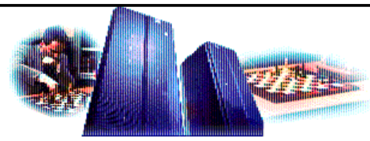
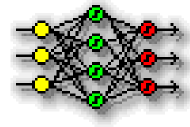




Sistemi Intelligenti Reinforcement Learning: Policy iteration

Alberto Borghese

Università degli Studi di Milano
Laboratorio di Sistemi Intelligenti Applicati (AIS-Lab)
Dipartimento di Informatica
alberto.borghese@unimi.it

Barto and Sutton, Capitoli 3 e 6

Sommario

Le equazioni di Bellman per policy deterministica

Le equazioni Bellman per policy stocastica

Iterative policy evaluation

Miglioramento della policy

The RL updated picture

Agent

What the world is like now (internal representation)?

s_t

What action should I choose now? (policy)

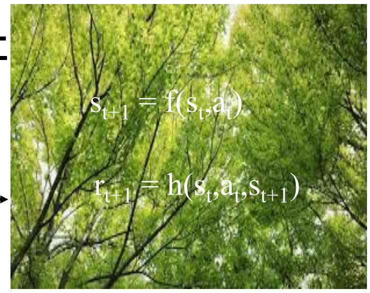
Which is the value of my action (value function)?

s_{t+1}

r_{t+1}

a_t

Environment



a_t, dipende dalla situazione!

Meccanismo di apprendimento nel RL

Inizializzazione: se l'agente non agisce sull'ambiente non succede nulla. Occorre specificare una policy iniziale.

Ciclo dell'agente (le tre fasi sono sequenziali):

1) Implemento una policy ($\pi(s,a)$)
2) Apprendo la sua Value function ($Q^{\pi}(s,a)$) = stima del reward totale
3) Miglioro la policy.

Itero i passi 2 e 3 fino a quando non raggiungo l'ottimo.

Meccanismo di apprendimento nel RL

Ciclo dell'agente (le tre fasi sono sequenziali):

1) Implemento una policy ($\pi(s,a)$)
2) Stimo la Value function ($Q^{\pi}(s,a)$) per tutte le coppie stato-azione
3) Miglioro la policy, $\pi(s,a)$.

Framework analizzati per calcolare $Q^{\pi}(s,a)$:

1. Stocastico completo. **L'agente conosce la statistica della dinamica dell'ambiente e dei reward**, scrive **le equazioni lineari** che legano le value function in stati diversi e **calcola** i valori di $Q^{\pi}(.)$.

2. Stocastico con aggiornamento. **L'agente conosce la statistica della dinamica dell'ambiente e dei reward**. **Procede da uno stato iniziale a quello finale.** Da ogni stato **esplora in parallelo tutti i possibili stati prossimi** e aggiorna i valori delle $Q^{\pi}(.)$.

3. Stocastico con interazione singola e con la scelta di una singola azione. **L'agente NON conosce la statistica della dinamica dell'ambiente e dei reward**. **Procede da uno stato iniziale a quello finale**. Da ogni stato esplora una sola azione e **un SOLO solo stato prossimo**. Aggiorna i valori di $Q^{\pi}(.)$.

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.

Goal: collezionare il maggior numero di lattine.

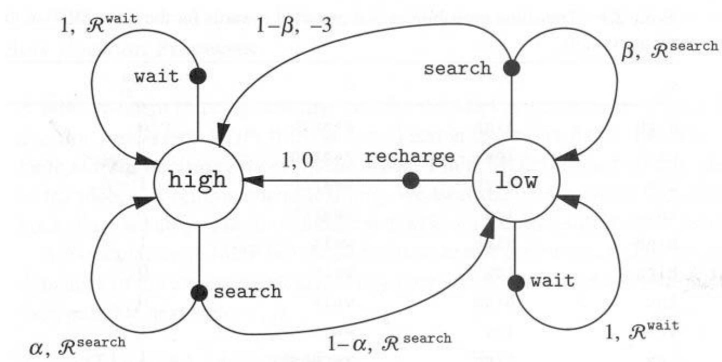
Azioni ammissibili (policy):
a(s = high) = {Search, Wait}
a(s = low) = {Search, Wait, Recharge}

Esempio di calcolo della Value function

Policy deterministica
a(high) = wait
a(low) = search

Value function
Q(high, wait) = ?
Q(low, search) = ?

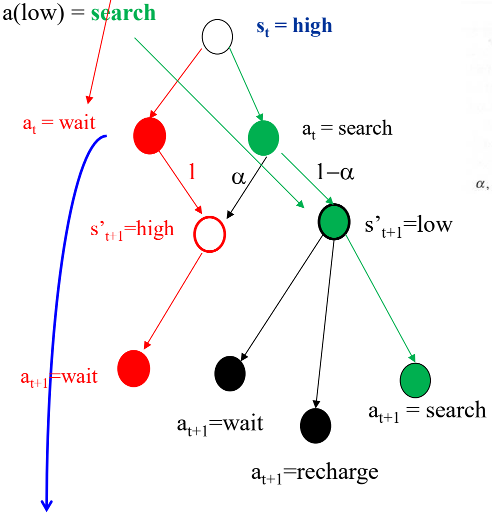
α=Pr($s_{t+1} = High | s_t = High, a_t = Search$) =0.4
β=Pr($s_{t+1} = Low | s_t = Low, a_t = Search$)=0.1,
γ=0.8, R^{search}=3, R^{wait}=1 R^{dead} = -3, R^{auto} = 0

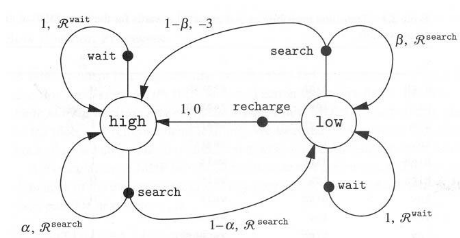


7

Analisi a un passo al tempo t – s_t = high

Policy deterministica
a(high) = **wait**
a(low) = **search**





$$Q^{\pi}(s_t, a_t) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

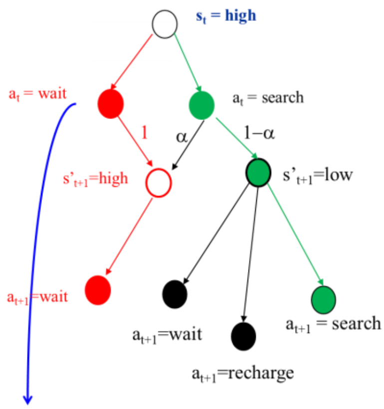
$$Q^{\pi}(s_t, a_t) = E_{\pi}\{R_t | s_t = s, a_t = a\}$$

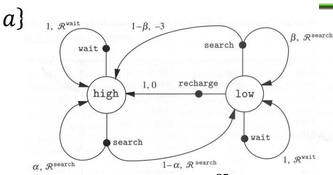
8

Analisi a un passo al tempo t – s_t = high

Policy deterministica
a(high) = wait
a(low) = search

$$Q^\pi(s_t, a_t) = E_\pi\{R_t | s_t = s, a_t = a\}$$





$$Q^\pi(high, wait) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} =$$

$$R^{wait} + \sum_{k=1}^{\infty} \gamma^k r_{t+k+1} =$$

$$R^{wait} + \gamma \sum_{k=1}^{\infty} \gamma^{k-1} r_{t+k+1} =$$

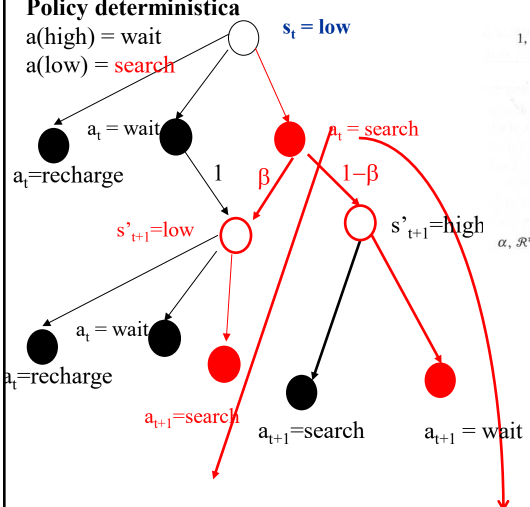
$$R^{wait} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2}$$

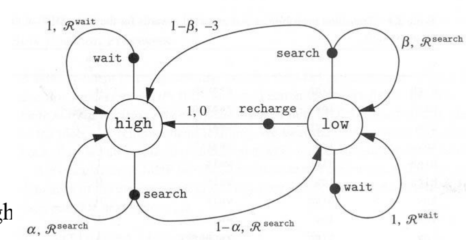
$$Q^\pi(high, wait) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} = R^{wait} + \gamma Q^\pi(high, wait)$$

$$Q^\pi(h,w) = [1 + 0.8\, Q^\pi(h,w)]$$

Analisi a un passo al tempo t – s_t = low

Policy deterministica
a(high) = wait
a(low) = search



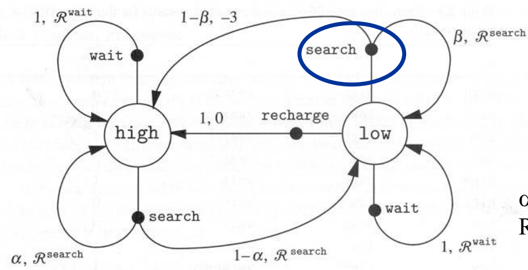


$$Q^\pi(s_t, a_t) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

2 cammini possibili!!

$$Q^\pi(s_t, a_t) = E_\pi\{R_t | s_t = s, a_t = a\}$$

Policy deterministica – s_t = low



α=0.4, β=0.1, γ=0.8,
R^{search}=3, R^{wait}=1 R^{dead} = -3, R^{auto} = 0

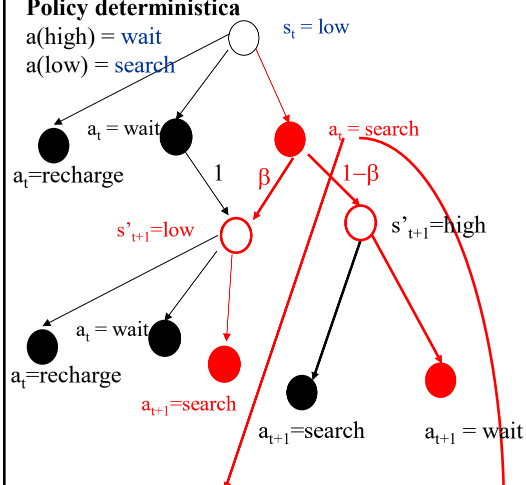
s = High - a = Wait;
s = Low - **a = Search;**

$$Q^{\pi}(\text{low,search}) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} = \beta\,(R^{search} + \gamma Q^{\pi}(low, search) + (1 - \beta)\,(R^{dead} + \gamma Q^{\pi}(high, wait)$$

$$Q^{\pi}(\text{low,search}) = 0.1\text{x}[3+0.8\text{x}Q^{\pi}(\text{low,search})] + 0.9\text{x}[-3+0.8\, Q^{\pi}(\text{high,wait})]$$

Analisi a un passo dal tempo t

Policy deterministica
a(high) = wait
a(low) = search

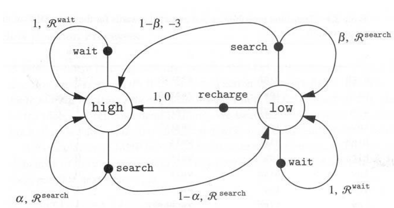


$$Q^{\pi}(low, search) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} =$$

2 cammini possibili per a_t = search !!

1) $R^{search} + \gamma Q^{\pi}(low, search)$

2) $R^{dead} + \gamma Q^{\pi}(high, wait)$



$$Q^{\pi}(low, search) = \beta\left(R^{search} + \gamma Q^{\pi}(low, search)\right) + \quad (1-\beta)\left(R^{dead} + \gamma Q^{\pi}(high, wait)\right)$$

Q(l,s) = 0.1x[3+0.8xQ(l,s)]+0.9x[-3+0.8 Q(h,w)]

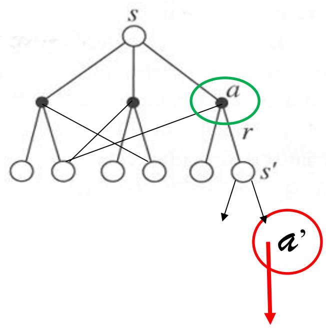
Contiene la probabilità di ricevere un reward γQ(s',a), condizionata a s_{t+1} = s'!

Calcolo ricorsivo della Value function

$$Q^\pi(s_t, a_t) = E_\pi\{R_t | s_t = s, a_t = a\} = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

$$Q^\pi(s_{t+1}, a_{t+1}) = E_\pi\{R_t | s_{t+1} = s', a_{t+1} = a'\} = \sum_{k=0}^{\infty} \gamma^k r_{t+k+2}$$

Relazione tra $Q^\pi(s,a)$ e $Q^\pi(s_{t+1},a_{t+1})$?



Calcolo ricorsivo della Value function

$$Q^\pi(s_t, a_t) = E_\pi\{R_t | s_t = s, a_t = a\} = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

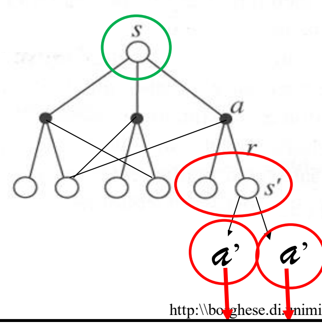
Isolo il reward ad un passo (k = 0), nella serie dei reward.

$$Q^\pi(s_t, a_t) = E_\pi\{\gamma^0 r_{t+1} + \sum_{k=1}^{\infty} \gamma^k r_{t+k+1} \,| s_t = s, a_t = a\} \Rightarrow$$

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

Io termine
(a un passo)

IIo termine
(passi futuri)

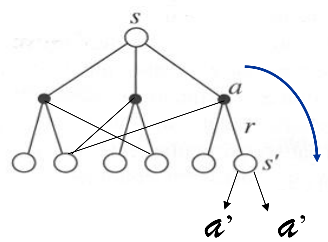


$Q^{\pi}(s_t, a_t)$: primo termine

$$P_{s \to s'|a} \triangleq \Pr(s_{t+1} = s' | s_t = s, a_t = a)$$

$$E_{\pi}\{r_{t+1} | s_{t+1} = s', s_t = s, a_t = a\} = \sum_{s'} P_{s \to s'|a} R_{s,s',a}$$

Per ogni coppia stato-azione devo valutare:
- Più stati prossimi
- Reward stocastici nella transizione ad un passo



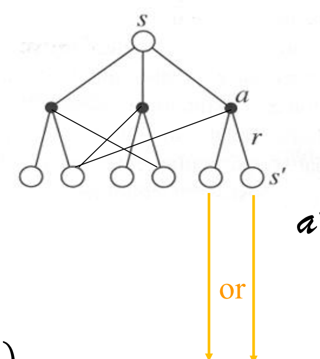
Visione Statistica: Probabilità di ottenere il reward: condizionata all'arrivare nello stato s':

$$R_{s \to s'|a_j}$$

$Q^{\pi}(s_t, a_t)$: secondo termine

$$E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^{k+1} r_{t+k+2} | s_t = s, a_t = a\right\}$$

$$P_{s \to s'|a} \triangleq \Pr(s_{t+1} = s' | s_t = s, a_t = a)$$



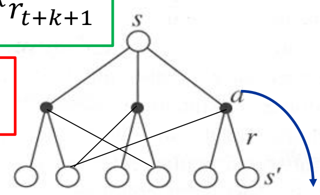
$$E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^{k+1} r_{t+k+2} | s_t = s, a_t = a\right\}$$

$$= \gamma \sum_{s'} P_{s \to s'|a} E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^{k} r_{t+k+2} | s_{t+1} = s'\right\}$$

Sommo su tutti gli s'

Putting all together

$$\boxed{Q^\pi(s_t,a_t) = E_\pi\{R_t | s_t = s, a_t = a\} = \textstyle\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}}$$



$$\boxed{Q^\pi(s_{t+1},a_{t+1}) = E_\pi\{R_t | s_{t+1} = s', a_{t+1} = a'\}}$$

$$\textstyle\sum_{s'} \mathrm{P}_{s\to s'|a} R_{s,s',a} +$$
$$\gamma \textstyle\sum_{s'} \mathrm{P}_{s\to s'|a}\, E_\pi\{\textstyle\sum_{k=0}^{\infty}\gamma^k r_{t+k+2} \,| s_{t+1} = s'\} =$$

Not yet there

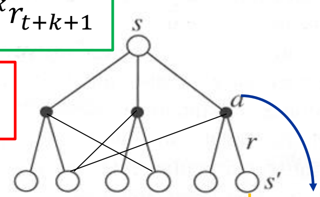
$$\textstyle\sum_{s'} \mathrm{P}_{s\to s'|a}\left\{R_{s,s',a} + \gamma E_\pi\{\textstyle\sum_{k=0}^{\infty}\gamma^k r_{t+k+2} \,| s_{t+1} = s'\}\right\}$$

Io termine
(a un passo)

IIo termine
(passi futuri)

Formulazione ricorsiva – policy deterministica

$$\boxed{Q^\pi(s_t,a_t) = E_\pi\{R_t | s_t = s, a_t = a\} = \textstyle\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}}$$



$$\boxed{Q^\pi(s_{t+1},a_{t+1}) = E_\pi\{R_t | s_{t+1} = s', a_{t+1} = a'\}}$$

$$\textstyle\sum_{s'} \mathrm{P}_{s\to s'|a} R_{s,s',a} +$$
$$= \gamma \textstyle\sum_{s'} \mathrm{P}_{s\to s'|a}\, E_\pi\, \{\textstyle\sum_{k=0}^{\infty}\gamma^k r_{t+k+2} \,| s_{t+1} = s'\} =$$

a'

$$\textstyle\sum_{s'} \mathrm{P}_{s\to s'|a}\{R_{s,s',a} + \gamma\, \{E_\pi\{\textstyle\sum_{k=0}^{\infty}\gamma^k r_{t+k+2} \,| s_{t+1} = s', a_{t+1} = a'\}\}\}$$

$$\boxed{Q^\pi(s_t,a_t) = \sum_{s'} \mathrm{P}_{s\to s'|a}\{R_{s,s',a} + \gamma Q^\pi(s',a')\}}$$

Io termine
(a un passo)

IIo termine
(passi futuri, per ogni azione a_{t+1})

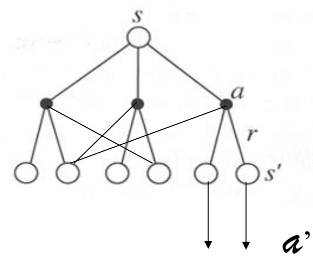
Osservazioni

$$Q^{\pi}(s_t, a_t) = \sum_{s'} \mathrm{P}_{s \to s'|a}\{R_{s,s',a} + \gamma Q^{\pi}(s', a')\}$$

Devo considerare i reward a un passo che portano da s a tutti gli stati prossimi s' che possono venire visitati.

A partire da ogni s', devo considerare il reward a lungo termine che si può accumulare nell'interazione con l'ambiente, $Q^{\pi}(s', a')$.

Abbiamo considerate una policy deterministica: a = g(s).

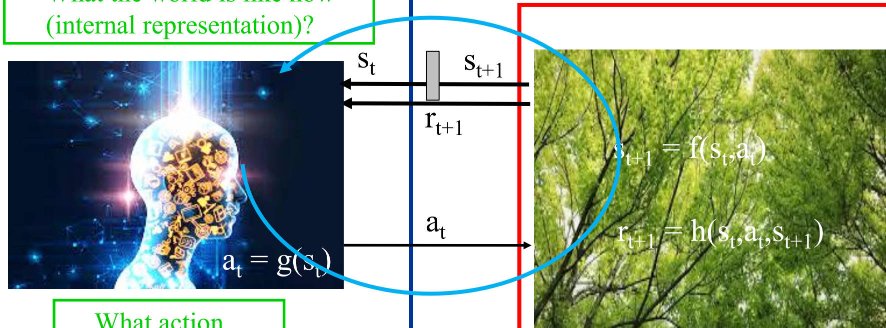


Un ciclo di interazione

Agent

What the world is like now (internal representation)?

Environment



What action should I choose now? (policy)

Which is the value of my action (value function)?

Dobbiamo completare un ciclo con la scelta dell'azione!

Sommario

Le equazioni di Bellman per policy deterministica

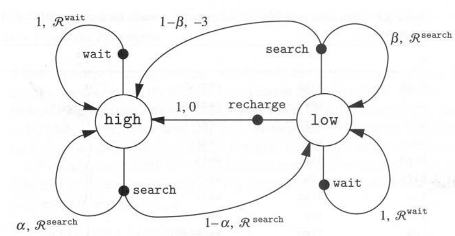
Le equazioni Bellman per policy stocastica

Iterative policy evaluation

Miglioramento della policy

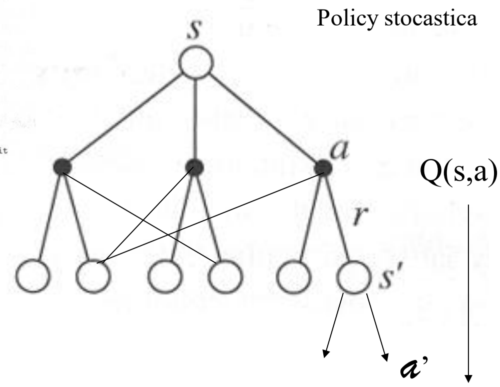
Valutazione policy stocastica

Nel valutare Q(s,a) dobbiamo valutare tutti i cammini che partono da ogni s'.

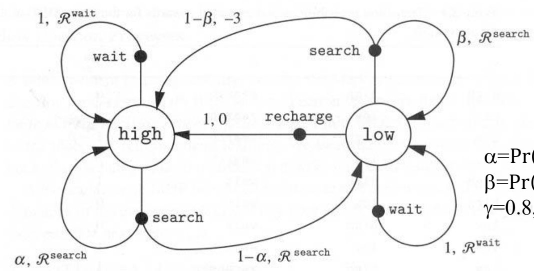


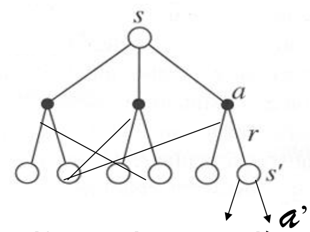
π(s,a) stocastica

Policy stocastica



Policy stocastica





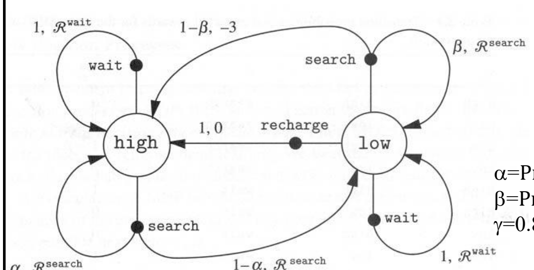
α=Pr($s_{t+1} = High|s_t = High, a_t = Search$) =0.4
β=Pr($s_{t+1} = Low|s_t = Low, a_t = Search$)=0.1,
γ=0.8, R^{search}=3, R^{wait}=1 R^{dead} = -3, R^{auto} = 0

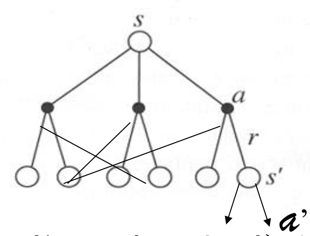
Deterministica (1 azione scelta in s'):
Q(high,wait) = 1 x {R^{wait} + γ [**Q(high,wait)**]}

Stocastica (più azioni scelte in s'):
Q(high,wait) = 1 x {R^{wait} + γ [(Pr(a'=search|high)Q(high,search)+ (Pr(a'=wait|high) Q(high,wait)]}

Q(high,wait) = 1x {1+0.8[Pr(a'=search|high) **Q(high,search)** + Pr(a'=wait|high) **Q(high,wait)**]}

Policy stocastica





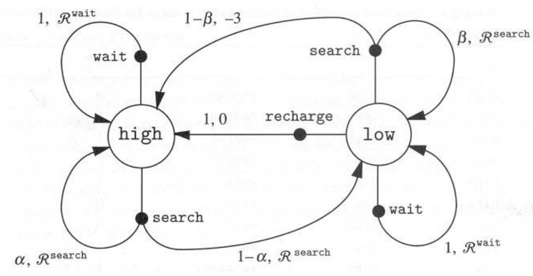
α=Pr($s_{t+1} = High|s_t = High, a_t = Search$) =0.4
β=Pr($s_{t+1} = Low|s_t = Low, a_t = Search$)=0.1,
γ=0.8, R^{search}=3, R^{wait}=1 R^{dead} = -3, R^{auto} = 0

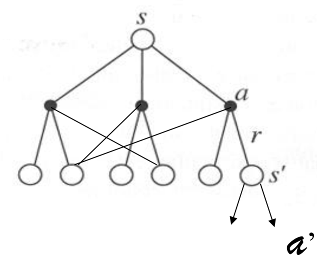
Deterministica (1 azione scelta in s'):
Q(high,search) = Pr($s_{t+1} = High|s_t = High, a_t = Search$) x {$R^{search}$+γ **Q(high,search)**} +
Pr($s_{t+1} = Low|s_t = High, a_t = Search$)) x {$R^{search}$+γ **Q(low,search)**}

Stocastica (più azioni scelte in s'):
Q(high,search) = Pr($s_{t+1} = High|s_t = High, a_t = Search$) x
{R^{search}+γ [(Pr(a'=search|high) **Q(high,search)** + Pr(a'=wait|high) **Q(high,wait)**]} +
Pr($s_{t+1} = Low|s_t = High, a_t = Search$) x
{R^{search}+γ [Pr(a'=search|low)**Q(low,search)**+Pr(a'=wait|low)**Q(low,wait)**+Pr(a'=recharge|low)**Q(low,rech)]}**

Q(high,search) = 0.4x{3+0.8[(Pr(a'=search|high)**Q(high,search)**+Pr(a'=wait|high) **Q(high,wait)**]} +
0.6x{3+0.8[Pr(a'=search|low)**Q(low,search)**+Pr(a'=wait|low)**Q(low,wait)**
+Pr(a'=recharge|low)**Q(low,rech)]}**

Policy stocastica





α=0.4, β=0.1, γ=0.8,
R^{search}=3, R^{wait}=1 R^{dead} = -3, R^{auto} = 0

Q(low,wait) = 1 x {R^{wait}+γ [Pr(a'=search) **Q(low,search)**+ Pr(a'=wait) **Q(low,wait)** + Pr(a'=recharge) **Q(low,reharge)**]}

Q(low,search) = β x {R^{search}+γ [(Pr(a'=search) **Q(high,search)**+ Pr(a'=wait) **Q(high,wait)+** Pr(a'=recharge) **Q(low,reharge)**]} +
(1-β) x {R^{dead}+γ [Pr(a'=search) **Q(high,search)**+ Pr(a'=wait) **Q(high,wait)]**}

Q(low,recharge) = 1 x {R^{auto}+γ [(Pr(a'=search) **Q(high,search)**+ Pr(a'=wait) **Q(high,wait)**]}

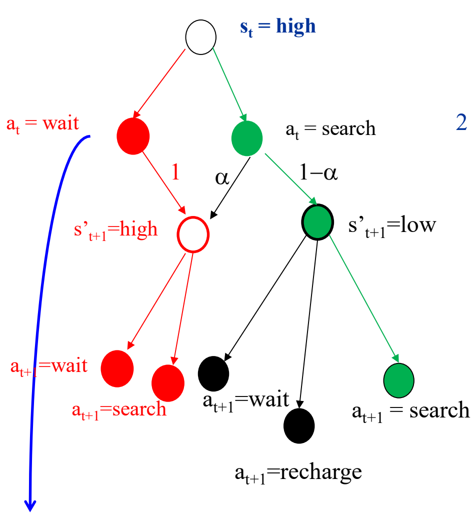
5 equazioni in 5 incognite

Analisi a un passo al tempo t

Policy stocastica

$$Q^{\pi}(s_t, a_t) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

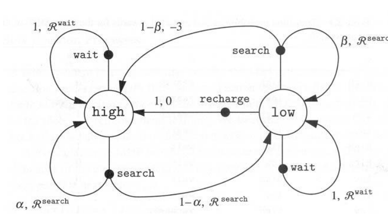
$$Q^{\pi}(s_t, a_t) = E_{\pi}\{R_t | s_t = s, a_t = a\}$$



2 cammini possibili quando scelgo wait in s_t !!

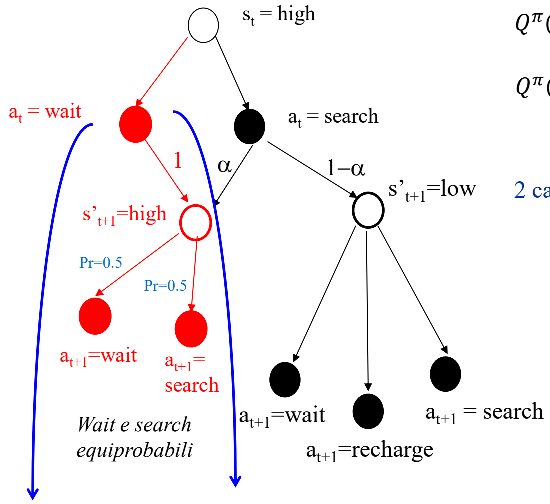
1) $R^{wait} + \gamma Q^{\pi}(high, wait)$

2) $R^{wait} + \gamma Q^{\pi}(high, search)$



Analisi a un passo al tempo t

Policy stocastica (uniforme)



$$Q^{\pi}(s_t, a_t) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

$$Q^{\pi}(s_t, a_t) = E_{\pi}\{R_t | s_t = s, a_t = a\}$$

2 cammini possibili!!

1) $R^{wait} + \gamma Q^{\pi}(high, wait)$

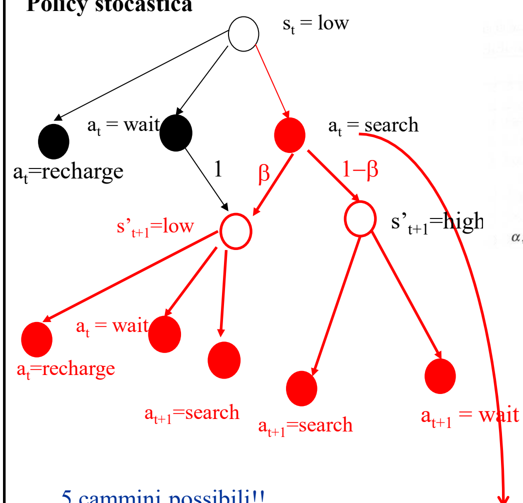
2) $R^{wait} + \gamma Q^{\pi}(high, search)$

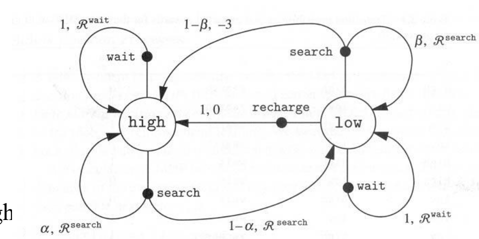
$$Q^{\pi}(high, wait) = R^{wait} + 0.5\,\gamma\, Q^{\pi}(high, wait) + 0.5\gamma Q^{\pi}(high, search)$$

27

Analisi a un passo al tempo t

Policy stocastica





$$Q^{\pi}(s_t, a_t) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

$$Q^{\pi}(s_t, a_t) = E_{\pi}\{R_t | s_t = s, a_t = a\}$$

5 cammini possibili!!

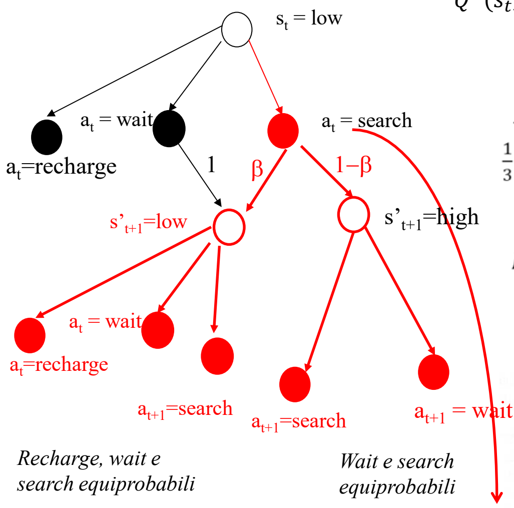
$$Q^{\pi}(low, search) = E_{\pi}\{R_t | s_t = low, a_t = search\}$$

28

Analisi a un passo al tempo t

Policy stocastica (equiprobabile)

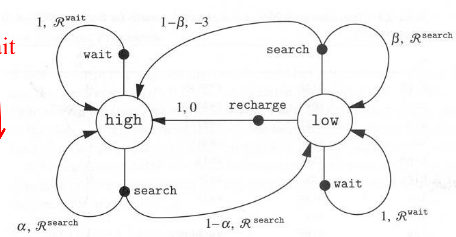
$Q^\pi(s_t, a_t) = E_\pi\{R_t | s_t = s, a_t = a\}$



5 cammini possibili!!

$A)\ R^{search} + \gamma[\frac{1}{3}Q^\pi(low, search) + \frac{1}{3}Q^\pi(low, wait) + \frac{1}{3}Q^\pi(low, recharge)]$

$B)\ R^{dead} + \gamma[\frac{1}{2}Q^\pi(high, search) + \frac{1}{2}Q^\pi(high, wait)]$

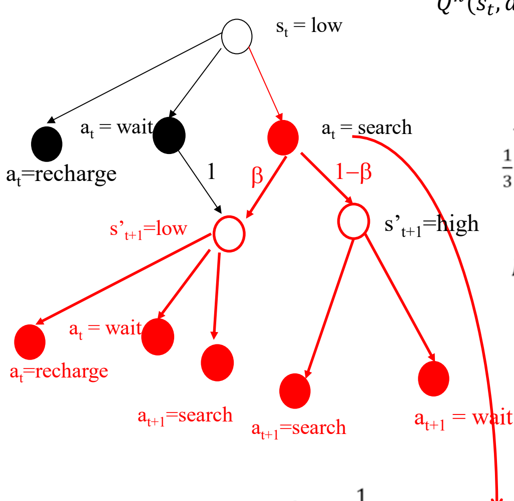


29

Analisi a un passo al tempo t

Policy stocastica (equiprobabile)

$Q^\pi(s_t, a_t) = E_\pi\{R_t | s_t = s, a_t = a\}$



5 cammini possibili da a_t = search!!

$A)\ R^{search} + \gamma[\frac{1}{3}Q^\pi(low, search) + \frac{1}{3}Q^\pi(low, wait) + \frac{1}{3}Q^\pi(low, recharge)]$

$B)\ R^{dead} + \gamma[\frac{1}{2}Q^\pi(high, search) + \frac{1}{2}Q^\pi(high, wait)]$

$Q^\pi(low, search) = \beta[R^{search} + \gamma(\frac{1}{3}Q^\pi(low, search) + \frac{1}{3}Q^\pi(low, wait) + \frac{1}{3}Q^\pi(low, recharge))] +$
$(1-\beta)\ [R^{dead} + \gamma(\frac{1}{2}Q^\pi(high, search) + \frac{1}{2}Q^\pi(high, wait))]$

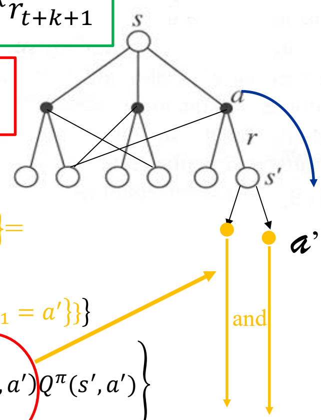
5 equazioni in 5 incognite

30

Formulazione ricorsiva – policy stocastica

$$\boxed{Q^\pi(s_t, a_t) = E_\pi\{R_t | s_t = s, a_t = a\} = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}}$$

$$\boxed{Q^\pi(s_{t+1}, a_{t+1}) = E_\pi\{R_t | s_{t+1} = s', a_{t+1} = a'\}}$$



$\sum_{s'} P_{s\to s'|a} R_{s,s',a} +$

$= \gamma \sum_{s'} P_{s\to s'|a} E_\pi \{\sum_{k=0}^{\infty} \gamma^k r_{t+k+2} | s_{t+1} = s'\} =$

$\sum_{s'} P_{s\to s'|a} \{R_{s,s',a} + \gamma P_{a'|s'} \{E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+2} | s_{t+1} = s', a_{t+1} = a'\}\}\}$

$$Q^\pi(s_t, a_t) = \sum_{s'} P_{s\to s'|a} \left\{ R_{s,s',a} + \gamma \sum_{a'} \pi(s', a') Q^\pi(s', a') \right\}$$

Io termine
(a un passo)

IIo termine
(passi futuri, per ogni azione a_{t+1})

Sommario

Le equazioni di Bellman per policy deterministica

Le equazioni Bellman per policy stocastica

Iterative policy evaluation

Miglioramento della policy

Fondamenti del metodo

• Supponiamo di essere all'istante t. In questo istante t, siamo in s_t e da s_t si può passare a un certo insieme di stati: $\{s'_{t+1}\}$.

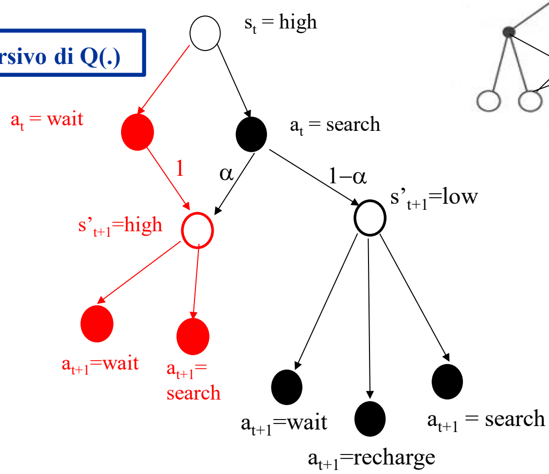
•Analizziamo un solo passo: cosa succede nella transizione da t a t+1.

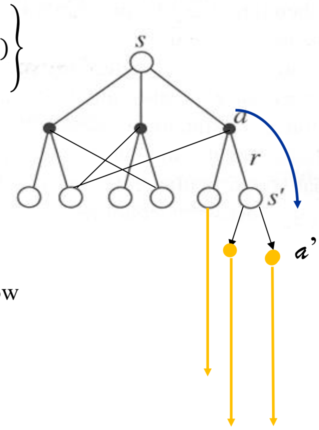
• Migliorare la stima della nostra Value Function ad ogni iterazione.

Equazioni di Bellman

$$Q^\pi(s_t,a_t) = \sum_{s'} P_{s\to s'|a}\left\{R_{s\to s'|a} + \gamma\sum_{a'}\pi(s',a')Q^\pi(s',a')\right\}$$

Calcolo ricorsivo di Q(.)

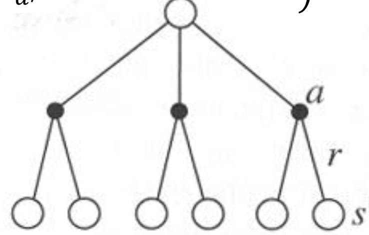




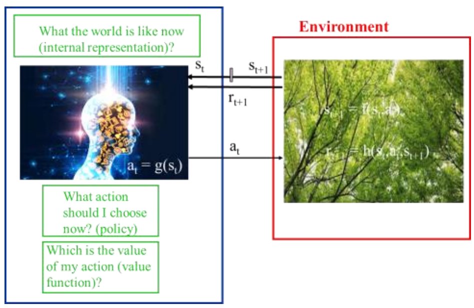
$$Q^\pi(high, wait) = R^{wait} + 0.5\,\gamma\, Q^\pi(high, wait) + 0.5\gamma Q^\pi(high, search)$$

Osservazioni

$$Q^{\pi}(s_t, a_t) = \sum_{s'} \mathrm{P}_{s \to s'|a} \left\{ R_{s,s',a} + \gamma \sum_{a'} \pi(s', a') Q^{\pi}(s', a') \right\}$$

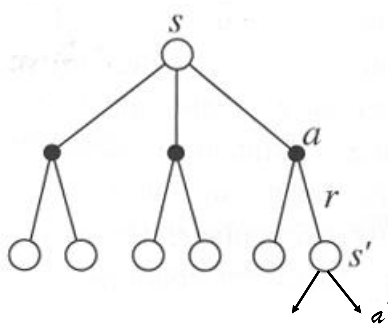


Calcolo ricorsivo di Q(.)



Passo da t a t+1 poi guardo backwards in time

Tecnica full-back



Back-up

π(s,a) fissata

t+1

Conosciamo $Q^{\pi}(s_t,a_t)$ $\forall s_t,a_t$ anche per $\{s'_{t+1}, a'_{t+1}\}$ quindi:

- Analizziamo la transizione da $\{s_t,a_t\}$ -> $\{s'_{t+1},a'_{t+1}\}$
- Calcoliamo un nuovo valore di Q^{π} per {s,a: $Q^{\pi}(s_t, a_t)$ congruente con:

  $Q^{\pi}(s_t, a_t)$ ed r_{t+1}

Full backup se esaminiamo tutti gli s' e a' (cf. DP).

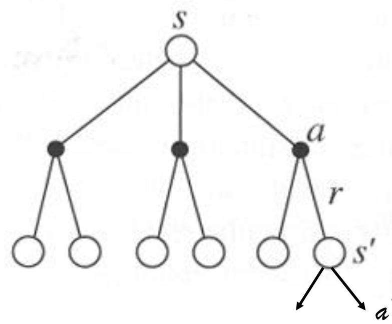
Da {s', a'} mi guardo indietro e aggiorno Q^{π}(s,a)

π fissata

Tecnica full-back

Back-up

π(s,a) fissata



t+1

$$Q^{\pi}(s_t, a_t) = \sum_{s'} P_{s \to s'|a} \left\{ R_{s,s',a} + \gamma \sum_{a'} \pi(s', a') Q^{\pi}(s', a') \right\}$$

Conosciamo $Q^{\pi}(s_t,a_t)$ $\forall s_t,a_t$ anche per $\{s'_{t+1}, a'_{t+1}\}$ quindi:

- Analizziamo la transizione da $\{s_t,a_t\}$ -> $\{s'_{t+1},a'_{t+1}\}$
- Calcoliamo un nuovo valore di Q^{π} per {s,a: $Q^{\pi}(s_t, a_t)$ congruente con:
  $Q^{\pi}(s_t, a_t)$ ed r_{t+1}

$\boxed{\pi \text{ fissata}}$

Full backup se esaminiamo tutti gli s' e a' (cf. DP).
Da {s', a'} mi guardo indietro e aggiorno Q^{π}(s,a).

http:\\borghese.di.unimi.it\

37

Algoritmo per "iterative policy evaluation", versione batch

Partiamo da una politica π(s,a) data, supponiamo deterministica.
Definiamo una soglia di convergenza τ
Inizializziamo Q(s,a) = 0 $\forall$s, $\forall$a, compreso gli stati finali.

```
Repeat
{   Δ = 0;
    for s = 1 : NS                        // ∀s, ≠ TS
    {       a = policy(s);
         {   Temp_Q(s,a) = 0;
             for s_next = 1 : NS                      // for all possible next states
             {      Pr_s_next = NextState(s,a);        // compute the probability that s_t+1 = s'
                    reward = ComputeReward(s,a,s_next);   // Compute average 1 step reward
                    a_next = policy(s_next);              // Next action
                    Temp_Q(s,a) = Pr_s_next * (reward + γQ(s_next,a_next);
    }    }  }                             // Until End of states - End of an episode
    for s=1:NS;
    {      a = policy(s);
           if ( | Temp_Q(s,a) – Q(s,a) | > Δ )
           {   Δ = | Temp_Q(s,a) – Q(s,a) | ;
           }
               Q(s,a) = Temp_Q(s,a);
   }
}  Until (Δ < τ);
```

1 step Forwards
Pass for all states

1 step Backwards
Pass for all states
(full backup)

http:\\borghese.di.unimi.it\

38

Interpretazione dell'update (batch o trial)

$$Q^{\pi}(s_t, a_t) = \sum_{s'} \mathrm{P}_{s \to s'|a} \left\{ R_{s,s',a} + \gamma \sum_{a'} Q^{\pi}(s', a') \right\}$$

Al termine dell'aggiornamento dei $Q^{\pi}(s,a)$ per tutti gli stati, $Q^{\pi}(s,a) = Q^{\pi}_{new}(s,a)$. **Aggiornamento batch.**

In alternativa, utilizzerò in parte già il nuovo valore di $Q^{\pi}(s,a)$ all'interno dell'equazione di aggiornamento. **Aggiornamento per trial.**

Entrambe le modalità di aggiornamento convergono.

Algoritmo per "iterative policy evaluation", versione per trial

Partiamo da una politica $\pi(s,a)$ data, deterministica.
Definiamo una soglia **relativa** di convergenza τ.
Inizializziamo $V(s) = 0$ $\forall$s, compreso gli stati finali.

1 step Forwards
Pass for all states

```
Repeat
{       Δ = 0;
        for s = 1 : NS                     // ∀s, ≠ TS
        {    a = policy(s);
             Value = Q(s,a);
             for s_next = 1 : NS                      // for all next states
             {    Pr_s_next = NextState(s,a);          // compute the probability that s_t+1 = s'
                  reward = ComputeReward(s,a,s_next);   // Compute average 1 step reward
                  a_next = policy(s_next);              // Next action
                  Q(s,a) = Pr_s_next * (reward + γQ(s_next,a_next));
                  Δ = max(Δ, (| Value – Q(s,a) |));
    }    }  }                          // Until end of States – End of an episode

} Until (Δ < τ);
```

Problematiche legate al calcolo di V(s): problema di policy evaluation

3 assunzioni:

1) Conoscenza della dinamica dell'ambiente: P(s->s'| a)
2) Conoscenza della policy (eventualmente stocastica), $\pi(s, a)$
3) Potenza di calcolo sufficiente
4) Proprietà Markoviane dell'ambiente (definizione di uno stato).

Le equazioni contengono dei termini statistici (valori attesi).

Soluzione di un sistema lineare in N incognite (numero di stati).

Come mai posso determinare la Value function per la policy $\pi(.)$, se questa si basa sul reward che riceverò negli istanti futuri?

C'e' poca interazione con l'ambiente e molta simulazione (cf. metodi Montecarlo).

Riassunto

Posso determinare la Value function in modo ricorsivo. Per ogni stato, sarà funzione dell'output dell'ambiente in quell'istante (attraverso la funzione stato prossimo ed il reward istantaneo) e della policy scelta in quell'istante e dei reward a lungo termine attesi negli stati in cui l'ambiente mi porta.

Per scegliere la policy devo esaminare il reward a lungo termine che mi si prospetta nello stato in cui mi trovo e scegliere l'azione che lo massimizza.

Sommario

Le equazioni di Bellman per policy deterministica

Le equazioni Bellman per policy stocastica

Iterative policy evaluation

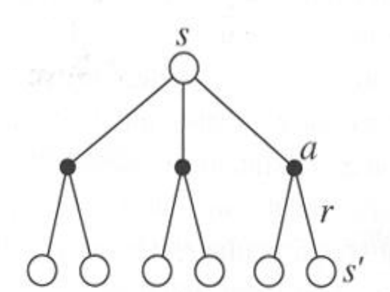
Miglioramento della policy

Meccanismo di apprendimento nel RL

Inizializzazione: se l'agente non agisce sull'ambiente non succede nulla. Occorre specificare una policy iniziale.

Ciclo dell'agente (le tre fasi sono sequenziali):

1) Implemento una policy (π(s,a))
2) Aggiorno la Value function (Q^π(s,a)
3) Aggiorno la policy.



Miglioramento della policy

Tutti gli stati sono valutati in funzione di una policy data.

Condizioni di funzionamento dell'agente:
•Policy **deterministica**: $a = \pi(s)$.
•Ambiente **stocastico**.

Cosa succede se cambiamo la policy per un certo stato s_m? $a_{new} \neq \pi(s_m)$.
Cosa viene influenzato?

Scelgo a_{new} in s_m, visiterò una certa sequenza di stati, per questi stati seguirò la policy precedente per $s \neq s_m$. Cosa viene influenzato?

Come faccio a valutare se miglioro la policy o no?

Dimostrazione del teorema del miglioramento della policy

Analizziamo la seguente condizione:

$\pi' = \pi$ $\forall s$ tranne che per s_m per il quale si applica l'azione:
$$a_{new} = \pi'(s_m)$$

Risulta che il reward a lungo termine è maggiore per $a_{new} = \pi'(s)$.

$V^{\pi'}(s) = Q^{\pi'}(s, a_{new} = \pi'(s)) >= Q^{\pi}(s, a = \pi(s)) = V^{\pi}(s)$

Tesi: π' è meglio di π. Cioè: $V^{\pi'}(s) >= V^{\pi}(s)$ $\forall s$ (ed in particolare per gli altri stati s)

Effetto del cambiamento della policy

Cambia, a, cambiano i possibili stati successivi ad s_m, $\{s_{t+k}\}$, ed il reward a lungo termine ($V^\pi(s_{t+1}) = Q^\pi(s_{t+1}, a_{t+1})$) per policy deterministica):

$$Q^\pi(s_m, a_{new}) = E_\pi\{r_{t+1} + \gamma V^\pi(s_{t+1}) \mid s_t = s_m, a_t = a_{new} \neq \pi(s_m)\} =$$

$$\sum_{s'} P^{a_{new}}_{s_m -> s'}\left[R^{a_{new}}_{s_m -> s'} + \gamma V^\pi(s')\right]$$

?

$$Q^\pi(s_m, a_{new}) >=< Q^\pi(s_m, a = \pi(s_m)) \forall s, a?$$

Se il reward fosse migliore con a_{new}, sceglierò sempre a_{new} in s_m.

Il reward a lungo termine può essere maggiore (minore) solamente se aumenta (diminuisce) il reward totale "visto" a un passo (reward del passo + reward successivo).

Enunciato del teorema del miglioramento della policy

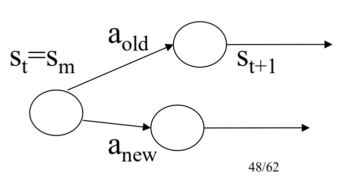
$$Q^\pi(s, a) = \sum_k P_{s \to s_k | a}\left[R_{s \to s_k | a} + \gamma V^\pi(s_k)\right]$$

Ipotesi: π and π' are deterministic policies $Q^\pi(s_m, \pi'(s_m)) \geq V^\pi(s_m)$

$$Q^\pi(s, a_{new} = \pi'(s_m)) = \sum_k P_{s_m \to s_k | a_{new}}\left[R_{s_m \to s_k | a_{new}} + \gamma V^\pi(s_k)\right]$$

Tesi: π' è meglio di π. Cioè: $V^{\pi'}(s,a(s)) >= V^\pi(s,a(s))$ $\forall s$.

$Q^{\pi'}(s,a_{new}) >= Q^\pi(s,a_{old})$



Dimostrazione del teorema del miglioramento della policy

Hp: $Q^\pi(s,\pi'(s)) >= V^\pi(s)$ $\forall s$ $\pi'(s,a)$ è migliore per almeno uno stato

$V^\pi(s) \leq \quad Q^\pi(s,\pi'(s))$

$$= E_{\pi'}\{r_{t+1} + \gamma V^\pi(s_{t+1}) | s_t = s\}$$

$$<= E_{\pi'}\{r_{t+1} + \gamma Q^\pi(s_{t+1}, \pi'(s_{t+1})) | s_t = s\}$$

$$<= E_{\pi'}\{r_{t+1} + \gamma E_{\pi'}(r_{t+2} + \gamma V^\pi(s_{t+2})) | s_t = s\}$$

$$= E_{\pi'}\{r_{t+1} + \gamma r_{t+2} + \gamma^2 V^\pi(s_{t+2})) | s_t = s\}$$

Sostituisco ancora $Q^{\pi *}(.)$

$$<= E_{\pi'}\{r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + ... | s_t = s\}$$

Th: $V^{\pi'}(s) >= V^\pi(s)$

Osservazioni

$s = s_m \qquad Q^\pi(s_m, \pi'(s)) \geq Q^\pi(s_m, \pi(s))$

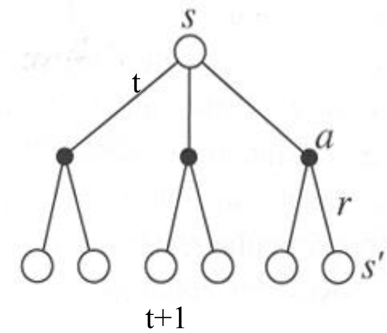
$s \neq s_m \qquad Q^\pi(s,a) = E_{\pi'}\{r_{t+1} + \gamma V^\pi(s_{t+1}) | s_t = s\}$

$$= E_{\pi'}\{r_{t+1} + \gamma Q^\pi(s_{t+1}, \pi(s_{t+1})) | s_t = s\}$$

Se $s_{t+k} = s_m$ miglioro la Q(s,a).

Se nessun $s_{t+k} = s_m$. Non varia la Q(s,a).

Visione grafica del miglioramento



Ogni volta che sono in uno stato, s , scelgo un'azione che migliora il reward a lungo termine ottenuto da quell'istante/stato in poi.

Per gli altri stati, il reward a lungo termine non viene modificato ogni volta che l'albero uscente da s' passa per s.

Ottimizzazione policy

Per ogni stato scelgo le azioni secondo la policy: $\pi(s,a)$.

Posso ordinare la Value function Q(s,a) in ordine decrescente, in funzione delle azioni scelte in s (policy).

Si definisce una policy, π_1, migliore di un'altra, π_2, se e solo se:
$Q^{\pi 1}(s,a(s)) > =Q^{\pi 2}(s,a(s))\ \forall s$.

In particolare si definisce una politica ottima, π^*, se e solo se:
$Q^*(s,a(s)) > =V^{\pi}(s,a(s))\ \forall s$

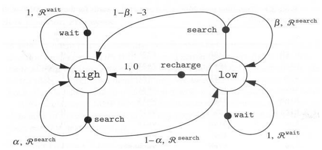
$Q^*(s,a(s)) > =Q^{\pi}(s,a(s))\ \forall [s,a]$

.

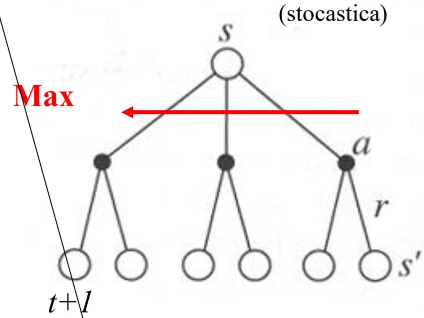
Q(s,a) – Osservazioni

$$Q^{\pi}(s_t, a_t) = \sum_{s'} \mathrm{P}_{s \rightarrow s'|a} \left\{ R_{s,s',a} + \gamma \sum_{a'} \pi(s', a') Q^{\pi}(s', a') \right\}$$

Policy nota (stocastica)



Per ogni stato devo valutare con informazioni esclusivamente racchiuse in 1 passo l'azione migliore a lungo termine



$$a_{new} : \max_{a} Q^{\pi}(s, a)$$

E' supposto noto il funzionamento dell'ambiente (simulazione)

Calcolo ricorsivo della Value function ottima::confronti

$$Q^{\pi}_{k+1}(s_t, a_t) = \sum_{s'} \mathrm{P}_{s \rightarrow s'|a} \left\{ R_{s,s',a} + \gamma\, \pi(s', a') \sum_{a'} Q^{\pi}(s', a') \right\}$$

Q*(s,a) di uno stato-azione, quando viene scelta la policy ottima, deve essere uguale al valore atteso del reward Totale per l'azione migliore per lo stato s.

$$Q^{*}\,(s_t, a_t) = \max_{a'} \sum_{s'} \mathrm{P}_{s \rightarrow s'|a} \left\{ R_{s,s',a} + \gamma\, \pi(s', a') \sum_{a'} Q^{\pi}(s', a') \right\}$$

Politica greedy: scelgo l'azione ottimale.
Ha senso per il robot raccogli-lattine?

Policy iteration

Iterazione tra:

• Calcolo iterativo della Value function (iterative policy evaluation)
• Miglioramento della policy (policy improvement)

$$\pi_0 \rightarrow Q^{\pi 0} \rightarrow \pi_1 \rightarrow Q^{\pi 1} \rightarrow \pi_2 \rightarrow Q^{\pi 2} \rightarrow \ldots\ldots.. \rightarrow \pi^* \rightarrow Q^*$$

Converge velocemente ad una buona politica

Algoritmo

Inizialization

Q(s,a) = 0;
π(s,a) = random (e.g. equiprobabile) o deterministica;

Repeat
  Policy evaluation.
  Policy improvement.
until policy_stable

Algoritmo per "iterative policy evaluation", versione per trial

Partiamo da una politica $\pi(s,a)$ data, deterministica.
Definiamo una soglia **relativa** di convergenza τ.
Inizializziamo $V(s) = 0\ \forall s$, compreso gli stati finali.

1 step Forwards
Pass for all states

```
Repeat
{       Δ = 0;
        for s = 1 : NS                    // ∀s, ≠ TS
        {   a = policy(s);
            Value = Q(s,a);
            for s_next = 1 : NS                      // for all next states
            {   Pr_s_next = NextState(s,a);          // compute the probability that s_{t+1} = s'
                reward = ComputeReward(s,a,s_next);  // Compute average 1 step reward
                a_next = policy(s_next);             // Next action
                Q(s,a) = Pr_s_next * (reward + γQ(s_next,a_next);
                Δ = max(Δ, (| Value – Q(s,a) |));
    }   }   }                              // Until end of States – End of an episode

} Until (Δ < τ);
```


Policy improvement

policy_stable = true;
for s = 1:NS // in alternativa, scelgo uno stato
a_old = π(s);
a_new = $arg\max_{a'} \sum_{s'} P_{s \to s'|a} \left\{ R_{s,s',a} + \gamma\, \pi(s',a') \sum_{a'} Q^{\pi}(s',a') \right\}$
if (a_new ≠ a_old)
policy_stable = false;
end;

Operazione di max hard o «soft» -> policy ε-greedy, pursuit, ...

Iterative policy evaluation sulla value function V(s)

$$Q^{\pi}_{k+1}(s_t, a_t) = \sum_{s'} \mathrm{P}_{s \to s'|a} \left\{ R_{s,s',a} + \gamma\, \pi(s', a') \sum_{a'} Q^{\pi}_{k}(s'_{t+1}, a'_{t+1}) \right\}$$

Converge al limite a $Q^{\pi}(s,a)$. Come facciamo a troncare?

Value iteration

$$Q^{\pi}_{k+1}(s_t, a_t) = \sum_{s'} \mathrm{P}_{s \to s'|a} \left\{ R_{s,s',a} + \gamma\, \pi(s', a') \sum_{a'} Q^{\pi}_{k}(s'_{t+1}, a'_{t+1}) \right\}$$

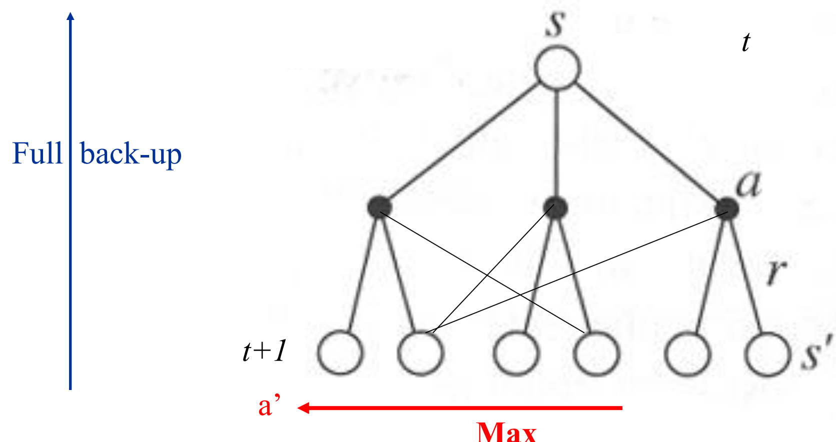
Invece di considerare una policy stocastica, consideriamo l'azione migliore:

$$Q_{k+1}(s_t, a_t) = \max_{a'} \sum_{s'} \mathrm{P}_{s \to s'|a} \left\{ R_{s,s',a} + \gamma\, \pi(s', a') \sum_{a'} Q_{k}\ (s', a') \right\}$$

$\forall s$

Visualizzazione grafica

$$Q_{k+1}(s_t, a_t) = \max_{a'} \sum_{s'} \mathrm{P}_{s \to s'|a} \left\{ R_{s,s',a} + \gamma\, \pi(s', a') \sum_{a'} Q_k\ (s', a') \right\}$$



Sommario

Le equazioni di Bellman per policy deterministica

Le equazioni Bellman per policy stocastica

Iterative policy evaluation

Miglioramento della policy