Hard and soft constraints

Let S be the state space, A be the action space, $T : S \times A \rightarrow S$ be the transition function, $R : S \times A \rightarrow R$ be the reward function.

- K Hard safety constraints: For each $s \in S$, $1 \leq k \leq K$, let $\mathcal{C}_k(s)$ be the indicator function of the event where s violates the $k-$th safety constraint.
- J Soft constraints: For each $a \in A$, $1 \leq j \leq J$, let $\mathcal{G}_j(a) \leq 0$ denotes the preferred domain of the action space and, in principle, we want $\mathcal{G}_j(a)$ as negative as possible.

Constrained Optimization Formulation 1

The optimization problem can be formulated as

$$\begin{aligned} &\max_{\pi} \mathbf{E}\left[\sum_{t=0}^{T-1} \gamma^t \mathcal{R}(s_t, a_t) - \sum_{j=1}^{J} \nu_j q_j\right], \\ &\text{s.t.} \sum_{t=0}^{T-1}\sum_{k=1}^{K} \mathcal{C}_k(s_t) = 0, \text{ and } \sum_{t=0}^{T-1} \mathcal{G}_j(a_t) \le q_j, \ \forall 1 \le j \le J. \end{aligned} \tag{1}$$

In this formulation, ν_j's indicate how much we care about the soft constraints. It is equivalent to the following problem:

$$\begin{aligned} \min_{\substack{\lambda_0 \\ \lambda_j \ge 0, \forall 1 \le j \le J}} \max_{\pi} \mathbf{E}\Bigg[&\sum_{t=0}^{T-1} \gamma^t \mathcal{R}(s_t, a_t) + \sum_{j=1}^{J} (\lambda_j - \nu_j) q_j \\ &-\lambda_0 \sum_{t=0}^{T-1}\sum_{k=1}^{K} \mathcal{C}_k(s_t) - \sum_{j=1}^{J} \lambda_j \sum_{t=0}^{T-1} \mathcal{G}_j(a_t)\Bigg]. \end{aligned} \tag{2}$$

Constrained Optimization Formulation 2

The optimization problem can be formulated as

$$\begin{aligned} &\max_{\pi} \mathbf{E}\left[\sum_{t=0}^{T-1} \gamma^t \mathcal{R}(s_t, a_t)\right], \\ &\text{s.t.} \sum_{t=0}^{T-1} \sum_{k=1}^{K} \mathcal{C}_k(s_t) = 0, \text{ and } \mathbf{E}\sum_{t=0}^{T-1} \mathcal{G}_j(a_t) \leq q_j, \ \forall 1 \leq j \leq J. \end{aligned} \tag{3}$$

In this formulation, q_j's need to be determined beforehand. To incorporate the discount factor, the last condition is further restricted to

$$\mathbf{E}\sum_{t=0}^{T-1} \gamma^t \mathcal{G}_j(a_t) \leq q_j(1-\gamma), \ \forall 1 \leq j \leq J, \tag{4}$$

for T not too large and γ close to 1. To include the discount factor on the right hand side, [?] we further restrict the condition:

$$\mathbf{E}\sum_{t=0}^{T-1} \gamma^t \mathcal{G}_j(a_t) \leq \sum_{t=0}^{T-1} \gamma^t q_j(1-\gamma)^2, \ \forall 1 \leq j \leq J. \tag{5}$$

Constrained Optimization Formulation 2: continued

It is equivalent to the following problem:

$$\min_{\substack{\lambda_0 \\ \lambda_j \geq 0, \forall 1 \leq j \leq J}} \max_{\pi} \mathbf{E}\left[\sum_{t=0}^{T-1} \gamma^t \hat{\mathcal{R}}(s_t, a_t)\right]. \tag{6}$$

where

$$\hat{\mathcal{R}}(s_t, a_t) = \mathcal{R}(s_t, a_t) - \lambda_0 \sum_{k=1}^{K} \mathcal{C}_k(s_t) - \sum_{j=1}^{J} \lambda_j(\mathcal{G}_j(a_t) - q_j(1-\gamma)^2) \tag{7}$$

Teacher Advice in function approximation: Intuition

We can learn a model by solving the value function of action a at state s: $Q_a(s)$. Then an advice can be presented as an if-then rule:

$$\text{if} \quad s \in S_{\mathrm{adv}} \quad \text{then} \quad Q_{a_s}(s) \geq c, \tag{8}$$

for certain $a_s \in A$ and $c > 0$. Alternatively, an advice can be given as preference of one action over the other:

$$\text{if} \quad s \in S_{\mathrm{adv}} \quad \text{then} \quad Q_{a_p}(s) \geq Q_{a_n}(s) + c, \tag{9}$$

for a pair $(a_p, a_n) \in A^2$ and $c > 0$.

Soft constraints via function approximation

For example let $A = \{a_1, ..., a_m\}$ be a finite set and $S = \mathbb{R}^n$. We want to find w_a and b_a, $a \in A$, such that

$$Q_a(s) \approx w_a \cdot s + b_a. \tag{10}$$

Suppose there are R pieces of advice. Let $S^r_{\mathrm{adv}} = \{B_r s \leq d_r\}$, $1 \leq r \leq R$. (In general, the advice domain does not have to be linear.) Then the advice is given by

$$\begin{aligned} &\text{if} \quad B_r s \leq d_r \\ &\text{then} \quad w_{a_p} \cdot s + b_{a_p} - w_{a_n} \cdot s - b_{a_n} - c \geq 0. \end{aligned}$$

Equivalent form of the if-then advice

Proposition

Assume the domain $S_{\mathrm{adv}} = \{Bs \leq d\}$ is non-empty. Then, the if-then advice is equivalent to the following system having a non-negative solution u:

$$\begin{aligned} Bu + w_{a_p} - w_{a_n} &= 0 \\ -d \cdot u + b_{a_p} - b_{a_n} &\geq c \end{aligned}$$

Soft constraints via function approximation formulation

Suppose (M_a, y_a), $a \in A$ be the training sets. Then the approximation of the value function Q can be formulated as:

$$
\begin{aligned}
\min_{w_a, b_a, q_a, z_r, \zeta_r, u_r \geq 0, \zeta_r + u_r > 0} \quad & \sum_{a=1}^{m} (\|w_a\| + \nu |b_a| + C\|q_a\|) + \sum_{r=1}^{R} (\mu_1 \|z_r\| + \mu_2 \zeta_r) \\
\text{subject to} \quad & |M_a w_a + b_a - y_a| \leq q_a, \quad \forall a \in A, \\
& |w_{a_p} - w_{a_n} + B_r u_r| \leq z_r, \quad \forall 1 \leq r \leq R, \\
& - d_r \cdot u_r + \zeta_r \geq c_r - b_{a_p} + b_{a_n}, \quad \forall 1 \leq r \leq R.
\end{aligned}
\tag{11}
$$

Algorithm

Algorithm 1 PPO-Clip

Input: initial policy parameters θ_0, initial value function parameters ϕ_0

for $k = 0, 1, 2, \ldots$ **do**

Collect set of trajectories $\mathcal{D}_k = \{\tau_i\}$ by running policy $\pi_k = \pi(\theta_k)$ in the environment.

Compute rewards-to-go $\hat{R}_t$.

Compute advantage estimates, $\hat{A}_t$ (using any method of advantage estimation) based on the current value function V_{ϕ_k}.

Update the policy by maximizing the PPO-Clip objective:

$$\theta_{k+1} = \arg\max_{\theta} \frac{1}{|\mathcal{D}_k| T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^{T} \min\left(\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_k}(a_t|s_t)} A^{\pi_{\theta_k}}(s_t, a_t), \;\; g(\epsilon, A^{\pi_{\theta_k}}(s_t, a_t)) \right),$$

typically via stochastic gradient ascent with Adam.

Fit value function by regression on mean-squared error:

$$\phi_{k+1} = \arg\min_{\phi} \frac{1}{|\mathcal{D}_k| T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^{T} \left(V_\phi(s_t) - \hat{R}_t \right)^2,$$

typically via some gradient descent algorithm.

end for