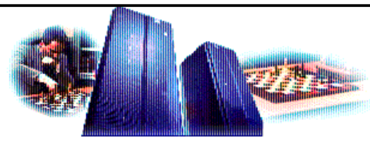
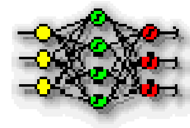




Sistemi Intelligenti Reinforcement Learning: Equazioni di Bellman

Alberto Borghese

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

Sommario

Determinazione della value function. Esempio.

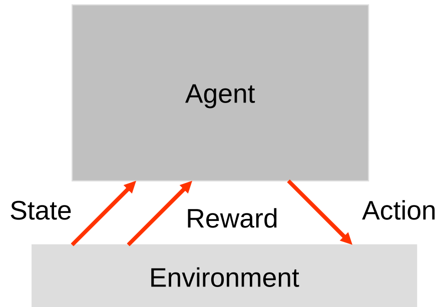
Le equazioni di Bellman

Reinforcement Learning Problem

Given: Repeatedly…
- Executed action
- Observed state
- Observed reward

Learn action policy π: $S \rightarrow A$
- Maximizes life reward
  $r_0 + \gamma r_1 + \gamma^2 r_2 \ldots$
  from any start state.
- Discount: $0 < \gamma < 1$

Note:
- Unsupervised learning
- Delayed reward



$$s_0 \xrightarrow[r_0]{a_0} s_1 \xrightarrow[r_1]{a_1} s_2 \xrightarrow[r_2]{a_2} s_3$$

Goal: Learn to choose actions that maximize life reward

$$r_0 + \gamma r_1 + \gamma^2 r_2 \ldots$$

Il nostro filo logico

La Value function ci serve per decidere l'azione migliore.
Per calcolare la Value function devo collezionare reward futuro.

Come se ne esce?

Determinazione algebrica della Value Function
Determinazione "esplorativa" della Value Function
Determinazione della Value Function e scelta della policy via via sempre più intrecciate tra loro.

Esempio: AIBO search

Azioni:
1) Rimanere fermo e aspettare che qualcuno getti nel cestino una lattina vuota.
2) Muoversi attivamente in cerca di lattine.
3) Tornare alla sua base (recharge station) e ricaricarsi.
Stato:
1) Alto livello di energia.
2) Basso livello di energia.
Policy:
A(s = high) = {Search, Wait}
A(s = low) = {Search, Wait, Recharge}

Goal: collezionare il maggior numero di lattine.

Funzionamento del Robot

Funzione Stato prossimo: $P_{s->s'|a} = \Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$

Se il livello di energia è alto (s_t = alto):
se scelgo Wait – s_{t+1} = alto.
se scelgo Search, s_{t+1} avrà una certa probabilità di diventare low.

$$P_{high->low|Search} = \Pr\{s_{t+1} = low | s_t = high, a_t = Search\} = \alpha$$

Se il livello di energia è basso (s_t = basso):
se scelgo Wait – s_{t+1} = basso.
se scelgo Recharge - s_{t+1} = alto.
se scelgo Search, s_{t+1} avrà una certa probabilità di fermarsi.

$$P_{low->low|Search} = \Pr\{s_{t+1} = low | s_t = low, a_t = Search\} = \beta$$

Reward del Robot

Funzione Reward: $R_{s->s'|a} = E\{r_{t+1} = r' | s_t = s, a_t = a, s_{t+1} = s'\}$

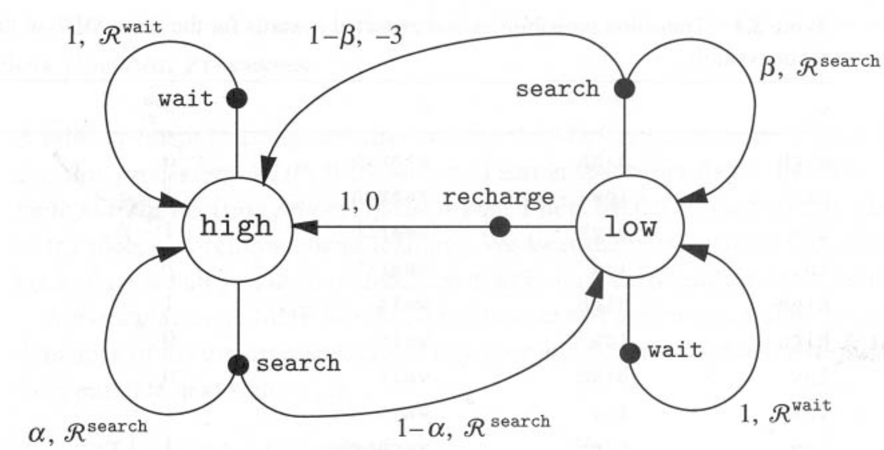
R^{search} reward se il robot sta cercando.

R^{wait} reward se il robot sta cercando.

R^{die} se occcorre portarlo a ricaricarsi.

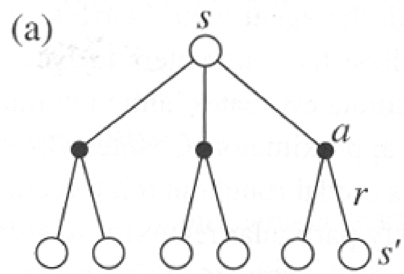
$R^{recharge}$ se il robot va autonomamente a ricaricarsi. $R^{search} > R^{wait}. > R^{recharge} > R^{die}$

State Transition Graph



Probabilistic Finite State Machine

Policy



La policy deve essere ancora determinata. Come fa l'agente a determinare la policy ottimale?

Archi multipli fuoriuscenti da un'azione sono associati alla probabilità di scegliere quel cammino (ambiente stocastico).

Archi multipli fuoriuscenti da uno stato, sono associati alla policy.

Value function & policy

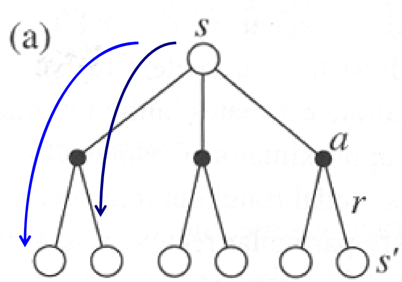
Nulla è detto sulla policy: dato uno stato, in quale nodo azione mi sposto?

Vogliamo costruire agenti lungimirar

State-Value function
(function of the policy):

$$V^{\pi}(s) = E_{\pi}\{R_t \mid s_t = s\} =$$

$$E_{\pi}\left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s \right\}$$



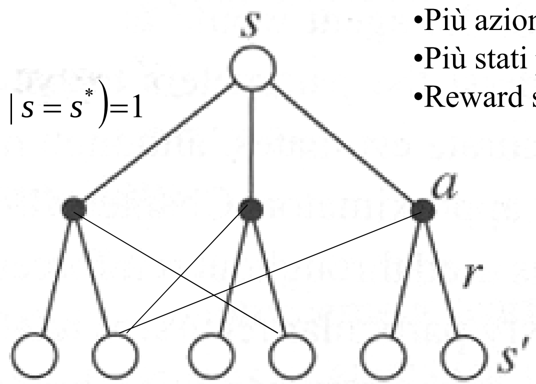
Massimizzo la ricompensa a lungo termine, V(.). Dipende dalla policy:

Value function e modelli markoviani

Anche la policy può essere stocastica.

Per ogni stato devo valutare:
- Più azioni.
- Più stati prossimi
- Reward stocastici.

$$\sum_{j=1}^{N\_azioni} \Pr\left(a_j \mid s = s^*\right) = 1$$



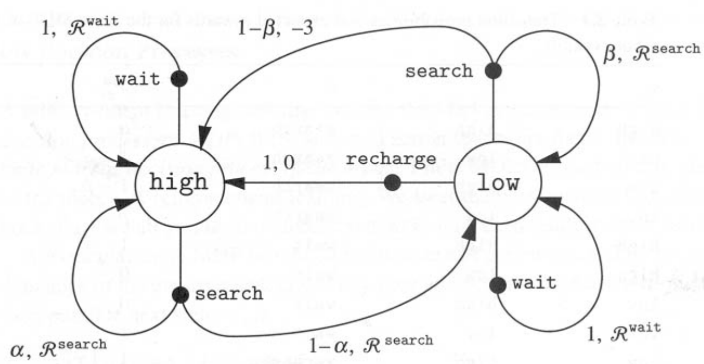
$$\sum_{k=1}^{N\_stati} \Pr\left(s_{t+1} = s_k \mid s_t = s'; a_t = a_j\right) = 1$$

$$V^{\pi}(s) = E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s\right\}$$

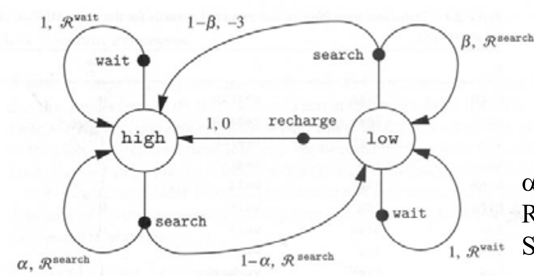
Esempio di calcolo della Value function

Value function
V(high) = ?
V(low) = ?

Policy
a(high) = ?
a(low) = ?



Policy deterministica



$\alpha=0.4$, $\beta=0.1$, $\gamma=0.8$,
$R^{search}=3$, $R^{wait}=1$ $R^{recharge} = -3$, $R^{auto} = 0$
S = H - a = Wait; s = L - a = Search;

Vh = [1+0.8Vh]

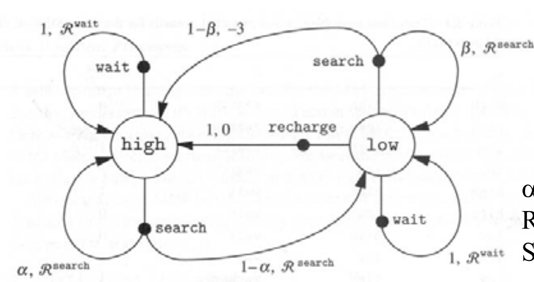
Wait

Vl = 0.1x[3+0.8xVl]+0.9x[-3+0.8Vh]

Search -> Low *Search -> High*

Sistema lineare di 2 equazioni nelle 2 incognite: Vh e Vl
Come calcolo Vh e Vl?

Policy deterministica



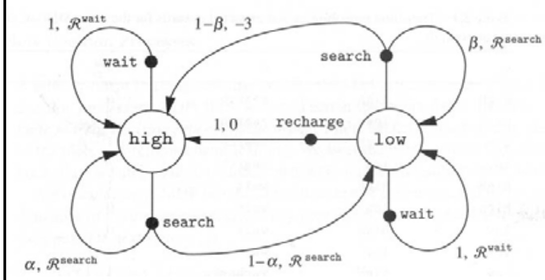
$\alpha=0.4$, $\beta=0.1$, $\gamma=0.8$,
$R^{search}=3$, $R^{wait}=1$ $R^{recharge} = -3$, $R^{auto} = 0$
S = H - a = Wait; s = L - a = Search;

Vh = [1+0.8Vh] → Vh = 5

Vl = 0.1x[3+0.8xVl]+0.9x[-3+0.8Vh]

Vl = 0.3 + 0.08 Vl -0.27 +0.72Vh → Vl(1-0.08) = 0.3-0.27+3.6
→ Vl = 3,945

Esempio di calcolo della funzione valore



α=0.4, β=0.1, γ=0.8, R^{search}=3, R^{wait}=1

Vh = Pr(W)x1x[1+0.8Vh] + Pr(S)x0.4x[3+0.8xVh]+Pr(S)x0.6x[3+0.8Vl]

Wait *Search -> high* *Search -> low*

Vl = Pr(W)x1x[1+0.8Vl] + Pr(S)x0.1x[3+0.8xVl]+Pr(S)x0.9x[-3+0.8Vh]+

Pr(R)x1x[0+0.8Vh]

Wait *Recharge* *Search -> High*

Search -> Low

Sistema lineare di 2 equazioni nelle 2 incognite: Vh e Vl

Come calcolo Vh e Vl?

Esempio di calcolo della funzione valore - I

Vh = Pr(W)x1x[1+0.8Vh] + Pr(S)x0.4x[3+0.8xVh]+Pr(S)x0.6x[3+0.8Vl]

Wait *Search -> high* *Search -> low*

Vl = Pr(W)x1x[1+0.8Vl] + Pr(S)x0.1x[3+0.8xVl]+Pr(S)x0.9x[-3+0.8Vh]+

Pr(R)x1x[0+0.8Vh]

Wait *Recharge* *Search -> High*

Search -> Low

Devo specificare una policy (stocastica):

s = high [Pr(W) = 0.4 Pr(S) = 0.6]

s = low [Pr(W) = 0.4 Pr(S) = 0.5 Pr(R) = 0.1]

Vh = 0.4x1x[1+0.8Vh] + 0.6x0.4x[3+0.8xVh]+0.6x0.6x[3+0.8Vl]

Vl = 0.4x1x[1+0.8Vl] + 0.5x0.1x[3+0.8xVl]+0.5x0.9x[-3+0.8Vh]+

0.1x1x[0+0.8Vh]

Esempio di calcolo della funzione valore - II

Devo specificare una policy (stocastica):

s = high [Pr(W) = 0.4 Pr(S) = 0.6]
s = low [Pr(W) = 0.4 Pr(S) = 0.5 Pr(R) = 0.1]

Vh = 0.4x1x[1+0.8Vh] + 0.6x0.4x[3+0.8xVh]+0.6x0.6x[3+0.8Vl]
Vl = 0.4x1x[1+0.8Vl] + 0.5x0.1x[3+0.8xVl]+0.5x0.9x[-3+0.8Vh]+ 0.1x1x[0+0.8Vh]

0.488 Vh = 2.20 + 0.288 Vl => Vh ≈ 6,35
0.64Vl = -0.8 + 0.44 Vh => Vl ≈ 3,12

Cosa si può concludere? Vh > Vl. Ma non molto altro.

Esempio di calcolo della funzione valore - III

Stessa policy (stocastica) ma altro ambiente (robot che si scarica più difficilmente):
s = high [Pr(W) = 0.4 Pr(S) = 0.6]
s = low [Pr(W) = 0.4 Pr(S) = 0.5 Pr(R) = 0.1]

$\alpha=0.4$, $\beta=0.9$, $\gamma=0.8$, $R^{search}=3$, $R^{wait}=1$

Vh = 0.4x1x[1+0.8Vh] + 0.6x0.4x[3+0.8xVh]+0.6x0.6x[3+0.8Vl]
Vl = 0.4x1x[1+0.8Vl] + 0.5x0.9x[3+0.8xVl]+0.5x0.1x[-3+0.8Vh] + 0.1x1x[0+0.8Vh]

0.488 Vh = 2.20 + 0.288 Vl => Vh ≈ 30,93
0.32Vl = 1.9 + 0.04 Vh => Vl ≈ 6,904

Esempio di calcolo della funzione valore - IV

Devo specificare una policy (stocastica), robot più attivo:

s = high [Pr(W) = 0.2 Pr(S) = 0.8]
s = low [Pr(W) = 0.2 Pr(S) = 0.7 Pr(R) = 0.1]

$$Vh = 0.2x1x[1+0.8Vh] + 0.8x0.4x[3+0.8xVh]+0.8x0.6x[3+0.8Vl]$$
$$Vl = 0.2x1x[1+0.8Vl] + 0.7x0.1x[3+0.8xVl]+0.7x0.9x[-3+0.8Vh]+ 0.1x1x[0+0.8Vh]$$

$0.488\ Vh = 2.20 + 0.288\ Vl \quad => \quad Vh \approx -14.2$
$0.64Vl = -0.8 + 0.44\ Vh \quad => \quad Vl \approx -28.465$

E' una policy peggiore.

Esempio di calcolo della funzione valore: ambiente generoso e stessa policy aggressiva

Stessa policy (stocastica) ma altro ambiente (robot che si scarica più difficilmente):
s = high [Pr(W) = 0.2 Pr(S) = 0.8]
s = low [Pr(W) = 0.2 Pr(S) = 0.7 Pr(R) = 0.1]

$\alpha=0.4, \beta=0.9, \gamma=0.8, R^{search}=6, R^{wait}=1$

$$Vh = 0.2x1x[1+0.8Vh] + 0.8x0.4x[6+0.8xVh]+0.6x0.6x[6+0.8Vl]$$
$$Vl = 0.2x1x[1+0.8Vl] + 0.7x0.9x[6+0.8xVl]+0.7x0.1x[-3+0.8Vh] + 0.1x1x[0+0.8Vh]$$

$0.584\ Vh = 5.00 + 0.384\ Vl \quad => \quad Vh \approx 13{,}12$
$0.784\ Vl = -1.48 + 0.512\ Vh \quad => \quad Vl \approx 6{,}942$

Come utilizziamo V per determinare la policy ottimale?

Sommario

Determinazione della value function. Esempio.

Le equazioni di Bellman

Il modello markoviano

Il comportamento dell'ambiente è definito dallo stato: S = $\{s_j\}$
Per ogni stato l'agente sceglie un'azione: a = a(s) A = $\{a_k\}$
Policy di un agente: π(s, a) è quanto dobbiamo definire.

L'ambiente ha una evoluzione stocastica rappresentata da un MDP:

$$P_{s_t = s -> s_{t+1} = s' | a_t = a} = \Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$$

Inoltre, ad ogni istante fornisce un reward immediato associato alla transizione, stimato all'istante t come:

$$R_{s_t = s -> s_{t+1} = s' | a_t = a} = E\{r_{t+1} = r' | s_t = s, a_t = a, s_{t+1} = s'\}$$

$\forall$**s** $\in$**S;** $\forall$**a** $\in$**A**

Probabilità composte

Probabilità degli eventi. It is a way of expressing knowledge or belief that an event will occur or has occurred. It ranges between 0 and 1.

Probabilità composta nel caso di eventi indipendenti:
  Probabilità che due dadi diano 6 + 6 (probabilità **congiunta**)?
_ Probabilità che in una cassetta una mela sia gialla e bacata?
  Probabilità che uno di due dadi dia 2? (p(1=2) * p(2 = 2) – p(1=2)p(2=2)

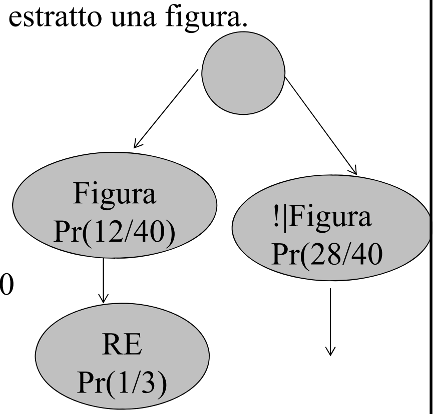
Probabilità condizionata

Probabilità condizionata - consideriamo un mazzo di 40 carte:
  vogliamo valutare quale sia la probabilità che una carta estratta a caso sia un re.
  vogliamo valutare quale sia la probabilità che una carta estratta a caso sia un re, sapendo di avere estratto una figura.

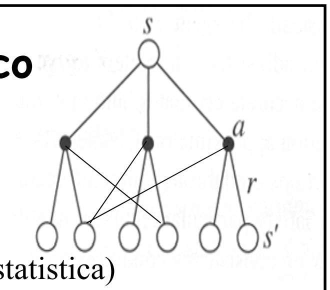
P(A) = probabilità che sia un re
P(B) = probabilità che sia una figura

p(A | B) = 1/3

p(A) = p(A/B) p(B) = 1/3 12/40 = 4/40



Reward stocastico



$$R_{s->s'|a} = E\{r_{t+1} = r' | s_t = s, a_t = a, s_{t+1} = s'\}$$

reward = 3 – stocastico (da una distribuzione statistica)

È in realtà un valore condizionato in s e vale:

Pr(reward = r | s’) = Pr(reward = r | s) / (Pr(a)Pr(s’))

Da Bayes:

Pr(reward = r | s) = Pr(reward = r | s’) * Pr(a | s) * Pr(s’ | s,a)
Questa è la probabilità congiunta di stato prossimo, azione e reward.

La value function

Nulla è detto sulla policy: dato uno stato, quale azione scegliere? In quale stato mi sposto?

Vogliamo costruire agenti lungimiranti.

State-Value function:

$$V^{\pi}(s) = E_{\pi}\{R_t \mid s_t = s\} = E_{\pi}\left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \,\middle|\, s_t = s \right\}$$

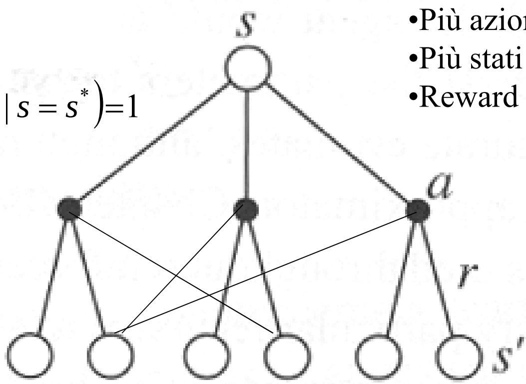
Massimizzo la ricompensa a lungo termine, V(.). Dipende dalla policy:

Value function e modelli markoviani

Anche la policy può essere stocastica.

Per ogni stato devo valutare:
- Più azioni.
- Più stati prossimi
- Reward stocastici.

$$\sum_{j=1}^{N\_azioni} \Pr\left(a_j \mid s = s^*\right) = 1$$



$$\sum_{k=1}^{N\_stati} \Pr\left(s_{t+1} = s_k \mid s_t = s'; a_t = a_j\right) = 1 \qquad V^{\pi}(s) = E_{\pi}\left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s \right\}$$

Calcolo ricorsivo della Value function

$$V^{\pi}(s) = E_{\pi}\{R_t \mid s_t = s\} = E_{\pi}\left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s \right\}$$

$$V^{\pi}(s') = E_{\pi}\{R_{t+1} \mid s_{t+1} = s'\}$$

Relazione?

$$V^{\pi}(s) = E_{\pi}\left\{ r_{t+1} + \gamma \sum_{k=1}^{\infty} \gamma^{k-1} r_{t+k+1} \middle| s_t = s \right\} =$$

$$V^{\pi}(s) = E_{\pi}\left\{ r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^{k} r_{t+k+2} \middle| s_t = s \right\}$$

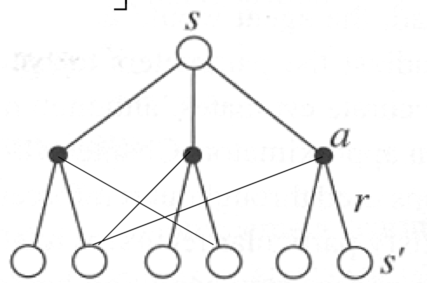
Io termine

IIo termine

$V^\pi(s)$: primo termine

$$E_\pi\{r_{t+1} \mid s_t = s\} = \left[\sum_{a_j} \pi(a_j, s)\right] \sum_{s'} P_{s \to s'|a_j} \left[R_{s \to s'|a_j}\right]$$

Per ogni stato devo valutare:
- •Più azioni.
- •Più stati prossimi
- •Reward stocastici nella transizione ad un passo

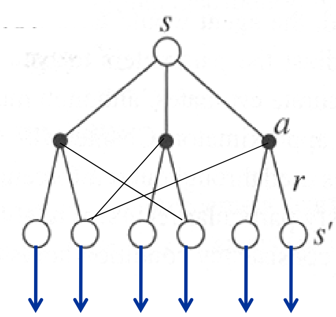


Visione Statistica: Probabilità di ottenere il reward: $R_{s \to s'|a_j}$ condizionata all'arrivare nello stato s', che a sua volta è condizionata allo scegliere l'azione a_j.

$V^\pi(s)$: secondo termine

$$V^\pi(s) = ... + E_\pi\left\{\gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \middle| s_t = s\right\} \qquad V^\pi(s') = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \middle| s_{t+1} = s'\right\}$$

Per ogni stato devo valutare:
- •Più azioni.
- •Più stati prossimi
- •Reward stocastici.

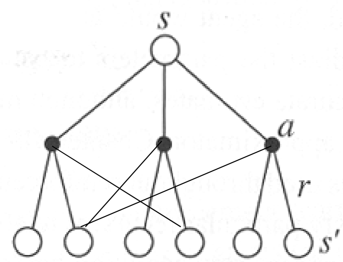


Nella valutazione dello stato s_t, sono considerati con un peso opportuno e scontati di γ, i reward a lungo termine che collezionerò a partire dai diversi stati s'.

$V^\pi(s)$: secondo termine

$$V^\pi(s) = ... + E_\pi\left\{\gamma\sum_{k=0}^{\infty}\gamma^k r_{t+k+2}\middle| s_t = s\right\} \qquad V^\pi(s') = E_\pi\left\{\sum_{k=0}^{\infty}\gamma^k r_{t+k+2}\middle| s_{t+1} = s'\right\}$$

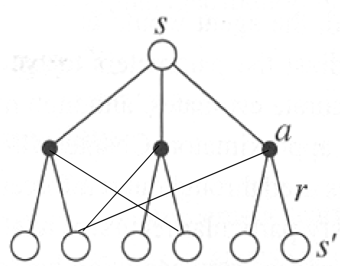
In s confluiranno i reward a lungo termine di tutti gli stati prossimi, s', ciascuno pesato con la probabilità di passare da s a s', ovverosia, in termini statistici, condizionati alla realizzazione della transizione di stato, s-> s'.



$$E_\pi\left\{\gamma\sum_{k=0}^{\infty}\gamma^k r_{t+k+2}\middle| s_t = s\right\} = \gamma\sum_{s'} E_\pi\{\sum_{k=0}^{\infty}\gamma^k r_{t+k+2} \mid s_{t+1} = s'\}\Pr(s_{t+1} = s' \mid s_t = s)$$

$V^\pi(s)$: secondo termine

$$E_\pi\left\{\gamma\sum_{k=0}^{\infty}\gamma^k r_{t+k+2}\middle| s_t = s\right\} = \gamma\sum_{t} E_\pi\{\sum_{k=0}^{\infty}\gamma^k r_{t+k+2} \mid s_{t+1} = s'_l\}\Pr(s_{t+1} = s_k \mid s_t = s)$$



$$E_\pi\left\{\gamma\sum_{k=0}^{\infty}\gamma^k r_{t+k+2}\middle| s_t = s\right\} = \left[\sum_{a_j}\pi\left(a_j, s\right)\right]\sum_{s'} P_{s\to s'|a_j}\left[\gamma V^\pi(s')\right]$$

Calcolo ricorsivo della Value function

$$V^{\pi}(s) = E_{\pi}\{R_t \mid s_t = s\} = E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s\right\}$$

$$V^{\pi}(s') = E_{\pi}\{R_{t+1} \mid s_{t+1} = s'\}$$

Legame?

$$P_{s->s'|a} = \Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$$

Policy Next-state

$$V^{\pi}(s) = \sum_{a_j} \pi(a_j, s) \sum_{s'} P_{s \to s'|a_j} R_{s \to s'|a_j} + E_{\pi}\left\{\gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \middle| s_t = s\right\}$$

$$\boxed{V^{\pi}(s) = \left\{\sum_{a_j} \pi(a_j, s) \sum_{s_l'} \left\{P_{s \to s_l'|a_j}\left[R_{s \to s_l'|a_j} + \gamma V^{\pi}(s_l')\right]\right\}\right\}}$$

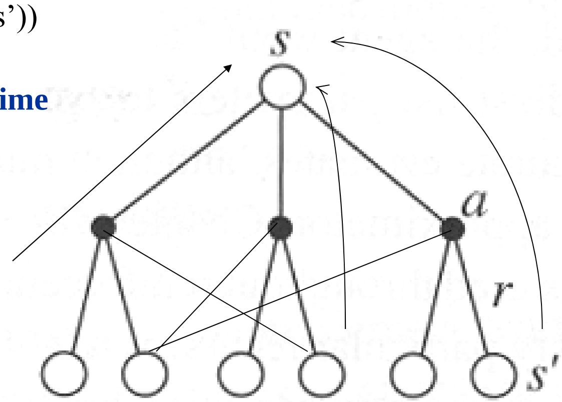
Bellman's equation

A.A. 2011-2012 33/36 http:\\homes.dsi.unimi.it\~borghese\

Osservazioni

$V^{\pi}(s) = \text{funz}(V^{\pi}(s'))$

Backwards in time



$$V^{\pi}(s) = \left\{\sum_{a_j} \pi(a_j, s) \sum_{s_l'} \left\{P_{s \to s_l'|a_j}\left[R_{s \to s_l'|a_j} + \gamma V^{\pi}(s_l')\right]\right\}\right\}$$

A.A. 2011-2012 34/36 http:\\homes.dsi.unimi.it\~borghese\

Confronto con il setting non associativo

	Setting non associativo	Setting associativo
Task	Azioni	Comportamenti (catena di azioni)
Reward	Reward istantaneo	Somma (scontata) dei reward collezionati lungo il task.
Max	Reward atteso sulla singola azione	Reward del comportamento
Orizzonte temporale del task	Finito (1 azione)	Finito / infinito per il singolo task
Policy	Stocastica	Stocastica
Stato	Non definito	Markoviano

Sommario

Determinazione della value function. Esempio.

Le equazioni di Bellman