

# **APRENDIZAGEM POR REFORÇO**

Luís Morgado

2015

# APRENDIZAGEM AUTOMÁTICA

**Aprendizagem = Melhoria de desempenho,  
para uma dada tarefa,  
com a experiência**

- Melhorar o desempenho para uma dada **tarefa  $T$**
- Com base numa medida de **desempenho  $D$**
- Com base na **experiência  $E$**

## EXEMPLOS:

### **Aprender a jogar xadrez**

**$T$ :** Jogar xadrez

**$D$ :** Percentagem de jogos ganhos

**$E$ :** Jogos realizados

### **Aprender a conduzir um veículo**

**$T$ :** Conduzir com base na informação proveniente de câmaras de vídeo

**$D$ :** Distância média percorrida sem erros

**$E$ :** Sequências de imagens e de comandos de condução obtidos através da observação de um condutor humano

# APRENDIZAGEM AUTOMÁTICA

## Aprendizagem $\neq$ Memorização

- Aprendizagem
  - **Generalização**
  - Formação de **abstracções** (modelos)
    - Protótipos
    - Conceitos
    - Padrões comportamentais

# **APRENDIZAGEM AUTOMÁTICA**

- **APRENDIZAGEM CONCEPTUAL**

- **O que é?**

- Conceito

- SUPERVISIONADA

- NÃO SUPERVISIONADA

- **APRENDIZAGEM COMPORTAMENTAL**

- **O que fazer?**

- Comportamento (acção)

- POR REFORÇO

# Aprendizagem por Reforço

---

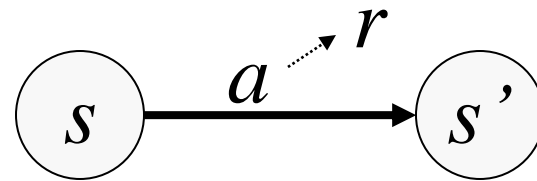
- Aprendizagem a partir da **interacção** com o ambiente

- **Estado**

- **Acção**

- **Reforço**

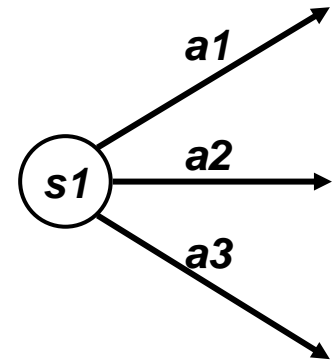
- Ganho / perda



- Aprendizagem de **comportamentos**

- O que fazer

- Relação entre situações e acções



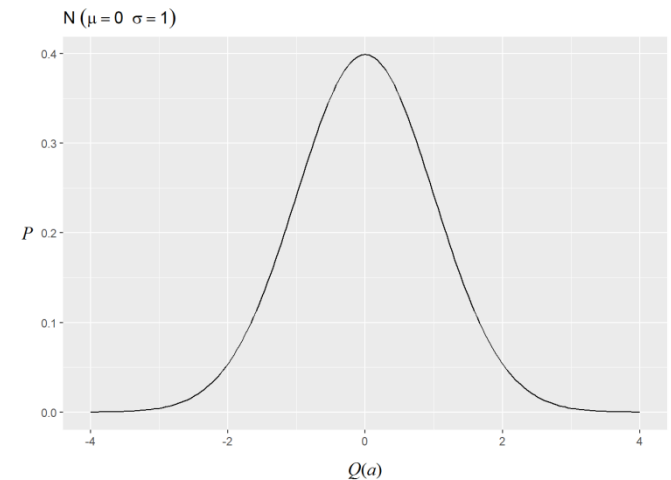
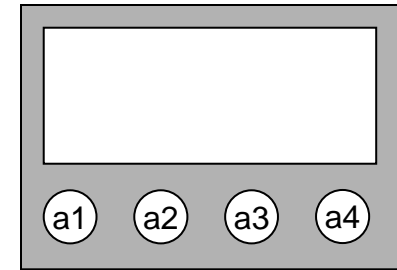
# Aprendizagem de Valor de Acção

- **Exemplo**

- Escolha repetida de diferentes acções
- Por cada acção é obtida uma recompensa
  - De acordo com uma determinada distribuição de probabilidades
- Resultado depende só da acção escolhida

- **Motivação**

- Maximizar a recompensa de longo prazo



# Aprendizagem de Valor de Acção

---

- Como determinar o valor  $Q(a)$  de cada acção?
- Valor médio para uma acção  $a$  após  $n$  tentativas
  - Cada tentativa produz uma recompensa  $r_n$

$$Q_n(a) = \frac{r_1^a + r_2^a + \dots + r_n^a}{n}$$

- De forma incremental

$$Q_n(a) = Q_{n-1}(a) + \frac{1}{n} [r_n^a - Q_{n-1}(a)]$$

# Aprendizagem de Valor de Acção

---

- **Problemas não estacionários?**
  - A distribuição de probabilidades muda com o tempo
- **Estimação por acumulação não linear**
  - Por exemplo, exponencialmente amortecida

$$Q(a)_n = Q(a)_{n-1} + \alpha[r_n^a - Q(a)_{n-1}]$$

$\alpha \in [0,1]$  - Factor de aprendizagem



# Dilema Explorar / Aproveitar (Explore / Exploit)

---

- **Quando é que se aprendeu o suficiente para começar a aplicar o que se aprendeu?**
- **Exploração** (*Exploration*)
  - Escolher uma acção que permita explorar o mundo para melhorar a aprendizagem
- **Aproveitamento** (*Exploitation*)
  - Escolher a acção que leva à melhor recompensa de acordo com a aprendizagem
  - Acção Sôfrega (*Greedy*)

# Estratégias de Selecção de Acção

---

- Estratégia *greedy*

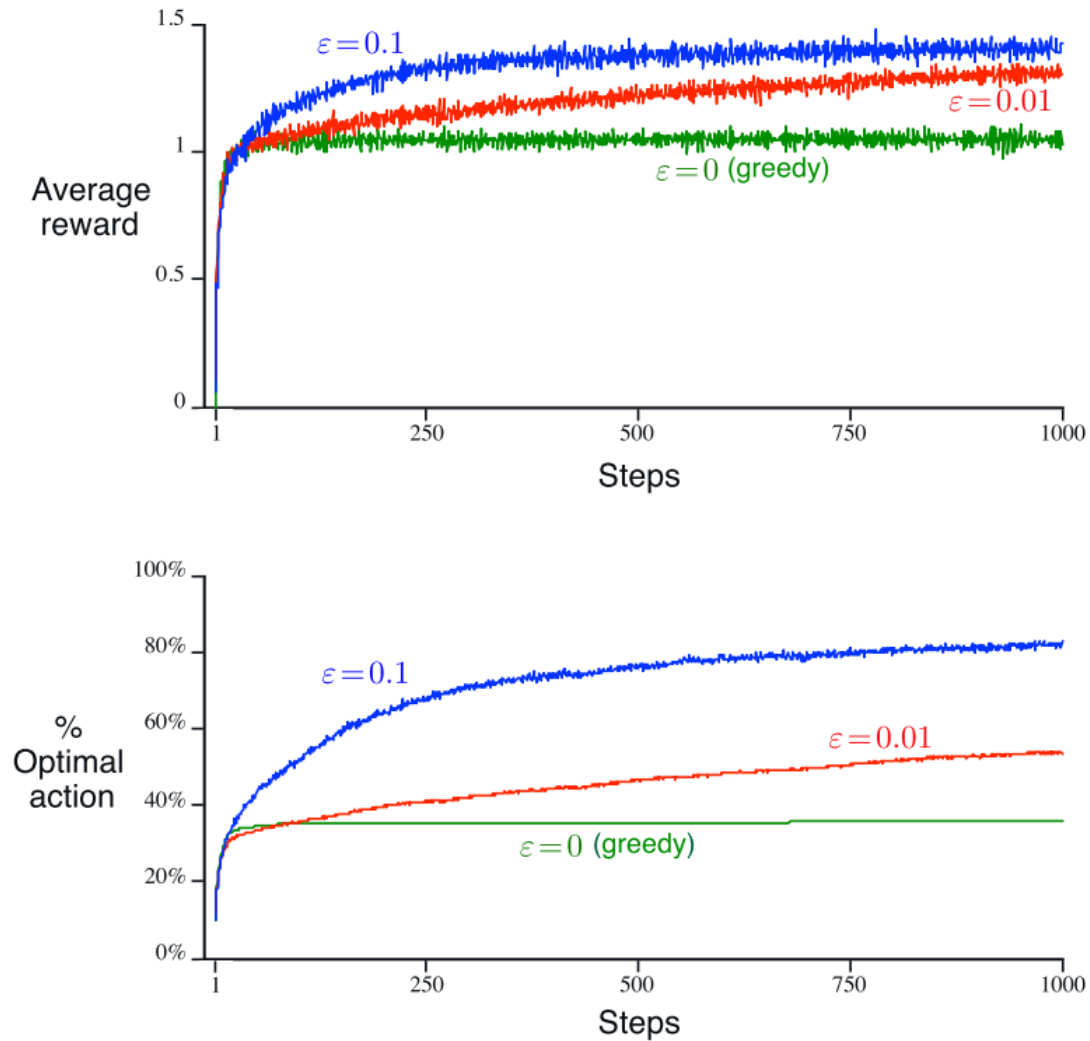
$$a_t = a_t^* = \operatorname{argmax}_a Q_t(a)$$

- Estratégia  $\varepsilon$ -*greedy*

$$a_t = \begin{cases} a_t^* & \text{com probabilidade } 1 - \varepsilon \\ \text{acção aleatória} & \text{com probabilidade } \varepsilon \end{cases}$$

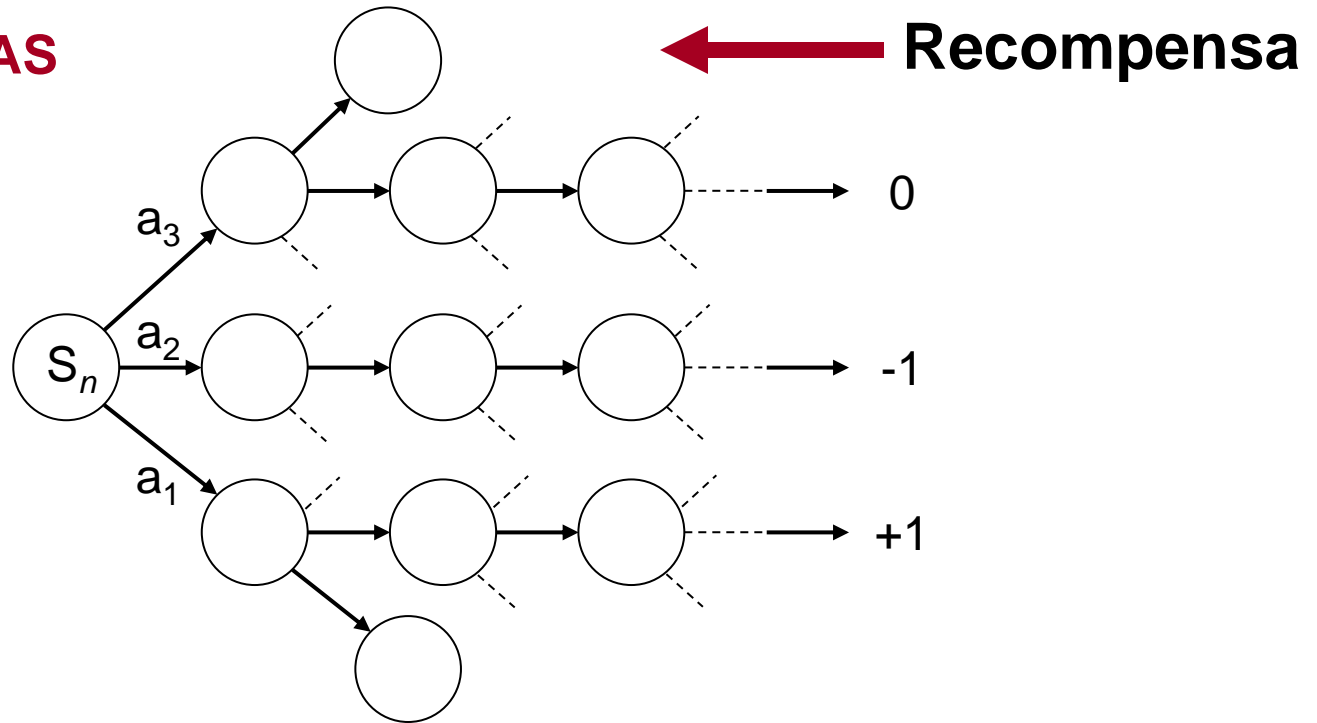
- Balanceamento de Exploração / Aproveitamento

# Exemplo



# Aprendizagem por Reforço

**RECOMPENSAS  
DIFERIDAS**



- Aprendizagem incremental a partir da experiência

$$s \rightarrow a \rightarrow r \rightarrow s' \rightarrow a' \rightarrow \dots$$

# Aprendizagem por Reforço

---

- Aprendizagem associativa

- Estados observados

- $s \in S$

$$s \xrightarrow{\textcolor{red}{q}} a$$

- Acções realizadas

- $a \in A$

$$q = Q(s, a)$$

- Reforços obtidos

- $r \in \mathbb{R}$

- **Valor de num estado realizar uma acção**

- $Q(s, a)$

# Aprendizagem por Diferença Temporal

**Valor de realizar uma acção num determinado estado**

*Valor Estado-Acção:  $Q(s,a)$*

$$Q(s,a) = r + \gamma Q(s',a')$$

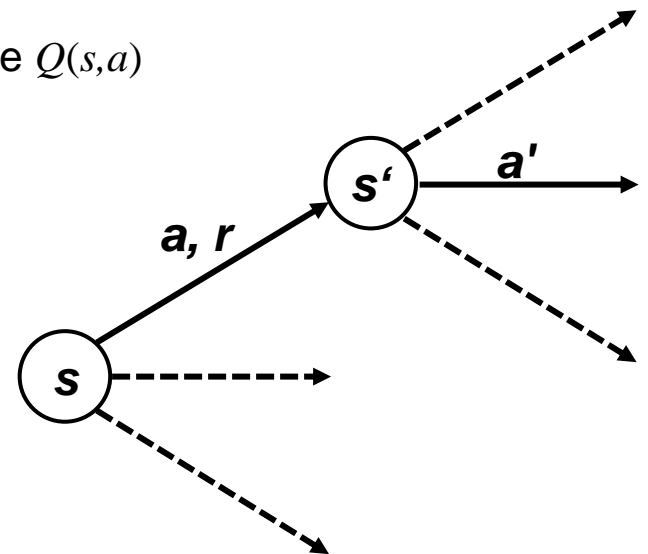
Reforço

↑

Estimativa actual de  $Q(s,a)$

Aprendizagem incremental a partir da experiência

$s \rightarrow a \rightarrow r \rightarrow s' \rightarrow a' \rightarrow \dots$



# Aprendizagem por Diferença Temporal

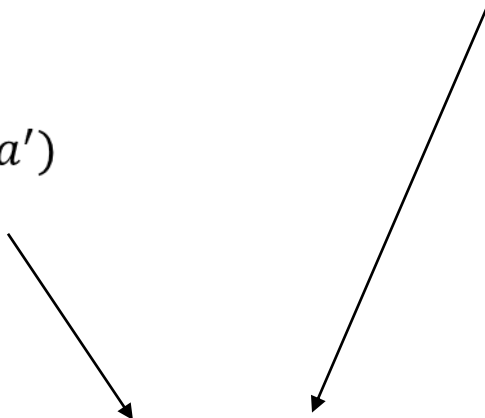
---

$$Q_n(a) = \frac{r_1^a + r_2^a + \dots + r_n^a}{n}$$

$$Q_n(a) = Q_{n-1}(a) + \frac{1}{n}[r_n^a - Q_{n-1}(a)]$$

$$Q_n(a) = Q_{n-1}(a) + \alpha[r_n^a - Q_{n-1}(a)]$$

$$Q'_n(s, a) = r_n + \gamma Q_{n-1}(s', a')$$


$$Q_n(s, a) = Q_{n-1}(s, a) + \alpha[r_n + \gamma Q_{n-1}(s', a') - Q_{n-1}(s, a)]$$

# Aprendizagem por Diferença Temporal

---

## Ambiente não estacionário

Actualização de uma **estimativa** de valor de estado com base na sua mudança (**diferença temporal**) entre instantes sucessivos

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)]$$

Diagram illustrating the components of the temporal difference learning equation:

- Reforço** (Reward) points to  $r$ .
- Estimativa anterior de  $Q(s,a)$**  (Previous estimate of  $Q(s,a)$ ) points to  $Q(s,a)$ .
- Estimativa actual de  $Q(s,a)$**  (Current estimate of  $Q(s,a)$ ) points to  $\gamma Q(s',a')$ .
- Diferença temporal** (Temporal difference) points to the entire term in brackets:  $r + \gamma Q(s',a') - Q(s,a)$ .



# Algoritmo SARSA

---

- Iniciar  $Q(s,a)$
- Repetir (por cada episódio)
  - Iniciar  $s$
  - Escolher  $a$  de acordo com  $s$  com base numa política derivada de  $Q$  (por exemplo  $\epsilon$ -greedy)
  - Repetir (por cada passo)
    - Executar acção  $a$ , observar  $r$  e  $s'$
    - Escolher  $a'$  de acordo com  $s'$  com base numa política derivada de  $Q$  (por exemplo  $\epsilon$ -greedy)
    - Actualizar  $Q$ :  
$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)]$$
    - $s \leftarrow s', a \leftarrow a'$
  - Até  $s$  ser um estado terminal

# Política Comportamental

---

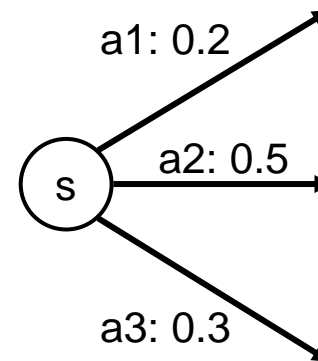
- Forma de representação do comportamento do agente
- Define qual a acção que deve ser realizada em cada estado (estratégia de acção)

- Política **determinista**

$$\pi : S \rightarrow A(s) ; s \in S$$

- Política **não determinista**

$$\pi : S \times A(s) \rightarrow [0,1] ; s \in S$$



# Política Comportamental Óptima

---

- Função valor de estado-acção

$$Q^{\pi}(s, a)$$

- Valor óptimo

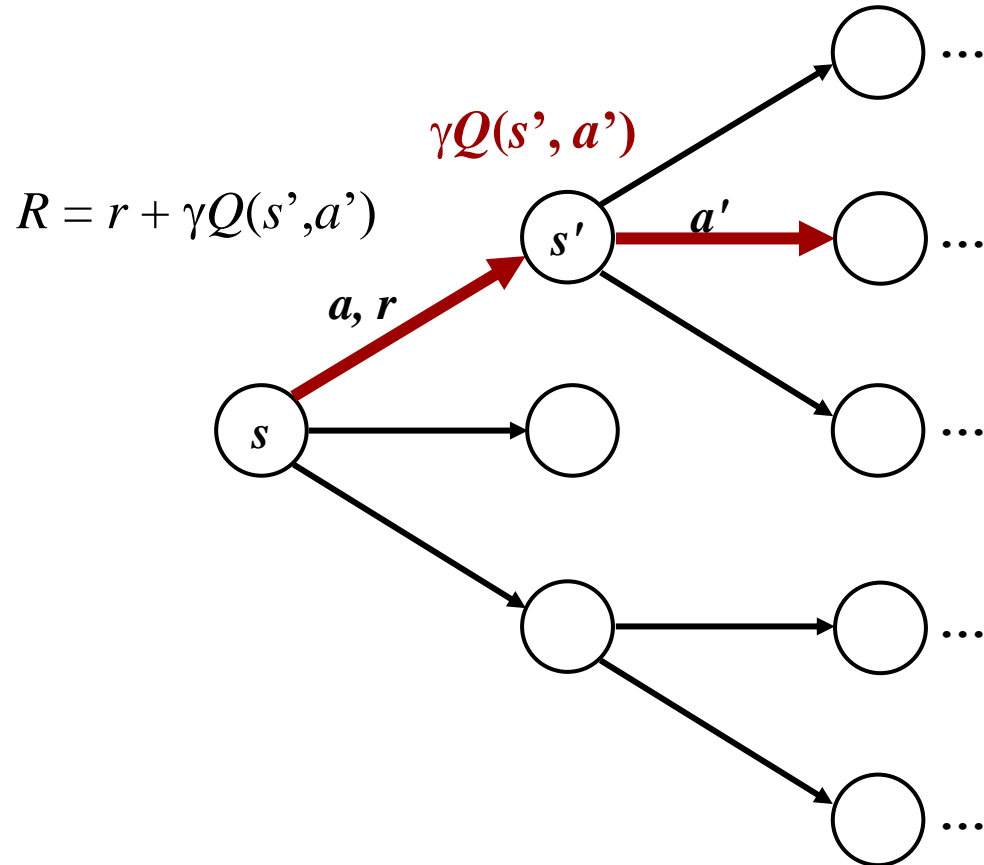
$$Q^*(s, a) = \max_{\pi} Q^{\pi}(s, a)$$

- Política óptima

$$\pi^*(s) = \arg \max_a Q^*(s, a)$$

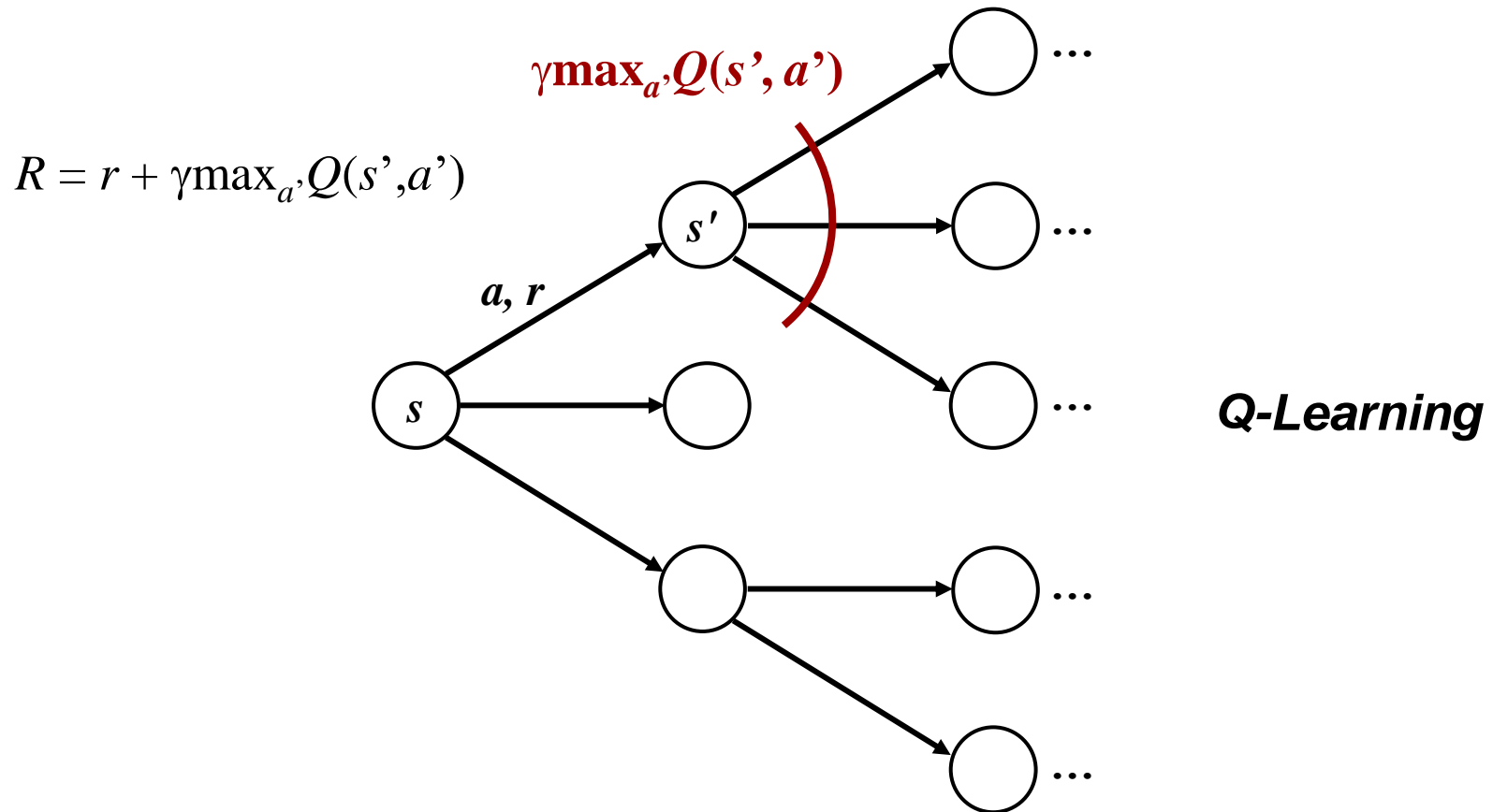
Política “*greedy*” em relação a  $Q^*$

Estimativa de retorno  $R$  (valor estimado de realizar a acção  $a$  num estado  $s$ )  
**considerando a acção  $a'$  seleccionada para realização (SARSA)**



$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)]$$

Estimativa de retorno  $R$  (valor estimado de realizar a acção  $a$  num estado  $s$ )  
considerando a acção  $a'$  correspondente à melhor estimativa  $Q(s', a')$



$$Q(s, a) \leftarrow Q(s, a) + \alpha [r + \gamma \max_a Q(s', a') - Q(s, a)]$$

# Algoritmo *Q-Learning*

---

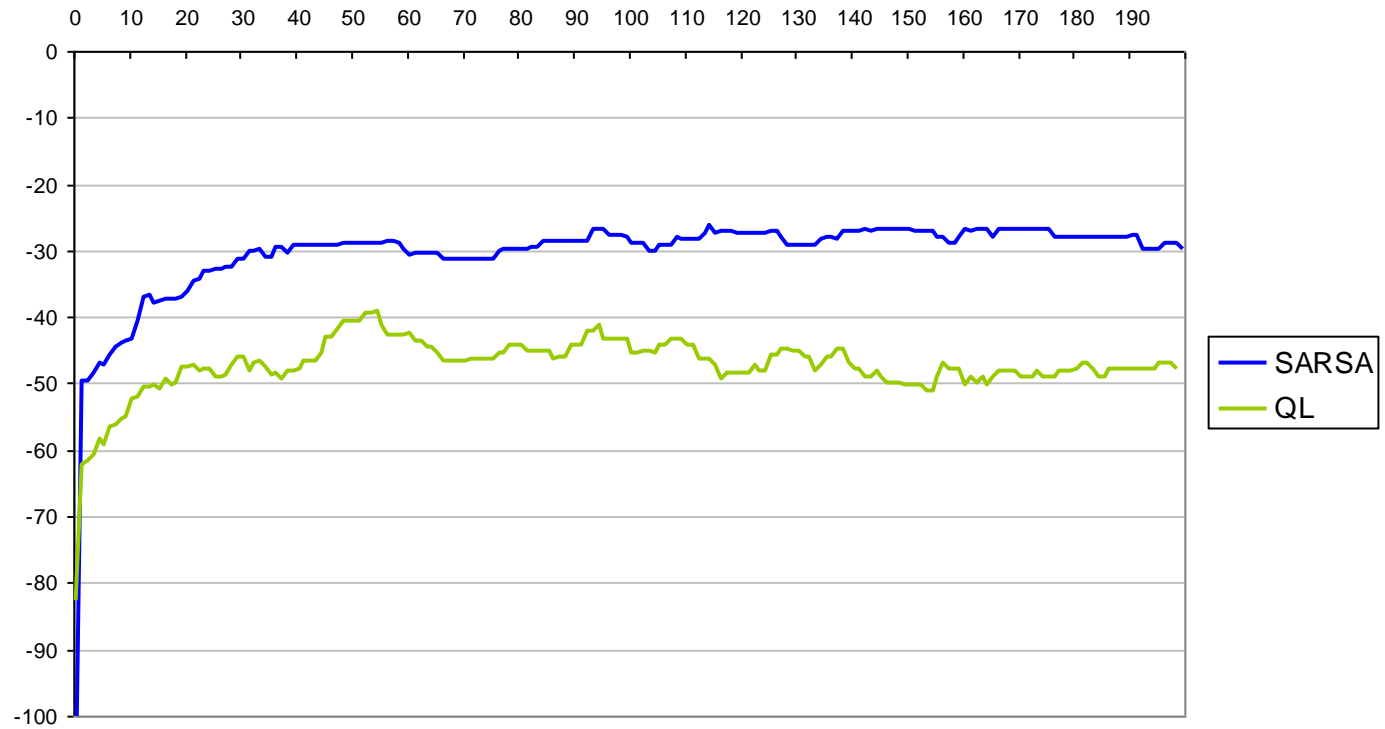
- Iniciar  $\mathbf{Q}(\mathbf{s}, \mathbf{a})$
- Repetir (por cada episódio)
  - Iniciar  $\mathbf{s}$
  - Repetir (por cada passo)
    - Escolher  $\mathbf{a}$  de acordo com  $\mathbf{s}$  com base numa política derivada de  $\mathbf{Q}$  (por exemplo  $\varepsilon$ -greedy)
    - Executar acção  $\mathbf{a}$ , observar  $\mathbf{r}$  e  $\mathbf{s}'$
    - Actualizar  $\mathbf{Q}$ :  
$$\mathbf{Q}(\mathbf{s}, \mathbf{a}) \leftarrow \mathbf{Q}(\mathbf{s}, \mathbf{a}) + \alpha[\mathbf{r} + \gamma \max_{\mathbf{a}'} \mathbf{Q}(\mathbf{s}', \mathbf{a}') - \mathbf{Q}(\mathbf{s}, \mathbf{a})]$$
    - $\mathbf{s} \leftarrow \mathbf{s}'$
  - Até  $\mathbf{s}$  ser um estado terminal

# Processo de Aprendizagem

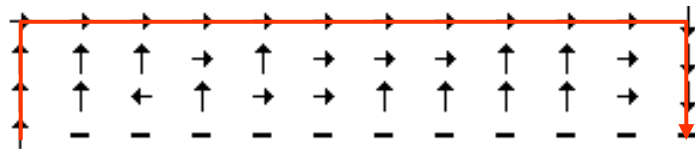
---

- Dois tipos de aprendizagem
  - **Política de selecção de acção única**  
**(On-policy)**
    - Utilização da mesma política de selecção de acção para comportamento e para propagação de valor
    - Exploração de todas as acções (e.g. política  $\epsilon$ -greedy)
  - **Políticas de selecção de acção diferenciadas**  
**(Off-policy)**
    - Utilização da mesma política de selecção de acção para comportamento e para propagação de valor
    - **Optimização da função valor  $Q(s,a)$**

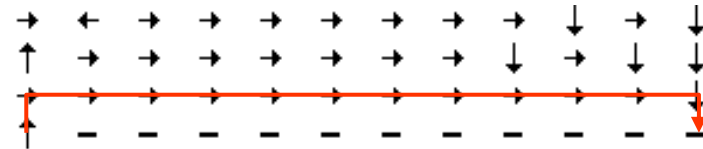
# SARSA vs. Q-Learning



SARSA



QL





# Dilema Explorar / Aproveitar

---

- Para convergir para o valor óptimo
  - Não se pode apenas explorar
  - Não se pode apenas aproveitar
- Estratégia Sôfrega (*Greedy*)
  - Mínimos/máximos locais
- Nunca se pode parar de explorar
  - Convergência assintótica
- Deve-se progressivamente reduzir a exploração
  - GLIE (*Greedy in the Limit of Infinite Exploration*)

# Referências

---

[Russel & Norvig, 2003]

S. Russell, P. Norvig, "Artificial Intelligence: A Modern Approach", 2nd Edition, Prentice Hall, 2003

[Russel & Norvig, 2020]

S. Russell, P. Norvig, "Artificial Intelligence: A Modern Approach", 4th Edition, Pearson, 2020

[Sutton & Barto, 1998]

R. Sutton, A. Barto, "Reinforcement Learning: An Introduction", MIT Press, 1998

[Fox *et al.*, 1994]

G. Fox, R. Williams, P. Messina, "Parallel Computing Works", Morgan Kaufmann, 1994

[Poole & Mackworth, 2010]

D. Poole, A. Mackworth, Artificial Intelligence: Foundations of Computational Agents, Cambridge University Press, 2010

[Scamell-Katz, 2009]

S. Scamell-Katz, "Breaking the Habit", Retail & Shopper, 2009

[Chris Barnard, 2003]

C. Barnard, "Animal Behaviour: Mechanism, Development, Ecology and Evolution", Prentice Hall, 2003