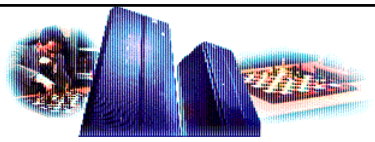
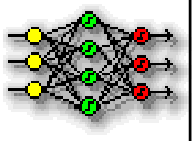




Sistemi Intelligenti
Reinforcement Learning: Eligibility Trace

Alberto Borghese

Università degli Studi di Milano
Laboratorio di Sistemi Intelligenti Applicati (AIS-Lab)
Dipartimento di Scienze dell'Informazione
borghese@dsi.unimi.it

Sommario

The elegibility trace

SARSA(λ) & Q(λ)

Proprietà del rinforzo

L'ambiente o l'interazione può essere complessa.

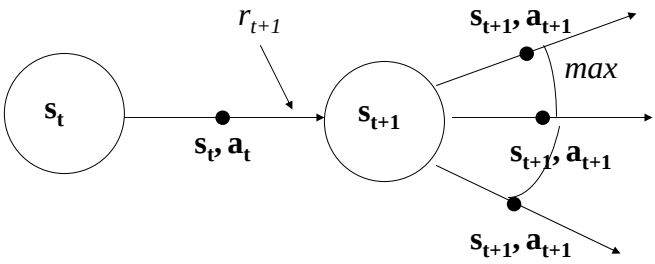
Il rinforzo può avvenire solo dopo una più o meno lunga sequenza di azioni (delayed reward).

E.g. agente = giocatore di scacchi.
ambiente = avversario.

Problemi collegati:
temporal credit assignement.
structural credit assignement.

L'apprendimento non è più da esempi, ma dall'osservazione del proprio comportamento nell'ambiente.

Q-learning: Rappresentazione grafica



$Q(s_t, a_t)$ $Q(s_{t+1}, a_{t+1})$

One step for **Q Iteration**

Viene migliorata la policy al tempo t+1.

Formulazione di TD(0) per Q-learning

$$Q_{k+1}(s_t,a_t)=Q_k(s_t,a_t)+\alpha\left[r_{t+1}+\gamma\max_{a_{t+1}}Q_k(s_{t+1},a_{t+1})-Q_k(s_t,a_t)\right]$$

$$\Delta Q(s_t,a_t)=+\alpha\delta_k \qquad \delta_k=\left[r_{t+1}+\gamma\max_{a_{t+1}}Q_k(s_{t+1},a_{t+1})-Q_k(s_t,a_t)\right]$$

$$V_{k+1}(s_t)=V_k(s_t)+\alpha\left[r_{t+1}+\gamma V_k(s_{t+1})-V_k(s_t)\right]$$

$$\Delta V(s_t)=+\alpha\,\delta_k \qquad \delta_k=\left[r_{t+1}+\gamma V_k(s_{t+1})-V_k(s_t)\right]$$

Cosa rappresenta la Eligibility trace

Buffer di memoria: contiene traccia di eventi passati (stati visitati, azioni…); la traccia evapora nel tempo.

Quando viene calcolato un errore usando metodi basati su TD, la eligibility trace suggerisce quali variabili aggiornare (credit assignement).

Amplia l'orizzonte temporale sul quale fare l'aggiornamento a più di 1 passo.

View of the idea

Considero il reward su orizzonti temporali più ampi:

$R_t = r_{t+1} + \gamma\, V(s_{t+1})$
$R_t = r_{t+1} + \gamma\, r_{t+1} + \gamma^2\, V(s_{t+2})$
$R_t = r_{t+1} + \gamma\, r_{t+2} + \gamma^2\, r_{t+3} + \gamma^3\, V(s_{t+3})$
……..

$$R_t = \frac{1}{M+1}\sum_{k=0}^{M}\gamma^k\left[r_{t+1+k} + \gamma V(s_{t+1+k})\right]$$

Aggiornamento della value function: $\Delta V(s) = \alpha\, \delta_t$

$$\delta_t = R_t - V_t(s_t) = \left(\frac{1}{M+1}\sum_{k=0}^{M}\gamma^k\left[r_{t+1+k} + \gamma V(s_{t+1+k})\right] - V(s_t)\right)$$

Estensione a più istanti di tempo

Considero il reward su orizzonti temporali più ampi:

$R_t = r_{t+1} + \gamma\, V(s_{t+1})$
$R_t = r_{t+1} + \gamma\, r_{t+1} + \gamma^2\, V(s_{t+2})$
$R_t = r_{t+1} + \gamma\, r_{t+1} + \gamma^2\, r_{t+2} + \gamma^3\, V(s_{t+3})$
……..

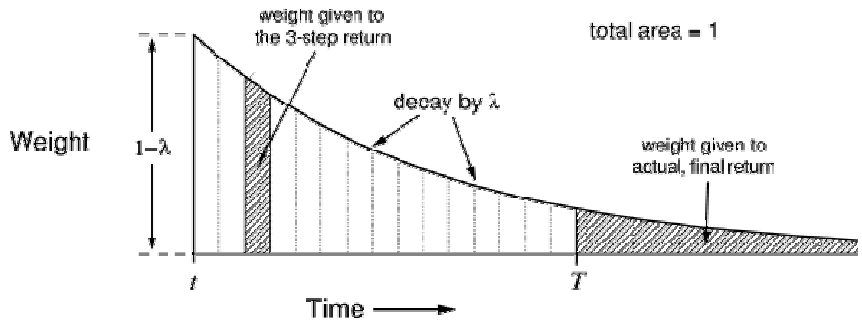
Ne faccio una media pesando di più i reward nell'immediato futuro:

$R^{\lambda}_t = K * (R_t^{(1)} + \lambda\, R_t^{(2)} + \lambda^2\, R_t^{(3)} + \ldots..)$

$$R^{\lambda}_t = (1-\lambda) * \sum_{k=1}^{+\infty}\lambda^{k-1} R_t^{(k)}$$

Dipende da λ e da γ!

Visualizzazione grafica



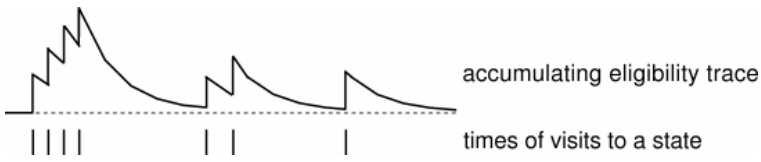
$\lambda = 0$ TD(0) $\Delta V_t = \alpha [R_t^{\lambda} - V_t(s_t)]$

λ regola la velocità di decremento del peso del reward.

Problemi: TD(λ) in questa forma è non causale.

Eligibility trace

$$e_t(s) = \begin{cases} \gamma\lambda e_{t-1}(s) & \text{if } s \neq s_t; \\ \gamma\lambda e_{t-1}(s) + 1 & \text{if } s = s_t, \end{cases}$$



$\lambda < 1;\ \gamma < 1;\quad e(s) \geq 0$

Come utilizzare la eligibility trace

TD(0) Learning:

$$V_{k+1}(s_t) = V_k(s_t) + \alpha\left[r_{t+1} + \gamma V_k(s_{t+1}) - V_k(s_t)\right]$$

Errore: δ_t

$$V_{k+1}(s_t) = V_k(s_t) + \alpha\delta_t$$ Per 1 stato

$$V_{k+1}(s) = V_k(s) + \alpha\delta_t e_t(s)$$ Per tutti gli stati

Eleggibilità: $e_t(s)$

Algoritmo

Initialize $V(s)$ arbitrarily and $e(s) = 0$, for all $s \in \mathcal{S}$
Repeat (for each episode):
  Initialize s
  Repeat (for each step of episode):
    $a \leftarrow$ action given by π for s
    Take action a, observe reward, r, and next state, s'
    $\delta \leftarrow r + \gamma V(s') - V(s)$
    $e(s) \leftarrow e(s) + 1$
    For all s:
      $V(s) \leftarrow V(s) + \alpha\delta e(s)$
      $e(s) \leftarrow \gamma\lambda e(s)$
    $s \leftarrow s'$
  until s is terminal

Sommario

The elegibility trace

SARSA(λ) & Q(λ)

Sarsa(λ)

Ci focalizziamo su Q(s,a) -> Eligibility trace: e(s,a)

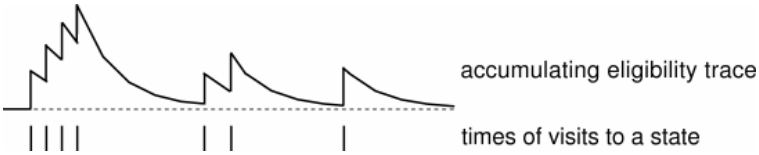
Equazione di apprendimento:

$$Q_{k+1}(s_t,a_t)=Q_k(s_t,a_t)+\alpha\delta_k e(s_t,a_t) \qquad \forall s, a$$

$$\delta_k=[r_{t+1}+\gamma Q_k(s_{t+1},a_{t+1})-Q_k(s_t,a_t)]$$

Aggiornamento dell'eligibility trace

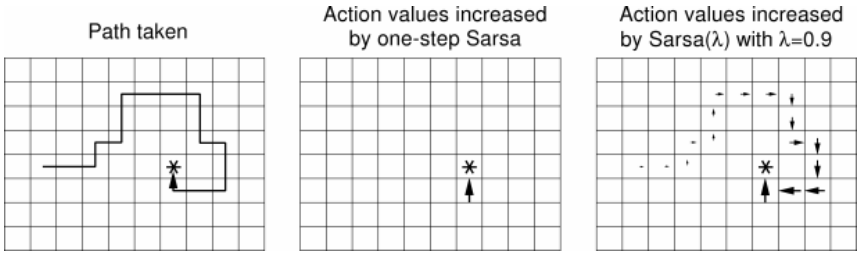
$$e_t(s,a) = \begin{cases} \gamma\lambda e_{t-1}(s,a) + 1 & \text{if } s = s_t \text{ and } a = a_t; \\ \gamma\lambda e_{t-1}(s,a) & \text{otherwise.} \end{cases} \qquad \text{for all } s, a$$



Algoritmo

Initialize $Q(s,a)$ arbitrarily and $e(s,a) = 0$, for all s, a
Repeat (for each episode):
  Initialize s, a
  Repeat (for each step of episode):
    Take action a, observe r, s'
    Choose a' from s' using policy derived from Q (e.g., ε-greedy)
    $\delta \leftarrow r + \gamma Q(s',a') - Q(s,a)$
    $e(s,a) \leftarrow e(s,a) + 1$
    For all s, a:
      $Q(s,a) \leftarrow Q(s,a) + \alpha\delta e(s,a)$
      $e(s,a) \leftarrow \gamma\lambda e(s,a)$
    $s \leftarrow s'$; $a \leftarrow a'$
  until s is terminal

Esempio



Con il semplice costo di una variabile per ogni stato,
ho un aggiornamento graduale della funzione valore di più stati.

Q(s,a) inizializzati ad un valore leggermente negativo.
r = 0 per ogni stato prossimo, tranne lo stato finale, per il quale r = +1.

Q-learning is off-policy

La Q(s,a) che viene appresa può non essere la stessa utilizzata per scegliere le azioni, perchè viene modificata la policy durante l'interazione.

In particolare, spesso Q(s,a) si riferisce alla politica greedy, mentre la scelta delle azioni può essere ε-greedy rispetto a Q.

Quando apprendiamo Q(s,a), ci riferiamo ad una certa politica della quale vogliamo misurare il reward a lungo termine. Supponiamo che questa politica sia la politica greedy. Cosa succede, se ad un passo, scegliamo un'azione non-greedy?

Q(λ)

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left[r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)\right]$$

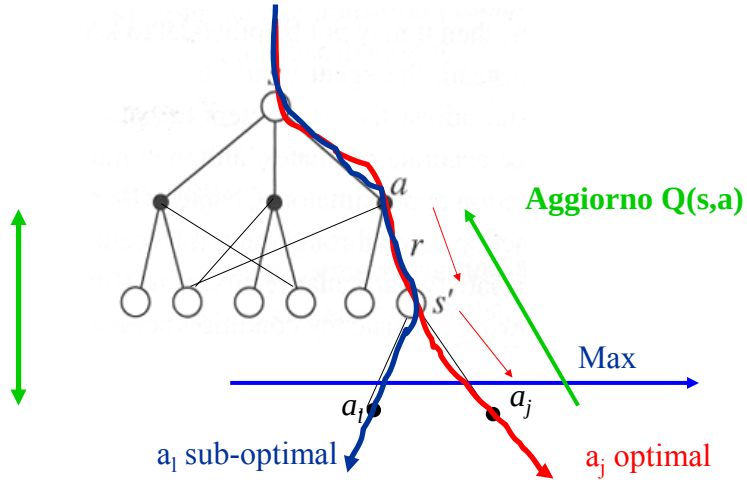
Quanto posso guardare in avanti (look-ahead)?

Suppongo di scegliere a' = a_{t+1} azione esplorativa.

Posso sempre calcolare $Q(s_t, a_t)$, scegliendo il $\max(Q(s_{t+1}, a_{t+1}))$. Questo vuole dire ipotizzare di scegliere a_{max} = argmax(max(Q(st+1, at+1)), che in questo caso: $a_{max} \neq a'$.

Ma poi devo ripartire da capo.

Analisi grafica delle mosse ε-eplorative



Q-learning

$$e_t(s,a) = \mathcal{I}_{ss_t} \cdot \mathcal{I}_{aa_t} + \begin{cases} \gamma\lambda e_{t-1}(s,a) & \text{if } Q_{t-1}(s_t,a_t) = \max_a Q_{t-1}(s_t,a); \\ 0 & \text{otherwise,} \end{cases}$$

Aggiorno Q:

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha\,\delta_t\, e_t(s,a),$$

$$\delta_t = r_{t+1} + \gamma \max_{a'} Q_t(s_{t+1},a') - Q_t(s_t,a_t).$$

Scelta di a:

Se scelgo a_{max}, continuo come SARSA, altrimenti e(s,a) = 0.

Algoritmo

Initialize $Q(s,a)$ arbitrarily and $e(s,a) = 0$, for all s,a
Repeat (for each episode):
  Initialize s, a
  Repeat (for each step of episode):
    Take action a, observe r, s'
    Choose a' from s' using policy derived from Q (e.g., ε-greedy)
    $a^* \leftarrow \arg\max_b Q(s',b)$ (if a' ties for the max, then $a^* \leftarrow a'$)
    $\delta \leftarrow r + \gamma Q(s',a^*) - Q(s,a)$
    $e(s,a) \leftarrow e(s,a) + 1$
    For all s,a:
      $Q(s,a) \leftarrow Q(s,a) + \alpha\delta e(s,a)$
      If $a' = a^*$, then $e(s,a) \leftarrow \gamma\lambda e(s,a)$
                  else $e(s,a) \leftarrow 0$
    $s \leftarrow s'$; $a \leftarrow a'$
  until s is terminal

Sommario

The elegibility trace

SARSA(λ) & Q(λ)