Aprendizado por Reforço

Métodos de Monte Carlo (parte 3)

Recapitulação das aulas passadas

Algorithm 5.1: MC Basic (a model-free variant of policy iteration)

Initialization: Initial guess π_0.
Goal: Search for an optimal policy.

For the kth iteration ($k = 0, 1, 2, \ldots$), do
    For every state $s \in \mathcal{S}$, do
        For every action $a \in \mathcal{A}(s)$, do
            Collect sufficiently many episodes starting from (s, a) by following π_k
            Policy evaluation:
            $q_{\pi_k}(s, a) \approx q_k(s, a) =$ the average return of all the episodes starting from (s, a)
        Policy improvement:
        $a_k^*(s) = \arg\max_a q_k(s, a)$
        $\pi_{k+1}(a|s) = 1$ if $a = a_k^*$, and $\pi_{k+1}(a|s) = 0$ otherwise

Recapitulação das aulas passadas

Algorithm 5.2: MC Exploring Starts (an efficient variant of MC Basic)

Initialization: Initial policy $\pi_0(a|s)$ and initial value $q(s,a)$ for all (s,a). Returns$(s,a) = 0$ and Num$(s,a) = 0$ for all (s,a).
Goal: Search for an optimal policy.

For each episode, do

Episode generation: Select a starting state-action pair (s_0, a_0) and ensure that all pairs can be possibly selected (this is the exploring-starts condition). Following the current policy, generate an episode of length T: $s_0, a_0, r_1, \ldots, s_{T-1}, a_{T-1}, r_T$.
Initialization for each episode: $g \leftarrow 0$
For each step of the episode, $t = T-1, T-2, \ldots, 0$, do

$g \leftarrow \gamma g + r_{t+1}$
Returns$(s_t, a_t) \leftarrow$ Returns$(s_t, a_t) + g$
Num$(s_t, a_t) \leftarrow$ Num$(s_t, a_t) + 1$
Policy evaluation:
$q(s_t, a_t) \leftarrow$ Returns(s_t, a_t)/Num(s_t, a_t)
Policy improvement:
$\pi(a|s_t) = 1$ if $a = \arg\max_a q(s_t, a)$ and $\pi(a|s_t) = 0$ otherwise

Recapitulação das aulas passadas

- Tanto o MC Básico quanto o MC com inícios exploratórios (*Exploring Starts*) exigem que haja um número suficientemente grande de episódios iniciando de **cada par** estado-ação (s, a).

Algorithm 5.1: MC Basic (a model-free variant of policy iteration)

Initialization: Initial guess π_0.
Goal: Search for an optimal policy.

For the kth iteration ($k = 0, 1, 2, \ldots$), do
    For every state $s \in \mathcal{S}$, do
        For every action $a \in \mathcal{A}(s)$, do

Algorithm 5.2: MC Exploring Starts (an efficient variant of MC Basic)

Initialization: Initial policy $\pi_0(a|s)$ and initial value $q(s, a)$ for all (s, a). Returns$(s, a) = 0$ and Num$(s, a) = 0$ for all (s, a).
Goal: Search for an optimal policy.

For each episode, do
    Episode generation: Select a starting state-action pair (s_0, a_0) and ensure that all pairs can be possibly selected (this is the exploring-starts condition). Following the

Como dispensar a condição de que todos os pares (estado, ação) sejam utilizados como ponto de partida dos episódios?

MC ϵ-guloso (ϵ-*greedy*)

- Política suave
  - Política estocástica.
  - Possui uma probabilidade positiva de escolher qualquer ação em qualquer estado.
  - Em um único episódio suficientemente longo é possível visitar todos os pares (estado, ação) várias vezes.
  - Condição de inícios exploratórios pode ser descartada, pois não é necessário gerar muitos episódios iniciando de diferentes pares (estado, ação).
  - **Política ϵ-gulosa (ϵ-*greedy*)**
    - Possui maior probabilidade de escolher a ação gulosa (com maior valor de ação).
    - Mantém uma mesma probabilidade não nula de escolher qualquer outra ação.

MC ϵ-guloso (ϵ-*greedy*)

- **Política ϵ-gulosa (ϵ-*greedy*)**
  - Formulação

$$\pi(a|s) = \begin{cases} 1 - \dfrac{\epsilon}{|\mathcal{A}(s)|}(|\mathcal{A}(s)| - 1), & ação\ gulosa \\ \dfrac{\epsilon}{|\mathcal{A}(s)|}, & qualquer\ outra\ das\ (|\mathcal{A}(s)| - 1)\ ações \end{cases}$$

  - $|\mathcal{A}(s)|$: número de ações disponíveis no estado s.
  - $\epsilon = 0$: política ϵ-gulosa se torna puramente gulosa.
  - $\epsilon = 1$: probabilidade de escolher qualquer ação é igual a $\frac{\epsilon}{|\mathcal{A}(s)|}$ (distribuição uniforme).
  - $0 \leq \epsilon \leq 1$: probabilidade da ação gulosa é sempre maior ou igual ao de escolher qualquer outra ação.

$$1 - \frac{\epsilon}{|\mathcal{A}(s)|}(|\mathcal{A}(s)| - 1) = 1 - \epsilon + \frac{\epsilon}{|\mathcal{A}(s)|} \geq \frac{\epsilon}{|\mathcal{A}(s)|}$$

MC ϵ-guloso (ϵ-*greedy*)

- **Política ϵ-gulosa (ϵ-*greedy*)**
  - Seleção de ação:
    - Sorteia-se um número aleatório $x \in [0,1]$ de uma distribuição uniforme.
      - $x \geq \epsilon$: ação gulosa
      - $x < \epsilon$: ação aleatória (incluindo a ação gulosa) com probabilidade $\frac{\epsilon}{|\mathcal{A}(s)|}$
      - Probabilidade total de selecionar a ação gulosa:

$$1 - \epsilon + \frac{\epsilon}{|\mathcal{A}(s)|}$$

      - Probabilidade de selecionar qualquer outra ação:

$$\frac{\epsilon}{|\mathcal{A}(s)|}$$

MC ϵ-guloso (ϵ-*greedy*)

- **Melhoria da política**
  - MC Básico ou MC com inícios exploratórios

$$\pi_{k+1}(s) = \underset{\pi \in \Pi}{\operatorname{argmax}} \sum_{a} \pi(a|s)\, q_{\pi_k}(s,a)$$

- Π: conjunto de todas as políticas possíveis
- Solução: política determinística gulosa

$$\pi_{k+1}(a|s) = \begin{cases} 1, & a = a_k^*(s) \\ 0, & a \neq a_k^*(s) \end{cases}, \qquad a_k^*(s) = \underset{a}{\operatorname{argmax}}\, q_{\pi_k}(s,a)$$

MC ϵ-guloso (ϵ-*greedy*)

- **Melhoria da política**
  - MC ϵ-guloso

$$\pi_{k+1}(s) = \underset{\pi \in \Pi_\epsilon}{\operatorname{argmax}} \sum_a \pi(a|s)\, q_{\pi_k}(s,a)$$

- Π_ϵ: conjunto de todas as políticas ϵ-gulosas possíveis para um dado valor de ϵ.
- Solução:

$$\pi_{k+1}(a|s) = \begin{cases} 1 - \dfrac{|\mathcal{A}(s)| - 1}{|\mathcal{A}(s)|}\epsilon, & a = a_k^*(s) \\ \dfrac{\epsilon}{|\mathcal{A}(s)|}, & a \neq a_k^*(s) \end{cases}, \qquad a_k^*(s) = \underset{a}{\operatorname{argmax}}\, q_{\pi_k}(s,a)$$

MC ϵ-guloso (ϵ-greedy)

Algorithm 5.3: MC ϵ-Greedy (a variant of MC Exploring Starts)

Initialization: Initial policy $\pi_0(a|s)$ and initial value $q(s,a)$ for all (s,a). Returns$(s,a) = 0$ and Num$(s,a) = 0$ for all (s,a). $\epsilon \in (0,1]$

Goal: Search for an optimal policy.

For each episode, do

Episode generation: Select a starting state-action pair (s_0, a_0) (the exploring starts condition is not required). Following the current policy, generate an episode of length T: $s_0, a_0, r_1, \ldots, s_{T-1}, a_{T-1}, r_T$.

Initialization for each episode: $g \leftarrow 0$

For each step of the episode, $t = T-1, T-2, \ldots, 0$, do

$g \leftarrow \gamma g + r_{t+1}$

Returns$(s_t, a_t) \leftarrow$ Returns$(s_t, a_t) + g$

Num$(s_t, a_t) \leftarrow$ Num$(s_t, a_t) + 1$

Policy evaluation:

$q(s_t, a_t) \leftarrow$ Returns(s_t, a_t)/Num(s_t, a_t)

Policy improvement:

Let $a^* = \arg\max_a q(s_t, a)$ and

$$\pi(a|s_t) = \begin{cases} 1 - \frac{|\mathcal{A}(s_t)|-1}{|\mathcal{A}(s_t)|}\epsilon, & a = a^* \\ \frac{1}{|\mathcal{A}(s_t)|}\epsilon, & a \neq a^* \end{cases}$$

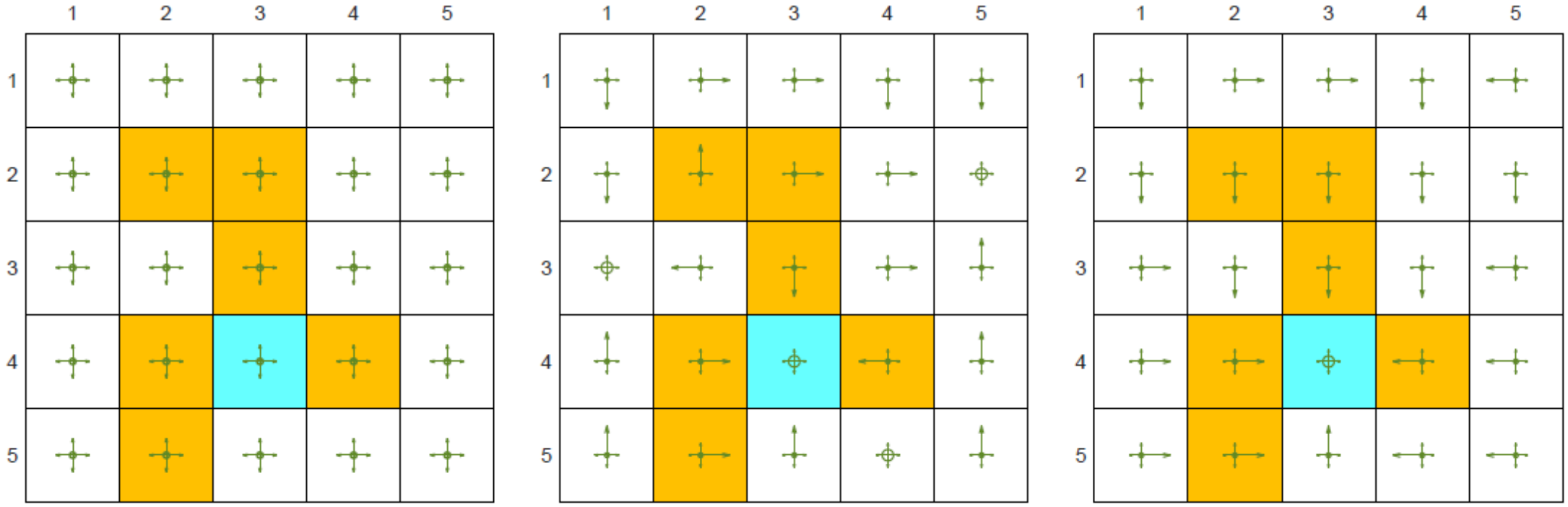
Podemos garantir a obtenção de políticas ótimas?

MC ϵ-guloso (ϵ-*greedy*)

- Podemos garantir a obtenção de políticas ótimas?
  - O algoritmo pode convergir para uma política ϵ-gulosa que é ótima no conjunto Π_ϵ.
  - A política será apenas ótima dentro de Π_ϵ, mas pode não ser ótima no conjunto Π.

MC ϵ-guloso (ϵ-*greedy*)

- Exemplo do algoritmo MC ϵ-guloso
  - Mundo em grade de 5x5
  - $r_{forbidden} = r_{boundary} = -1, r_{target} = 1, \gamma = 0.9, \epsilon = 0.5.$
  - 1 episódio por iteração, cada episódio com 10^6 passos



(a) Initial policy (b) After the first iteration (c) After the second iteration

MC ϵ-guloso (ϵ-*greedy*)

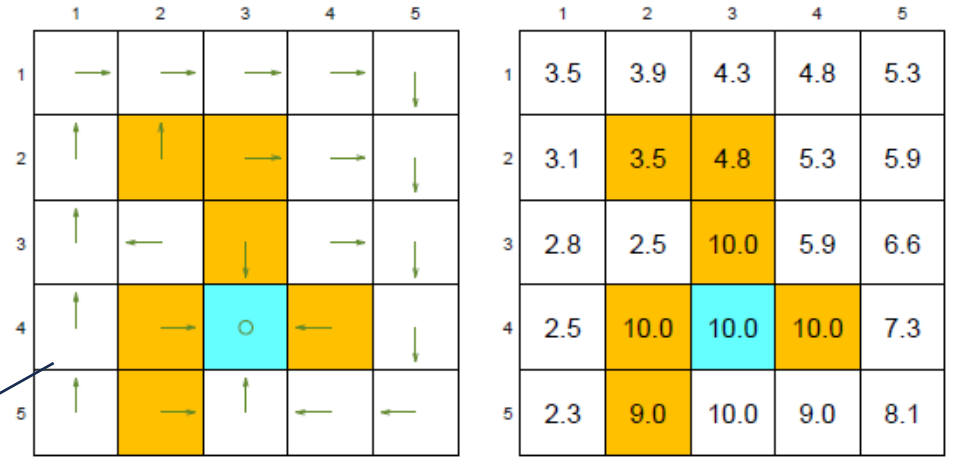
- Explorar versus Usufruir
  - Dilema fundamental no aprendizado por reforço.
  - **Explorar**: permitir que a política experimente diferentes ações, de modo que todas possam ser avaliadas.
  - **Usufruir**: seguir a ação com o maior valor de ação (ação gulosa).
- A estimativa atual pode ser imprecisa por exploração insuficiente.
  - É necessário continuar explorando enquanto se aproveitam as melhores estimativas, para evitar descartar ações que poderiam ser ótimas.
- Políticas ϵ-greedy
  - Propõem um equilíbrio entre esses dois objetivos.
  - Tendem a escolher a melhor ação conhecida (usufruir), mas ainda oferecem uma chance de selecionar outras ações (explorar).
  - Balancear explorar/usufruir: sacrificar um pouco um objetivo para alcançar o outro.

MC ϵ-guloso (ϵ-greedy)

- Mundo em grade de 5x5
- $r_{forbidden} = -1, r_{boundary} = -10, r_{target} = 1, \gamma = 0.9, \epsilon = 0.5.$
- 1 episódio por iteração, cada episódio com 10^6 passos

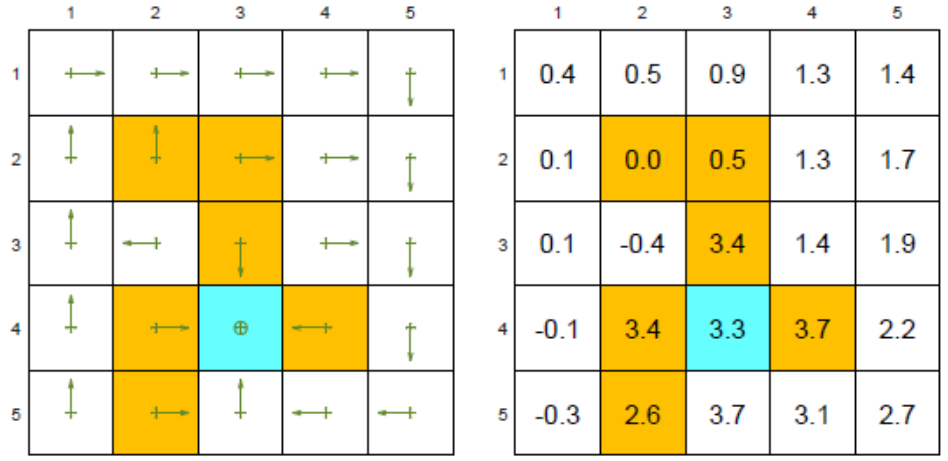
- Exemplo
- O que podemos observar?

Política ótima gulosa



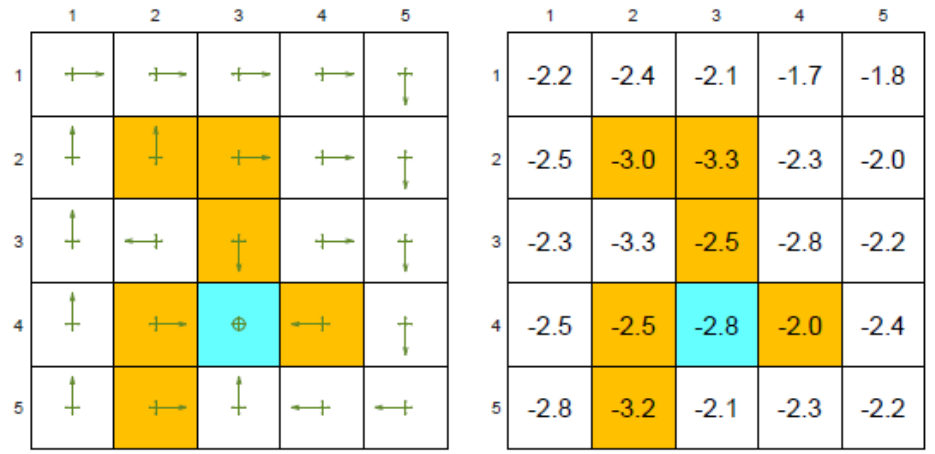
	1	2	3	4	5
1	3.5	3.9	4.3	4.8	5.3
2	3.1	3.5	4.8	5.3	5.9
3	2.8	2.5	10.0	5.9	6.6
4	2.5	10.0	10.0	10.0	7.3
5	2.3	9.0	10.0	9.0	8.1

(a) A given ϵ-greedy policy and its state values: $\epsilon = 0$



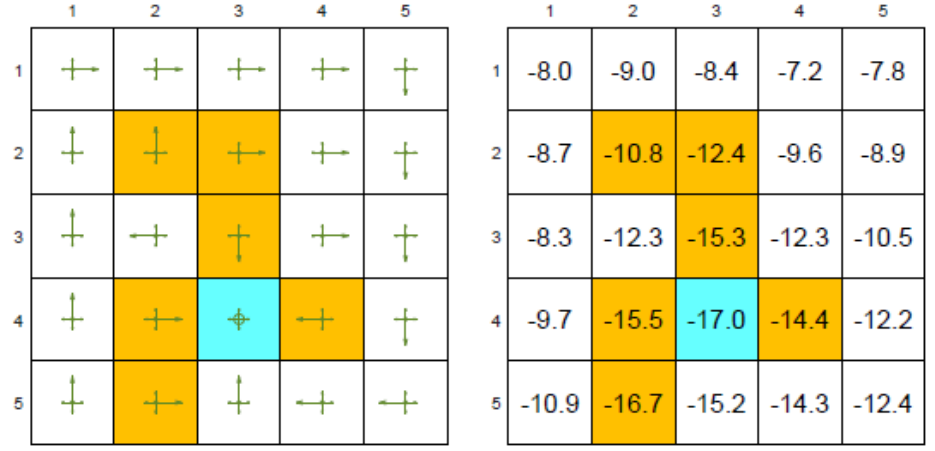
	1	2	3	4	5
1	0.4	0.5	0.9	1.3	1.4
2	0.1	0.0	0.5	1.3	1.7
3	0.1	-0.4	3.4	1.4	1.9
4	-0.1	3.4	3.3	3.7	2.2
5	-0.3	2.6	3.7	3.1	2.7

(b) A given ϵ-greedy policy and its state values: $\epsilon = 0.1$



	1	2	3	4	5
1	-2.2	-2.4	-2.1	-1.7	-1.8
2	-2.5	-3.0	-3.3	-2.3	-2.0
3	-2.3	-3.3	-2.5	-2.8	-2.2
4	-2.5	-2.5	-2.8	-2.0	-2.4
5	-2.8	-3.2	-2.1	-2.3	-2.2

(c) A given ϵ-greedy policy and its state values: $\epsilon = 0.2$



	1	2	3	4	5
1	-8.0	-9.0	-8.4	-7.2	-7.8
2	-8.7	-10.8	-12.4	-9.6	-8.9
3	-8.3	-12.3	-15.3	-12.3	-10.5
4	-9.7	-15.5	-17.0	-14.4	-12.2
5	-10.9	-16.7	-15.2	-14.3	-12.4

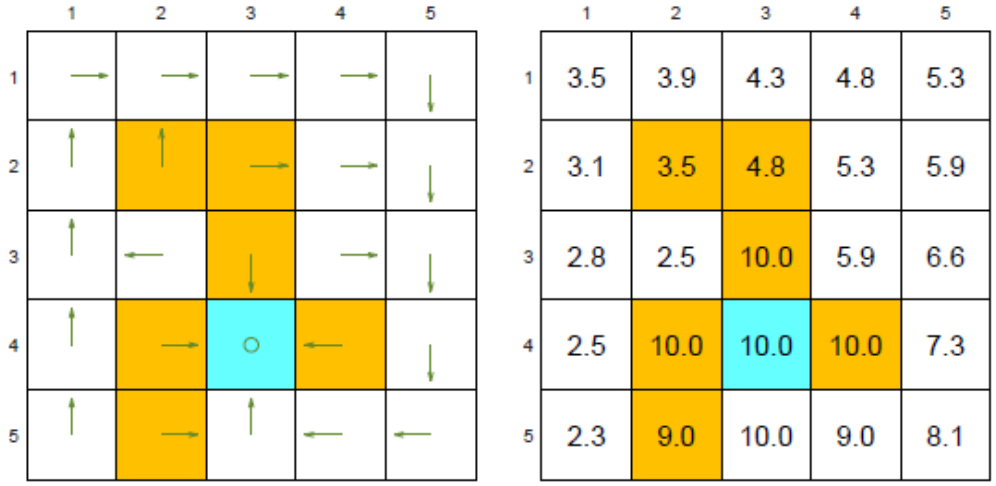
(d) A given ϵ-greedy policy and its state values: $\epsilon = 0.5$

Políticas ϵ-gulosas **consistentes:** ações com as maiores probabilidades são as mesmas.

MC ϵ-guloso

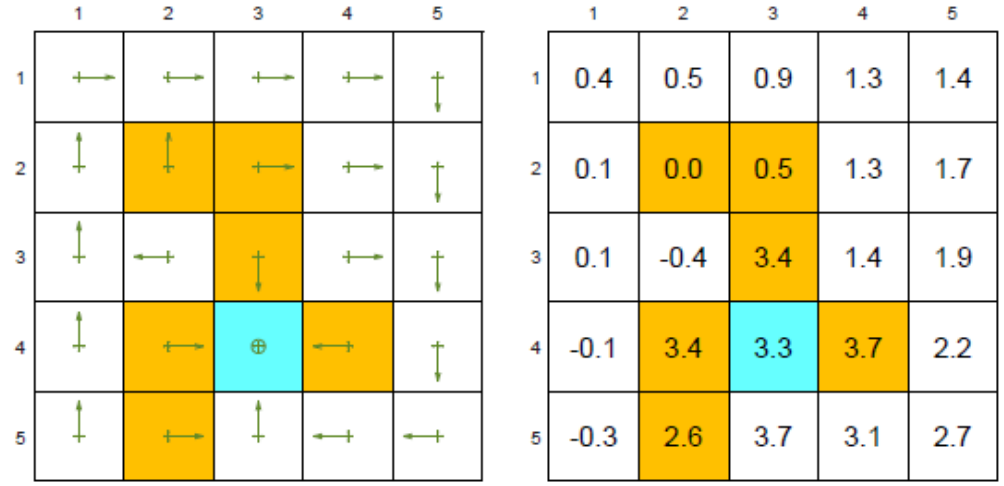
- Mundo em grade de 5x5
- $r_{forbidden} = -1, r_{boundary} = -10, r_{target} = 1, \gamma = 0.9.$

- Exemplo
- O que podemos observar?



	1	2	3	4	5
1	3.5	3.9	4.3	4.8	5.3
2	3.1	3.5	4.8	5.3	5.9
3	2.8	2.5	10.0	5.9	6.6
4	2.5	10.0	10.0	10.0	7.3
5	2.3	9.0	10.0	9.0	8.1

(a) The optimal ϵ-greedy policy and its state values: $\epsilon = 0$



	1	2	3	4	5
1	0.4	0.5	0.9	1.3	1.4
2	0.1	0.0	0.5	1.3	1.7
3	0.1	-0.4	3.4	1.4	1.9
4	-0.1	3.4	3.3	3.7	2.2
5	-0.3	2.6	3.7	3.1	2.7

(b) The optimal ϵ-greedy policy and its state values: $\epsilon = 0.1$

	1	2	3	4	5
1	-1.1	-1.5	-1.1	-0.6	-0.6
2	-1.5	-2.2	-2.3	-1.0	-0.6
3	-1.2	-2.4	-2.2	-1.5	-0.6
4	-1.6	-2.3	-2.6	-1.4	-1.1
5	-2.0	-3.0	-1.8	-1.4	-1.0

(c) The optimal ϵ-greedy policy and its state values: $\epsilon = 0.2$

	1	2	3	4	5
1	-4.3	-5.5	-4.5	-2.6	-2.3
2	-5.6	-7.7	-7.7	-4.1	-2.4
3	-5.5	-9.0	-8.0	-5.6	-2.8
4	-6.8	-8.9	-9.4	-5.5	-4.2
5	-7.9	-10.1	-6.7	-5.1	-3.7

(d) The optimal ϵ-greedy policy and its state values: $\epsilon = 0.5$

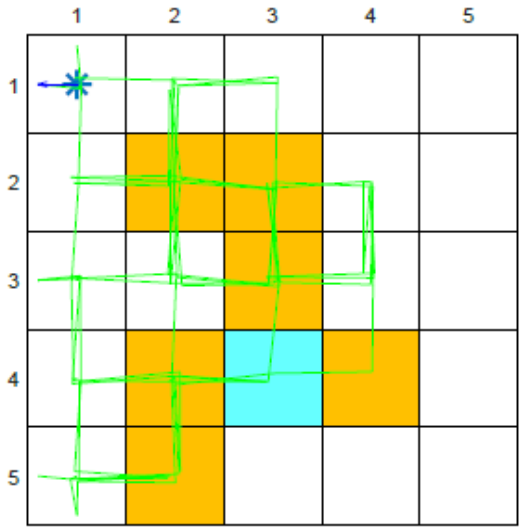
Políticas ϵ-gulosas ótimas

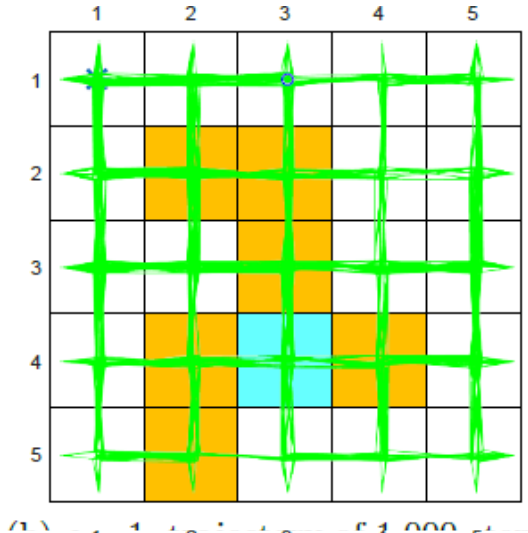
MC ϵ-guloso (ϵ-*greedy*)

- Exemplo
  - A otimalidade das políticas ϵ-gulosas piora com o aumento de ϵ (valores de estado são menores).
    - Valor de estado do estado alvo diminui quando ϵ assume valores maiores.
    - Em estados bons, aumenta ϵ aumenta a chance de visitar regiões ruins.
  - Com o aumento de ϵ, políticas ϵ-gulosas ótimas tornam-se inconsistentes com a gulosa ótima.

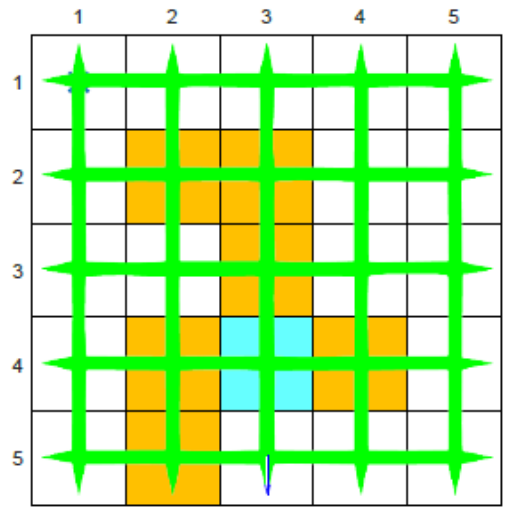
MC ϵ-guloso

- Exemplo – Exploração
- O que podemos observar?

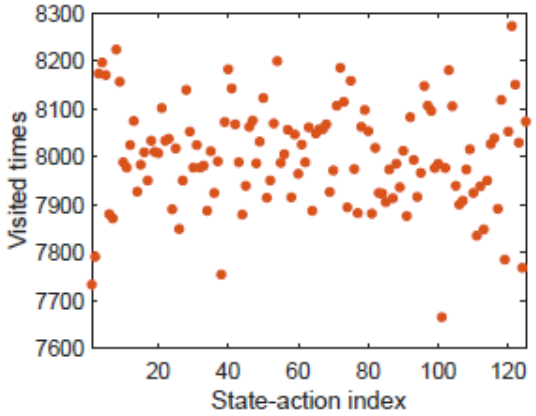




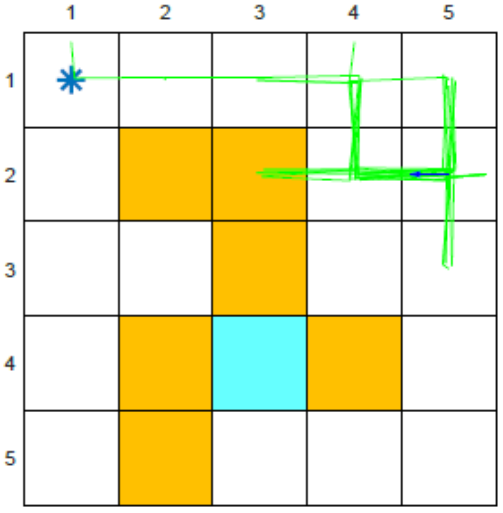
(b)



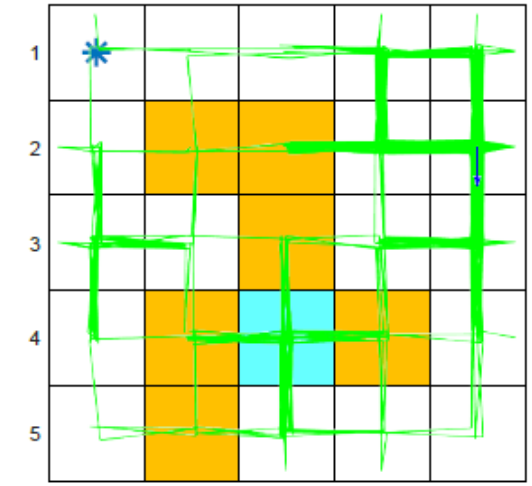
(c) $\epsilon = 1$, trajectory of 10,000 steps



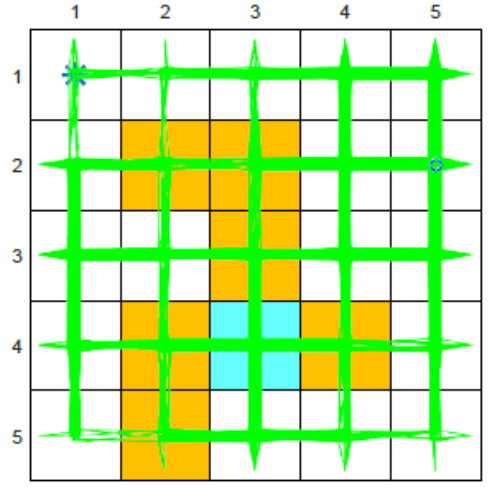
(d) $\epsilon = 1$, number of times each action is visited within 1 million steps



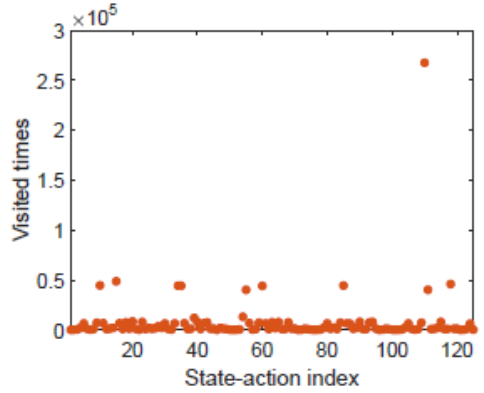
(e) $\epsilon = 0.5$, trajectory of 100 steps



(f) $\epsilon = 0.5$, trajectory of 1,000 steps



(g) $\epsilon = 0.5$, trajectory of 10,000 steps



(h) $\epsilon = 0.5$, number of times each action is visited within 1 million steps

MC ϵ-guloso (ϵ-*greedy*)

- Exemplo – Exploração
- A capacidade de exploração diminui quando ϵ diminui.
- Estratégia útil:
  - Começar com ϵ alto e reduzir gradualmente
  - Garante exploração inicial e assegura a otimalidade da política final.

Resumo

- Algoritmos MC
  - MC Básico
    - Substitui avaliação de política baseada em modelo por uma estimação sem modelo.
  - MC com inícios exploratórios
    - Variante do MC Básico
    - Usa a estratégia de primeira ou todas as visitas para melhorar a utilização das amostras.
  - MC ϵ-guloso:
    - Variante MC com inícios exploratórios
    - Busca por políticas ϵ-gulosas (estocásticas) ao invés de gulosas (determinísticas).
    - Remove necessidade da condição de inícios exploratórios.
- Balanço explorar vs. usufruir
  - $\uparrow \epsilon$: mais exploração, menos usofruto
  - $\downarrow \epsilon$: mais usofruto, menos exploração

Referências

- Shiyu Zhao. Mathematical Foundations of Reinforcement Learning. Springer Singapore, 2025. [capítulo 5]
  - disponível em: https://github.com/MathFoundationRL/Book-Mathematical-Foundation-of-Reinforcement-Learning

- Richard S. Sutton e Andrew G. Barto. An Introduction Reinforcement Learning, Bradford Book, 2018. [capítulo 5]
  - disponível em: http://incompleteideas.net/book/the-book-2nd.html

Slides construídos com base nos livros supracitados, os quais estão disponibilizados publicamente pelos seus respectivos autores.