Curriculum Learning for Reinforcement Learning

Runxuan Jiang

1 Deterministic MDP

We first consider the case of using curriculum learning on an MDP with deterministic reward and transitions.

Proposition (Existence and bound for deterministic finite MDP.)**.** Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R, \rho)$ be a deterministic MDP with sparse reward, and that $R(s, a) = \mathcal{X}(s = g)$ where for some $g \in A$.

Then there exists a finite sequence of H MDP's $\mathcal{M}_i = (\mathcal{S}, \mathcal{A}, H, P_i, R_i, \rho_i)$ such that the following algorithm

1. Initialize π_0 to be a random policy.
2. For $1 \leq i \leq H$, Follow π_{i-1} with an ϵ_i-greedy policy until an optimal policy is found.

will find an optimal policy for $\mathcal{M}$ in $O(H^2 k)$ episodes in expectation, where $k = |\mathcal{A}|$.

Proof. Let $s_0, s_1, \ldots, s_H = g$ be a trajectory that the optimal policy for $\mathcal{M}$ takes. Construct $\mathcal{M}_i$ such that $R_i(s, a) = \mathcal{X}(s = s_i)$ and $P_i(s_i|s_i, a) = 1 \ \forall a \in A$, otherwise $P_i = P$ for $1 \leq i \leq H$.

Let $p_i(\epsilon)$ denote the probability that the ϵ-greedy algorithm will achieve the optimal policy for MDP $\mathcal{M}_i$. Then we have

$$p_i(\epsilon) \geq f(\epsilon) := (1 - \epsilon)^{i-1}(\frac{\epsilon}{k}).$$

This comes from the fact that the optimal trajectory can be achieved by playing the greedy option $i - 1$ times, and then choosing the exploratory option and choosing the action that reaches the correct state, which has probability $\frac{\epsilon}{k}$.

For $i \geq 2$, f is concave, so we can find the maximal probability by setting the derivative to 0.

Note. Will need a prove on why this is concave.

$$p_i'(\epsilon) = 0$$

$$\Rightarrow \frac{(1-\epsilon)^{i-1}}{k} - (i-1)(1-\epsilon)^{i-2}(\frac{\epsilon}{k}) = 0$$
$$\Rightarrow \frac{(1-\epsilon)^{i-1}}{k} = (i-1)(1-\epsilon)^{i-2}(\frac{\epsilon}{k})$$
$$\frac{1-\epsilon}{k} = (i-1)(\frac{\epsilon}{k}) \Rightarrow \epsilon = \frac{1}{i}.$$

By plugging in this value for ϵ in f for $\mathcal{M}_i$, p is bounded below by

$$(1-\frac{1}{i})^{i-1}\frac{1}{ki}.$$

So the expected number of episodes to reach the optimal policy for $\mathcal{M}_i$ is

$$\frac{1}{(1-\frac{1}{i})^{i-1}(\frac{1}{ki})}$$
$$= \frac{ki}{(1-\frac{1}{i})^{i-1}} = (i-1)k(\frac{i}{i-1})^i$$

Notice that since $i \geq 2$ and $x \mapsto (\frac{x}{x-1})^x$ is decreasing for $x > 0$, the above expression is bounded above by

$$4k(i-1).$$

So the number of episodes for the full curriculum is in expectation

$$\sum_{j=2}^{H} 4k(j-1) \leq \frac{H}{2}(4kH)$$

which is $O(kH^2)$. ■

Remark (Tighter bound)**.** Note that the above proposition only considers the case where for each $\mathcal{M}_i$, the ϵ-greedy algorithm is greedy for $i-1$ steps, and then gets the correct state for the single final exploratory step. However, the curriculum learning also helps in the situation where some of the steps are exploratory and "lucky" enough to get the same action as the optimal policy, while the remaining steps are greedy.

However, even using this the bound is still $O(kH^2)$, so this is the tightest bound we can achieve under these conditions (probably).

Proof. Using a similar setup as the above proof, we now consider using the probability for achieving an optimal policy:

$$p(\epsilon) \geq f(\epsilon) := \binom{i-1}{0}(1-\epsilon)^{i-1}(\frac{\epsilon}{k}) + \binom{i-1}{1}(1-\epsilon)^{i-2}(\frac{\epsilon}{k})^2 + \ldots + \binom{i-1}{i-1}(\frac{\epsilon}{k})^j$$

$$= (\frac{\epsilon}{k}) \sum_{j=0}^{i-1} \binom{i-1}{j} (1-\epsilon)^{i-1-j} (\frac{\epsilon}{k})^j$$
$$= (\frac{\epsilon}{k})(1 - \epsilon + \frac{\epsilon}{k})^{i-1}.$$

Note that this is concave on $(0, 1)$.

Note. Proof of concavity needed.

Since it is concave, we can set the derivative to 0 to find the ϵ that maximizes f. We have

$$f'(\epsilon) = \frac{(1 - \epsilon + \frac{\epsilon}{k})^{i-1}}{k} + (\frac{\epsilon}{k})(i-1)(1 - \epsilon + \frac{\epsilon}{k})^{i-2}(\frac{1}{k} - 1) = 0$$
$$\Rightarrow \frac{(1 - \epsilon + \frac{\epsilon}{k})^{i-1}}{k} = (1-i)(\frac{\epsilon}{k})(\frac{1}{k} - 1)(1 - \epsilon + \frac{\epsilon}{k})^{i-2}$$
$$\Rightarrow (1 - \epsilon + \frac{\epsilon}{k})^{i-1} = \epsilon(1-i)(\frac{1}{k} - 1)(1 - \epsilon + \frac{\epsilon}{k})^{i-2}$$
$$\Rightarrow 1 - \epsilon + \frac{\epsilon}{k} = \epsilon(1-i)(\frac{1}{k} - 1)$$
$$\Rightarrow 1 - \epsilon + \frac{\epsilon}{k} = \frac{\epsilon}{k} - \epsilon + \frac{\epsilon i}{k} + i\epsilon$$
$$\Rightarrow 1 = \epsilon(i - \frac{i}{k})$$
$$\Rightarrow \epsilon = \frac{1}{i - \frac{i}{k}} = \frac{k}{i(k-1)}$$

We then plug this value of ϵ back into f to get the maximal point of f:

$$(\frac{\frac{k}{i(k-1)}}{k})(1 - \frac{k}{i(k-1)} + \frac{\frac{k}{i(k-1)}}{k})^{i-1} = \frac{1}{i(k-1)}(1 - \frac{k-1}{i(k-1)}) = \frac{1 - \frac{1}{i}}{i(k-1)}$$

and take the reciporocal to get the expected number of episodes for $\mathcal{M}_i$, which is

$$\frac{i(k-1)}{1 - \frac{1}{i}} = \frac{i^2(k-1)}{i-1}$$

.

Note that $x \leq \frac{x^2}{x-1} \leq 4x$, so since $2 \leq i \leq H$ we get that

$$\sum_{j=2}^{H} \frac{j^2(k-1)}{j-1} = (k-1 \sum_{j=2}^{H} \frac{j^2}{j-1})$$

which is $O(kH^2)$.

To prove this is a tight bound, we can show that the probability given by $p(\epsilon)$ is a tight lower bound to the actual probability. ■

Proposition (Existence and bound for deterministic finite MDP)**.** Let $\mathcal{M} = (S, A, H, P, R, \rho)$ be a deterministic MDP with sparse reward, and that $R(s, a) = \mathcal{X}(s = g)$ for some $g \in A$.

Then there exists a finite sequence of H MDP's $\mathcal{M}_i = (S, A, H, P_i, R_i, \rho_i)$ such that the following algorithm

1. Initialize π_0 to be a random policy.
2. For $1 \leq i \leq H$, Follow π_{i-1} with an ϵ_i-greedy policy until an optimal policy is found.

will find an optimal policy with probability $1 - \delta$ in at least x episodes.

Note. TODO: Prove this theorem (same as above but finding the number of episodes to get an optimal policy with high probability).

2 Curriculum Learning with Model-Based algorithms

Given a model-based reinforcement learning algorithm (such as RMAX or UCB), we want to show that we can solve MDP's using the curriculum learning paradigm on the MDP's used by the algorithm to "model" the MDP to be solved. Formally, such algorithms generate a sequence of MDP's $M_1, \ldots, M_n$ with corresponding optimal policies $\pi_1^*, \ldots, \pi_n^*$. We want to show that by using π_{i-1}^* with ϵ-greedy on M_i we can achieve optimal regret bounds for M_i. Through induction, we therefore have a "curriculum" for learning the MDP.

2.1 RMAX

We start with an episodic nonstationery MDP $M = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}, R, \rho)$. Let L be a collection of states and actions $L \subset \mathcal{S} \times \mathcal{A} \times [H]$ corresponding to the set of states marked unknown in the RMAX algorithm.

Definition (M_L)**.** Let M and L be defined as above. We define M_L as a MDP with the same states and actions as M, and with the same rewards and transitions for $(s, a, h) \notin L$. For $(s, a, h) \in L$, the the reward will be equal to R_{max} and the transition will lead to an absorbing state that always gives reward R_{max}.

2.1.1 Original R-Max Algorithm

We first describe the RMAX algorithm for MDP's:

Algorithm 1 R-max for MDP's

Require: MDP $M = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}, R, \rho), \epsilon > 0, 0 \leq \delta \leq 1, R_{max} > 0$.

procedure R-MAX($M, \epsilon, \delta, R_{max}$)
  $Rewards[S][A][H]$ be a new array.
  $Transitions[S][A][H]$ be a new array.
  $Known[S][A][H]$ be a new array.
  $L \leftarrow \mathcal{S} \times \mathcal{A} \times [H]$
  $M_{curr} \leftarrow M_L$
  for $i = 1, \ldots, S * A * H$ **do**
    while No new entry is known and the current policy has not been run for more than K_2 steps **do**
      Obtain trajectories by running the optimal policy for M_{curr} on M.
      if a state (s, a, h) has been visited more than K_1 times **then**
        Mark (s, a, h) as known
        Set the estimated transition probabilities and rewards based on the average of the observed transitions and rewards for (s, a, h).
        Replace the transition dynamics and reward in (s, a, h) in M_{curr} by the new estimated dynamics.
      end if
    end while
  end for
  Return The optimal policy for M_{curr}
end procedure

Remark (Is this curriculum learning?)**.** Notice that this algorithm appears similar to curriculum learning. However there are some differences. In curriculum learning, the algorithm is given a sequence of tasks and trains an agent on the current task by using a (near)-optimal policy for the previous task. So given a sequence of MDP's $M_1, \ldots, M_n$ you explore M_j with the aid of the policy π_{j-1}. However, in the R-max algorithm we are exploring M directly using π_j.

The RMAX algorithm works by starting with $L_0 = \mathcal{S} \times \mathcal{A}$, and using the optimal policy for M_{L_0} on M. Once enough samples have been collected from a state-action pair (s, a, h), it is removed from L_0 to get $L_1 = L_0 \setminus \{(s, a, h)\}$, and then using the optimal policy for M_{L_1}. This is done until there are no unknown states left (i.e., L is empty).

So we have

$$\mathcal{S} \times \mathcal{A} \times [H] = L_0 \supset L_1 \supset \ldots \supset L_{SAH} = \emptyset$$

We will also denote M_{L_j} as M_j.

Note. In the original R-max algorithm, the optimal policy for each M_j is not used since M_j is not know fully. Instead, the optimal policy for the estimated MDP for M_j, which is based on the samples collected previously, is used.

2.1.2 R-max Algorithm with Curriculum Learning

We first prove some preliminary results.

Theorem. Let M be a MDP and let M' be another MDP such that $||P_M - P_{M'}||_\infty < \frac{\epsilon}{SH^2R_{max}}$. Then for all policies π,

$$|V_M(\pi) - V_{M'}(\pi)| \leq \epsilon$$

Proof. Lemma 4 of the R-max paper. ■

Lemma. Let M be an MDP. To approximate the transition probabilility for a state s, a, h in M with error less than ϵ with probability higher than $1-\delta$, we need $K_1 \geq \frac{-\ln(\delta/2S)}{2\epsilon^2}$ samples.

Proof. Let n be the number of samples collected where the state (s, a, h) leads to the state s'. Let X_i be an indicator variable that is 1 if (s, a, h) leads to $(s', h+1)$ and 0 otherwise in ith sample. Let p be the actual transition probability of transitioning from (s, a, h) to $(s', h+1)$. Let $S = \frac{X_1+...+X_n}{n}$. Note that $0 \leq \frac{X_i}{n} \leq \frac{1}{n}$. Since there are S possibilities for (s', h), we want the probability that $|S - p| \geq \epsilon$ to be less than $\frac{\delta}{S}$. By Hoeffding's inequality we have

$$P(|S - p| \geq \epsilon) \leq 2\exp(\frac{-2\epsilon^2}{n(\frac{1}{n})^2}) = 2\exp(-2n\epsilon^2).$$

We want

$$2\exp(-2n\epsilon^2) \leq \delta/S$$

$$\Rightarrow -2n\epsilon^2 \leq \ln(\frac{\delta}{2S})$$

$$\Rightarrow n \geq \frac{-\ln(\frac{\delta}{2S})}{2\epsilon^2}.$$

■

The curriculum learning algorithm utilizing R-max is as follows:

Algorithm 2 Curriculum Learning with R-max

Require: MDP $M = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}, R, \rho), \epsilon > 0, 0 \leq \delta \leq 1, R_{max} > 0$.

1: **procedure** CURRICULUM_R-MAX($M, \epsilon, \delta, R_{max}$)
2: $L \leftarrow \mathcal{S} \times \mathcal{A} \times [H]$
3: $M_1 \leftarrow M_L$
4: $\hat{M} \leftarrow M_L$
5: $\pi_1 \leftarrow$ optimal policy for $\hat{M}$
6: $i \leftarrow 1$
7: **while** Running π_i on M explores a state in L with probability higher than $\frac{\alpha}{H * R_{max}}$ **do**
8: $(s, a, h) \leftarrow$ the state in L with the highest probability of being reached by running π_i on M_i.
9: $L \leftarrow L \setminus \{(s, a, h)\}$
10: $M_{i+1} \leftarrow M_L$
11: Run π_i and ϵ-greedy on M_{i+1} until K_1 trajectories passing through (s, a, h) are collected.
12: Update the reward and transitions for (s, a, h) in $\hat{M}$ based on the average values of the samples collected.
13: $\pi_{i+1} \leftarrow$ optimal policy for $\hat{M}$
14: $i \leftarrow i + 1$
15: **end while**
16: **Return** π_i
17: **end procedure**

Note. Line 10 requires us to know the transition probabilities beforehand, which makes the problem of solving the MDP trivial. But if we don't know the MDP beforehand then we can't use curriculum learning because we wouldn't be able to create a sequence of MDP's from scratch.

Theorem. Fix $0 < \delta < 1, \epsilon > 0$. Then we can find an ϵ-optimal policy with probability higher than $1 - \delta$ by running the above algorithm with $K_1 = \frac{-ln(\frac{\delta}{2S})}{2(\frac{\epsilon}{SH^2R_{max}})^2}$ and $\alpha < \frac{\epsilon}{SH^2R_{max}}$.

Furthermore, the sample complexity of the algorithm is TBD.

Lemma. Let $\alpha > 0$ and $1 \leq j \leq SA$. Denote (s, a, h) to be the entry in L_{j-1} but not in L_j. Consider playing the policy π^*_{j-1} with ϵ-greedy on M_j. Let $V_M(\pi)$ denote the value function on MDP M of policy π, and let V^*_M be the value of any optimal policy on M. Then either

1. $V_{M_j}(\pi_{j-1}) > V^*_{M_j} - \alpha$
2. (s, a, h) will be played with probability of at least $\frac{\alpha}{H * R_{max}}$.

Proof. This proof follows the proof of lemma 6 in the original R-max paper.

Suffices to show that if (2) does not hold then (1) holds. Suppose that (2) does not hold. Let T be the set of all possible trajectories on M_j and let T' be the set of all trajectories

that pass through (s, a, h). Let $P_M^\pi(t)$ denotes the probability that the trajectory t will be reached when playing policy π on the MDP M. Since (2) does not hold we have that

$$\sum_{t \in T'} P_{M_j}^{\pi_{j-1}}(t) < \frac{\alpha}{HR_{max}}.$$

We want to show that

$$V_{M_j}^* - V_{M_j}(\pi_{j-1}) < \alpha.$$

Note that the only difference between M_j and M_{j-1} is that in M_{j-1}, the state (s, h) is an absorbing state and any action from this state gives a reward of R_{max}, but in M_j the reward is replaced by the actual reward and the transition probabilities are also replaced by the actual transition probabilities from M. Thus we have

$$V_{M_j}^* \leq V_{M_{j-1}}^*.$$
$$\Rightarrow V_{M_j}^* - V_{M_j}(\pi_{j-1}) \leq V_{M_{j-1}}^* - V_{M_j}(\pi_{j-1})$$

We now further decompose each of these value functions over all possible trajectories. Let $V_M(t)$ denote the total reward of going through trajectory t on MDP M. Then we have

$$|V_{M_j}(\pi_{j-1}) - V_{M_{j-1}}^*| = |\sum_{t \in T} P_{M_j}^{\pi_{j-1}}(t) V_{M_j}(t) - \sum_{t \in T} P_{M_{j-1}}^{\pi_{j-1}}(t) V_{M_{j-1}}(t)|$$

$$= |\sum_{t \in T'} P_{M_j}^{\pi_{j-1}}(t) V_{M_j}(t) + \sum_{t \in T \setminus T'} P_{M_j}^{\pi_{j-1}}(t) V_{M_j}(t) - \sum_{t \in T'} P_{M_{j-1}}^{\pi_{j-1}}(t) V_{M_{j-1}}(t) - \sum_{t \in T \setminus T'} P_{M_{j-1}}^{\pi_{j-1}}(t) V_{M_{j-1}}(t)$$

$$\leq |\sum_{t \in T'} P_{M_j}^{\pi_{j-1}}(t) V_{M_j}(t) - \sum_{t \in T'} P_{M_{j-1}}^{\pi_{j-1}}(t) V_{M_{j-1}}(t)| + |\sum_{t \in T \setminus T'} P_{M_j}^{\pi_{j-1}}(t) V_{M_j}(t) - \sum_{t \in T \setminus T'} P_{M_{j-1}}^{\pi_{j-1}}(t) V_{M_{j-1}}(t)|$$

Since the rewards and transition probabilities for $t \in T \setminus T'$ are the same for MDPs M_j and M_{j-1}, the second expression above is equal to 0. Thus we are left with the first expression

$$= |\sum_{t \in T'} P_{M_j}^{\pi_{j-1}}(t) V_{M_j}(t) - \sum_{t \in T'} P_{M_{j-1}}^{\pi_{j-1}}(t) V_{M_{j-1}}(t)|.$$

Finally note that since the probabilities for all states before step h have the same transition probability for M_j and M_{j-1}, we have that $\sum_{t \in T'} P_{M_j}^{\pi_{j-1}}(t) = \sum_{t \in T'} P_{M_{j-1}}^{\pi_{j-1}}(t)$. We also have that $\forall t, 0 \leq V_{M_j}(t) \leq HR_{max}$ and $0 \leq V_{M_{j-1}}(t) \leq HR_{max}$. Thus we have

$$|\sum_{t \in T'} P_{M_j}^{\pi_{j-1}}(t) V_{M_j}(t) - \sum_{t \in T'} P_{M_{j-1}}^{\pi_{j-1}}(t) V_{M_{j-1}}(t)| \leq \sum_{t \in T'} P_{M_j}^{\pi_{j-1}} HR_{max}.$$

However from our assumption we have that

$$\sum_{t \in T'} P_{M_j}^{\pi_{j-1}}(t) < \frac{\alpha}{HR_{max}}.$$

Substituting this into the previous expression we have that

$$V_{M_j}^* - V_{M_j}(\pi_{j-1}) < \alpha$$

Which is the same statement as (1). ■

We will now prove a bound on the number of samples needed in line 11 in the above algorithm.

Lemma. Line 11 in the above algorithm will require (with probability $1-\delta$) less than TBD samples.

Proof. Fix i from the algorithm and consider π_i and M. By line 7, the probability of exploring a state in L when running π_i on M is greater than $\frac{\alpha}{HR_{max}}$. Since M_{i+1} is the same as M but with some states in L replaced by absorbing states, we must also have that the probability of exploring a state L when running π_i on M_{i+1} is $\geq \frac{\alpha}{HR_{max}}$.

Since we choose (s,a,h) to be the state in L with the highest probability of being reached by running π_i on M_i, by the pigeonhole principle the probability of reaching (s,a,h) when running π_i on M_{i+1} is $\leq \frac{1}{SAH}\frac{\alpha}{HR_{max}} = \frac{\alpha}{SAH^2R_{max}}$.

Let n be the number of trajectories we collect and let X_i be an indicator random variable that is 1 if the ith trajectory reaches (s,a,h) and 0 otherwise. Let $S = x_1 + \ldots + x_n$. Then we want

$$P(S \geq K_1) \geq 1-\delta$$

$$\Rightarrow \sum_{i=0}^{K_1-1} \binom{n}{i} (\frac{\alpha}{SAH^2R_{max}})^i (1 - \frac{\alpha}{SAH^2R_{max}})^{n-i} \geq 1-\delta$$

We can then solve for δ or use some concentration inequality.

■

Proof (of Theorem). Follows from previous lemmas. ■

2.2 UCRL