INCREMENTAL REINFORCEMENT LEARNING

Abstract. Gradient methods fundamentally intertwine exploration and exploitation. For these techniques, determining if any gradient direction (in policy space) will improve the performance of the current policy can be arbitrarily difficult. To overcome these difficulties, an alternative performance measure which does not suffer from variance convergence problems is proposed and analyzed. This alternative performance measure has the same global optima as the average reward and can be optimized without decreasing the average reward given a sufficiently flexible function approximator.

1. Introduction

In recent years, there has been a growing interest in direct policy-gradient methods for approximate planning in large Markov decision problems (MDP). Direct policy-gradient methods seek to find a good policy π among some restricted class of policies, by following the gradient of the future reward. Gradient methods are essentially on-policy and so intertwine the fundamental exploration-exploitation tradeoff. We argue that this entanglement could lead to highly inefficient policy improvement and suggest how to incorporate exploration into these methods.

Earlier work has argued that random policies could lead to learning that is exponentially slow in the size of the state space [?]. For example, consider the simple MDP shown in Figure 1.1A (adapted from [?]). Under a policy that gives equal probability to all actions, the expected time to the goal from the left most state is $3(2^n - n - 1)$, and with $n = 50$, the expected time to the goal is about 10^{15}. Any gradient method initialized to this policy would have to run at least this long to obtain just one accurate estimate. The inherent problem is that in gradient methods, policy improvement at a state is tied to the current probability of visiting that state. Importance sampling methods have been formulated [?, ?] but these cannot salvage the situation when there is an exponentially low probability event with a significant contribution to the expectation of interest.

In online value function methods, this problem is seen as a lack of exploration. Active exploration is necessary to improve the policy at states that are not visited often in order to efficiently propagate rarely visited state value information to commonly visited states (see [?, ?]). Dynamic programming also improves the policy at all states: unlikely states will exhibit policy improvement even though this improvement might have little impact on the average reward.

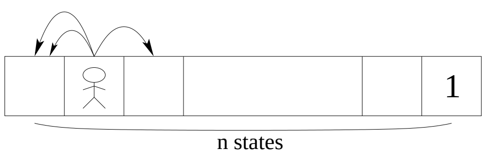

Figure 1.1. Example 1: Two actions move agent to the left and one actions moves agent to the right.

The average reward is what we ultimately seek to maximize. However, this performance measure has a fundamental problem: it is insensitive to performance improvement at highly unlikely states. Improvement at these states can lead to drastic policy improvement (as in figure 1.1). We formalize this by showing that it can be very difficult to improve a policy which has significantly sub-optimal performance. What is meant by difficult is that estimating the gradient *direction* (in policy space) when the gradient *magnitude* is small can be (at least) exponentially difficult (in the number of states). Although being close to an optimum implies the gradient has a small magnitude, a small magnitude does *not* imply being close to an optimum. The crucial detail is that the magnitude of the gradient can be small at plateaus far away from the optima.

- In section 2, the usual definition of the reinforcement learning problem is stated.
- In section 3, the problems with gradient methods are exhibited in more detail.
- In section 4, we discuss general incremental policy improvement.
- In section 5, an alternative performance measure is proposed which is sensitive to performance improvement at unlikely states (unlike the average reward). This alternative can (in principle) be improved with far fewer samples than the standard performance measure.
- In section 6, we discuss why richer function approximation schemes (such as non-parametric schemes) are necessary.

2. Preliminaries

First, let us state the standard formulation of a reinforcement learning problem. A reinforcement learning problem is a finite Markov decision process (MDP) with rewards which is a tuple (S, μ, A, R, P) where: S is finite set of states, μ is the starting state distribution, A is a finite set of actions, R is a reward function $R : S \times A \to [0, R_{\max}]$, and P is the transition model in which p^a_{ij} is the transition probability from state i to j under action a. The agent's decision making procedure is characterized by a stochastic policy $\pi(a|s)$, which is the probability of taking action a in state s.

If we make the assumption that *every policy π is ergodic, ie* has a well-defined stationary distribution p_{stat}, then the *average reward* is $\sum_{s,a} p_{\text{stat}}(s)\pi(a|s)R(s,a)$. Although in many situations, the goal of the agent is to maximize the average reward, we only consider the case of maximizing the γ- discounted average reward from the starting state distribution μ. This approximation allows us to avoid mathematical difficulties while arriving at a similar solution for processes that "restart" on a reasonable timescale [**?**].

We define the state-action value as

$$Q_\pi(s,a) \equiv E_\pi\{(1-\gamma)\sum_{t=0}^{\infty} \gamma^t R(s_t,a_t)|s_0 = s, a_0 = a\} \tag{2.1}$$

and the value function as

$$J_\pi(s) \equiv E_{\pi(a'|s)}\{Q_\pi(s,a')\}\ , \tag{2.2}$$

where and s_t and a_t are the state and action at time t. Thus, the goal of the agent is to maximize the discounted average reward:

$$\eta(\pi) = E_\mu\{J_\pi(s)\}\,.$$

A solution exists which simultaneously maximizes $J_\pi(s)$ for all states, and dynamic programming finds this solution. One last definition is required. The advantage is defined by:

$$A_\pi(s,a) \equiv Q_\pi(s,a) - J_\pi(s)\,. \tag{2.3}$$

We will sometimes suppress the π subscript when the policy is clear from context.

3. What is Wrong with Policy Gradients?

Before discuss the problems, a few definitions are in order. Given a fixed starting state distribution μ, we can define the γ- discounted future distribution of states as:

$$d_\pi(s) \equiv (1-\gamma)\sum_{t=0}^{\infty}\gamma^t E_{\pi,s_0\sim\mu}\Pr(s_t = s|s_0)$$

which is analogous to the stationary distribution in the infinite time setting. As $\gamma \to 1$, $d_\pi(s)$ tends to the stationary distribution, $p_{\text{stat}}(s)$.

The gradient of the discounted average reward from a starting state s_0 chosen according to μ is:

$$\nabla\eta(\pi) = \frac{1}{1-\gamma}\sum_{s,a} d_\pi(s)Q_\pi(s,a)\nabla\pi(a|s)$$

as shown in [?]. Let us define the following distribution on states and actions:

$$\nu(s,a) \equiv d_\pi(s)\pi(s,a)\,.$$

The gradient can be re-written as:

$$\nabla\eta(\pi) = \frac{1}{1-\gamma}E_{s,a\sim\nu}\left[\frac{\nabla\pi}{\pi}Q_\pi\right]\,, \tag{3.1}$$

which is a more convenient form with regards to sampling. Note that the expectation:

$$E_\nu x x' = \sum_{sa}\nu(s,a)x(s,a)x'(s,a)$$

can be seen as an inner product for vectors x and x' in the state-action space under the state-action distribution $d_\pi(s)\pi(a|s)$, and this view will be useful throughout this paper.

Let us consider the complexity of the most uninformed initial policy, namely a random or *undirected* policy, in MDPs where the task is to reach some goal state as quickly as possible. Undirected policies are of interest since they are common initializations. If a random actions are more likely to increase the distance to the goal state rather than decreasing it, a fundamental theorem by Whitehead [?, ?]is now applicable:

Theorem 3.1. *(by Whitehead) In any finite homogeneous problem solving task, the expected time using undirected exploration (such as random walk exploration) required to identify an optimal policy is bounded below by an expression exponential in the depth of the state space, if the number of actions leading away from the goal is larger than the the number of actions leading closer to the goal.*

Can this be stated mathematically? -John

The proof of the theorem shows that the expected time to the goal state on the first trial is exponential in the size of the state space. We share the belief with Whitehead and others ([?]) that many real-world problems fulfill the condition that a random action is expected to worsen the situation. As a special case of this theorem, policy gradient methods will be intractable for these classes of problems.

More generally let us understand the situations in which estimating the gradient direction will be difficult. In order to determine if gradient descent with respect to the i-th parameter, $\nabla_i \eta$, will improve the policy, it is crucial to obtain the correct sign of $E_\nu Q \frac{\nabla_i \pi}{\pi}$. Intuitively, we expect the difficulty of determining the sign to be proportional to the ration of the "noise" Σ to the "signal" μ.

Fundamentally, the only process available to us is running the MDP of the reinforcement learning problem. Given certain assumptions (see section 8), it is reasonable to suppose that we can obtain samples from $\nu(s,a)$ and that we can calculate a quantity $\hat{Q}_\pi(s,a)$ which is a nearly unbiased estimate of $Q_\pi(s,a)$.

Theorem 3.2. *Assume that samples s, a are drawn from some distribution $\nu(s,a)$ and that $\frac{\nabla_i \pi}{\pi}$ exists. For each sample, we are given an unbiased estimate $\hat{Q}(s,a)$, of $Q(s,a)$ with variance $\sigma^2(s,a)$. Let $\sigma_{min} = \min_{s,a} \sigma(s,a)$. Then the ratio of the variance, Σ, to the mean, μ, of the estimate $\hat{Q}(s,a)\frac{\nabla_i \pi}{\pi}$ of $\nabla_i \eta$ satisfies:*

$$\frac{\Sigma^2}{\mu^2} \geq \frac{\sigma^2_{min}}{E_\nu A^2} - 1\,.$$

Proof. Without loss of generality, we will assume that $\frac{\nabla_i \pi}{\pi}$ is normalized, *ie* that $E_\nu (\frac{\nabla_i \pi}{\pi})^2 = 1$. Since the $\hat{Q}$ are unbiased, the mean μ of the estimate $\hat{Q}(s,a)\frac{\nabla_i \pi}{\pi}$ is also unbiased. The variance of this estimate is:

$$\Sigma^2 = E_\nu \left(\frac{\nabla_i \pi}{\pi}\right)^2 \hat{Q}^2 - \mu^2 \tag{3.2}$$

$$= \sum_{s,a} \nu(s,a) \int dP(\hat{Q}(s,a))\hat{Q}^2(s,a) \left(\frac{\nabla_i \pi(a|s)}{\pi(a|s)}\right)^2 - \mu^2 \tag{3.3}$$

$$= \sum_{s,a} \nu(s,a) \left(\frac{\nabla_i \pi(a|s)}{\pi(a|s)}\right)^2 (Q^2(s,a) + \sigma^2(s,a)) - \mu^2\,. \tag{3.4}$$

We can lower bound Σ as follows:

$$\begin{aligned}
\Sigma^2 &\geq \sum_{s,a} \nu(s,a) \left(\frac{\nabla_i \pi(a|s)}{\pi(a|s)}\right)^2 \sigma^2(s,a) - \mu^2 \\
&\geq \sigma^2_{\min} \sum_{s,a} \nu(s,a) \left(\frac{\nabla_i \pi(a|s)}{\pi(a|s)}\right)^2 - \mu^2 \\
&= \sigma^2_{\min} - \mu^2
\end{aligned}$$

where have used the unit length of $\left(\frac{\nabla_i \pi(s,a)}{\pi(s,a)}\right)^2$.

Note that since $\sum_a \nabla \pi(a|s) = 0$ then $E_\nu \left[Q_\pi \frac{\nabla_i \pi}{\pi}\right] = E_\nu \left[A_\pi \frac{\nabla_i \pi}{\pi}\right]$. It follows that $\frac{\Sigma^2}{\mu^2} \geq \frac{\sigma^2_{\min}}{(E_\nu A \frac{\nabla_i \pi}{\pi})^2} - 1$, and so the $\frac{\nabla_i \pi}{\pi}$ that maximizes $E_\nu \left[A_\pi \frac{\nabla_i \pi}{\pi}\right]$ also minimizes the

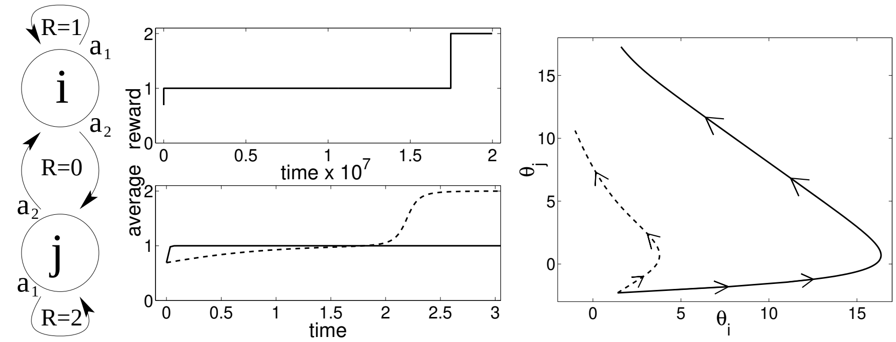

FIGURE 3.1. Example 2: A) See text. B top) The average reward vs. time (on a 10^7 scale) of a policy under standard gradient descent using the sigmoidal policy parameterization ($\pi(1; s, \theta_i) \propto \exp(\theta_i)/(1+\exp(\theta_i))$, with the initial conditions $\pi(i,1) = .8$ and $\pi(j,1) = .1$. B bottom) The average reward vs. time (unscaled) under standard gradient descent (solid line) and natural gradient descent (dashed line) for an early window of the above plot. D) Phase space plot for the standard gradient case (the solid line) and the natural gradient case (dashed line).

Sham, I don't understand the notation in this caption. What is s? What is "1"? -John

lower bound (subject to the normalization constraint). This occurs when $\frac{\nabla_i \pi}{\pi} = \frac{A}{\sqrt{E_\nu A^2}}$. Substitution leads to the result. □

Consider the case in which the large advantages are at states that are visited infrequently, so $E_\nu A^2$ will be small. This theorem shows provides evidence that it may be difficult to determine if the gradient will improve the policy for *any parameterization*.

Let us give an additional example demonstrating the problems for a simple two state MDP as shown in Figure 3.1B. Our example will use the common Gibbs table-lookup distribution, $\tilde{\Pi}_G = \{\pi_\theta : \pi(a|s) \propto \exp(\theta_{sa})\}$. The gradient of the discounted average reward (as shown later) is:

$$\partial_{sa}\eta(\pi) = \frac{1}{1-\gamma} d_\pi(s)\pi(a|s)A_\pi(s,a) \tag{3.5}$$

Increasing the chance of a self-loop at i decreases the stationary probability of j, which hinders the learning at state j. Under an initial policy that has the stationary distribution $\rho(i) = .8$ and $\rho(j) = .2$, learning at state i reduces the learning at state j leading to an an extremely flat plateau of improvement at 1 average reward shown in Figure 3.1C (see figure text). Figure 3.1D shows that this problem is so severe that $\rho(j)$ drops as low as 10^{-7} from it's initial probability of .2, which makes sampling the gradient difficult. Figure 3.1E shows where this occurs and gives a clearer picture (see figure text).

Sham, fill in the missing figures. -John

One question immediately arises: are we suffering from numerical error problems? The answer is no, as can be shown by a careful (and somewhat hairy) analysis of reinforcement learning with the Gibbs distribution. The analysis is given in section 7.

This example is remarkably discouraging for policy gradient approaches. We can easily make it even worse by using a few extra states in a chain as in figure 1.1.

4. Optimizing a Different Performance Measure

The fundamental problem with policy gradients is that η, which is what we ultimately seek to optimize, is insensitive to policy improvement at unlikely states. As shown in theorem 3.2, the gradient will be difficult to estimate in these cases where the advantages are hidden to the agent. However, policy improvement at these unlikely states might be necessary in order for the agent to achieve near optimal payoff.

We desire an alternative performance measure that does not down weight advantages at unlikely states or unlikely actions. The performance measure we choose will *require* access to an exploration policy or simulator which can achieve a more "uniform" distribution over states. A natural candidate for a performance measure is to weight the improvement from all states and all actions more uniformly (rather than by μ), such as

$$\omega(\pi) = \sum_s \sum_a \mu_e(s,a) Q_\pi(s,a)$$

where μ_e is some 'exploratory' starting *state-action* distribution. Note that for the special case when $\mu_e = \mu(s)\pi(a|s)$ then $\omega = \eta$. Under this performance measure, even if a state is not visited often (like the right most states in example 1), importance will be given to improving the future reward from this unlikely state.

Note to self: say more about why ω is sensible.

Since the policy that maximizes ω also simultaneously maximizes the future reward from every state, then maximizing ω also maximizes η. However, the problem with using a restricted class of parameterized policies, Π_θ, is that a policy which optimizes ω does not necessarily have high return on η. In order to exploit the benefits of exploration, the class of policies we consider must allow us to improve the policy in different areas without ultimately preventing us from improving η.

The subsections are outlined as follows:

- in 4.1, we state general conditions for policy improvement.
- in 4.2, we bound the sample size necessary to improve ω and argue why using an exploratory distribution μ_e is important
- in 4.3, we discuss the implications of improving ω rather than η

4.1. Incremental improvement.

For now, let us consider general conditions for improvement without regards to any restricted class of policies. Consider the incremental update rule:

$$\pi(a|s) \leftarrow \pi(a|s) + \alpha f(s,a) \tag{4.1}$$

where f is some vector in the space of state-actions. We later address how to choose an f at each point in time. Note that this update rule is to first order equal to the update rule $\pi(a|s) \propto \pi(a|s) \exp\left(\alpha \frac{f(s,a)}{\pi(s,a)}\right)$, which enforces positivity. Also, note that using an infinite α returns a greedy policy based on f. If f is equal to the true advantages A, then the probability of activ jons with largest advantages increases the fastest in a ratio sense.

The relevant γ- discounted future distribution of states is:

$$d_e(s) \equiv (1-\gamma)\sum_{t=0}^{\infty}\gamma^t E_{\pi, s_0 a_0 \sim \mu_e} \Pr(s_t = s | s_0, a_0)$$

where the starting state-action distribution is μ_e.

Theorem 4.1. *Let $\nu(s,a) = d_e(s)\pi(a|s)$ and let A be the true advantages. If the update rule is*

$$\pi(a|s) \leftarrow \pi(a|s) + \alpha f(s,a)$$

then the performance improvement of ω is:

$$\Delta\omega = \frac{\alpha}{1-\gamma} E_\nu A f + O(\alpha^2)\,. \tag{4.2}$$

Before we provide the proof, let us compare this to the gradient update rule. Note that the update rule for a gradient method is:

$$\pi(a|s) \leftarrow \pi(a|s) + \frac{\partial \pi}{\partial \theta}\Delta\theta$$

and so $f = \frac{\partial \pi}{\partial \theta}$. To first order, the performance change specified by the gradient, equation 3.1, is identical to the previous theorem.

Proof. For small enough α to ensure positivity, the normalized policy is

$$\pi_\alpha(a|s) = \frac{\pi(a|s) + \alpha f(s,a)}{1 + \alpha \sum_{a'} \pi(a'|s) f(s,a')}\,,$$

and so

$$\frac{\partial \pi_\alpha(a|s)}{\partial \alpha}|_{\alpha=0} = \pi(a|s)(f(s,a) - \sum_{a'} \pi(a'|s) f(s,a')\,.$$

Since the probabilities are normalized $\sum_a \nabla\pi(a|s) = 0$ and so $\sum_a Q(s,a)\nabla\pi(a|s) = \sum_a (A(s,a) + J(s))\nabla\pi(a|s) = \sum_a A(s,a)\nabla\pi(a|s)$. Hence,

$$\nabla\eta(\pi) = \frac{1}{1-\gamma} E_{s,a\sim\nu}\left[\frac{\nabla\pi}{\pi} Q_\pi\right] = \frac{1}{1-\gamma} E_\nu A \frac{\nabla\pi}{\pi}$$

It follows that, to first order in α,

$$\begin{aligned}
\Delta\eta &= \alpha\frac{\partial\eta}{\partial\alpha}|_{\alpha=0} \\
&= \frac{\alpha}{1-\gamma}\sum_{s,a} d_\pi(s)\pi(a|s)A(s,a)\frac{\pi(a|s)(f(s,a) - \sum_{a'}\pi(a'|s)f(s,a'))}{\pi(a|s)} \\
&= \frac{\alpha}{1-\gamma}\sum_{s,a} d_\pi(s)\pi(a|s)A(s,a)f(s,a) - (\sum_{s,a} d_\pi(s)\pi(a|s)A(s,a))(\sum_{a'}\pi(a'|s)f(s,a')) \\
&= \frac{\alpha}{1-\gamma}\sum_{s,a}\nu(s,a)A(s,a)f(s,a)
\end{aligned}$$

where we have used $\sum_a \pi(a|s)A(s,a) = 0$. □

Thus, to improve the policy it is only necessary that f must have a positive inner product with A. *Perhaps make a statement here comparing this to greedy dynamic programming*.

Note that this theorem implies that if we define $\tilde{f}(s,a) \equiv f(s,a) - \sum_{a'} \pi(a'|s) f(s,a')$ then:

$$\Delta\omega = \alpha E_\nu Q \tilde{f} + O(\alpha^2) .$$

This form is convenient since we can directly estimate $E_{d\pi} Q \tilde{f}$ using Monte-Carlo estimates of the state-action values. We now address how many samples are required to improve a sub-optimal policy.

4.2. **The difficulty of improving a sub-optimal policy.** Now let us consider for an arbitrary μ_e how difficult it is to improve ω using a direction given to us by some vector $f(s,a)$ for some sub-optimal policy. More precisely, given an f how many samples are sufficient to determine if the direction f or $-f$ will improve the policy? We again consider the sampling model used in theorem 3.2.

In order to do this, we must determine, with some certainty, the sign of $E_\nu Q \tilde{f}$, where

$$\tilde{f}(s,a) \equiv f(s,a) - \sum_{a'} \pi(a'|s) f(s,a')$$

Note that this constrained direction has the property that for all states s, $E_{\pi(a|s)} \tilde{f}(s,a) = 0$.

In the following theorem, θ is the angle between the vectors A and f in the state action space. The inner product then satisfies $E_\nu A f = \sqrt{(E_\nu A^2)(E_\nu f^2)} \cos\theta$.

Theorem 4.2. *Let* $\cos\theta = \frac{E_\nu A f}{\sqrt{(E_\nu A^2)(E_\nu f^2)}}$ *and assume f is a vector in the state-action space. Under the assumptions and sampling model of theorem 3.2, then given k independent samples, the estimate* $X = \frac{1}{k}\sum_{i=1}^{k} \hat{Q}(s,a)\tilde{f}(s,a)$ *has the*

$$\Pr(sign(X) \neq sign(E_\nu A f)) \leq \exp(-\frac{2k(E_\nu A)^2 \cos^2\theta}{5R_{max}^2})$$

Proof. The variance is:

$$\begin{aligned} \Sigma^2 &= E_\nu((\tilde{f}\hat{Q})^2) - \mu^2 \\ &\leq E_\nu(\tilde{f}^2) E_\nu(\hat{Q}^2) \end{aligned}$$

We can upper bound $E_\nu \hat{Q}^2$ by noting that each sample $\hat{Q}(s,a)$ is drawn from the interval $[0, R_{\max}]$ and $E\hat{Q}(s,a) = Q(s,a) \leq R_{\max}$. This implies that the maximum variance for any state-action is $\frac{R_{\max}^2}{4}$. Therefore, $E_\nu(\hat{Q}^2) = \sigma(s,a)^2 + Q(s,a)^2 \leq \frac{5R_{\max}^2}{4}$ and so:

$$\Sigma^2 \leq E_\nu(\tilde{f}^2) \frac{5R_{\max}^2}{4} .$$

Hoeffding's inequality for k independent identical random variables, $X \in [0, R_{\max}]$ states:

$$\Pr\left(|EX - \frac{1}{k}\sum_i X_i| > \epsilon \right) \leq e^{-\frac{2k\epsilon^2}{5R_{\max}^2}}$$

setting $\epsilon = EX = E_\nu A f = \sqrt{(E_\nu A^2)(E_\nu f^2)} \cos\theta$ gives the $\Pr\left(|E_\nu A f - \frac{1}{k}\sum_i X_i| > E_\nu A f \right)$ and the result follows. □

Note that if $E_\nu A^2$ is large then few samples are needed. In particular, the number of samples required to achieve a fixed error rate is $O\left(\frac{R_{\max}^2}{E_\nu A^2 \cos^2\theta}\right)$. This suggests that a uniform distribution is sensible since it allows $E_\nu A^2$ to have a significant value as long as any advantage is large. A uniform distribution in ν may be difficult to achieve by controlling μ_e.

Note that if the policy has near optimal performance, then all the advantages are small and we expect improvement to be difficult regardless of the performance measure (*ie* μ_e) being used since $E_\nu A^2$ is small. This is the common situation that close to a maxima it is difficult to determine if any direction will improve the function due to local flatness. The ideal situation is that the estimates only have high variance when the policy has near optimal performance. Unfortunately, as demonstrated by our two examples, this is not the case for η. The important question we now address is "How difficult is it to determine if f improves a sub-optimal policy?"

We measure the degree of sub-optimality by:

$$\varepsilon = \max_s |J(s) - J^*(s)|$$

which is the standard ∞-measure used in dynamic programming (see [?]). Under this definition, η could be at most ε-far from η^* in a worst case scenario.

The following theorem states how far from optimal a given policy is in terms of the largest advantage.

Theorem 4.3. *Let* $\varepsilon = \max_s |J(s) - J^*(s)|$ *and* $A_\infty = \max_{s,a} A(s,a)$. *Then*

$$\varepsilon \leq A_\infty\,. \tag{4.3}$$

Proof. By definition of the advantages, if we deviate from the current policy at any state s by taking action a, we gain, on average, $A(s,a)$ future reward. The change in $J_\pi(s)$ when we switch to any new policy π' is

$$J_{\pi'}(s) - J_\pi(s) = (1-\gamma)E_{a\sim\pi', s_t\sim P(s_t|s_o=s)}(\sum_t \gamma^t A_\pi(s_t,a_t))$$

, since the new policy deviates from the old at every time step. Each term in this sum is at most A_∞, so for any policy $J_{\pi'}(s) - J_\pi(s) \leq A_\infty$. The result follows from the special case when π' is optimal. $\square$

This bound is tight and is the best one could hope for some A_∞. As a lower bound, consider the case in which one state-action pair has a large advantage. The agent stands to gain that advantage by taking that action and can do so forever if the action returns the agent to the same state.

Let us consider for a given $E_\nu A^2$, how large A_∞ could be. Let (s',a') be the state-action pair with the largest advantage, so $E_\nu A^2 \geq \nu(s',a')A_\infty^2 \geq \nu_0 A_\infty^2$. Hence,

$$A_\infty \leq \sqrt{\frac{E_\nu A^2}{\nu_0}}$$

, where the bound is tight for the worst case scenario of 0 advantages everywhere except for the state with lowest probability. In practice, the reinforcement learning problem is (hopefully) not this difficult.

The importance of using exploration is that $E_{\nu_e} A^2$ will be small only when all advantages are small, where ν_e is the distribution implied by the exploratory distribution μ_e. Using the worst case bounds,

$$E_\nu A^2 \le \nu_0 \varepsilon^2 (1-\gamma)^2 \,. \tag{4.4}$$

The number of sample required to improve ω is $O(\frac{R_{\max}^2}{E_\nu A^2})$, neglecting the θ dependence. Hence, in the worst case scenario, when the policy is ε-near optimal, we need at most $O\left(\frac{1}{\varepsilon^2 \nu_0}\right)$ samples to estimate the gradient. If we use a μ_e that implies a uniform state-action distribution, then we need at most $O\left(\frac{n}{\varepsilon^2}\right)$ (where n is the number of state-actions). In contrast, for the on-policy ν, $E_\nu A^2$ can be exponentially small for the same policy, thus requiring many more samples to improve the policy.

Although we ultimately seek to improve η, the appropriate performance measure to use is one which weights $J(s)$ more uniformly. This stems from the fact that if $|J(s) - J^*(s)|$ is large for just one state, then all other states stand to gain significant improvement.

4.3. **What about η?** What has been lost by improving ω instead of η? Our real goal is improving η. Note that the end point of improving ω will be a policy which is optimal for criteria η. An important question is: "Can η decrease as we search for this near-optimal policy?" In general, the answer is "yes" because we are only guaranteed to increase ω. To see this, consider a f which improves the policy under the the distribution $d_\pi(s)$ yet $-f$ improves the policy under $\tilde{d}_\pi(s)$. Such f can exist when the initial distributions μ and μ_e differ significantly. For these f, optimizing according to ω will decrease η. There *always* exist $f(s,a)$ for which optimizing $\omega(\pi)$ improves $\eta(\pi)$: consider $f(s,a) = A(s,a)$ everywhere.

In general, determining if any f or $-f$ improves η at every step is impractical, as shown previously. This is why we considered ω to begin with. The problematic cases for improving f are when $E_\nu(A^2)$ is small, where ν is the 'on-policy' distributions. However, if $E_\nu(A^2)$ is not small, then it is possible to determine if f or $-f$ improves η.

We now focus on the important $\cos\theta$ dependence, and show how it is possible to improve both ω and η in non-problematic case.

5. Policy Improvement (**incomplete**)

The question now is how to choose an f to improve the policy. There are two alternatives to improve the policy, which we now discuss. There is the brute force method of just choosing different f, possibly at random, to optimize ω. The second and more powerful method is to use function approximation to find an f that is aligned with A.

5.1. **The Brute Force Method.** We could just try to use a gradient method to improve ω with some parameterized class of policies Π_θ. There are two problems with doing this. The first is that the parameterized class of polices must be tailored such that f always overlaps with A. In many problems, such information might not be available. The second and more serious problem is that Π_θ might not be a rich enough set such that a near-optimal policy is contained, so when θ converges to a local optimum η might still be small. In order to achieve near-optimal performance, the "size" of the set Π_θ must be increased. For instance, if the class is parameterized by a neural network and θ converges to point where η is insufficiently small, more hidden units must be added.

5.2. **Function approximation and the need for exploration.** We now pose the policy improvement issue as a supervised learning problem with reasonable requirements. The function approximation strategy is to align f with A, and then use this f in an incremental update rule.

In practice, the l_2-error of f's approximation to A is a convenient error to minimize. Consider the case in which the advantages are fit such that

$$E_\nu (f - A)^2 \leq \delta \,.$$

Since $E_\nu (f - A)^2 = E_\nu A^2 - E_\nu f A + E_\nu f^2$, then $E_\nu f A \geq E_\nu A^2 - E_\nu (f - A)^2$. Therefore, if

$$E_\nu A^2 > \delta$$

then $E_v A f > 0$. This suggests the following incremental algorithm.

(1) Fit the advantages with mean squared error no more than δ.
(2) Perform a line search with the following update rule:

$$\pi(a|s) \leftarrow \pi(a|s)(1 + \alpha f(s, a))$$

(3) Terminate if no the ω does not increase.

Note that the previous algorithm will return a policy with the property that $E_\nu A^2 < \delta$. In the worst case situation, the ε degree of sub-optimality is given by equation 4.4, which implies $\nu_0 \varepsilon^2 (1 - \gamma)^2 < \delta$ or

$$\varepsilon < \sqrt{\frac{\delta}{\nu_0 (1 - \gamma)^2}}$$

Therefore, in the worst case, if ν is uniform, then a polynomially small δ is required to be ε close to optimal. Unfortunately, if ν is based on the current policy, then ν_0 could be exponentially small, which implies that an exponentially small generalization error is necessary to be ε close to optimal. Not surprisingly, we see that exploration is also necessary for function approximation.

Need to mention non-param. function approximation is necessary to achieve condition 1.

If we desire both ω and η to improve most of the time, let us consider a slightly different error measure. Let ν and $\tilde{\nu}$ be the distributions implied by μ and μ_e respectively. Consider

$$E_{\tilde{\nu}} (f - A)^2 + E_\nu (f - A)^2 \leq \delta$$

so that the error under both error metrics is δ small. **Need to say more about joint improvement.**

6. Discussion

(1) discuss sensible supervised learning methods.
(2) how to represent compact policies?
(3) discuss why the assumption that we have access to an exploration policy is reasonable.

References

7. Appendix

The goal of this section is constructing a proof that policy gradient will *eventually* work on a Gibbs table lookup distribution. Note that the time scale of "eventually" can be remarkably long.

The Gibbs table-lookup distribution is $\tilde{\Pi}_G = \{\pi_\theta : \pi(a|s) \propto \exp(\theta_{sa})\}$. Using the notation $\partial_{sa} \equiv \partial/\partial\theta_{sa}$,

$$\partial_{sa}\eta(\pi) = d_\pi(s)\pi(a|s)(Q_\pi(s,a) - J_\pi(s)) = d_\pi(s)\pi(a|s)A_\pi(s,a) \tag{7.1}$$

Note that for any deterministic policy (*ie* $\forall s\, \exists a$ s.t. $\theta_{sa} = \infty$) the gradient is zero. Since the gradient is 0 at all points at infinity, this suggests the dismal possibility that the gradient descent could lead to arbitrary deterministic policy. It is straight forward to show that if the stationary distribution, of the asymptotic policy, is nonzero for all states then this asymptotic policy is optimal. However, the non-trivial question is can $\partial_{sa}\eta_\pi$ tend to 0 due to $d(s)$ tending to 0 in a manner such that such that the optimal policy is not achieved. Note that the gradient $\partial_{sa}\eta(\pi)$ is zero if either the advantage is zero or if $d(s)$ is zero. The following theorem states that this can never occur, and the key to proof is dealing with this case (where learning to one state could stop due to a vanishing stationary probability). We only consider the infinitesimal learning rate case where $\theta_t = \theta_0 + \int_0^t \nabla\eta^{\pi_t}\, dt$.

Theorem 7.1. *Assume that π_∞ is the single limiting policy corresponding to $\lim_{t\to\infty}\theta_t$. Then π_∞ is globally optimal.*

Proof. Let $S^* = \{s | \exists\pi \text{ s.t. } \rho_\pi(s) \neq 0\}$. Our proof that π_∞ globally maximizes the average reward consists of showing that for all $s \in S^*$, $J^{\pi_\infty}(s) = \max_a Q^{\pi_\infty}(s,a)$, *ie* π_∞ partially satisfies the Bellman equations. When clear, the subscript on π_t is dropped, *eg* $\rho^{\pi_\infty} = \rho^\infty$. Assume for now that as $t \to \infty$ then for all $s \in S^*$ and actions a either $\pi_t(a|s) \to 0$ or $A^t(s,a) \to 0$. For all $\tilde{a} \in \text{argmax}_a Q^{\pi_\infty}(s,a)$, $\partial_{sa}\eta^t$ is positive for sufficiently large t, so $\pi_\infty(a|s) \neq 0$ and $A^\infty(s,a) = 0$. Therefore, π_∞ satisfies the Bellman equations for $s \in S^*$. It follows that if $\tilde{a} \notin \text{argmax}_a Q^{\pi_\infty}(s,a)$ then $\pi_\infty(\tilde{a}, s) = 0$, and so if t is sufficiently large $\partial_{s\tilde{a}}\eta^t < 0$, else there must exist a partial extrema for some $\tilde{t} \geq t$.

It is straightforward to show that $\lim_{t\to\infty}\partial_{sa}\eta^t = 0$, and so as $t \to \infty$ either $\rho^t \to 0$, $\pi_t(a|s) \to 0$, or $A^t(s,a) \to 0$. We now show that even if if $\rho^\infty \neq 0$, then $\pi_\infty(a|s) = 0$ or $A^\infty(s,a) = 0$, which completes the proof. Assume that for some s that $\rho^\infty(s) = 0$. By stationarity, $\rho^t(s) = \sum_{s_0} P(s_\tau = s|s_0)\rho^t(s_0)$, and so for any trajectory $s_0, a_0, s_1, a_1 \ldots s$, $\rho^t(s) \geq \rho^t(s_0)p^{a_0}_{s_0 s_1}\pi_t(a_0|s_0)\ldots p^{a_{\tau-1}}_{s_{\tau-1}s}\pi_t(a_{\tau-1}|s_{\tau-1})$. For every τ-length trajectory with $p^{a_i}_{s_i s_{i+1}} \neq 0$ and $\rho^\infty(s_0) \neq 0$, there must exist at least one s_i, a_i such that $\pi_\infty(a_i|s_i) = 0$, else $\rho^\infty(s) \neq 0$ or $s \notin S^*$. It can be shown that there exist some $\tilde{s}$, $\tilde{a}$ and positive constant $\tilde{\alpha}$ such that $\forall t\, \rho^t(s) \geq \tilde{\alpha}\rho^t(\tilde{s})\pi_t(\tilde{a}|\tilde{s})$, $\rho^\infty(\tilde{s}) \neq 0$, and $\pi_\infty(\tilde{a}|\tilde{s}) = 0$ (using a trajectory such that $\pi_\infty(a_i|s_i) = 0$ only for $i = 0$). Assume that $\pi_\infty(s) \neq 0$ and $A^\infty(s) \neq 0$ (else the conjecture is true), then it follows from equation 7.1 and the previous inequality that for each a, $|\int_0^\infty \partial_{sa}\eta^t dt| \geq \alpha|\int_0^\infty \partial_{\tilde{s}\tilde{a}}\eta^t dt|$ for some positive constant α. However, $\pi_\infty(\tilde{s}) = 0$ forces $|\int_0^\infty \partial_{\tilde{s}\tilde{a}}\eta^t dt|$ to be ∞, and so $|\int_0^\infty \partial_{sa}\eta^t dt|$ is ∞, contradicting $\pi_\infty(s) \neq 0$ and $A^\infty(s) \neq 0$. $\square$

8. How to Monte Carlo

In order to determine which direction is the "right" direction, we need to know how to estimate this right direction. This is a subtle problem when you must (in general) sample from the underlying reinforced MDP forever in order to acquire an unbiased estimate of the discounted future reward, $\hat{Q}(s, a)$.

Intuitively, if we run the Markov process for a time long compared to the discount γ, we will obtain an estimate $\hat{Q}(s, a)$ which is "nearly" unbiased, simply due to the fact that the contribution to the reward from future actions will necessarily be small.