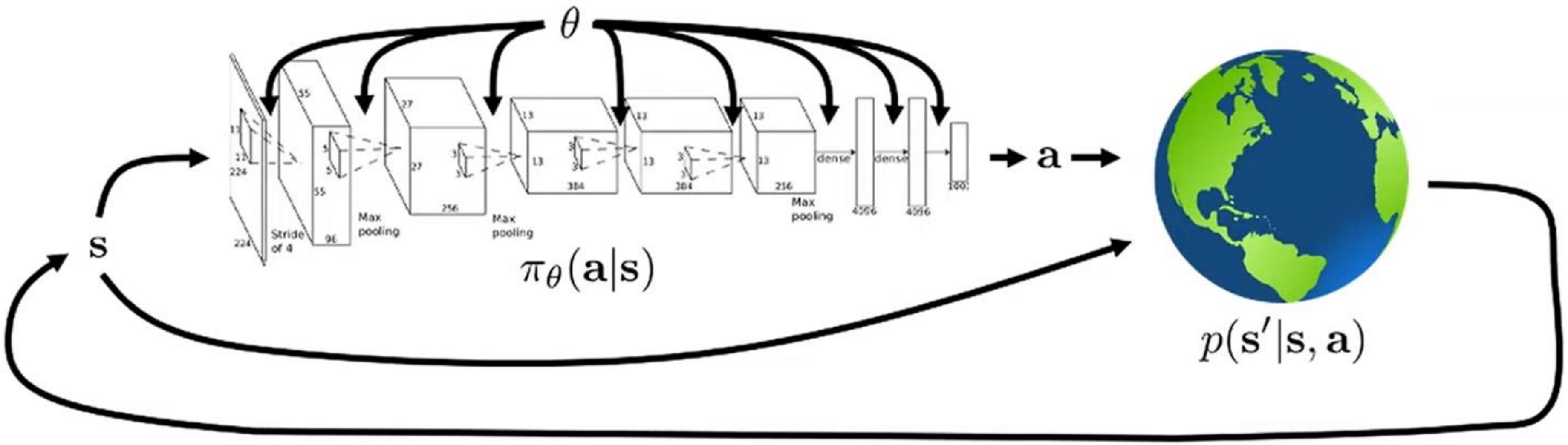
Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería de Transporte y Logística

Sistemas Urbanos Inteligentes

Optimización directa de políticas

Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación

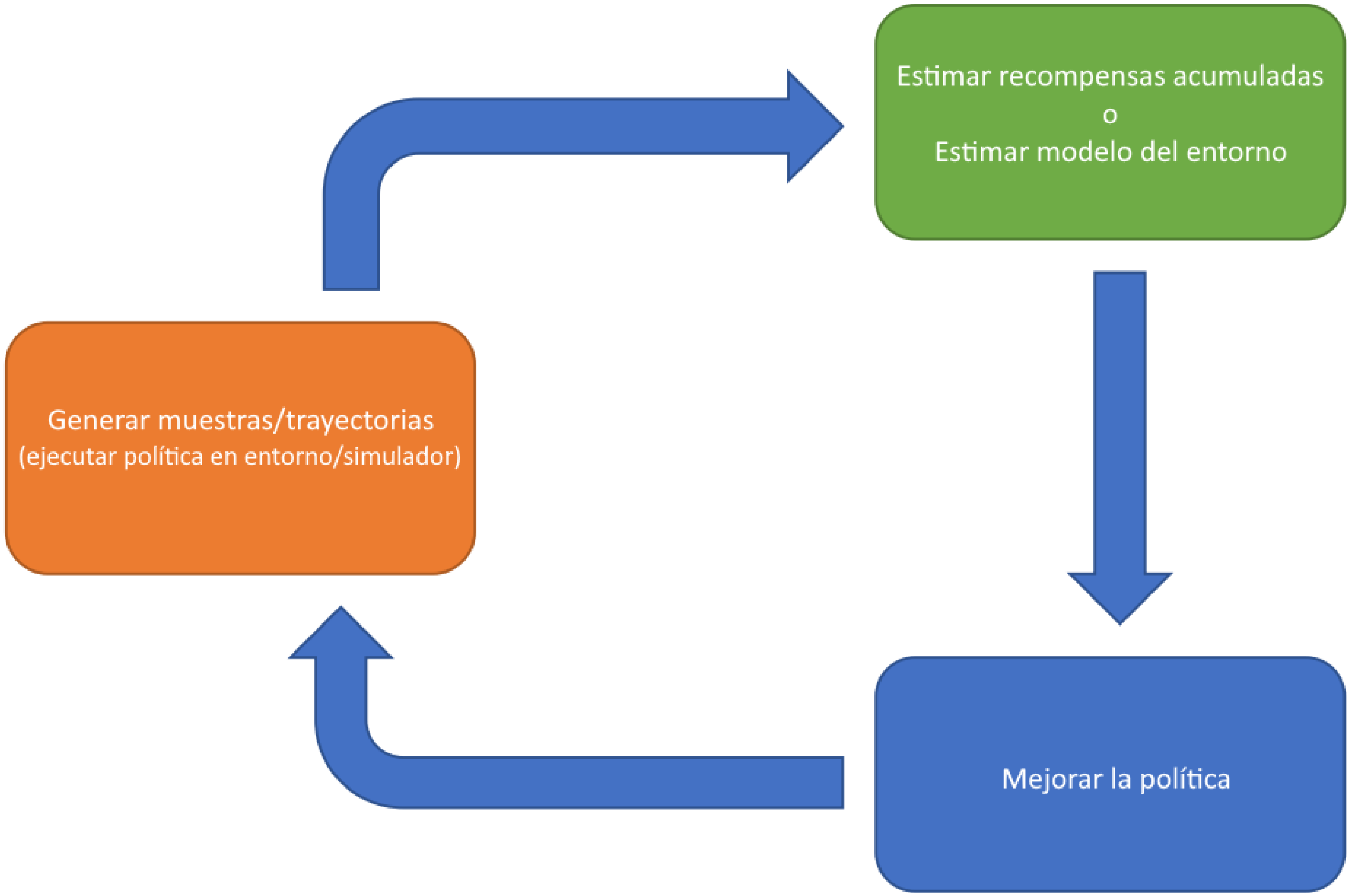
En (D)RL, buscamos la política que maximiza la recompensa esperada

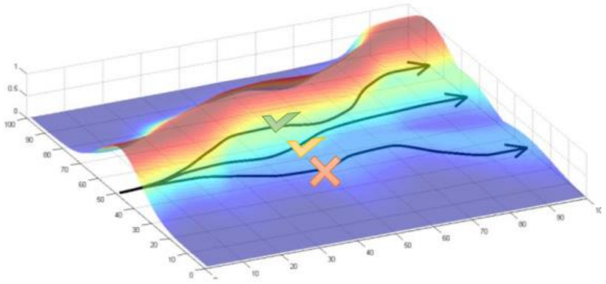


$$\underbrace{p_\theta(\mathbf{s}_1, \mathbf{a}_1, \ldots, \mathbf{s}_T, \mathbf{a}_T)}_{p_\theta(\tau)} = p(\mathbf{s}_1) \prod_{t=1}^{T} \pi_\theta(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$$

$$\theta^\star = \arg\max_\theta E_{\tau \sim p_\theta(\tau)} \left[\sum_t r(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1}) \right]$$

Por ejemplo, si queremos optimizar directamente la política…

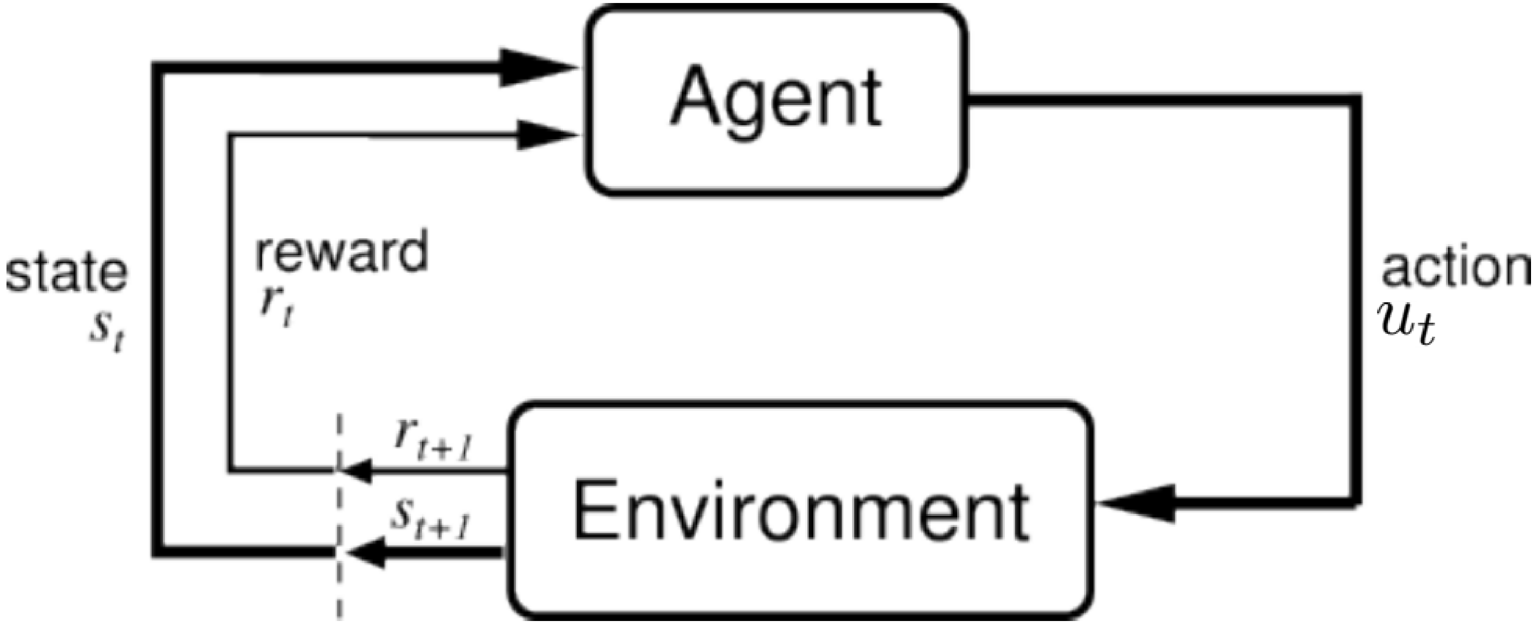




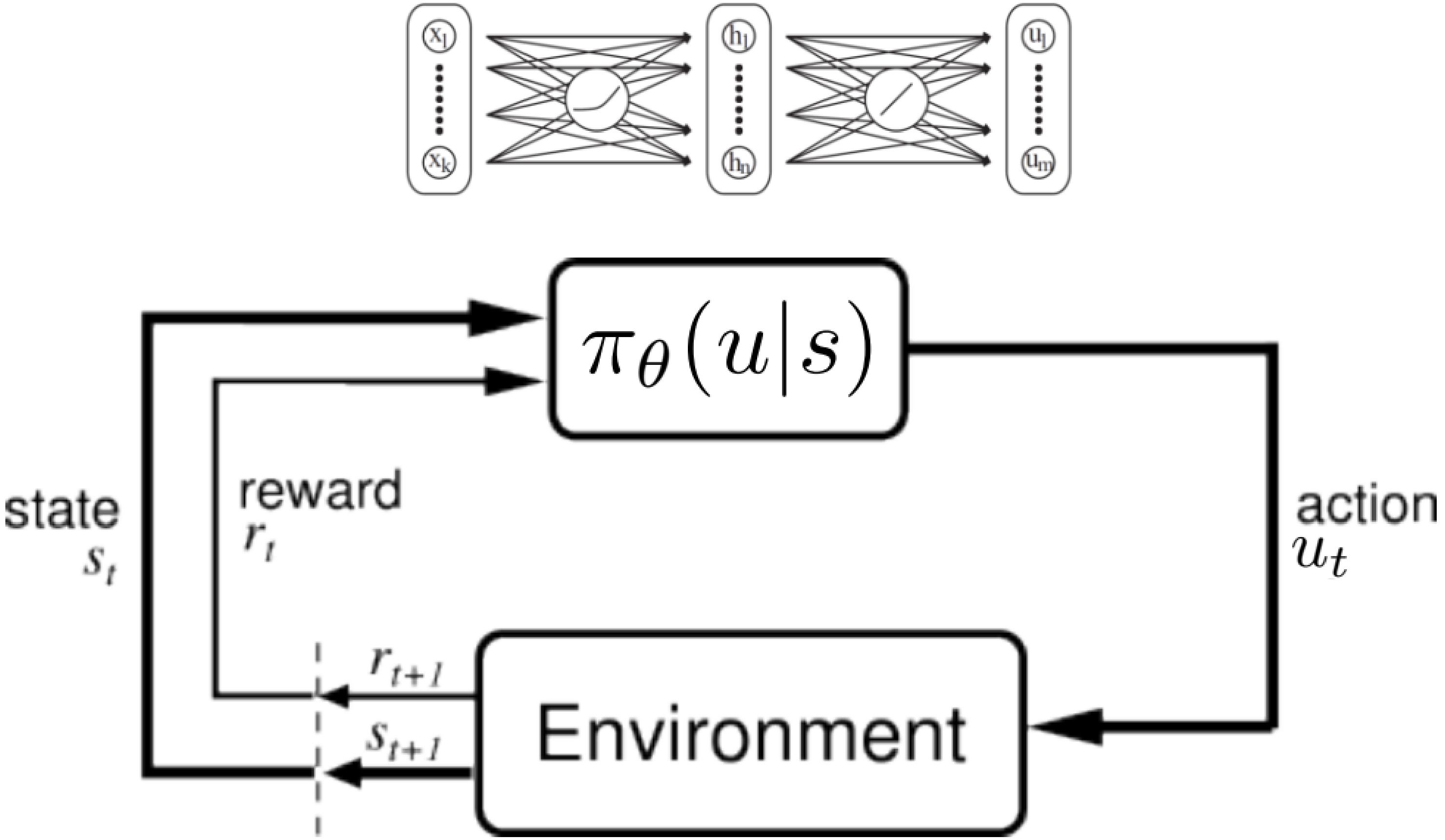
$$J(\theta) = E_\pi\left[\sum_t r_t\right] \approx \frac{1}{N}\sum_{i=1}^{N}\sum_t r_t^i$$

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$$

Introducimos un muy leve cambio en la notación



Introducimos un muy leve cambio en la notación



Veamos la formulación

Consideramos la optimización de una política de control parametrizada por un vector θ:

$$\max_{\theta} \quad \mathrm{E}[\sum_{t=0}^{H} R(s_t)|\pi_\theta]$$

Con el fin de "suavizar" el problema (para usar gradientes), utilizamos una política estocástica $\pi_\theta(u \mid s)$, que entrega la probabilidad de cada acción condicionada en el estado actual.

Veamos la formulación

Como primer paso de cualquier problema de optimización, definimos la función objetivo:

$$U(\theta) = \mathrm{E}[\sum_{t=0}^{H} R(s_t, u_t); \pi_\theta] = \sum_{\tau} P(\tau; \theta) R(\tau)$$

Buscamos los parámetros θ que cumplan lo siguiente:

$$\max_{\theta} U(\theta) = \max_{\theta} \sum_{\tau} P(\tau; \theta) R(\tau)$$

Veamos la formulación

Tomando el gradiente de la función objetivo, obtenemos lo siguiente:

$$\nabla_\theta U(\theta) = \nabla_\theta \sum_\tau P(\tau;\theta)R(\tau)$$
$$= \sum_\tau \nabla_\theta P(\tau;\theta)R(\tau)$$

$$p_\theta(\tau)\nabla_\theta \log p_\theta(\tau) = p_\theta(\tau)\frac{\nabla_\theta p_\theta(\tau)}{p_\theta(\tau)} = \nabla_\theta p_\theta(\tau)$$

Veamos la formulación

Tomando el gradiente de la función objetivo, obtenemos lo siguiente:

$$\begin{aligned}\nabla_\theta U(\theta) &= \nabla_\theta \sum_\tau P(\tau;\theta)R(\tau) \\ &= \sum_\tau \nabla_\theta P(\tau;\theta)R(\tau) \\ &= \sum_\tau \frac{P(\tau;\theta)}{P(\tau;\theta)}\nabla_\theta P(\tau;\theta)R(\tau) \\ &= \sum_\tau P(\tau;\theta)\frac{\nabla_\theta P(\tau;\theta)}{P(\tau;\theta)}R(\tau)\end{aligned}$$

$$p_\theta(\tau)\nabla_\theta \log p_\theta(\tau) = p_\theta(\tau)\frac{\nabla_\theta p_\theta(\tau)}{p_\theta(\tau)} = \nabla_\theta p_\theta(\tau)$$

Veamos la formulación

Tomando el gradiente de la función objetivo, obtenemos lo siguiente:

$$
\begin{aligned}
\nabla_\theta U(\theta) &= \nabla_\theta \sum_\tau P(\tau;\theta)R(\tau) \\
&= \sum_\tau \nabla_\theta P(\tau;\theta)R(\tau) \\
&= \sum_\tau \frac{P(\tau;\theta)}{P(\tau;\theta)} \nabla_\theta P(\tau;\theta)R(\tau) \\
&= \sum_\tau P(\tau;\theta)\frac{\nabla_\theta P(\tau;\theta)}{P(\tau;\theta)} R(\tau) \\
&= \sum_\tau P(\tau;\theta)\nabla_\theta \log P(\tau;\theta)R(\tau)
\end{aligned}
$$

$$
p_\theta(\tau)\nabla_\theta \log p_\theta(\tau) = p_\theta(\tau)\frac{\nabla_\theta p_\theta(\tau)}{p_\theta(\tau)} = \nabla_\theta p_\theta(\tau)
$$

Veamos la formulación

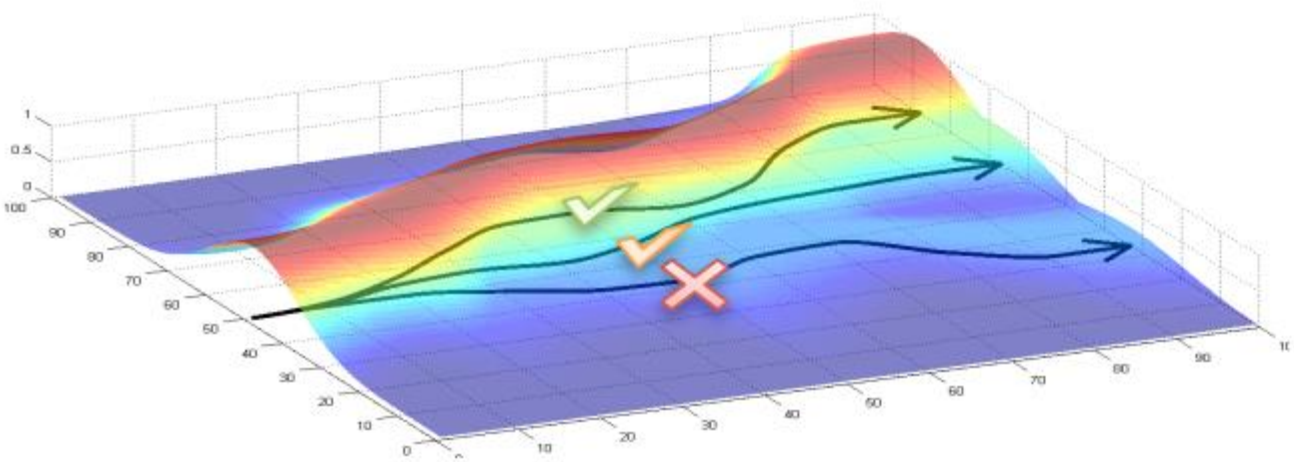
$$\nabla_\theta U(\theta) = \sum_\tau P(\tau;\theta) \nabla_\theta \log P(\tau;\theta) R(\tau)$$

Como en el caso de Q-Learning, podemos siempre aproximar el valor esperando a través de muestreo:

$$\nabla_\theta U(\theta) \approx \hat{g} = \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)};\theta) R(\tau^{(i)})$$

¿Qué hace un *policy gradient*?

- Esencialmente, todo depende de la recompensa: si esta es positiva, probabilidad de trayectoria sube, si no, baja
- Visto de otra forma, esta expresión formaliza un esquema de prueba y error



$$\nabla_\theta U(\theta) \approx \hat{g} = \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)}; \theta) R(\tau^{(i)})$$

Continuemos con la derivación

$$\nabla_\theta \log P(\tau^{(i)};\theta) = \nabla_\theta \log \left[\prod_{t=0}^{H} \underbrace{P(s_{t+1}^{(i)} | s_t^{(i)}, u_t^{(i)})}_{\text{dynamics model}} \cdot \underbrace{\pi_\theta(u_t^{(i)} | s_t^{(i)})}_{\text{policy}} \right]$$

Continuemos con la derivación

$$\nabla_\theta \log P(\tau^{(i)};\theta) = \nabla_\theta \log \left[\prod_{t=0}^{H} \underbrace{P(s_{t+1}^{(i)}|s_t^{(i)}, u_t^{(i)})}_{\text{dynamics model}} \cdot \underbrace{\pi_\theta(u_t^{(i)}|s_t^{(i)})}_{\text{policy}} \right]$$

$$= \nabla_\theta \left[\sum_{t=0}^{H} \log P(s_{t+1}^{(i)}|s_t^{(i)}, u_t^{(i)}) + \sum_{t=0}^{H} \log \pi_\theta(u_t^{(i)}|s_t^{(i)}) \right]$$

Continuemos con la derivación

$$\begin{aligned}
\nabla_\theta \log P(\tau^{(i)};\theta) &= \nabla_\theta \log \left[\prod_{t=0}^{H} \underbrace{P(s_{t+1}^{(i)}|s_t^{(i)}, u_t^{(i)})}_{\text{dynamics model}} \cdot \underbrace{\pi_\theta(u_t^{(i)}|s_t^{(i)})}_{\text{policy}} \right] \\
&= \nabla_\theta \left[\sum_{t=0}^{H} \log P(s_{t+1}^{(i)}|s_t^{(i)}, u_t^{(i)}) + \sum_{t=0}^{H} \log \pi_\theta(u_t^{(i)}|s_t^{(i)}) \right] \\
&= \nabla_\theta \sum_{t=0}^{H} \log \pi_\theta(u_t^{(i)}|s_t^{(i)}) \\
&= \sum_{t=0}^{H} \underbrace{\nabla_\theta \log \pi_\theta(u_t^{(i)}|s_t^{(i)})}_{\text{no dynamics model required!!}}
\end{aligned}$$

Un par de detalles para cerrar la formulación

1. Necesitamos recompensas negativas y positivas para reducir probabilidad de malas trayectorias (recordar que estamos muestreando):

$$\hat{g} = \frac{1}{m}\sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)};\theta)\underbrace{(R(\tau^{(i)}) - \overset{\text{baseline}}{b})}_{\text{advantage}}$$

Un par de detalles para cerrar la formulación

1. Podemos reducir la varianza si consideramos la causalidad acción/recompensa (acciones presentes no influyen en recompensas pasadas):

$$\hat{g} = \frac{1}{m}\sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)};\theta)(R(\tau^{(i)}) - b)$$

$$= \frac{1}{m}\sum_{i=1}^{m} \left(\sum_{t=0}^{H-1} \nabla_\theta \log \pi_\theta(u_t^{(i)}|s_t^{(i)})\right)\left(\sum_{t=0}^{H-1} R(s_t^{(i)}, u_t^{(i)}) - b\right)$$

$$= \frac{1}{m}\sum_{i=1}^{m} \left(\sum_{t=0}^{H-1} \nabla_\theta \log \pi_\theta(u_t^{(i)}|s_t^{(i)}) \left[\overset{0}{\cancel{(\sum_{k=0}^{t-1} R(s_k^{(i)}, u_k^{(i)}))}} + (\sum_{k=t}^{H-1} R(s_k^{(i)}, u_k^{(i)})) - b\right]\right)$$

Todos estos ingredientes nos permiten construir un algoritmo para obtener una política

Algorithm 1 "Vanilla" policy gradient algorithm

Initialize policy parameter θ, baseline b

for iteration$=1, 2, \ldots$ **do**

Collect a set of trajectories by executing the current policy

At each timestep in each trajectory, compute

the *return* $R_t = \sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'}$, and

the *advantage estimate* $\hat{A}_t = R_t - b(s_t)$.

Re-fit the baseline, by minimizing $\|b(s_t) - R_t\|^2$,

summed over all trajectories and timesteps.

Update the policy, using a policy gradient estimate $\hat{g}$,

which is a sum of terms $\nabla_\theta \log \pi(a_t \mid s_t, \theta)\hat{A}_t$

end for

La pregunta es ahora, ¿para qué todo esto?

Si bien conceptualmente todo esto es muy bonito, no es obvio que ganamos sobre los métodos basados en funciones de valor:

1. No dependemos de la existencia de un modelo conocido de transiciones entre estados del mundo
2. Obtenemos la política de manera directa (esto es importante, debido a la naturaleza estimada dada por el muestreo)
3. Podemos manejar acciones continuas (Q-Learning tiene problemas con esto)
4. Una red como política es más versátil (arquitecturas y objetivos auxiliares)
5. Menos eficiente en cuanto a muestras, pero eso tiene arreglo porque podemos simular…

https://youtu.be/9498aIQ1Bd4

https://www.youtube.com/watch?v=vHfc08KoUP4

https://www.youtube.com/watch?v=VMp6pq6_QjI

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería de Transporte y Logística

Sistemas Urbanos Inteligentes

Optimización directa de políticas

Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación