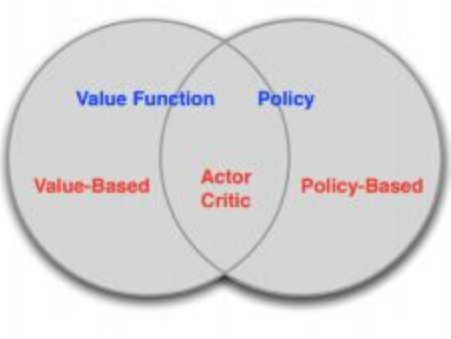
III. Actor-Critic

Intuition

Méthodes Actor-Critic :

- Critic : estime la fonction valeur (Q ou V)
  ↳ méthodes Value-Based
- Actor : met à jour la politique dans la direction suggérée par la critique (e.g. avec policy gradient)
  ↳ méthodes Policy-Based



Les fonctions Critic et Actor sont toutes les deux paramétrisées par des réseaux de neurones.

Forme générale des Policy Gradients

$\nabla_\theta J(\theta)$	Algorithme
$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{\lvert\tau\rvert-1}\nabla_\theta \log \pi_\theta(a_t\lvert s_t)\mathcal{R}(\tau)\right]$	REINFORCE
$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{\lvert\tau\rvert-1}\nabla_\theta \log \pi_\theta(a_t\lvert s_t)\sum_{t'=t}^{\lvert\tau\rvert-1} r(s_{t'},a_{t'})\right]$	REINFORCE causalité
$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{\lvert\tau\rvert-1}\nabla_\theta \log \pi_\theta(a_t\lvert s_t)\left(\left(\sum_{t'=t}^{\lvert\tau\rvert-1} r(s_{t'},a_{t'})\right)-b(s_t)\right)\right]$	REINFORCE causalité + baseline

$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{\lvert\tau\rvert-1}\nabla_\theta\log\pi_\theta(a_t\lvert s_t)\left(\left(\sum_{t'=t}^{\lvert\tau\rvert-1}\gamma^{t'-t}r(s_{t'},a_{t'})\right)-b(s_t)\right)\right]$	REINFORCE causalité + baseline + discount
$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{\lvert\tau\rvert-1}\nabla_\theta\log\pi_\theta(a_t\lvert s_t)(r_t+V^\pi(s_{t+1})-V^\pi(s_t))\right]$ avec $V^\pi(s_t)=\mathbb{E}_{s_{t+1:\infty},a_{t:\infty}}\left[\sum_{k=0}^{\infty}r_{t+k}\right]$	TD Actor-Critic
$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{\lvert\tau\rvert-1}\nabla_\theta\log\pi_\theta(a_t\lvert s_t)Q^\pi(s_t,a_t)\right]$ avec $Q^\pi(s_t,a_t)=\mathbb{E}_{s_{t+1:\infty},a_{t+1:\infty}}\left[\sum_{k=0}^{\infty}r_{t+k}\right]$	Q Actor-Critic
$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{\lvert\tau\rvert-1}\nabla_\theta\log\pi_\theta(a_t\lvert s_t)A^\pi(s_t,a_t)\right]$ avec $A^\pi(s_t,a_t)=Q^\pi(s_t,a_t)-V^\pi(s_t)$ (fonction avantage)	Advantage Actor-Critic

Policy Gradient :

- ⊕ biais faible
- ⊖ forte variance (un échantillon par estimation)

Actor-Critic :

- ⊕ plus faible variance (grâce à la critique)
- ⊖ biaisé (si la critique n'est pas parfaite)

Advantage Actor-Critic

En utilisant la fonction valeur V comme baseline, on obtient la fonction avantage :

$$A(s_t, a_t) = Q_w(s_t, a_t) - V_v(s_t)$$

Elle indique à quel point il est préférable d'effectuer l'action a_t par rapport à l'action moyenne effectuée en s_t.

En utilisant l'équation d'optimalité de Bellman,

$$Q(s_t, a_t) = \mathbb{E}[r_{t+1} + \gamma V(s_{t+1})]$$

la fonction avantage se réécrit comme :

$$A(s_t, a_t) = r_{t+1} + \gamma V_v(s_{t+1}) - V_v(s_t)$$

Ainsi, seul le réseau de neurones v correspondant à la fonction valeur V suffit.

On peut donc réécrire $\nabla_\theta J(\theta)$ de la manière suivante :

$$\begin{aligned}\nabla_\theta J(\theta) &\sim \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t)(r_{t+1} + \gamma V_v(s_{t+1}) - V_v(s_t)) \\ &= \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t) A(s_t, a_t)\end{aligned}$$

batch actor-critic algorithm:

1. sample $\{\mathbf{s}_i, \mathbf{a}_i\}$ from $\pi_\theta(\mathbf{a}|\mathbf{s})$ (run it on the robot)
2. fit $\hat{V}^\pi_\phi(\mathbf{s})$ to sampled reward sums
3. evaluate $\hat{A}^\pi(\mathbf{s}_i, \mathbf{a}_i) = r(\mathbf{s}_i, \mathbf{a}_i) + \gamma \hat{V}^\pi_\phi(\mathbf{s}'_i) - \hat{V}^\pi_\phi(\mathbf{s}_i)$
4. $\nabla_\theta J(\theta) \approx \sum_i \nabla_\theta \log \pi_\theta(\mathbf{a}_i|\mathbf{s}_i) \hat{A}^\pi(\mathbf{s}_i, \mathbf{a}_i)$
5. $\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$

online actor-critic algorithm:

1. take action $\mathbf{a} \sim \pi_\theta(\mathbf{a}|\mathbf{s})$, get $(\mathbf{s}, \mathbf{a}, \mathbf{s}', r)$
2. update $\hat{V}^\pi_\phi$ using target $r + \gamma \hat{V}^\pi_\phi(\mathbf{s}')$
3. evaluate $\hat{A}^\pi(\mathbf{s}, \mathbf{a}) = r(\mathbf{s}, \mathbf{a}) + \gamma \hat{V}^\pi_\phi(\mathbf{s}') - \hat{V}^\pi_\phi(\mathbf{s})$
4. $\nabla_\theta J(\theta) \approx \nabla_\theta \log \pi_\theta(\mathbf{a}|\mathbf{s}) \hat{A}^\pi(\mathbf{s}, \mathbf{a})$
5. $\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$

Generalized Actor Critic

$$\begin{aligned}
\hat{A}_t^{(1)} &:= \delta_t^V &&= -V(s_t) + r_t + \gamma V(s_{t+1}) \\
\hat{A}_t^{(2)} &:= \delta_t^V + \gamma\delta_{t+1}^V &&= -V(s_t) + r_t + \gamma r_{t+1} + \gamma^2 V(s_{t+2}) \\
\hat{A}_t^{(3)} &:= \delta_t^V + \gamma\delta_{t+1}^V + \gamma^2\delta_{t+2}^V &&= -V(s_t) + r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 V(s_{t+3})
\end{aligned}$$

$$\hat{A}_t^{(k)} := \sum_{l=0}^{k-1} \gamma^l \delta_{t+l}^V = -V(s_t) + r_t + \gamma r_{t+1} + \cdots + \gamma^{k-1} r_{t+k-1} + \gamma^k V(s_{t+k})$$

$$\hat{A}_t^{GAE(\gamma,\lambda)} = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_{t+l}^V$$

L'équation de mis à jour devient :

$$\theta \leftarrow \theta + \alpha \frac{1}{M} \sum_{\tau^{(i)}} \sum_{t=0}^{|\tau^{(i)}|-1} \hat{A}_t^{GAE(\gamma,\lambda)} \nabla_\theta \pi_\theta(a_t|s_t)$$

- Variance augmente lorsque λ augmente ($\lambda = 1 \rightarrow$ Monte-Carlo)
- Biais augmente lorsque λ diminue ($\lambda = 1 \rightarrow$ TD(0))

Traces d'éligibilité

On peut définir des traces d'éligibilité pour faire les mises à jour de θ au fur et à mesure du processus :

$$e_0 \leftarrow 0$$

$$e_t \leftarrow \lambda\gamma e_{t-1} + \nabla_\theta \log \pi_\theta(a_t|s_t)$$

On s'en sert pour pondérer le passé et faire des mises à jour à chaque étape de la trajectoire :

$$\delta_t = r_t + \gamma V_\phi(s_{t+1}) - V_\phi(s_t)$$

$$\theta \leftarrow \theta + \alpha\delta_t e_t$$