TRPO Notes

1

1.1

Markov decision process (infinite horizon): $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \rho_0, \gamma)$:
$\mathcal{S}$: finite set of states
$\mathcal{A}$: finite set of actions
P: transition probability
r: $\mathcal{S} \to \mathbb{R}$
ρ_0: $\mathcal{S} \to \mathbb{R}$ is the distribution of the initial state s_0
$\gamma \in (0, 1)$: discount factor

Let π denote a stochastic policy: $\mathcal{S} \times \mathcal{A} \to [0, 1]$, and let $\eta(\pi)$ denote its expected discounted reward:

$$\eta(\pi) = \mathbb{E}_{s_0, a_0, \ldots}[\sum_{t=0}^{\infty} \gamma^t r(s_t)],$$

where $s_0 \sim \rho(s_0)$, $a_t \sim \pi(a_t|s_t)$, $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$. Use the following definition:

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}, a_{t+1}, \ldots}[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l})]$$
$$V_\pi(s_t) = \mathbb{E}_{a_t, s_{t+1}, \ldots}[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l})]$$
$$A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s)$$

The following expresses the expected return of another policy $\tilde{\pi}$ in terms of the advantage over π, accumulated over timestamps:

$$\eta(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{s_0, a_0, \ldots \sim \tilde{\pi}}[\sum_{t=0}^{\infty} \gamma^t A_\pi(s_t, a_t)] \tag{1}$$

or simply:

$$\eta(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{\tau \sim \tilde{\pi}}[\sum_{t=0}^{\infty} \gamma^t A_\pi(s_t, a_t)] \tag{2}$$

The expectation is taken over trajectories $\tau := (s_0, a_0, s_1, a_0, ...)$

proof of **Lemma 1**:
First note that $A_\pi(s,a) = \mathbb{E}_{s' \sim P(s'|s,a)}[r(s) + \gamma V_\pi(s') - V_\pi(s)]$. Therefore,

$$
\begin{aligned}
&\mathbb{E}_{\tau\sim\tilde{\pi}}[\sum_{t=0}^{\infty}\gamma^t A_\pi(s_t,a_t)] \\
&= \mathbb{E}_{\tau\sim\tilde{\pi}}[\sum_{t=0}^{\infty}\gamma^t (r(s_t) + \gamma V_\pi(s_{t+1}) - V_\pi(s_t))] \\
&= \mathbb{E}_{\tau\sim\tilde{\pi}}[-V_\pi(s_0)) + r(s_1) + \gamma V_\pi(s_1) + \gamma r(s_t) + \gamma^2 V_\pi(s_{t+1}) - \gamma V_\pi(s_t) + \gamma^2 r(s_t) + \gamma^3 V_\pi(s_{t+1}) - \gamma^2 V_\pi(s_t) + ...] \\
&= \mathbb{E}_{\tau\sim\tilde{\pi}}[-V_\pi(s_0)) + \sum_{t=0}^{\infty}\gamma^t r(s_t)] \\
&= -\mathbb{E}_{s_0}[V_\pi(s_0)] + \mathbb{E}_{\tau\sim\tilde{\pi}}[\sum_{t=0}^{\infty}\gamma^t r(s_t)] \\
&= -\eta(\pi) + \eta(\tilde{\pi})
\end{aligned}
$$

Rearranging, the result follows. Define $\bar{A}(s)$ to be the expected advantage of $\tilde{\pi}$ over π at state s:

$$\bar{A}(s) = \mathbb{E}_{a\sim\tilde{\pi}(\cdot|s)[A_\pi(s,a)]} \tag{3}$$

Lemma 1 can then be rewritten as:

$$\eta(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{\tau\sim\tilde{\pi}}[\sum_{t=0}^{\infty}\gamma^t \bar{A}(s)] \tag{4}$$

Note that a new L_π can be written as:

$$L_\pi(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{\tau\sim\pi}[\sum_{t=0}^{\infty}\gamma^t \bar{A}(s)] \tag{5}$$

The difference being whether the states are sampled using π or $\tilde{\pi}$.

1.2

Let $\rho_\pi(s)$ be the discounted visitation frequencies:

$$\rho_\pi(s) = P(s_0 = s) + \gamma P(s_1 = s) + \gamma^2 P(s_2 = s) + ...,$$

where $s_0 \sim \rho_0$ and the actions are chosen according to π. We can rewrite the previous **Lemma 1** with a sum over states instead of timesteps:

$$\eta(\tilde{\pi}) = \eta(\pi) + \sum_{t=0}^{\infty} \sum_s P(s_t = s|\tilde{\pi}) \sum_a \tilde{\pi}(a|s)\gamma^t A_\pi(s,a) \tag{6}$$

$$= \eta(\pi) + \sum_s \sum_{t=0}^{\infty} \gamma^t P(s_t = s|\tilde{\pi}) \sum_a \tilde{\pi}(a|s) A_\pi(s,a) \tag{7}$$

$$= \eta(\pi) + \sum_s \rho_{\tilde{\pi}}(s) \sum_a \tilde{\pi}(a|s) A_\pi(s,a) \tag{8}$$

This equation implies that any policy update $\pi \to \tilde{\pi}$ has a nonnegative expected advantage at every state s. We introduce the following local approximation to η:

$$L_\pi = \eta(\pi) + \sum_s \rho_\pi(s) \sum_a \tilde{\pi}(a|s) A_\pi(s,a) \tag{9}$$

If we have a parameterized policy π_θ, where $\pi_\theta(a|s)$ is a differentiable function of the parameter vector θ, then L_π matches η to first order; that is, for any parameter value θ_0:

$$L_{\pi_{\theta_0}} = \eta(\pi_{\theta_0})$$
$$\nabla_\theta L_{\pi_{\theta_0}}|_{\theta=\theta_0} = \nabla_\theta \eta(\pi_{\theta_0})|_{\theta=\theta_0}$$

which implies that a sufficiently small step $\pi_{\theta_0} \to \tilde{(}\pi)$ that improves $L_{\pi_{\theta_{old}}}$ will also improve η, but does not give any guidance on how big of a step to take.
To bound the difference between $\eta(\tilde{\pi})$ and $L_\pi(\tilde{\pi})$, we will bound the difference arising from each timestep. We will couple the policies so they define a joint distribution over pairs of actions:

Definition 1 $(\pi, \tilde{\pi})$ is an $\alpha-$coupled policy pair if it defines a joint distribution $(a, \tilde{a})|s$, such that $P(a \neq \tilde{a})|s \leq \alpha$ for all s. π and $\tilde{\pi}$ will denote the marginal distributions of a and $\tilde{a}$, respectively.

Computationally, $\alpha-$coupling means that if we randomly choose a seed for our random number generator, and then we sample from each of π and $\tilde{\pi}$ after settling that seed, the results will agree for at least fraction $1 - \alpha$ of seeds.

Lemma 2: Given that π, $\tilde{\pi}$ are $\alpha-$coupled policies, for all s,

$$|\bar{A}(s)| \leq 2\alpha \max_{s,a} |A_\pi(s,a)| \tag{10}$$

proof:

$$\begin{aligned}\bar{A}(s) &= \mathbb{E}_{\tilde{a}\sim\tilde{\pi}}[A_\pi(s,\tilde{a})] \\ &= \mathbb{E}_{(a,\tilde{a})\sim(\pi,\tilde{\pi})}[A_\pi(s,\tilde{a}) - A_\pi(s,a)]\end{aligned}$$

due to the fact that $\mathbb{E}_{a\sim\pi}[A_\pi(s,a)] = 0$. We then have:

$$\begin{aligned}\bar{A}(s) &= P(a \neq \tilde{a}|s)\mathbb{E}_{(a,\tilde{a})\sim(\pi,\tilde{\pi})|a\neq\tilde{a}}[A_\pi(s,\tilde{a}) - A_\pi(s,a)] \\ |\bar{A}(s)| &\leq \alpha \cdot 2 \max_{s,a} |A_\pi(s,a)|\end{aligned}$$

Lemma 3: Let $(\pi, \tilde{\pi})$ be an $\alpha-$coupled policy pair, then:

$$|\mathbb{E}_{s_t\sim\tilde{\pi}}[\bar{A}(s_t)] - \mathbb{E}_{s_t\sim\pi}[\bar{A}(s_t)]| \leq 2\alpha \max_s \bar{A}(s) \leq 4\alpha(1-(1-\alpha)^t) \max_s |A_\pi(s,a)| \tag{11}$$

proof: Given the policy pair we can also define $\tau, \tilde{\tau}$ to be the generated trajectories obtained from the two policies. We will consider the advantage of $\tilde{\pi}$ over π at timestep t, and decompose this expression based on whether π agrees with $\tilde{\pi}$ at all timesteps $i < t$. Same random seed is used to generate both trajectories:

Let n_t denote the number of times that $a_i \neq \tilde{a_i}$, for $i < t$, that is, the number of times that the two policies disagree with each other before time t.

$$\mathbb{E}_{s_t\sim\tilde{\pi}}[\bar{A}(s_t)] = P(n_t = 0)\mathbb{E}_{s_t\sim\tilde{\pi}|n_t=0}[\bar{A}(s_t)] + P(n_t > 0)\mathbb{E}_{s_t\sim\tilde{\pi}|n_t>0}[\bar{A}(s_t)] \tag{12}$$

The expectation decomposes similarly for actions sampled using π:

$$\mathbb{E}_{s_t\sim\pi}[\bar{A}(s_t)] = P(n_t = 0)\mathbb{E}_{s_t\sim\pi|n_t=0}[\bar{A}(s_t)] + P(n_t > 0)\mathbb{E}_{s_t\sim\pi|n_t>0}[\bar{A}(s_t)] \tag{13}$$

Note that $n_t = 0$ terms are equal:

$$\mathbb{E}_{s_t\sim\tilde{\pi}|n_t=0}[\bar{A}(s_t)] = \mathbb{E}_{s_t\sim\pi|n_t=0}[\bar{A}(s_t)] \tag{14}$$

substracting equation 12 from equation 13 we get:

$$\mathbb{E}_{s_t\sim\tilde{\pi}} - \mathbb{E}_{s_t\sim\pi} = P(n_t > 0)(\mathbb{E}_{s_t\sim\tilde{\pi}|n_t>0}[\bar{A}(s_t)] - \mathbb{E}_{s_t\sim\pi|n_t>0}[\bar{A}(s_t)]) \tag{15}$$

by definition of α, $P(\pi, \tilde{\pi}$ agree at timestep i) $\geq 1-\alpha$, so $P(n_t = 0) \geq (1-\alpha)^t$, and

$$P(n_t > 0) \leq 1 - (1-\alpha)^t \tag{16}$$

We then have:

$$\begin{aligned} |\mathbb{E}_{s_t\sim\tilde{\pi}}[\bar{A}(s_t)] - \mathbb{E}_{s_t\sim\pi}[\bar{A}(s_t)]| &\leq |\mathbb{E}_{s_t\sim\tilde{\pi}}[\bar{A}(s_t)]| + |\mathbb{E}_{s_t\sim\pi}[\bar{A}(s_t)]| \\ &\leq 4\alpha \max_{s,a} |A_\pi(s,a)| \end{aligned}$$

where the second inequality follows from **Lemma 3**, and the first inequality is the direct consequence of **Lemma 2**.

Plug equation 16 into equation 15 we get:

$$|\mathbb{E}_{s_t\sim\tilde{\pi}}[\bar{A}(s_t)] - \mathbb{E}_{s_t\sim\pi}[\bar{A}(s_t)]| \leq 4\alpha(1-(1-\alpha)^t) \max_{s,a} |A_\pi(s,a)| \tag{17}$$

which bounds the difference in expected advantage at each timestep t. We can sum over time to bound the difference between $\eta(\tilde{\pi})$ and $L_{\tilde{\pi}}$. Substracting equation 4 and 5, and defining $\epsilon = \max_{s,a} |A_\pi(s,a)|$,

$$\begin{aligned} |\eta(\tilde{\pi}) - L_\pi(\tilde{\pi})| &= \sum_{t=0}^{\infty} \gamma^t |\mathbb{E}_{\tau\sim\tilde{\pi}}[\bar{A}(s_t)] - \mathbb{E}_{\tau\sim\pi}[\bar{A}(s_t)]| \\ &\leq \sum_{t=0}^{\infty} \gamma^t \cdot 4\epsilon\alpha(1-(1-\alpha)^t) \\ &= 4\epsilon\alpha(\frac{1}{1-\gamma} - \frac{1}{1-\gamma(1-\alpha)}) \\ &= \frac{4\alpha^2\gamma\epsilon}{(1-\gamma)(1-\gamma(1-\alpha))} \\ &\leq \frac{4\alpha^2\gamma\epsilon}{(1-\gamma)^2} \end{aligned}$$

Aforementioned sufficiently small step $\pi_{\theta_0} \rightarrow \tilde{\pi}$ that improves $L_{\pi_{\theta_{old}}}$ will also improve η, but does not give us any guidance on how big of a step to take.

Kakade & Langford (2002) proposed a policy update scheme called conservative policy iteration, for which they could provide explicit lower bounds on the improvement of η. Let π_{old} be denote the current policy, and let $\pi' = \arg\max_{\pi'} L_{\pi_{old}}(\pi')$. The new policy π_{new} is defined to be the following mixture:

$$\pi_{new}(a|s) = (1-\alpha)\pi_{old}(a|s) + \alpha\pi'(a|s) \tag{18}$$

Kakade & Langford derived the following lower bound:

$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - \frac{2\epsilon\gamma}{(1-\gamma)^2}\alpha^2 \tag{19}$$

where $\epsilon = \max_s |\mathbb{E}_{a\sim\pi'(a|s)}[A_\pi(s,a)]|$ which so far only applies to equation 18, which is a policy class that is unwieldy and restrictive in practice.

1.3

Define total variance divergence $D_{TV}(p||q) = \frac{1}{2}\sum_i |p_i - q_i|$, and define:

$$D_{TV}^{max}(\pi, \tilde{\pi}) = \max_s D_{TV}(\pi(\cdot|s)||\tilde{\pi}(\cdot|s)) \tag{20}$$

Theorem 1: Let $\alpha = D_{TV}^{max}(\pi(\cdot|s)||\tilde{\pi}(\cdot|s))$. Then the following bound holds:

$$\eta(\pi_{\text{new}}) \geq L_{\pi_{\text{new}}} - \frac{4\epsilon\gamma}{(1-\gamma)^2}\alpha^2 \tag{21}$$

where $\epsilon = \max_{s,a} |A_\pi(s,a)|$.

The proof is already proved in the previous section(1.2). There is another proof that uses perturbation theory.

Next, we note the following relationship between total variation divergence and the KL divergence, in which $D_{TV}(p||q)^2 \leq D_{\text{KL}}(p||q)$. Let $D_{\text{KL}}^{\max} = \max_s D_{\text{KL}}(\pi(\cdot|s)||\tilde{\pi}(\cdot|s))$. As a direct consequence of **Theorem 1**:

$$\eta(\tilde{\pi}) \geq L_{\tilde{\pi}} - CD_{\text{KL}}^{\max}(\pi, \tilde{\pi}) \tag{22}$$

where $C = \frac{4\epsilon\gamma}{(1-\gamma)^2}$ We thus give the following **Algorithm 1** based on policy improvement bound in equation 22. Note that for now, we assume evaluation of the advantage values A_π.

Algorithm 1 Policy iteration guaranteeing non-decreasing expected return η

Initialize π_0
for $i = 0, 1, 2, ...$ until convergence **do**
  Compute all advantage values $A_{\pi_i}(s,a)$
  Solve the constrained optimization problem
  $\pi_{i+1} = \arg\max_\pi [L_{\pi_i} - CD_{\text{KL}}^{\max}(\pi_i, \pi)]$, where $C = 4\epsilon\gamma/(1-\gamma)^2$
  and $L_{\pi_i} = \eta(\pi_i) + \sum_s \rho_{\pi_i}(s) \sum_a \pi(a|s) A_{\pi_i}(s,a)$
end for

It follows from **Theorem 1** that **Algorithm 1** is guaranteed to generate a monotonically improving sequence of policies $\eta(\pi_0) \leq \eta(\pi_1) \leq \eta(\pi_2) \leq ...$ To see this, Let $M_i(\pi) = L_{\pi_i}(\pi) - CD_{\text{KL}}^{\max}(\pi, \tilde{\pi})$, then:

$$\eta(\pi_{i+1}) \geq M_i(\pi_{i+1}) \tag{23}$$

given by equation 22. Further, we claim that $\eta(\pi_i) = M_i(\pi_i)$. This is true since when $\pi = \tilde{\pi}$, the term $CD_{\text{KL}}^{\max}(\pi, \tilde{\pi})$ will be effectively 0, thus the left and right side of the inequality will be the same. Then we have:

$$\eta(\pi_{i+1}) - \eta(\pi_i) \geq M_i(\pi_{i+1}) - M_i(\pi_i) \tag{24}$$

This algorithm is a type of minorization-maximization(MM), which is a class of methods that include EM(expectation-maximization).

1.4

Overload the previous notation to use functions of θ rather than π. Proceeding section shows that we are guaranteed to improve the true objective η by maximizing the following term:

$$\max_{\theta}[L_{\theta_{\text{old}}}(\theta) - CD_{\text{KL}}^{\max}(\theta_{\text{old}}, \theta)] \tag{25}$$

If we directly use the penalty coefficient C recommended by the theory above, the step size would be very small, one way to increase the step size in a robust way is to use a constraint on the KL divergence between the new policy and the old policy, i.e. a trust region constraint:

$$\max_{\theta} L_{\theta}(\theta) \tag{26}$$

subject to $D_{\text{KL}}^{\max}(\theta_{\text{old}}, \theta) \leq \delta$. In that case the step size which is included in C can be arbitrarily large as long as the contraint remains valid. The problem imposes a constraint that the KL divergence is bounded at every point in the state space. While it is motivated by the theory, the problem is impractical to solve due to a large number of constraints. Instead, we can use a heuristic approximation which considers the average KL divergence:

$$\bar{D}_{\text{KL}}^{\rho}(\theta_1, \theta_2) := \mathbb{E}_{s\sim\rho}[D_{\text{KL}}(\pi_{\theta_1}(\cdot|s)||\pi_{\theta_2}(\cdot|s))] \tag{27}$$

We therefore propose solving the following optimization problem to generate a policy update:

$$\max_{\theta} L_{\theta_{\text{old}}}(\theta) \tag{28}$$

subject to $\bar{D}_{\text{KL}}^{\rho}(\theta_{\text{old}}, \theta) \leq \delta$.

We seek to solve the following optimization problem, obtained by expanding $L_{\theta_{\text{old}}}$:

$$\max_{\theta} \sum_{s} \rho_{\theta_{\text{old}}}(s) \sum_{a} \pi_{\theta}(a|s) A_{\theta_{\text{old}}}(s, a) \tag{29}$$

subject to $\bar{D}_{\text{KL}}^{\rho_{\theta_{\text{old}}}}(\theta_{\text{old}}, \theta) \leq \delta$

We first replace $\sum_s \rho\theta_{\text{old}}(s)[...]$ by the expectation $\frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_{\theta_{\text{old}}}}[...]$. Next, replace the advantage value $A_{\theta_{\text{old}}}$ by the Q-value $Q_{\theta_{\text{old}}}$, which only changes the objective by a constant. Last, we replace the sum over the actions by an importance sampling estimator. Use q to denote the sampling distribution, the contribution of a single s_n to the loss function is:

$$\sum_{a}(a|s_n)A_{\theta_{\text{old}}}(s_n, a) = \mathbb{E}_{a\sim q}[\frac{\pi_{\theta}(a|s_n)}{q(a|s_n)}A_{\theta_{\text{old}}}(s_n|a)] \tag{30}$$

The optimization objective in equation 29 is equivalent to the following one:

$$\max_{\theta} \mathbb{E}_{s\sim\theta_{\text{old}}, a\sim q}[\frac{\pi_{\theta}(a|s)}{q(a|s)}Q_{\theta_{\text{old}}}(s|a)] \tag{31}$$

subject to $\mathbb{E}_{s\sim\rho_{\theta_{\text{old}}}}[D_{\text{KL}}(\pi_{\theta_{\text{old}}}(\cdot|s)||\pi_{\theta}(\cdot|s))] \leq \delta$
All we have to do is to replace the expectation by sample averages and replace the Q valueby an empirical estimate.

1.4.1 Single path

Collect a sequence of states by sampling $s_0\sim\rho_0$ and then simulating the policy $\pi_{\theta_{\text{old}}}$ to generate a trajectory $s_0, a_0, s_1, a_1, ..., s_{T-1}, a_{T-1}, s_T$, in which $q(a|s)$ is equivalent to $\pi_{\theta_{\text{old}}}(a|s)$. $Q_{\theta_{\text{old}}}(s, a)$ is computed at each state-action pair (s_t, a_t) by taking the discounted sum of future rewards along the trajectory.

1.4.2 Vine path

First sample $s_0 \sim \rho_0$ and simulate the policy π_{θ_i} to generate a number of trajectories. We then choose a subset of N states along these trajectories, called rollout set, $s_1, s_2, ..., s_N$. For each set s_n in the rollout set, we sample K actions according to $a_{n,k} \sim q(\cdot|s_n)$. Any choice of $q(\cdot|s_n)$ with a support that includes the $\pi_{\theta_i}(\cdot|s_n)$ will produce a consistent estimator. Use identical distribution of $\pi_{\theta_i}(\cdot|s_n)$ will work well on continuous problems such as robotics locomotion, while uniform distribution works well on discrete problems such as Atari games. For each action $a_{n,k}$ sampled, we estimate $\hat{Q}_{\theta_i}(s_n, a_{n,k})$ by performing a rollout(a short trajectory starting with s_n and action $a_{n,k}$). We can greatly reduce the variance of Q-value differences by using the same random number sequence and for the noise in each of the K rollouts, i.e. common random number.

In small finite space, we can generate a rollout for every possible action from a given state. The contribution to $L_{\theta_{\text{old}}}$ from a single state s_n is as follows:

$$L_n(\theta) = \sum_{k=1}^{K} \pi_\theta(a_k|s_n)\hat{Q}(s_n, a_k) \tag{32}$$

where the action space is $\mathcal{A} = a1, a2, ..., a_K$. In large or continuous state spaces, we can construct an estimator of the surrogate objective using importance sampling. The self-normalized estimator of $L_{\theta_{\text{old}}}$ obtained at a single state s_n is:

$$L_n(\theta) = \frac{\sum_{k=1}^{K} \frac{\pi_\theta(a_{n,k}|s_n)}{\pi_{\theta_{\text{old}}}(a_{n,k}|s_n)} \hat{Q}(s_n, a_k)}{\sum_{k=1}^{K} \frac{\pi_\theta(a_{n,k}|s_n)}{\pi_{\theta_{\text{old}}}(a_{n,k}|s_n)}} \tag{33}$$

Note that the gradient is unchanged by adding a constant to the Q-value. Averaging over $s_n \sim \rho(\pi)$, we obtain an estimator for $L_{\theta_{\text{old}}}$ as well as its gradient. Single path method can be directly applied on a physical system, while vine method while possessing lower variance for local estimate of Q-value, requires far more calls to the simulator for each of these advantage estimates, and requires the system to be reset to arbitrary states.

1.5

Practical algorithm:

- Use the single path or vine methods to collect a set of state-action pairs along with Monte Carlo estimates of their Q-values.
- By averaging over samples, construct the estimated objective and constraint in equation 31
- Approximately solve this constrained optimization problem to update the policy's parameter vector θ. We use the conjugate gradient algorithm followed by a line search, which is altogether only slightly more expensive than computing the gradient itself.

For the 3rd step, we compute the Fisher information matrix (FIM) by analytically computing the Hessian of the KL divergence, rather than using the covariance matrix of the gradients. That is, we estimate A_{ij} as $\frac{1}{N}\sum_{n=1}^{N} \frac{\partial^2}{\partial\theta_i \partial\theta_j} D_{\text{KL}}(\pi_{\theta_{\text{old}}}(\cdot|s_n)||\pi_\theta(\cdot|s_n))$ rather than using $\frac{1}{N}\sum_{n=1}^{N} \frac{\partial}{\partial_{\theta_i}} \log \pi_\theta(a_n|s_n) \frac{\partial}{\partial\theta_j} \log \pi_\theta(a_n|s_n)$, which has computational benefits in the large-scale setting, since it removes the need to store a dense Hessian or all policy gradients from a batch of trajectories. The rate of improvement of the policy is similar to empirical FIM.

1.5.1 Efficiently solving the Trust-Region Constrained Optimization Problem

To solve the following constrained optimization problem:

$$\max L(\theta) \text{ subject to } \bar{D}_{\text{KL}}(\theta_{\text{old}}, \theta) \leq \delta \tag{34}$$

- compute the search direction, using a linear approximation to the objective and quadratic approximation to the constraint
- perform a line search in that direction, ensuring that we improve the nonlinear objective while satisfying the nonlinear constraint

The search direction is computed by approximately solving the equation $Ax = g$, where A is the FIM matrix, i.e. the quadratic approximation to the KL divergence constraint: $\bar{D}_{\mathrm{KL}}(\theta_{\mathrm{old}}, \theta) \approx \frac{1}{2}(\theta - \theta_{\mathrm{old}})^T A(\theta - \theta_{\mathrm{old}})$ where $A_{ij} = \frac{\partial}{\partial \theta_i}\frac{\partial}{\partial \theta_j}\bar{D}_{\mathrm{KL}}$. In large-scale problems, it is prohibitively costly (with respect to computation and memory) to form the full matrix A or A^{-1}. However, the conjugate gradient algorithm allows us to approximately solve the equation $Ax = b$ without forming the full matrix, when we merely have access to a function that computes matrix-vector products $y \to Ay$.

1.5.2 Efficiently computing the Fisher-vector product

Supposed $\mu_\theta(x)$ parameterizes the distribution $\pi(u|x)$, now the KL divergence for a given input x can be written as follows:

$$D_{\mathrm{KL}}(\pi_{\theta_{\mathrm{old}}}(\cdot|x)||\pi_\theta(\cdot|x)) = kl(\mu_\theta(x), \mu_{\mathrm{old}}) \tag{35}$$

where kl is the KL divergence between the distributions corresponding to the two mean parameter vectors. Differentiating kl twice with respect to θ, we obtain:

$$\frac{\partial \mu_a(x)}{\partial \theta_i}\frac{\partial \mu_b(x)}{\partial \theta_j} kl''_{ab}(\mu_\theta(x), \mu_{\mathrm{old}}) + \frac{\partial^2 \mu_a(x)}{\partial \theta_i \partial \theta_j} kl'_a(\mu_\theta(x), \mu_{\mathrm{old}}) \tag{36}$$

where the primes (') indicate differentiation with respect to the first argument, and there is an implied summation over indices a, b. The second term vanishes, leaving just the first term. Let $J := \frac{\partial \mu_a(x)}{\partial \theta_i}$, then the FIM matrix can be written in terms of $J^T M J$, where $M = kl''_{ab}(\mu_\theta(x), \mu_{\mathrm{old}})$ is the FIM matrix of the distribution in terms of the mean parameter as opposed to the parameter θ.

Having computed the search direction $s \approx A^{-1}g$, we next need to compute the next step length β such that $\theta + \beta s$ will satisfy the KL divergence constraint. To do this, let $\delta = \bar{D}_{\mathrm{KL}} \approx \frac{1}{2}(\beta s)^T A(\beta s) = \frac{1}{2}\beta^2 s^T A s$. We then obtain $\beta = \sqrt{2\delta / s^T A s}$, where δ is the desired KL divergence.

Last, we use a line search to ensure improvement of surrogate objective and satisfaction of KL divergence constraint, both of which are nonlinear in the parameter vector θ (and thus depart from the linear and quadratic approximations used to compute the step). We perform the line search on the objective $L_{\theta_{\mathrm{old}}}(\theta) - \mathcal{X}[\bar{D}_{\mathrm{KL}}(\theta_{\mathrm{old}}, \theta) \leq \delta]$, where the function $\mathcal{X}[...]$ returns 0 is the argument is true and $+\infty$ if the argument is false. Starting the maximal value of step length β computed in the previous paragraph, we shrink β exponentially until the objective improves. Without this line search, the algorithm occasionally computes large steps that cause a catastrophic degradation of performance.
To summarize:

- optimize surrogate objective with a penalty on KL divergence. However, the large penalty coefficient C leads to prohibitively small steps, so we would like to decrease its multiplier. We choose a hard constraint δ
- The constraint $D^{\max}_{\mathrm{KL}}(\theta_{\mathrm{old}}, \theta)$ is hard for numerical optimization and estimation, so instead we constrain $D_{\mathrm{KL}}(\theta_{\mathrm{old}}, \theta)$

1.6

The policy, which is conditional probability distribution $\pi_\theta(a|s)$ can be parameterized with a neural network. This neural network maps from state vector s to a vector μ, which specifies a distribution over action space. Then we can compute the likelihood $p(a|\mu)$ and sample $a{\sim}p(a|\mu)$.

The policy is defined by the normal distribution $\mathcal{N}(\text{mean} = \text{NeuralNet}(s; W_i, b_i{}_{i=1}^{L}), \text{stdev} = \exp(r))$. For experiments with discrete actions (Atari), we use a factored discrete action space, where each factor is a parameterized as a categoritcal distribution. That is, the action consists of a tuple $(a_1, a_2, ..., a_K)$ of integers $a_k \in 1, 2, ..., N_k$, and each of the components is assumed to have a categorical distribution $\mu_k = [p_!, p_2, ..., p_{N_k}]$. Hence, μ is defined to be the concatenation of the factors' parameters: $\mu = [\mu_1, \mu_2, ..., \mu_K]$ and has dimension $\dim\mu = \sum_{k=1}^{K} N_k$.