Q-learning

Elaine C. R. Cândido

Pontifícia Universidade Católica de Minas Gerais

Março 2021

Conteúdo

Revisão

- Estado de Markov
- Cadeia de Markov ou processo de Markov
- Processo de Recompensa de Markov (MRP)
- Processo de Decisão de Markov (MDP)
- Equação de Bellman
  - Maximizar as recompensas
  - Encontrar $q_\star \rightarrow$ MDP ótimo
- Equação ótima de Bellman é não linear
  - Programação Dinamica
  - Métodos Monte Carlo
  - Métodos de Diferença Temporal (TD)

Abordagens

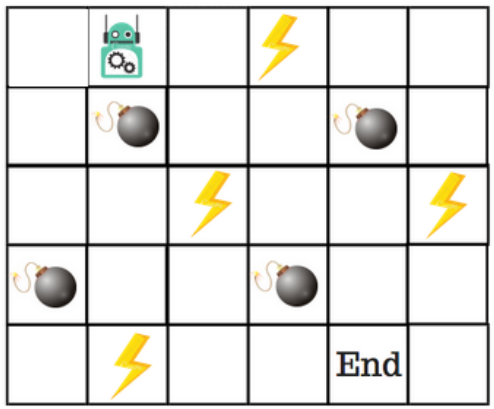
- Programação Dinamica
  - Aproximam $v_\star$ assumindo que há conhecimento perfeito do modelo do ambiente
  - Aprendizado *offline*
- Métodos Monte Carlo
  - Não exigem modelo, mas não obtém melhorias a todo passo
  - Deve ser aplicado em tarefas episódicas
- Métodos de Diferença Temporal (TD)
  - Não exigem modelo e são totalmente incrementais
  - *Q-learning*: uma das mais importantes descobertas

Q-learning

- Usa solução iterativa
  1. Inicializa estimativas
  2. Escolhe ação
  3. Obtém retorno do ambiente
  4. Melhora estimativas baseado na equação de Bellman

Inicializa estimativas

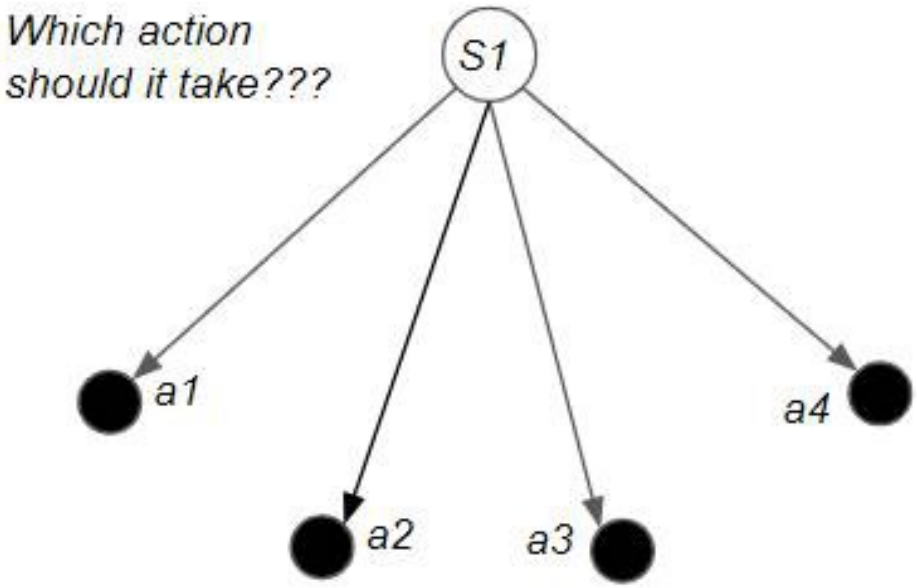
- Cria tabela q (q-table)
- Inicializa-a com zeros



Actions :	↑	→	↓	←
Start	0	0	0	0
Nothing / Blank	0	0	0	0
Power	0	0	0	0
Mines	0	0	0	0
END	0	0	0	0

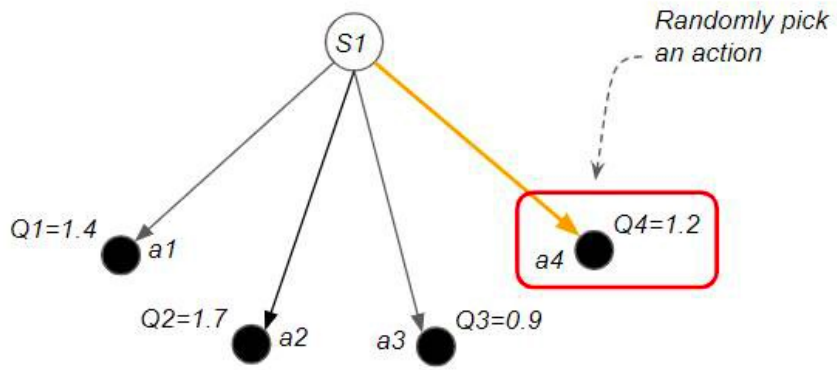
Escolhe ação

- Dado o estado, qual ação devo tomar?



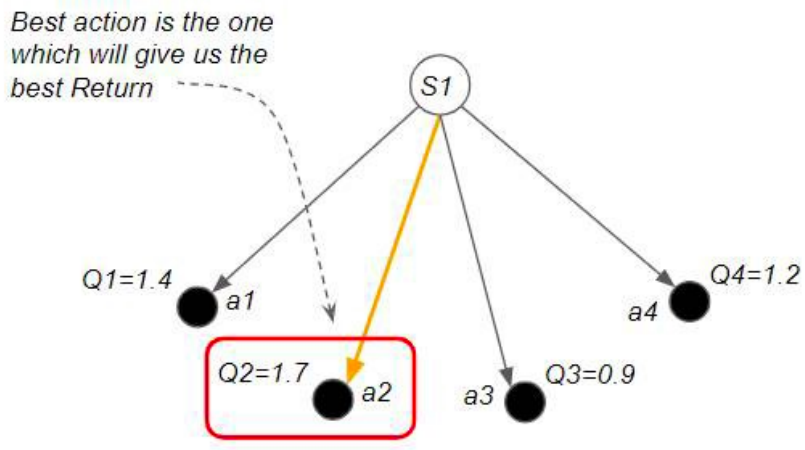
Escolhe ação - *Off-policy*

- Política de comportamento μ - **Explora**
  - A próxima ação é escolhida baseando-se por ela



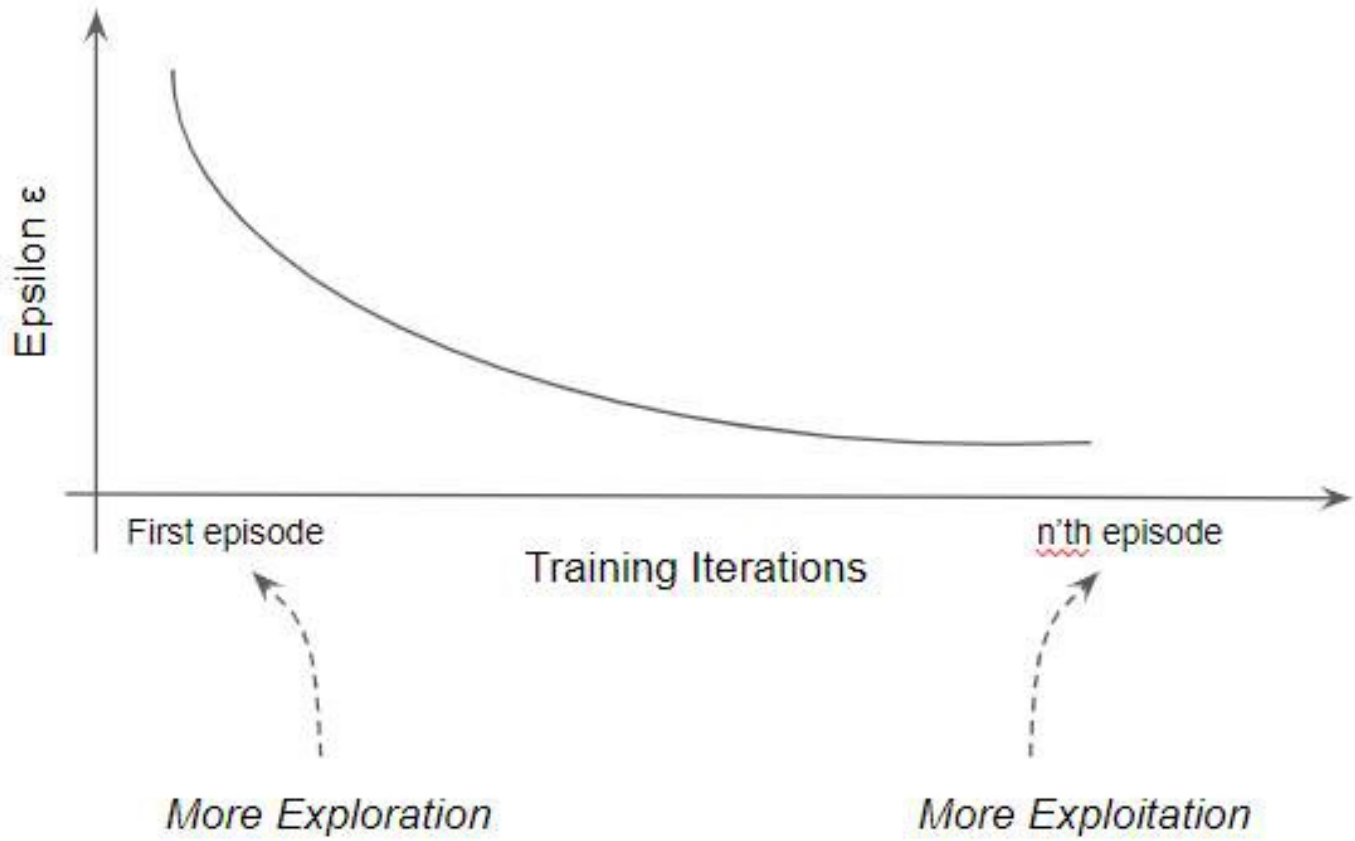
Escolhe ação - *Off-policy*

- Política alvo π - **Aproveita**
  - Referente a ação de maior valor



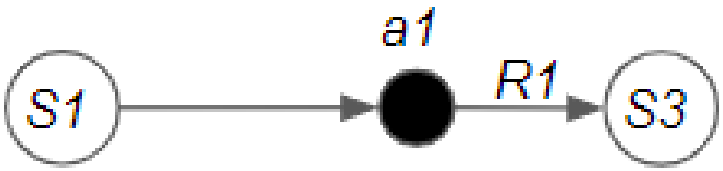
Exploração vs Aproveitamento

- $\epsilon - guloso$
  - taxa de exploração que ajusta o progresso de trinamento para garantir mais exploração no início de treinamento e muda para maior aproveitamento nos episódios finais



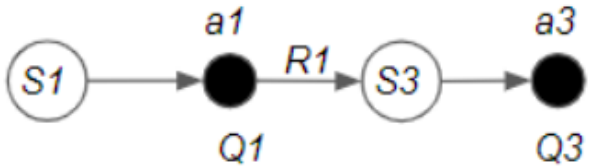
Obtém retorno do ambiente

- Agente recebe feedback do ambiente



Otimizar estimativas

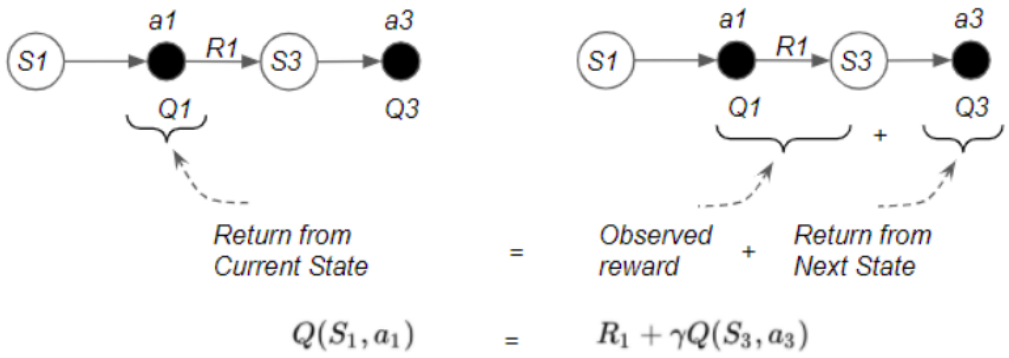
- Baseia-se na equação de Bellman
  - Se sabemos o valor q do próximo par estado-ação, então podemos computar o valor de q atual



$$Q(S_1, a_1) = \mathbb{E}[R_1 + \gamma Q(S_3, a_3)]$$

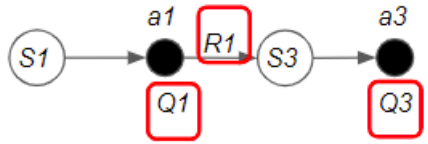
Otimizar estimativas

- Temos duas formas de computar o valor q:
  - Pelo valor q do estado atual
  - Pelo valor da recompensa + o valor q do próximo estado



Otimizar estimativas

- Usando valores estimados, podemos checar ao longo do treinamento quão corretas estão as estimativas



$$Q(S_1, a_1) = Q(S_1, a_1) + \alpha * Error$$

$$Q(S_1, a_1) = Q(S_1, a_1) + \alpha(R_1 + \gamma Q(S_3, a_3) - Q(S_1, a_1))$$

Q-learning - Algoritmo

Algoritmo 6 Controle por Q-Learning

Seja $Q(s, a)$ inicializado arbitrariamente, para todo s e a

1: **Repita**
2: Defina um estado inicial s
3: **Repita**
4: Escolha a de acordo com uma política derivada de Q (ex: ϵ-gulosa)
5: Execute a e observe a recompensa r e o estado sucessor, s'
6: $Q(s, a) \leftarrow Q(s, a) + \alpha(r + \gamma \max_{a'} Q(s', a') - Q(s, a))$
7: $s \leftarrow s'$
8: **até que** s seja um estado terminal
9: **fim repita**

Referências

- Livros
  - Ravichandiran, S., 2018. Hands-on Reinforcement Learning with Python: Master Reinforcement and Deep Reinforcement Learning Using OpenAI Gym and TensorFlow. Packt Publishing Ltd.
  - Lonza, A., 2019. Reinforcement Learning Algorithms with Python: Learn, understand, and develop smart algorithms for addressing AI challenges. Packt Publishing Ltd.

Referências

- Artigos
  - Ketan Doshi. Reinforcement Learning Explained Visually
    https://towardsdatascience.com/
    reinforcement-learning-explained-visually-part-3-model-fre
  - An introduction to Q-Learning: reinforcement learning
    https://www.freecodecamp.org/news/
    an-introduction-to-q-learning-reinforcement-learning-14ac0