Aprendizaje Reforzado
Clase 4

Julián Martínez

Dos problemas:

- Evaluar una política
- Mejorar una Política

Detalle computacional

$$\mu_k = \frac{\sum_{j=1}^{k} x_j}{k} = \mu_{k-1}\frac{(k-1)}{k} + \frac{1}{k}x_k$$

$$\boxed{= \mu_{k-1} + \frac{1}{k}\left(x_k - \mu_{k-1}\right)}$$

Media Incremental

A simple bandit algorithm

Initialize, for $a = 1$ to k:

$Q(a) \leftarrow 0$

$N(a) \leftarrow 0$

Loop forever:

$$A \leftarrow \begin{cases} \arg\max_a Q(a) & \text{with probability } 1 - \varepsilon \quad \text{(breaking ties randomly)} \\ \text{a random action} & \text{with probability } \varepsilon \end{cases}$$

$R \leftarrow bandit(A)$

$N(A) \leftarrow N(A) + 1$

$Q(A) \leftarrow Q(A) + \frac{1}{N(A)}\big[R - Q(A)\big]$

Evaluación de una política

$$v_{\pi}^{k+1}(s) = R(s) + \gamma \sum_{s'} v_{\pi}^{k}(s') \, P_{ss'}^{\pi}$$

Ecuaciones de Bellman

$$\boxed{v_{*}(s) = \max_{a} \left[\sum_{s'} \left\{ r(s,a,s') + \gamma \, v_{*}(s') \right\} P_{s,s'}^{a} \right]}$$

Mejora de una política

$$\pi_{k+1}(s) := \arg\max_{a} q_{\pi_k}(s,a)$$

Evaluación y mejora

$$\pi_k \longrightarrow v_{\pi_k} \longrightarrow q_{\pi_k} \longrightarrow \pi_{k+1}$$

Predicción libre de modelo

Episode: $s_0 \quad a_0 \quad s_1 \quad a_1 \quad s_2 \quad a_2 \quad \dots \quad s_j \quad a_j \dots s_T$

¿Podemos hacer evaluación, aprendiendo tan sólo de la experiencia?

Monte Carlo

$$v_\pi(s) = \mathbb{E}_\pi\left[G_t \mid S_t = s\right]$$

$$s_0^1 \quad a_0^1 \quad s_1^1 \quad a_1^1 \;\dots\; \overset{\Sigma}{\overrightarrow{\underset{\substack{\shortparallel\\ s}}{s_k^1}\; a_k^1 \;\dots\; \underset{\substack{\shortparallel\\ s}}{s_j^1}\; a_j^1 \;\dots\; s_T^1}}$$

$$s_0^2 \quad a_0^2 \quad s_1^2 \quad a_1^2 \;\dots\;\dots\; \overset{\Sigma}{\overrightarrow{\underset{\substack{\shortparallel\\ s}}{s_{k'}^2}\; a_{k'}^2 \;\dots\; \qquad s_T^2}}$$

$$s_0^n \quad a_0^n \quad s_1^n \quad a_1^n \;\dots\; \overset{\Sigma}{\overrightarrow{\underset{\substack{\shortparallel\\ s}}{s_{k''}^n} \;\dots\; \underset{\substack{\shortparallel\\ s}}{s_{j''}^n} \;\dots\; s_T^n}}$$

Ley de Los grandes Números (aproximando al esperanza)

$$\frac{\sum_{i=1}^{n} X_i}{n} \xrightarrow[n \to \infty]{} \mathbb{E}[X] \qquad X_i \quad \text{iid}$$

First/Every visit MC

$$G^{\ell}_{\tau^{\ell}_s} = \sum_{t=\tau^{\ell}_s+1}^{T} R^{\ell}_t \cdot \gamma^{t-\tau^{\ell}_s-1}$$

τ^{ℓ}_s = tiempo de la PRIMERA visita a s

$$v_{\pi}(s) \simeq \sum_{\ell=1}^{n} G^{\ell}_{\tau^{\ell}_s} \Big/ N(s)$$

$N(s)$ = # episodios donde s fue visitado

Sutton

First-visit MC prediction, for estimating $V \approx v_\pi$

Input: a policy π to be evaluated

Initialize:

  $V(s) \in \mathbb{R}$, arbitrarily, for all $s \in \mathcal{S}$

  $Returns(s) \leftarrow$ an empty list, for all $s \in \mathcal{S}$

Loop forever (for each episode):

  Generate an episode following π: $S_0, A_0, R_1, S_1, A_1, R_2, \ldots, S_{T-1}, A_{T-1}, R_T$

  $G \leftarrow 0$

  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

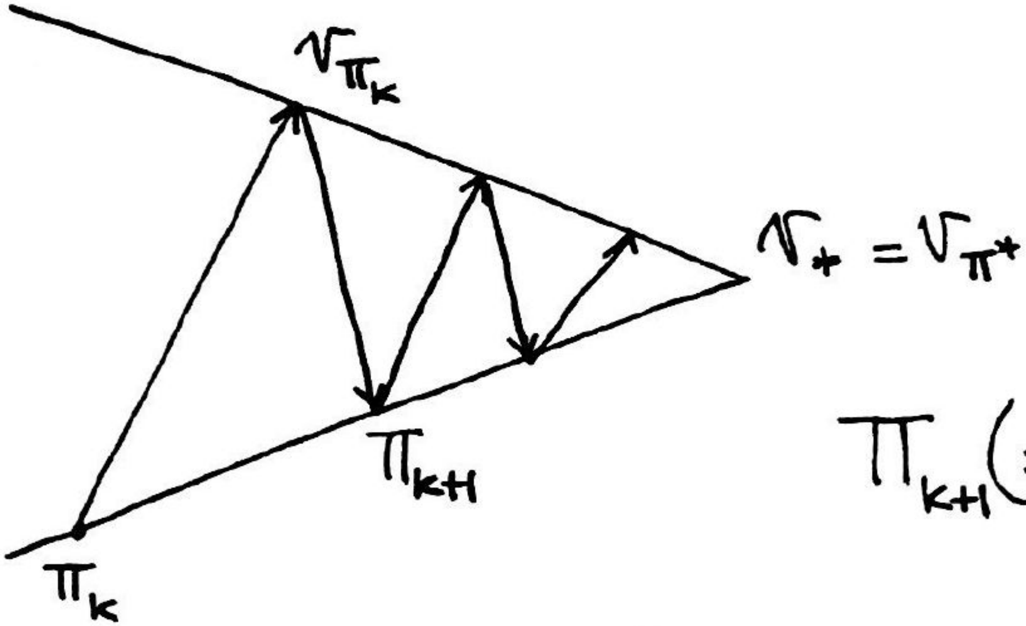
    $G \leftarrow \gamma G + R_{t+1}$

    Unless S_t appears in $S_0, S_1, \ldots, S_{t-1}$:

      Append G to $Returns(S_t)$

      $V(S_t) \leftarrow \text{average}(Returns(S_t))$

Evaluación y mejora



$$\pi_{k+1}(s) := \arg\max_a q_{\pi_k}(s,a)$$

$$\pi_k \longrightarrow v_{\pi_k} \longrightarrow q_{\pi_k} \longrightarrow \pi_{k+1}$$

Exploration

⚠ preciso reconstruir $q_\pi \Rightarrow \{P^{a}_{s,s'}\}_{s,s'} \quad \forall a$

$\Rightarrow$ preciso visitar los $|A| \times |S|$ estados!

- *Eploration starts*: Cada corrida comenzarla recorriendo todos los posibles pares (s,a)

Monte Carlo ES - Online

Monte Carlo ES (Exploring Starts), for estimating $\pi \approx \pi_*$

Initialize:

$\pi(s) \in \mathcal{A}(s)$ (arbitrarily), for all $s \in \mathcal{S}$

$Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

$Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Loop forever (for each episode):

Choose $S_0 \in \mathcal{S}$, $A_0 \in \mathcal{A}(S_0)$ randomly such that all pairs have probability > 0

Generate an episode from S_0, A_0, following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots, S_{t-1}, A_{t-1}$:

Append G to $Returns(S_t, A_t)$

$Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$

$\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$

Otra manera, on policy

- On policy MC **control**: En cada corrida, calcular la política mejorada, y con su versión ε-greedy samplear la siguiente corrida

Blackjack - David Silver

- States (200 of them):
  - Current sum (12-21)
  - Dealer's showing card (ace-10)
  - Do I have a "useable" ace? (yes-no)
- Action stick: Stop receiving cards (and terminate)
- Action twist: Take another card (no replacement)
- Reward for stick:
  - $+1$ if sum of cards $>$ sum of dealer cards
  - 0 if sum of cards $=$ sum of dealer cards
  - -1 if sum of cards $<$ sum of dealer cards
- Reward for twist:
  - -1 if sum of cards > 21 (and terminate)
  - 0 otherwise
- Transitions: automatically twist if sum of cards < 12

Comentarios sobre MC

- La aproximación para un estado no depende de la aproximación para los otros estados. Por lo tanto puedo aproximar la función de valor para sólo algunos estados.

- En algunos casos, inclusive conociendo la distribución del ambiente, puede ser inclusive más conveniente que usar ecuaciones de Bellman.

Off Policy Prediction via Importance Sampling

- Busco una política **óptima (*evaluation*),** pero para eso preciso **aproximar bien** la dinámica del ambiente (***exploration***).

Tengo dos políticas: la que quiero obtener, **π** (*target*) y la que voy a usar para explorar b (*behavior*).

Por ahora, sólo quiero evaluar la política **π**
¿Cómo estimar **π usando muestras generadas con b?**

Haz lo que yo digo, pero no lo que yo hago.

Importance Sampling

$$\mathbb{E}_p[f(X)] = \sum_x f(x)\,p(x)$$

puede que $p(x) >> 0$ PERO $|f(x)| \simeq 0$

$\Rightarrow$ MC no funciona bien!

- p puede ser difícil de samplear o tener varianza grande.

Aproximar la esperanza con respecto a p mediante muestras obtenidas con q

$$E_p[f(X)] = \sum_x f(x) \cdot q(x) \cdot \frac{p(x)}{q(x)}$$

$$= E_q\left[f(X) \cdot \frac{p(X)}{q(X)}\right]$$

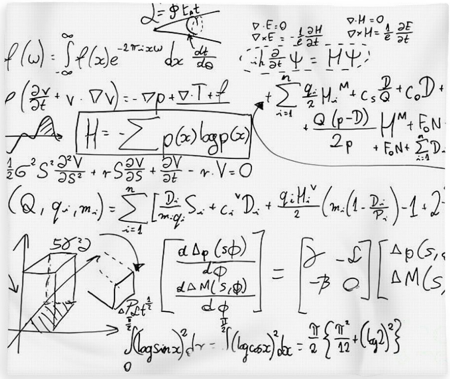
$$\simeq \frac{\sum_{i=1}^{n} f(X_i) \frac{p(X_i)}{q(X_i)}}{n} \quad ; \; X_i \sim q$$

$$w_s = \frac{p(X_s)}{q(X_s)}$$

IMPORTANCE WEIGHTS

$$V(\bar{X}) = \frac{1}{n} \cdot V(X)$$

Posible criterio para determinar q :



$$\min_{q} V_q\left(f(X)\,w(X)\right)$$

$$q^*(x) = \frac{|f(x)|\,p(x)}{\sum_x |f(x)|\,p(x)}$$

Referencia: Machine Lerning a Probabilistic Perspective: https://www.cs.ubc.ca/~murphyk/MLbook/

Volviendo al problema original

$$P_\pi(A_t = a, S_{t+1} = s_{t+1}, A_{t+1} = a_{t+1}, \dots, S_T = s_T \mid S_t = s)$$

$$= P_\pi(A_t = a, S_{t+1:T} = s_{t+1:T}, A_{t+1:T} = a_{t+1:T} \mid S_t = s)$$

$$= \pi(a|s)\, p^{a}_{s,s_{t+1}}\, \pi(a_{t+1}|s_{t+1})\, p^{a_{t+1}}_{s_{t+1},s_{t+2}} \cdots p^{a_{T-1}}_{s_{T-1},s_T}$$

$$\Rightarrow \quad \omega_{t:T-1} = \pi(a|s) \cdot \prod_{k=t+1}^{T-1} \frac{\pi(a_k|s_k)}{b(a_k|s_k)}$$

Volviendo al problema original

$$E_\pi[G_t \mid S_t = s] = E_\pi[\varphi(A_t, S_{t+1}, A_{t+1}, \dots, S_T) \mid S_t = s]$$

$$= E_b[\varphi(A_t, S_{t+1:T}, A_{t+1:T-1}) \cdot \omega(A_{t:T-1})]$$

- Tiene mayor varianza que hacer MC directamente y converge más lento.
- Me permite usar experiencia generada con otra política.

Hacia Temporal difference (Evaluation)

$$v_\pi(s) = E\left[R_{t+1} + \gamma E\left[G_{t+2} \mid S_{t+1}\right] \mid S_t = s\right]$$

$$= E\left[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s\right]$$

$$= v_\pi(s) + E\left[\left\{R_{t+1} + \gamma v_\pi(S_{t+1})\right\} - v_\pi(s) \mid S_t = s\right]$$

La máquina de hacer chorizos

$$v_\pi(s) = E\left[R_{t+1} + \gamma\, E\left[G_{t+2} \mid S_{t+1}\right] \mid S_t = s\right]$$

$$= E\left[R_{t+1} + \gamma\, v_\pi(S_{t+1}) \mid S_t = s\right]$$

$$= v_\pi(s) + E\left[\left\{R_{t+1} + \gamma\, v_\pi(S_{t+1})\right\} - v_\pi(s) \mid S_t = s\right]$$

Monte-Carlo

Temporal Difference

$$v_\pi(s) + E\left[\{R_{t+1} + \gamma v_\pi(S_{t+1})\} - v_\pi(s) \middle| S_t = s\right]$$

Monte-Carlo

$$v^{k+1}(s) = v^k(s) + \frac{\left(G^{k+1} - v^k(s)\right)}{N^k(s) + 1}$$

Temporal Difference

$$v^{t+1}(S_t) = v^t(S_t) + \alpha\left[\underbrace{\overbrace{R_{t+1} + \gamma v^t(S_{t+1})}^{\textit{Estimated Return}} - v^t(S_t)}_{\textit{TD Error}}\right]$$

MC vs TD (Varianza vs Sesgo)

- Monte Carlo usa todo el episodio para tener una nueva aproximación.

- Temporal difference en cada paso, genera una nueva aproximación

El lado oscuro...

If both TD and Monte Carlo methods converge asymptotically to the correct predictions, then a natural next question is "Which gets there first?" In other words, which method learns faster? Which makes the more efficient use of limited data? At the current time this is an open question in the sense that no one has been able to prove mathematically that one method converges faster than the other. In fact, it is not even clear what is the most appropriate formal way to phrase this question! In practice, however, TD methods

Learning to predict by the methods of **temporal differences, Sutton, 1988:**

https://link.springer.com/content/pdf/10.1007/BF00115009.pdf


TENGO HASTA AHÍ...
GENERADORMEMES.COM