INSTITUTO DE MATEMATICA E ESTATÍSTICA
UNIVERSIDADE DE SÃO PAULO

VERÃO IME-USP

XLIX Programa de Verão (2020) - Introdução ao Aprendizado por Reforço

Actor-Critic

Ângelo Gregório Lovatto
aglovatto@ime.usp.br

IME - USP, 14/02/2019
LIAMF: Grupo PAR (Planejamento e Aprendizado por Reforço)

Aula 4 - Actor-Critic

Agenda

1. Revisão e Funções vantagem
2. Regularização por retornos descontados
3. Retornos truncados
4. Arquitetura A2C
5. Generalized Advantage Estimation (GAE)

Objetivos

- Familiarizar-se com a família de algoritmos Actor-Critic
- Entender o papel da função valor como *critic*
- Entender o compromisso entre viés e variância
- Implementar um algoritmo Actor-Critic (A2C)

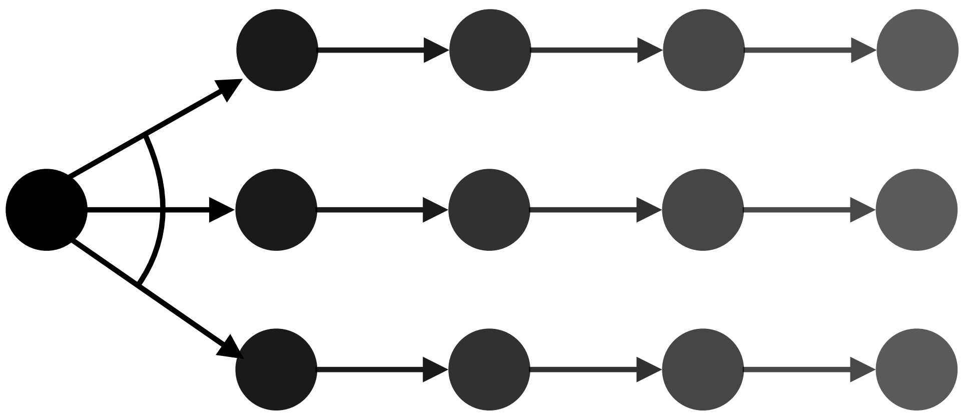
Policy Gradient + Reward-to-go + baseline (1/2)

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t)\left(\underbrace{\sum_{t'=t}^{T-1} r_{t'+1}}_{\textit{Reward-to-Go}} - \underbrace{V^{\pi_\theta}(s_t)}_{\textit{baseline}}\right)\right]$$

- Na aula passada derivamos o estimador REINFORCE com *reward-to-go* e *baseline*
- Uma escolha natural para o *baseline* é a função valor

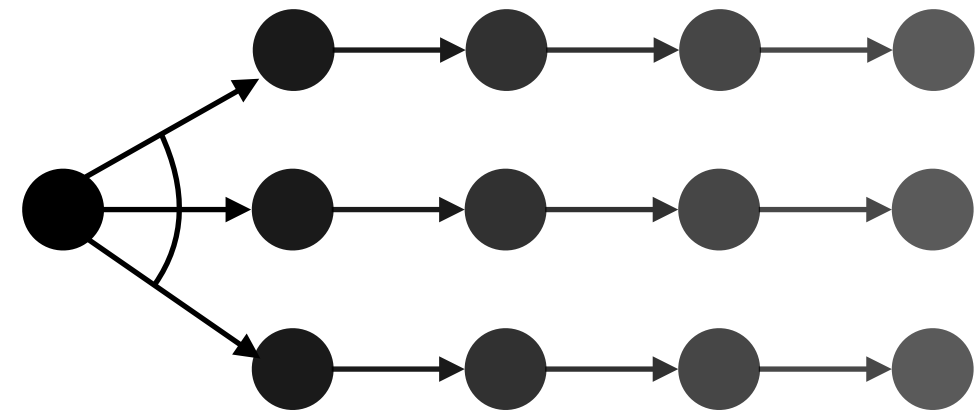
Função Ação-Valor $Q^{\pi}(s, a)$

$$V^{\pi_\theta}(s) = \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T-1} r_{t+1} \middle| s_0 = s\right]$$

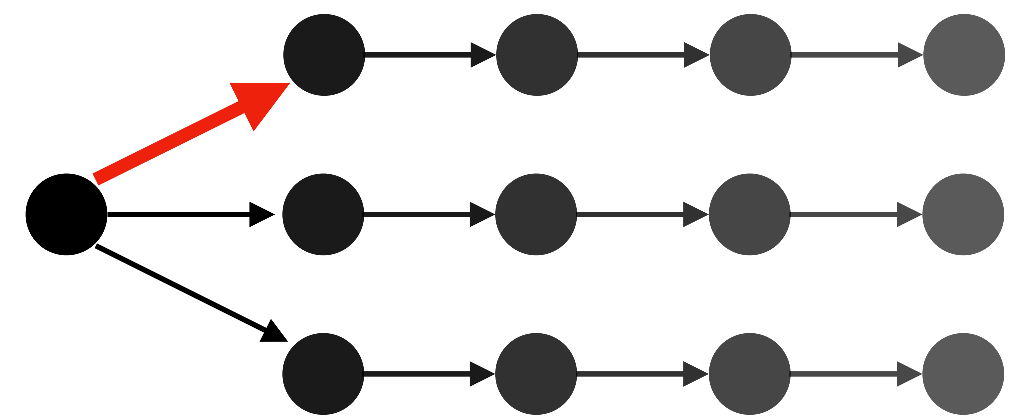


Função Ação-Valor $Q^{\pi}(s, a)$

$$V^{\pi_\theta}(s) = \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T-1} r_{t+1} \middle| s_0 = s\right]$$

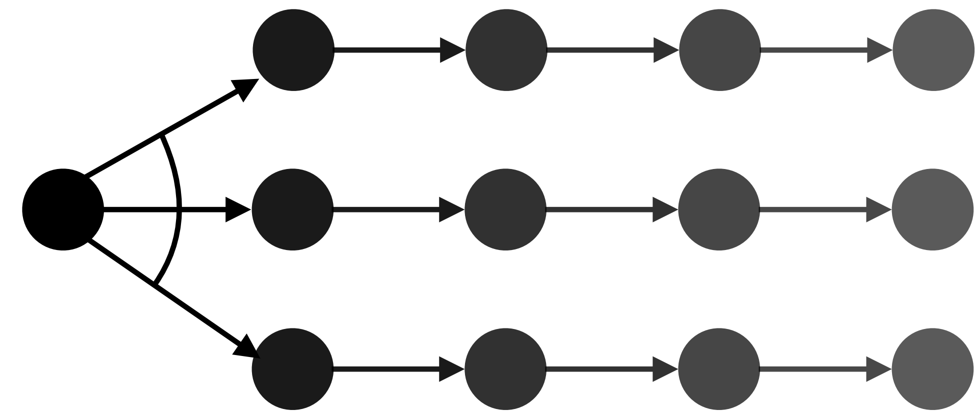


$$Q^{\pi_\theta}(s, a) = \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T-1} r_{t+1} \middle| s_0 = s, a_0 = a\right]$$



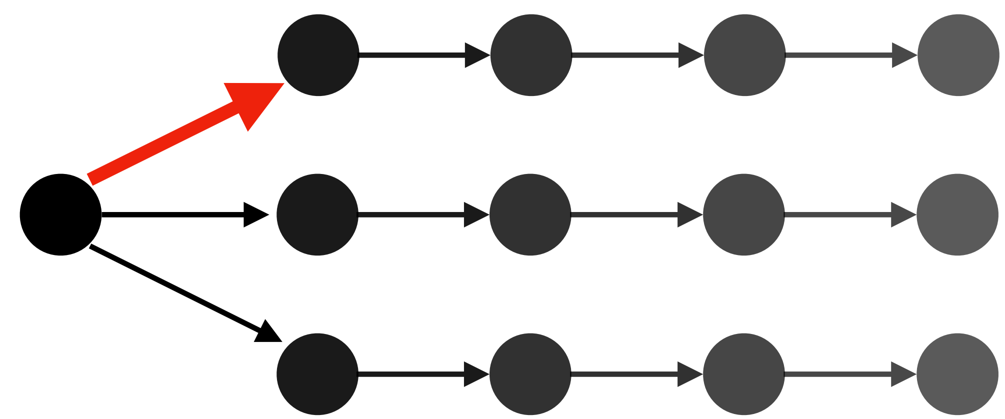
Função Ação-Valor $Q^{\pi}(s, a)$

$$V^{\pi_\theta}(s) = \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T-1} r_{t+1} \middle| s_0 = s\right]$$



$$Q^{\pi_\theta}(s, a) = \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T-1} r_{t+1} \middle| s_0 = s, a_0 = a\right]$$

Reward-to-Go



Policy Gradient + Reward-to-go + baseline (2/2)

$$Q^{\pi_\theta}(s,a) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1} r_{t+1} \middle| s_0 = s, a_0 = a\right]$$

$$\begin{aligned}\mathbb{E}_{\tau\sim\pi_\theta}&\left[\sum_{t=0}^{T-1}\nabla_\theta \log \pi_\theta(a_t|s_t)\left(\sum_{t'=t}^{T-1} r_{t'+1} - V^{\pi_\theta}(s_t)\right)\right]\\ &= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\nabla_\theta \log \pi_\theta(a_t|s_t)\left(Q^{\pi_\theta}(s_t,a_t) - V^{\pi_\theta}(s_t)\right)\right]\end{aligned}$$

- Note que o *reward-to-go* é o retorno observado dado um estado e ação iniciais
- A esperança desse valor é dada pela função ação-valor $Q^\pi(s,a)$
- Se tivéssemos essa função, poderíamos substituir o reward-to-go por $Q^\pi(s,a)$

Função Vantagem (*Advantage*)

$$\boxed{A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)}$$

$$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t)\left(Q^{\pi_\theta}(s_t,a_t) - V^{\pi_\theta}(s_t)\right)\right]$$
$$= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t) A^{\pi_\theta}(s_t,a_t)\right]$$

Função Vantagem (*Advantage*)

$$A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)$$

$$\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t)\left(Q^{\pi_\theta}(s_t,a_t) - V^{\pi_\theta}(s_t)\right)\right]$$

$$= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t) A^{\pi_\theta}(s_t,a_t)\right]$$

$A^{\pi_\theta}(s_t, a_t) > 0$	$\uparrow \pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)$	**Reforço positivo**
$A^{\pi_\theta}(s_t, a_t) < 0$	$\downarrow \pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)$	**Reforço negativo**

- A diferença entre a função ação-valor e a função valor é conhecida como função *vantagem*
- Intuitivamente, ela nos diz o valor da ação em relação à média naquele estado
- Ações acima da média são reforçadas. O contrário acontece com as abaixo da média

Limitações de REINFORCE

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta} \left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t) A^{\pi_\theta}(s_t, a_t) \right]$$

$$A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)$$

$$\approx \qquad \approx$$

$$\hat{A}_t = \left(\sum_{t'=t}^{T-1} r_{t'+1} \right) - V_\phi(s_t)$$

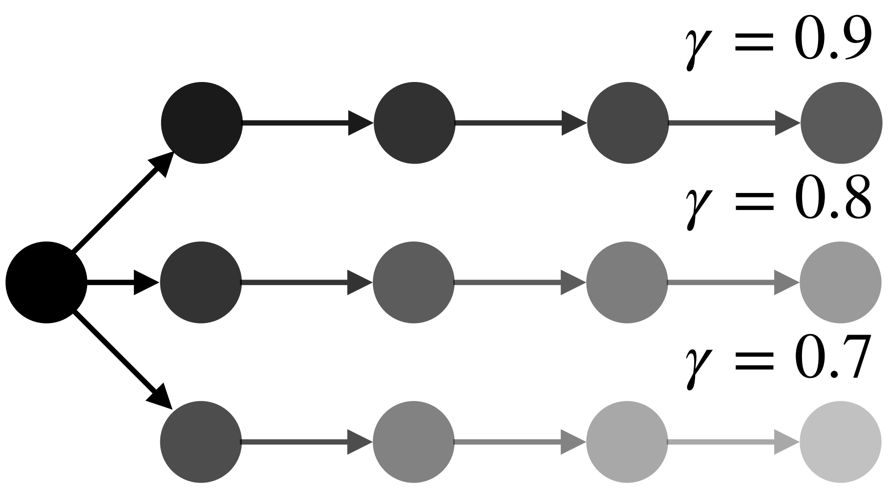
- Precisamos esperar que a trajetória termine para estimar o gradiente
- O estimador da vantagem leva em conta todas as recompensas futuras, levando a alta variância

Reduzindo a variância: temporal *discounting* (1/3)

$$V^{\pi_\theta,\gamma}(s) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s\right]$$

$$Q^{\pi_\theta,\gamma}(s,a) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s, a_0 = a\right]$$

$$A^{\pi_\theta,\gamma}(s,a) = Q^{\pi_\theta,\gamma}(s,a) - V^{\pi_\theta,\gamma}(s)$$



- Consideramos versões descontadas das funções valor vistas até agora
- *Discounting* funciona ao reduzir o peso de recompensas futuras
  - Reflete a noção de que dinheiro hoje vale mais que dinheiro amanhã
  - Ignora o efeito de dependências de longo prazo entre ações e recompensas

Reduzindo a variância: temporal *discounting* (2/3)

$$\begin{aligned}
\nabla J(\theta) &= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\nabla_\theta \log \pi_\theta(a_t \mid s_t)\sum_{t'=0}^{T-1} r_{t'+1}\right] \\
&= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\nabla_\theta \log \pi_\theta(a_t \mid s_t)\left(\underbrace{\cancel{\sum_{t'=0}^{t-1} r_{t'+1}}}_{\text{passado}} + \underbrace{\sum_{t'=t}^{T-1} r_{t'+1}}_{\text{futuro}}\right)\right] \\
&= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\nabla_\theta \log \pi_\theta(a_t \mid s_t)\sum_{t'=t}^{T-1} r_{t'+1}\right] \\
&\approx \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\nabla_\theta \log \pi_\theta(a_t \mid s_t)\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'+1}\right]
\end{aligned}$$

Anteriormente cortarmos as recompensas passadas. Agora também damos menos peso para recompensas futuras.

Reduzindo a variância: temporal *discounting* (3/3)

Consideramos uma versão enviesada do *policy gradient* original

$$\nabla J(\theta) \approx \mathbb{E}_{\tau \sim \pi_\theta} \left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t) A^{\pi_\theta,\gamma}(s_t, a_t) \right]$$

$$\begin{array}{ccccc} A^{\pi_\theta,\gamma}(s,a) = & Q^{\pi_\theta,\gamma}(s,a) & - & V^{\pi_\theta,\gamma}(s) \\ & \approx & & \approx \\ \hat{A}_t = & \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'+1} \right) & - & V_\phi(s_t) \end{array}$$

- Recompensas recebidas muito depois da escolha de uma ação tem peso menor para sua vantagem
- γ controla essa dependência temporal, ao custo de maior viés do estimador
- Cada vez mais veremos esse compromisso entre viés e variância

Estimando a vantagem: retornos de n passos (1/5)

$$\boxed{V^{\pi_\theta,\gamma}(s) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s\right] \qquad Q^{\pi_\theta,\gamma}(s,a) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s, a_0 = a\right]}$$

$$A^{\pi_\theta,\gamma}(s,a) = Q^{\pi_\theta,\gamma}(s,a) - V^{\pi_\theta,\gamma}(s)$$

- O valor de uma ação pode ser quebrado em recompensa imediata e valor do próximo estado
- A recompensa somada ao valor do próximo estado é conhecida como o **retorno de 1 passo**
    - Por um passo observamos a recompensa obtida
    - Sumarizamos o retorno esperado a partir do próximo passo com a função valor

Estimando a vantagem: retornos de n passos (1/5)

$$\boxed{V^{\pi_\theta,\gamma}(s) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum\nolimits_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s\right] \qquad Q^{\pi_\theta,\gamma}(s,a) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum\nolimits_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s, a_0 = a\right]}$$

$$\begin{aligned} A^{\pi_\theta,\gamma}(s,a) &= Q^{\pi_\theta,\gamma}(s,a) - V^{\pi_\theta,\gamma}(s) \\ &= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1} - V^{\pi_\theta,\gamma}(s_0)\middle| s_0 = s, a_0 = a\right] \end{aligned}$$

- O valor de uma ação pode ser quebrado em recompensa imediata e valor do próximo estado
- A recompensa somada ao valor do próximo estado é conhecida como o **retorno de 1 passo**
    - Por um passo observamos a recompensa obtida
    - Sumarizamos o retorno esperado a partir do próximo passo com a função valor

Estimando a vantagem: retornos de n passos (1/5)

$$\boxed{V^{\pi_\theta,\gamma}(s) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s\right] \qquad Q^{\pi_\theta,\gamma}(s,a) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s, a_0 = a\right]}$$

$$\begin{aligned}
A^{\pi_\theta,\gamma}(s,a) &= Q^{\pi_\theta,\gamma}(s,a) - V^{\pi_\theta,\gamma}(s) \\
&= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1} - V^{\pi_\theta,\gamma}(s_0)\middle| s_0 = s, a_0 = a\right] \\
&= \mathbb{E}_{\tau\sim\pi_\theta}\left[r_1 + \sum_{t=1}^{T-1}\gamma^t r_{t+1} - V^{\pi_\theta,\gamma}(s_0)\middle| s_0 = s, a_0 = a\right]
\end{aligned}$$

- O valor de uma ação pode ser quebrado em recompensa imediata e valor do próximo estado
- A recompensa somada ao valor do próximo estado é conhecida como o **retorno de 1 passo**
  - Por um passo observamos a recompensa obtida
  - Sumarizamos o retorno esperado a partir do próximo passo com a função valor

Estimando a vantagem: retornos de n passos (1/5)

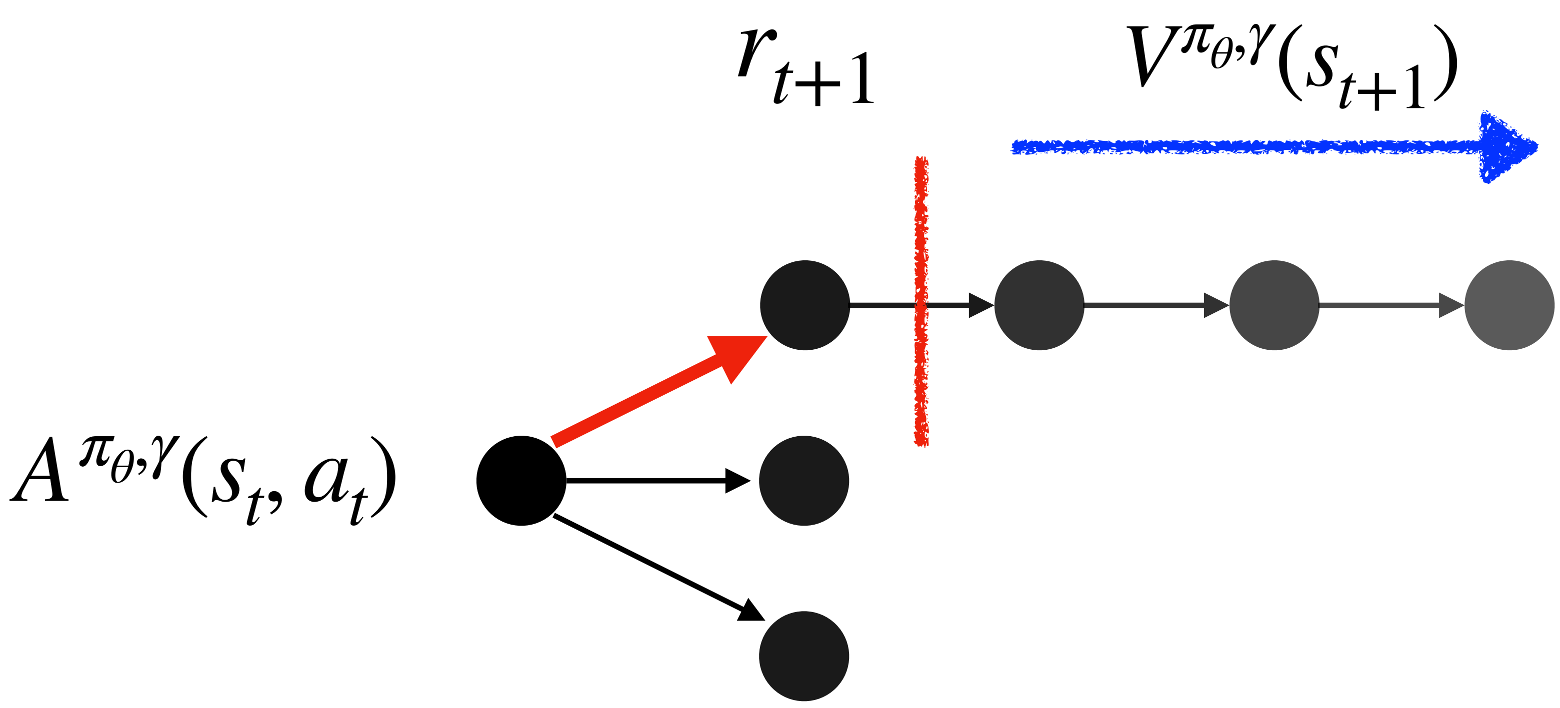
$$\boxed{V^{\pi_\theta,\gamma}(s) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s\right] \qquad Q^{\pi_\theta,\gamma}(s,a) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1}\middle| s_0 = s, a_0 = a\right]}$$

$$\begin{aligned}
A^{\pi_\theta,\gamma}(s,a) &= Q^{\pi_\theta,\gamma}(s,a) - V^{\pi_\theta,\gamma}(s) \\
&= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_{t+1} - V^{\pi_\theta,\gamma}(s_0)\middle| s_0 = s, a_0 = a\right] \\
&= \mathbb{E}_{\tau\sim\pi_\theta}\left[r_1 + \sum_{t=1}^{T-1}\gamma^t r_{t+1} - V^{\pi_\theta,\gamma}(s_0)\middle| s_0 = s, a_0 = a\right] \\
&= \mathbb{E}_{\tau_{0:1}\sim\pi_\theta}\left[r_1 + \gamma V^{\pi_\theta,\gamma}(s_1) - V^{\pi_\theta,\gamma}(s_0) | s_0 = s, a_0 = a\right]
\end{aligned}$$

- O valor de uma ação pode ser quebrado em recompensa imediata e valor do próximo estado
- A recompensa somada ao valor do próximo estado é conhecida como o **retorno de 1 passo**
  - Por um passo observamos a recompensa obtida
  - Sumarizamos o retorno esperado a partir do próximo passo com a função valor

Estimando a vantagem: retornos de n passos (2/5)

$$A^{\pi_\theta,\gamma}(s,a) = \mathbb{E}_{\tau_{0:1}\sim\pi_\theta}\left[r_1 + \gamma V^{\pi_\theta,\gamma}(s_1) - V^{\pi_\theta,\gamma}(s_0) \middle| s_0 = s, a_0 = a\right]$$



Estimando a vantagem: retornos de n passos (3/5)

$$
\begin{aligned}
A^{\pi_\theta,\gamma}(s,a) &= \mathbb{E}_{\tau_{0:1}\sim\pi_\theta}\left[r_1 + \gamma V^{\pi_\theta,\gamma}(s_1) - V^{\pi_\theta,\gamma}(s_0) \middle| s_0 = s, a_0 = a\right] \\
&= \mathbb{E}_{\tau_{0:2}\sim\pi_\theta}\left[r_1 + \gamma r_2 + \gamma^2 V^{\pi_\theta,\gamma}(s_2) - V^{\pi_\theta,\gamma}(s_0) \middle| s_0 = s, a_0 = a\right] \\
&= \mathbb{E}_{\tau_{0:3}\sim\pi_\theta}\left[r_1 + \gamma r_2 + \gamma^2 r_3 + \gamma^3 V^{\pi_\theta,\gamma}(s_3) - V^{\pi_\theta,\gamma}(s_0) \middle| s_0 = s, a_0 = a\right] \\
&\vdots \\
&= \mathbb{E}_{\tau_{0:n}\sim\pi_\theta}\left[r_1 + \cdots + \gamma^{n-1} r_n + \gamma^n V^{\pi_\theta,\gamma}(s_n) - V^{\pi_\theta,\gamma}(s_0) \middle| s_0 = s, a_0 = a\right]
\end{aligned}
$$

- Qualquer quantidade de passos antes de truncar a trajetória é válida
- O truncamento do retorno total com a função valor é conhecido como **bootstrapping**
- Note que quanto menos passos consideramos, menor o número de variáveis aleatórias
- Isso contribui para a diminuição da variância do estimador

Estimando a vantagem: retornos de n passos (4/5)

$$
\begin{aligned}
\hat{A}_t^{(1)} &= r_{t+1} + \gamma V_\phi(s_{t+1}) - V_\phi(s_t) \\
\hat{A}_t^{(2)} &= r_{t+1} + \gamma r_{t+2} + \gamma^2 V_\phi(s_{t+2}) - V_\phi(s_t) \\
\hat{A}_t^{(3)} &= r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 V_\phi(s_{t+3}) - V_\phi(s_t) \\
&\vdots \\
\hat{A}_t^{(n)} &= r_{t+1} + \cdots + \gamma^{n-t-1} r_n + \gamma^{n-t} V_\phi(s_n) - V_\phi(s_t)
\end{aligned}
$$

- Para amostrar esses retornos precisamos de uma estimativa da função valor
- Podemos usar o mesmo aproximador aprendido para o *baseline*
- Isso nos permite estimar a vantagem para um passo de tempo sem ter a trajetória completa!

Estimando a vantagem: retornos de n passos (4/5)

$$
\begin{aligned}
\hat{A}_t^{(1)} &= r_{t+1} + \gamma V_\phi(s_{t+1}) - V_\phi(s_t) \\
\hat{A}_t^{(2)} &= r_{t+1} + \gamma r_{t+2} + \gamma^2 V_\phi(s_{t+2}) - V_\phi(s_t) \\
\hat{A}_t^{(3)} &= r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 V_\phi(s_{t+3}) - V_\phi(s_t) \\
&\vdots \\
\hat{A}_t^{(n)} &= r_{t+1} + \cdots + \gamma^{n-t-1} r_n + \gamma^{n-t} V_\phi(s_n) - V_\phi(s_t)
\end{aligned}
$$

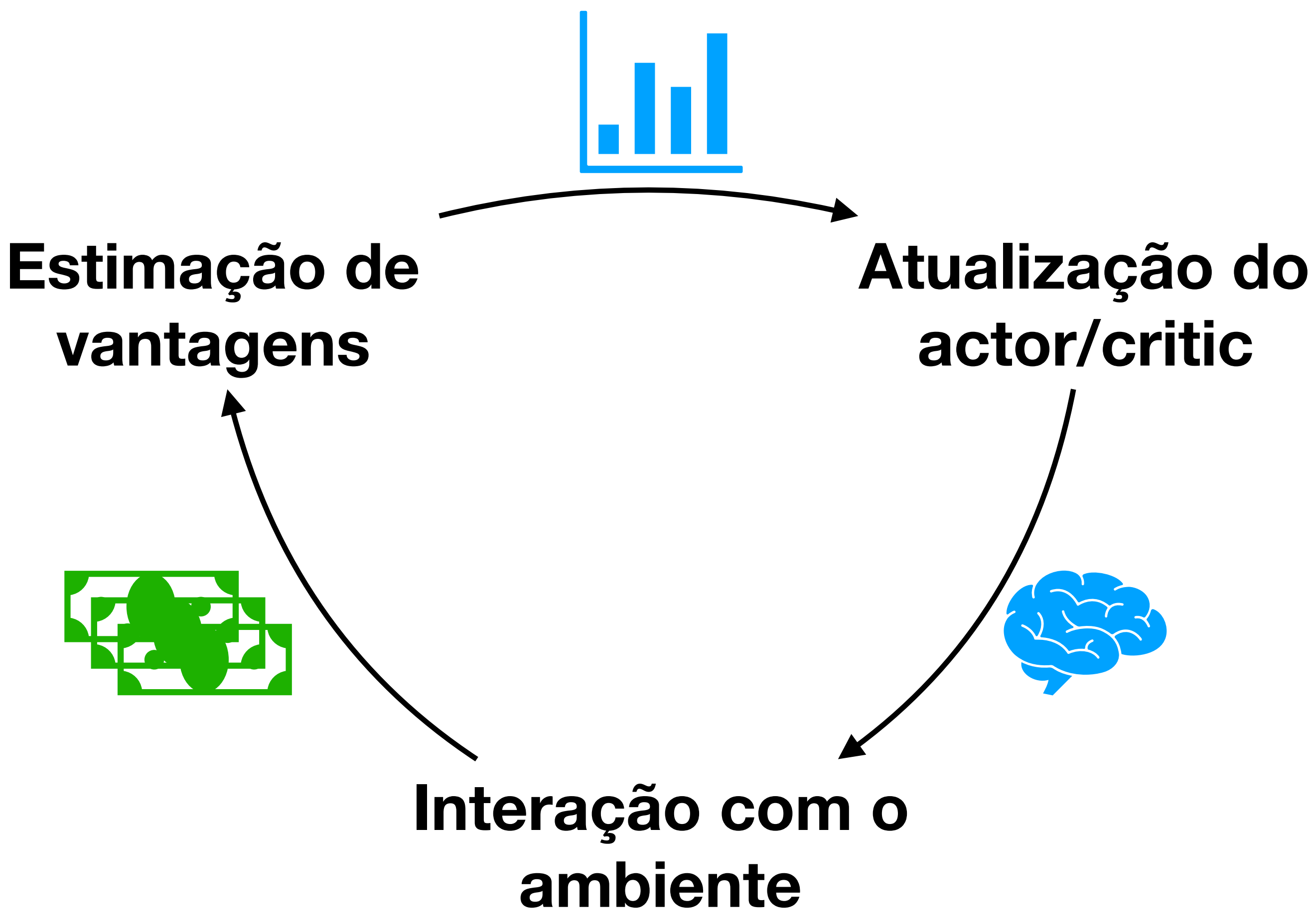
Vantagens

- Permitem trabalhar com trajetórias parciais
- Menor variância das estimativas

Desvantagens

- Necessitam o aprendizado de uma função-valor
- Maior dependência na qualidade do aproximador

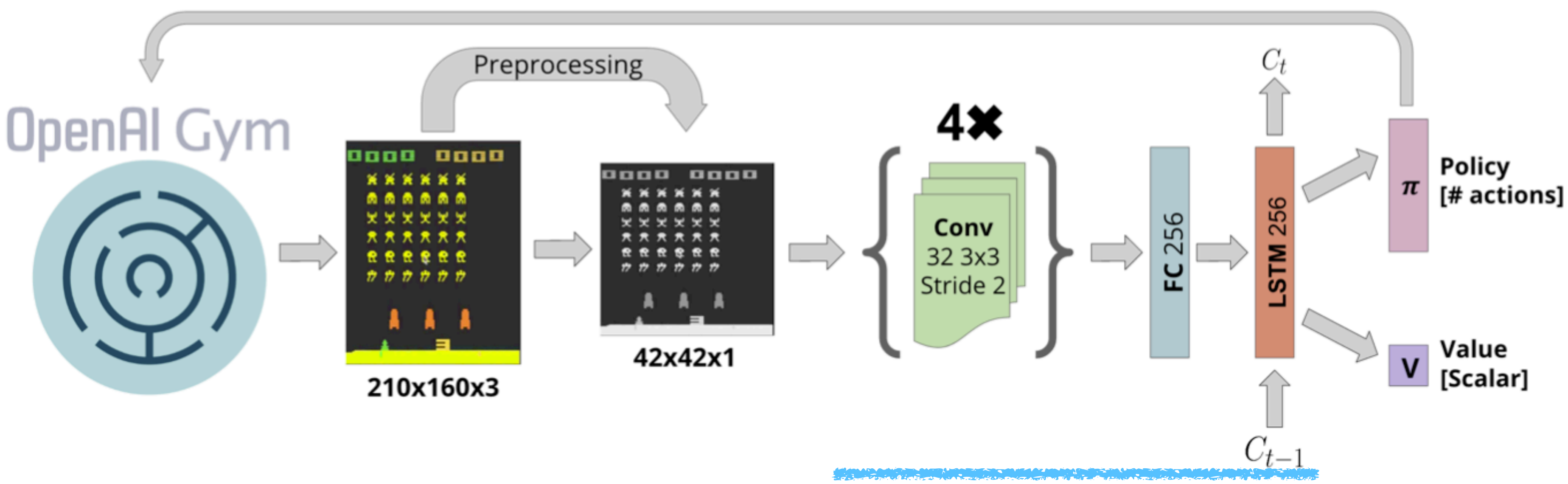
Advantage Actor-Critic (A2C)



- O algoritmo conhecido como A2C incorpora os conceitos vistos até agora
- **Actor-Critic** vem da interpretação intuitiva dos dois principais componentes do agente
    - A política $\pi_\theta(\cdot \mid s)$ recomenda ações para cada estado, portanto é vista como "*actor*"
    - A função $V_\phi(s)$ avalia o retorno esperado sob a política, portanto é vista como "*critic*"

Advantage Actor-Critic (A2C)

A arquitetura do modelo compartilha parâmetros entre actor e critic



Extrator de features

O modelo é atualizado *end-to-end*: definimos um objetivo conjunto entre *actor* e *critic*

"Joint Loss"

$$[\theta, \phi] \leftarrow [\theta, \phi] + \alpha \nabla_{\theta,\phi} \left(L_{\text{actor}}(\theta) + L_{\text{critic}}(\phi) \right)$$

Advantage Actor-Critic (A2C)

O objetivo do *actor* é a função de custo do *Policy Gradient*, com as vantagens calculadas por bootstrapping

$$\boxed{L_{\text{actor}}(\theta) = -\frac{1}{K}\sum_{t=1}^{K} \log \pi_\theta(a_t|s_t)\hat{A}_t^{(n)}}$$

$$\hat{A}_t^{(n)} = r_{t+1} + \cdots + \gamma^{n-t-1} r_n + \gamma^{n-t} V_\phi(s_{t+n}) - V_\phi(s_t)$$

O objetivo do *critic* é mesmo de antes: prever os retornos sob a política (com bootstrapping)

$$\boxed{L_{\text{critic}}(\phi) = \frac{1}{K}\sum_{t=1}^{K} (V_\phi(s_t) - \hat{R}_t)^2}$$

$$\hat{R}_t = r_{t+1} + \cdots + \gamma^{n-t-1} r_n + \gamma^{n-t} V_\phi(s_{t+n})$$

$$\boxed{\nabla L_{\textbf{actor}}(\theta) \approx -\nabla J(\theta)}$$

Advantage Actor-Critic (A2C)

Algoritmo 1 A2C

Entrada: parâmetros da política, θ, parâmetros da função-valor, ϕ

enquanto não satisfeito **faça**

  Colete N passos no ambiente $\{(s_i, a_i, r_i, s_{i+1})\}_{i=1}^{N}$

  Calcule os retornos truncados $\hat{R}_t = r_{t+1} + \cdots + \gamma^{n-t-1} r_n + \gamma^{n-t} V_\phi(s_{t+n})$

  Calcule as vantagens com *bootstrapping* $\hat{A}_t = \hat{R}_t - V_\phi(s_t)$

  Calcule o objetivo do *actor*

$$L_{\text{actor}}(\theta) = -\frac{1}{N} \sum_{t=1}^{N} \log \pi_\theta(a_t|s_t) \hat{A}_t$$

  Calcule o objetivo do *critic*

$$L_{\text{critic}}(\phi) = \frac{1}{N} \sum_{t=1}^{N} (V_\phi(s_t) - \hat{R}_t)^2$$

  Atualize os parâmetros do modelo

$$[\theta, \phi] \leftarrow [\theta, \phi] + \alpha \nabla_{\theta,\phi} \left(L_{\text{actor}}(\theta) + L_{\text{critic}}(\phi) \right)$$

fim enquanto

devolve π_θ

Generalized Advantage Estimation (GAE) (1/4)

Como escolher o número de passos antes de realizer o bootstrapping?

- Quanto mais cedo truncarmos a trajetória, mais dependente da função valor seremos
- Quanto mais tarde, mais estaremos sujeitos à variância das recompensas

GAE sugere usar uma média exponencial de todos eles

$$\hat{A}_t^{\mathrm{GAE}(\gamma,\lambda)} = (1-\lambda)(\hat{A}_t^{(1)} + \lambda \hat{A}_t^{(2)} + \lambda^2 \hat{A}_t^{(3)} + \dots)$$

Onde $\lambda \in (0,1)$ controla o grau de *bootstrapping* desejado:

- Com $\lambda \to 0$, o estimador se aproxima do estimador de um passo
- Com $\lambda \to 1$, o estimador se aproxima do *reward-to-go* menos *baseline*

Generalized Advantage Estimation (GAE) (2/4)

Como calcular o estimador GAE de maneira eficiente?

Note que a vantagem de n passos pode ser quebrada em n vantagens de 1 passo

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

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

Exercício: mostrar essa equivalência para n passos

$$\hat{A}_t^{(n)} = \delta_t + \gamma\delta_{t+1} + \cdots + \gamma^{n-t-1}\delta_{t+n}$$

Generalized Advantage Estimation (GAE) (3/4)

$$\hat{A}_t^{\mathrm{GAE}(\gamma,\lambda)} = (1-\lambda)(\hat{A}_t^{(1)} + \lambda \hat{A}_t^{(2)} + \lambda^2 \hat{A}_t^{(3)} + \dots)$$

$$\begin{aligned}
\hat{A}_t^{(1)} &= \delta_t \\
\hat{A}_t^{(2)} &= \delta_t + \gamma \delta_{t+1} \\
\hat{A}_t^{(3)} &= \delta_t + \gamma \delta_{t+1} + \gamma^2 \delta_{t+2} \\
\hat{A}_t^{(4)} &= \delta_t + \gamma \delta_{t+1} + \gamma^2 \delta_{t+2} + \gamma^3 \delta_{t+3}
\end{aligned}$$

Generalized Advantage Estimation (GAE) (3/4)

$$\hat{A}_t^{\text{GAE}(\gamma,\lambda)} = (1-\lambda)(\hat{A}_t^{(1)} + \lambda\hat{A}_t^{(2)} + \lambda^2\hat{A}_t^{(3)} + \dots)$$

$$\begin{aligned}
&(1-\lambda) && \delta_t \\
+&(1-\lambda)\lambda && (\delta_t + \gamma\delta_{t+1}) \\
+&(1-\lambda)\lambda^2 && (\delta_t + \gamma\delta_{t+1} + \gamma^2\delta_{t+2}) \\
+&(1-\lambda)\lambda^3 && (\delta_t + \gamma\delta_{t+1} + \gamma^2\delta_{t+2} + \gamma^3\delta_{t+3}) \\
&\vdots
\end{aligned}$$

Generalized Advantage Estimation (GAE) (3/4)

$$\hat{A}_t^{\mathrm{GAE}(\gamma,\lambda)} = (1-\lambda)(\hat{A}_t^{(1)} + \lambda\hat{A}_t^{(2)} + \lambda^2\hat{A}_t^{(3)} + \dots)$$

$$\begin{aligned}
&(1-\lambda)\ \delta_t \\
&+(1-\lambda)(\lambda\delta_t\ +\lambda\gamma\delta_{t+1}) \\
&+(1-\lambda)(\lambda^2\delta_t+\lambda^2\gamma\delta_{t+1}+\lambda^2\gamma^2\delta_{t+2}) \\
&+(1-\lambda)(\lambda^3\delta_t+\lambda^3\gamma\delta_{t+1}+\lambda^3\gamma^2\delta_{t+2}\ +\lambda^3\gamma^3\delta_{t+3} \\
&\quad\vdots
\end{aligned}$$

Generalized Advantage Estimation (GAE) (3/4)

$$\sum_{k=0}^{\infty} \lambda^k = \frac{1}{1-\lambda}, \quad \lambda \in (0,1)$$

$$\begin{array}{ll}
(1-\lambda) & \delta_t \\
+(1-\lambda) & (\lambda\delta_t + \lambda\gamma\delta_{t+1}) \\
+(1-\lambda) & (\lambda^2\delta_t + \lambda^2\gamma\delta_{t+1} + \lambda^2\gamma^2\delta_{t+2}) \\
+(1-\lambda) & (\lambda^3\delta_t + \lambda^3\gamma\delta_{t+1} + \lambda^3\gamma^2\delta_{t+2} + \lambda^3\gamma^3\delta_{t+3} \\
\vdots & \quad\vdots \\
 & \dfrac{1}{1-\lambda}\delta_t
\end{array}$$

Generalized Advantage Estimation (GAE) (3/4)

$$\sum_{k=0}^{\infty} \lambda^k = \frac{1}{1-\lambda}, \quad \lambda \in (0,1)$$

$$
\begin{aligned}
&(1-\lambda)\ \delta_t \\
&+(1-\lambda)(\lambda\delta_t + \lambda\gamma\delta_{t+1}) \\
&+(1-\lambda)(\lambda^2\delta_t + \lambda^2\gamma\delta_{t+1} + \lambda^2\gamma^2\delta_{t+2}) \\
&+(1-\lambda)(\lambda^3\delta_t + \lambda^3\gamma\delta_{t+1} + \lambda^3\gamma^2\delta_{t+2} + \lambda^3\gamma^3\delta_{t+3} \\
&\vdots \qquad\qquad\qquad \vdots
\end{aligned}
$$

$$\frac{\lambda}{1-\lambda}\gamma\delta_{t+1}$$

Generalized Advantage Estimation (GAE) (3/4)

$$\sum_{k=0}^{\infty} \lambda^k = \frac{1}{1-\lambda}, \quad \lambda \in (0,1)$$

$$\begin{aligned}
&(1-\lambda)\ \delta_t \\
&+(1-\lambda)(\lambda\delta_t\ +\lambda\gamma\delta_{t+1}) \\
&+(1-\lambda)(\lambda^2\delta_t+\lambda^2\gamma\delta_{t+1}+\boxed{\lambda^2\gamma^2\delta_{t+2}}) \\
&+(1-\lambda)(\lambda^3\delta_t+\lambda^3\gamma\delta_{t+1}+\boxed{\lambda^3\gamma^2\delta_{t+2}}\ +\lambda^3\gamma^3\delta_{t+3} \\
&\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots
\end{aligned}$$

$$\frac{\lambda^2}{1-\lambda}\gamma^2\delta_{t+2}$$

Generalized Advantage Estimation (GAE) (3/4)

$$
\begin{aligned}
&(1-\lambda)\ \delta_t \\
&+(1-\lambda)(\lambda\delta_t\ +\lambda\gamma\delta_{t+1}) \\
&+(1-\lambda)(\lambda^2\delta_t+\lambda^2\gamma\delta_{t+1}+\lambda^2\gamma^2\delta_{t+2}) \\
&+(1-\lambda)(\lambda^3\delta_t+\lambda^3\gamma\delta_{t+1}+\lambda^3\gamma^2\delta_{t+2}\ +\lambda^3\gamma^3\delta_{t+3} \\
&\vdots
\end{aligned}
$$

$$(1-\lambda)\left(\frac{1}{1-\lambda}\delta_t+\frac{\lambda}{1-\lambda}\gamma\delta_{t+1}+\frac{\lambda^2}{1-\lambda}\gamma^2\delta_{t+2}+\ldots\right)$$

Generalized Advantage Estimation (GAE) (3/4)

$$(1-\lambda)\left(\frac{1}{1-\lambda}\delta_t + \frac{\lambda}{1-\lambda}\gamma\delta_{t+1} + \frac{\lambda^2}{1-\lambda}\gamma^2\delta_{t+2} + \dots\right)$$

$$\delta_t + \gamma\lambda\delta_{t+1} + \gamma^2\lambda^2\delta_{t+2} + \dots$$

$$\boxed{\hat{A}_t^{\mathrm{GAE}(\gamma,\lambda)} = \sum_{t=0}^{\infty}(\gamma\lambda)^t\delta_t}$$

Generalized Advantage Estimation (GAE) (4/4)

Casos especias do estimador GAE(γ, λ):

$$\hat{A}_t^{\mathrm{GAE}(\gamma,\lambda)} = \sum_{t=0}^{\infty} (\gamma\lambda)^t \delta_t$$

$$\mathrm{GAE}(\gamma, 0): \quad \hat{A}_t := \delta_t \qquad = r_t + \gamma V(s_{t+1}) - V(s_t)$$

$$\mathrm{GAE}(\gamma, 1): \quad \hat{A}_t := \sum_{l=0}^{\infty} \gamma^l \delta_{t+l} = \sum_{l=0}^{\infty} \gamma^l r_{t+l} - V(s_t)$$

GAE nos permite interpolar entre retornos de N passos e Monte Carlo

*$GAE(\gamma, \lambda)$ *é análogo ao estimador de temporal difference,* $TD(\lambda)$*, em programação dinâmica.*

Referências

(1) **Generalized Advantage Estimation**

- http://arxiv.org/abs/1506.02438

(2) **Intuitive RL (Reinforcement Learning): Introduction to Advantage-Actor-Critic (A2C)**

- https://sudonull.com/post/32170-Intuitive-RL-Reinforcement-Learning-Introduction-to-Advantage-Actor-Critic-A2C

(3) **CS 285 - Deep Reinforcement Learning (UC Berkeley)**

- http://rail.eecs.berkeley.edu/deeprlcourse/static/slides/lec-6.pdf