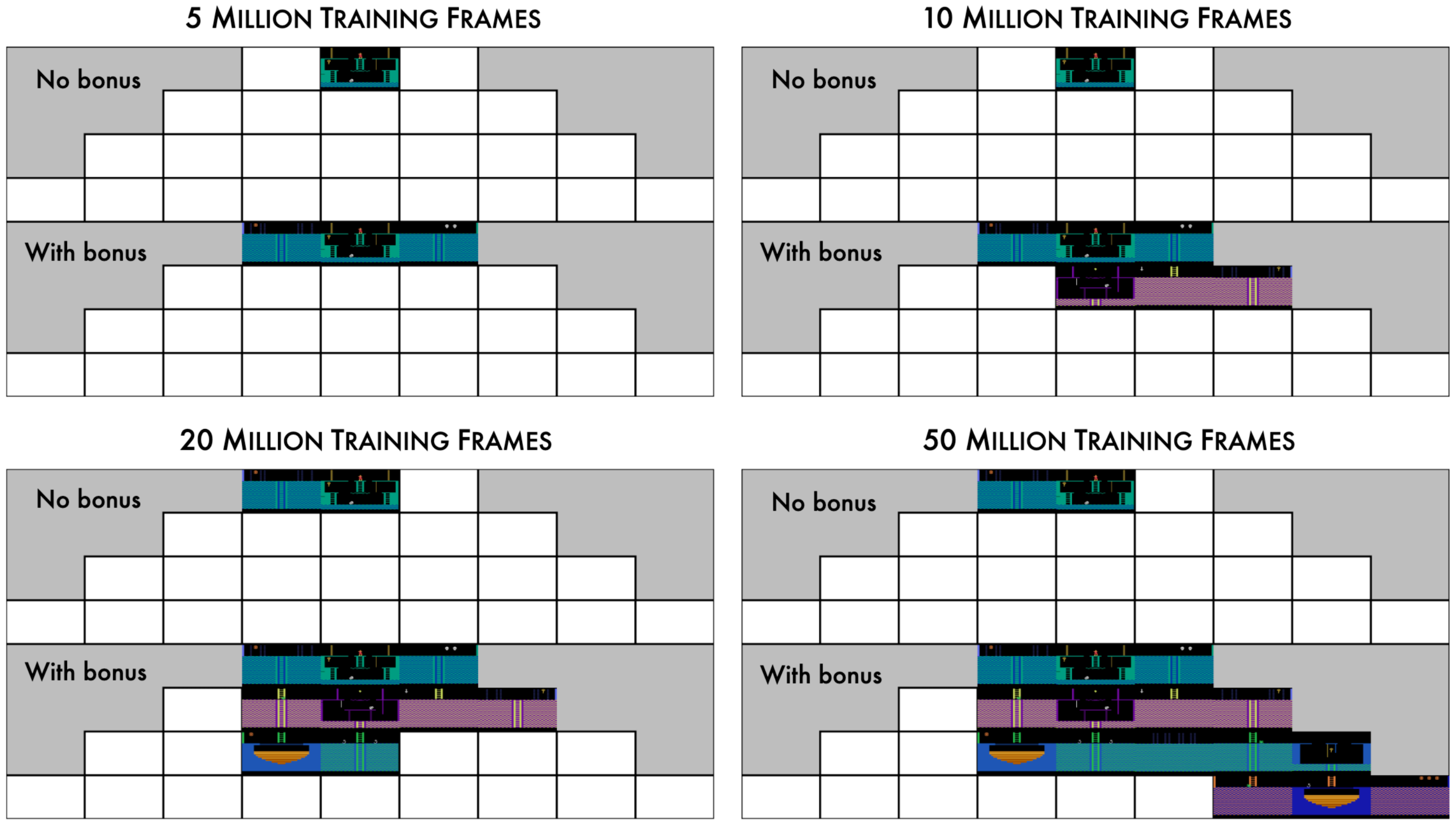
$$p_\phi(s) = \frac{n(s)+1}{N+1}$$

- To recover the pseudo-count:
  - $p_{\phi'} \leftarrow$ mock-update the density model with another visit of s
  - Compute

$$\hat{N} = \frac{1 - p_{\phi'}(s)}{p_{\phi'}(s) - p_\phi(s)} p_\phi(s) \qquad \hat{n}(s) = \hat{N} p_\phi(s)$$

Exploration bonus

- What's a good exploration bonus?
- In bandits: Upper Confidence Bound (UCB)
  - $$r_e(n(s)) = \sqrt{\frac{2 \ln N}{n(s)}}$$
- In RL, often:
  - $$r_e(n(s)) = \sqrt{\frac{1}{n(s)}}$$



[Bellemare et al., 2016]

Thompson sampling for RL

- Keep a distribution over models $p_\theta(\phi)$
- What's our “model”? Idea 1: MDP; Idea 2: Q-function
- Thompson sampling over Q-functions:
  - Sample $Q \sim p_\theta$
  - Roll out an episode with the greedy policy $\pi(s) = \arg\max_a Q(s, a)$
  - Update p_θ to be more likely for Q' that gives low empirical Bellman error
  - Repeat

Optimal exploration: simple settings

- Multi-Armed Bandits (MAB): single state, one-step horizon
  - Exploration–exploitation tradeoff very well understood
- Contextual bandits: random state, one-step horizon
  - Also has good theory (Online Learning)
- Tabular RL
  - Some good heuristics, recent theoretical guarantees
- Deep RL
  - Only few exploratory ideas and heuristics

Recap

- Online learning = getting good rewards while learning
  - In contrast: learn however, but deploy good policy
- Online learning requires trading off exploration–exploitation
  - Don't overfit to too little data
  - Don't be late to use what you've learned
- Optimism under uncertainty: exploration bonus for novelty
- Thompson sampling: coordinated exploration actions
- Same principles hold in RL

Today's lecture

Multi-Armed Bandits

Exploration in Deep RL

Sparse rewards

Relation between RL and IL

- What makes RL harder than IL?
  - IL: teacher policy $\pi_e(a \mid s)$ indicates a good action to take in s
  - RL: $r(s, a)$ does not indicate a globally good action; $Q^*(s, a)$ does, but it's nonlocal
- But didn't we see an equivalence between RL and IL?
  - NLL loss in BC: $\nabla_\theta \mathbb{E}[\log \pi_\theta(a \mid s)]$
    - s and a sampled from teacher distribution
  - PG loss: $\nabla_\theta \mathbb{E}[\log \pi_\theta(a \mid s) R]$
    - s and a sampled from learner distribution

Informational quantities: refresher

- Entropy: $\mathbb{H}[p(a)] = -\mathbb{E}_{a\sim p}[\log p(a)] = -\sum_a p(a)\log p(a)$
- Conditional entropy: $\mathbb{H}[\pi\,|\,s] = -\mathbb{E}_{a\sim\pi}[\log \pi(a\,|\,s)]$
- Expected conditional entropy: $\mathbb{H}[\pi] = \mathbb{E}_{s\sim p_\pi}[\mathbb{H}[\pi\,|\,s]] = -\mathbb{E}_{s,a\sim p_\pi}[\log \pi(a\,|\,s)]$
- Expected relative entropy: $\mathbb{D}[\pi\|\pi'] = \mathbb{E}_{s,a\sim p_\pi}\left[\log \frac{\pi(a\,|\,s)}{\pi'(a\,|\,s)}\right]$
- Expected cross entropy (aka NLL): $-\mathbb{E}_{s,a\sim p_\pi}[\log \pi'(a\,|\,s)]$
  - $\mathbb{D}[\pi\|\pi'] = \text{NLL} - \mathbb{H}[\pi]$

IL as sparse-reward RL

- NLL BC: maximize $\mathbb{E}_{s,a\sim p_e}[\log \pi_\theta(a\,|\,s)] = -\mathbb{D}[\pi_e \| \pi_\theta] - \mathbb{H}[\pi_e]$

  constant in θ

  - Experience from teacher distribution p_e
    - RL: experience from learner distribution p_θ
  - "Return" $R = 1_{\text{success}}$ for successful trajectory
    - RL: $r_t = r(s_t, a_t)$ in every step
- Sparse reward = most rewards are 0 $\implies$ rare learning signal
  - $R = 1$ on success = very sparse; but doesn't IL provide dense learning signal?

IL as dense-reward RL

- What if instead we minimize the other relative entropy?

teacher labeling of learner states/actions as in DAgger

$$\mathbb{D}[\pi_\theta \| \pi_e] = -\mathbb{E}_{s,a\sim p_\theta}[\log \pi_e(a|s)] - \mathbb{H}[\pi_\theta]$$

  - This is exactly the RL objective, with $r(s,a) = \log \pi_e(a|s)$ and entropy regularizer
  - Now $r(s,a)$ does give global information on optimal action
  - In fact, with deterministic teacher, $r(s,a) = -\infty$ for any suboptimal action
- The same return can be viewed as dense reward or sum of sparse rewards
  - Can we do the same in proper RL?

Reward shaping

- Ideal reward: $r(s, a) = -\infty$ for any suboptimal action $\implies$ as hard to provide as π^*
  - We need supervision signal that's sufficiently easy to program $\implies$ generate more data
- Sparse reward functions may be easier than dense ones
  - E.g., may be easy to identify good goal states, safety violations, etc.
- Reward shaping: art of adjusting the reward function for easier RL; some tips:
  - Reward "bottleneck states": subgoals that are likely to lead to bigger goals
  - Break down long sequences of coordinated actions $\implies$ better exploration
    - E.g. reward beacons on long narrow paths, for exploration to stumble upon