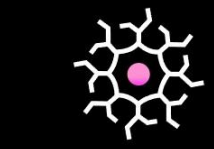
Introdução ao

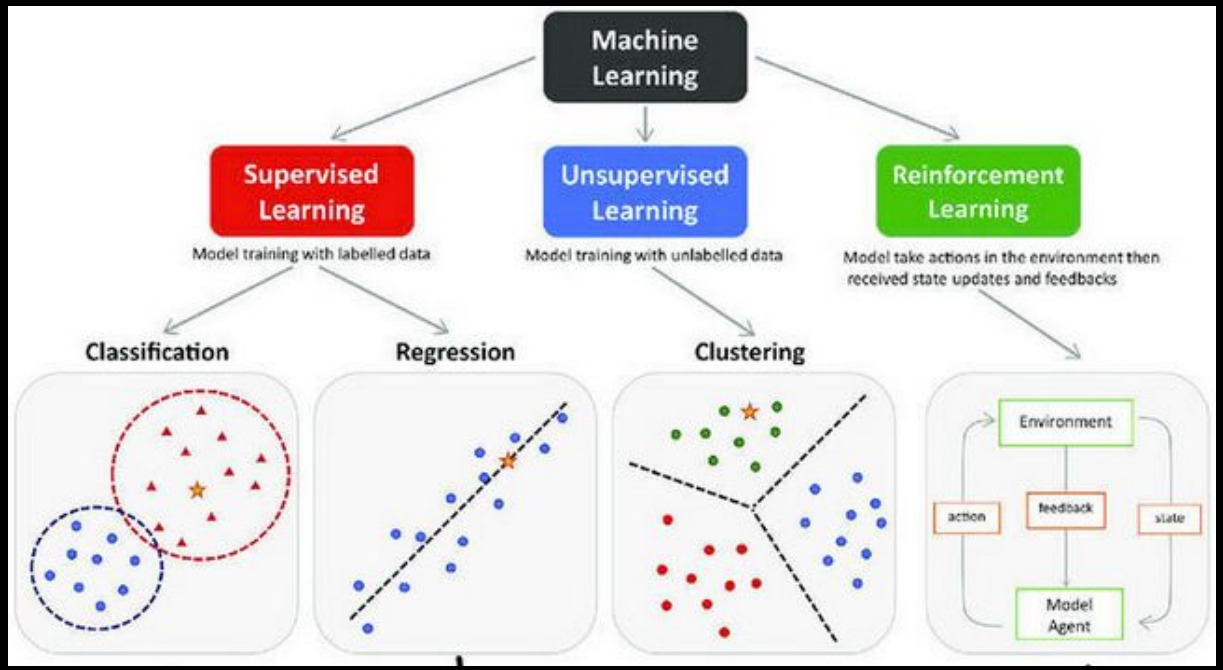
Reinforcement Learning



ACTION
CODE

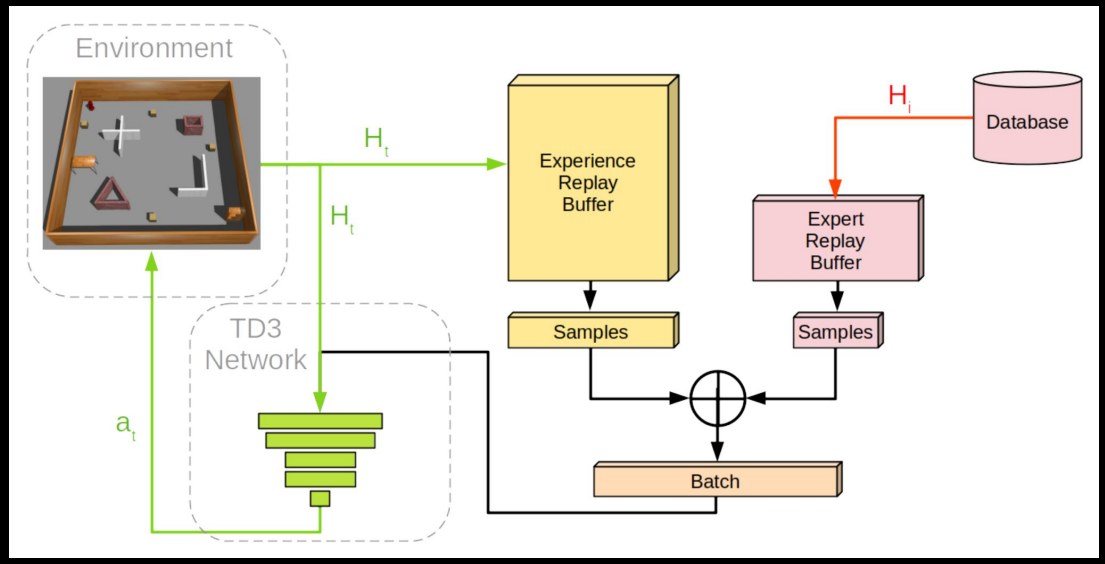
Introdução ao Reinforcement Learning

- Tipos de aprendizado de máquina: supervisionado, não-supervisionado, por reforço
- Peculiaridades do RL: feedback por tentativa, dependência de ambiente, retorno atrasado, *moving target*, *exploration vs. exploitation*



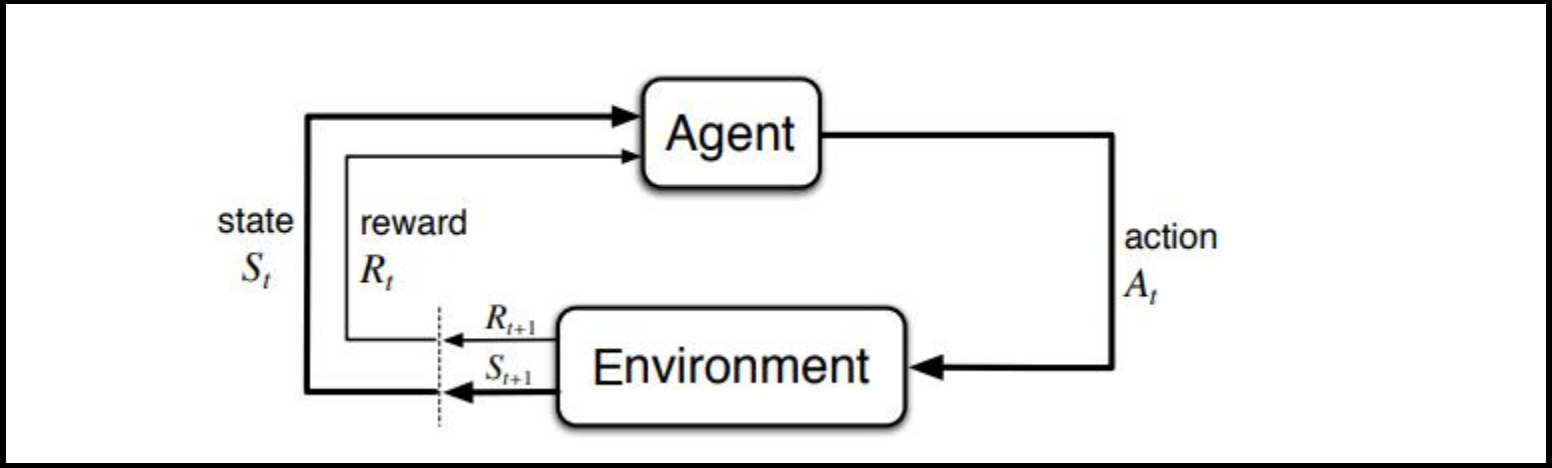
Introdução ao Reinforcement Learning

- Aplicações de RL:
  - *jogos,*
  - *robótica,*
  - *recomendação,*
  - *finanças,*
  - *llm, etc.*



Elementos de RL

- Agente: quem aprende
- Ambiente: onde ocorre a interação (environment)
- Estado ou observação: o que o agente percebe (state)
- Recompensa: o sinal de feedback (reward)
- Taxa de desconto (γ): peso das recompensas futuras (discount)



Interação Agente-Ambiente

- Em cada passo t:
  - O agente observa o estado s_t
  - Escolhe uma ação a_t
  - O ambiente devolve nova observação s_{t+1} e recompensa r_{t+1}
- Objetivo: aprender uma política ótima que maximize a soma de TODAS as recompensas (não apenas a imediata!).

```
episodes = 5
for episode in range(1, episodes+1):
    state = env.reset()
    done = False
    score = 0

    while not done:
        env.render()
        action = random.choice([0,1,2,3,4,5])
        n_state, reward, done, info = env.step(action)
        score+=reward
    print('Episode:{} Score:{}'.format(episode, score))
env.close()
```

16] ⊗ 0.5s

Dynamic Programming

- Requer modelo completo do ambiente (transições e recompensas)
- Algoritmos: *GPI (General Policy Iteration): evaluation* e *improvement*
- Usa *Bellman Equations* para avaliar e melhorar políticas

Bellman Equations

Expectation

$$V(s) = \mathbb{E}\left[G_t \mid S_t = s\right],$$

where:

$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-t} R_T.$$

$$V(s) = \mathbb{E}\left[R_{t+1} + \gamma G_{t+1} \mid S_t = s\right]. \tag{1}$$

$$V(s) = \sum_a \pi(a \mid s) \sum_{s',r} p(s', r \mid s, a)\left(r + \gamma V(s')\right).$$

Optimallity

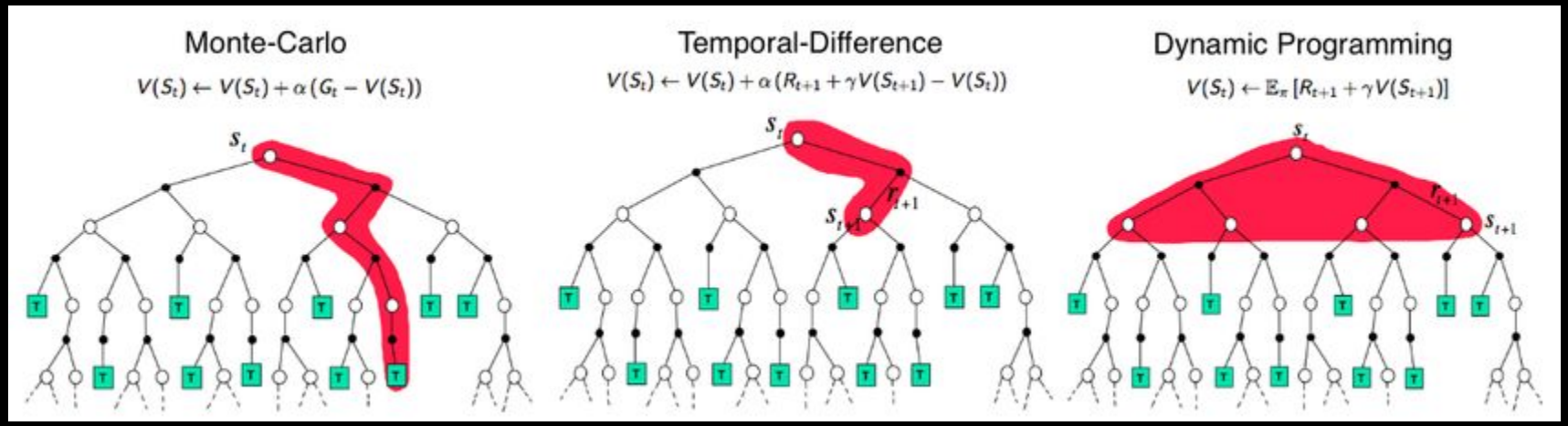
$$v_*(s) = \max_a R_s^a + \gamma \sum_{s' \in S} \mathcal{P}_{ss'}^a v_*(s')$$

$$q_*(s,a) = R_s^a + \gamma \sum_{s' \in S} \mathcal{P}_{ss'}^a \max_{a'} q_*(s', a')$$

Monte Carlo e TD Learning

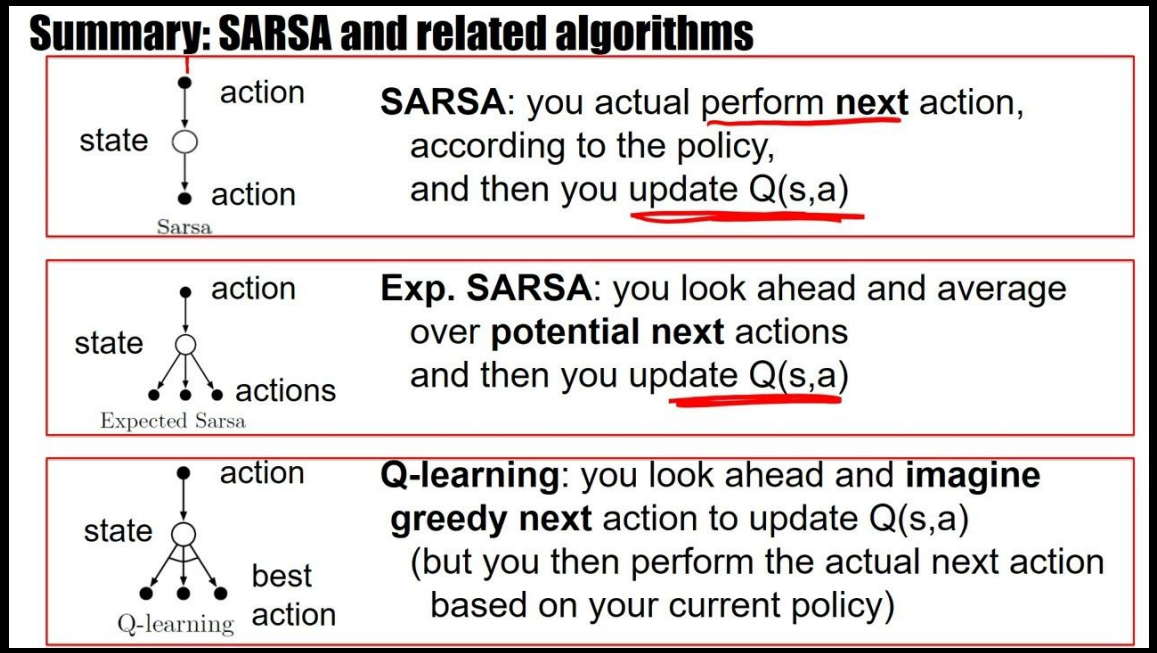
Model free learning, ambos usam value function:

- Monte Carlo: aprende por episódios completos, sem modelo
- Temporal Difference (TD): atualiza após cada passo (*bootstrapping*)



SARSA vs. Q-Learning

- SARSA: *on-policy* (segue a política atual)
- *Q-Learning: off-policy* (usa a melhor ação na atualização)
- Ambos usam TD para atualizar Q(s,a)
- Muito usados em ambientes com ações discretas (necessário) e estados com pouca dimensionalidade



Resumo comparação métodos de RL

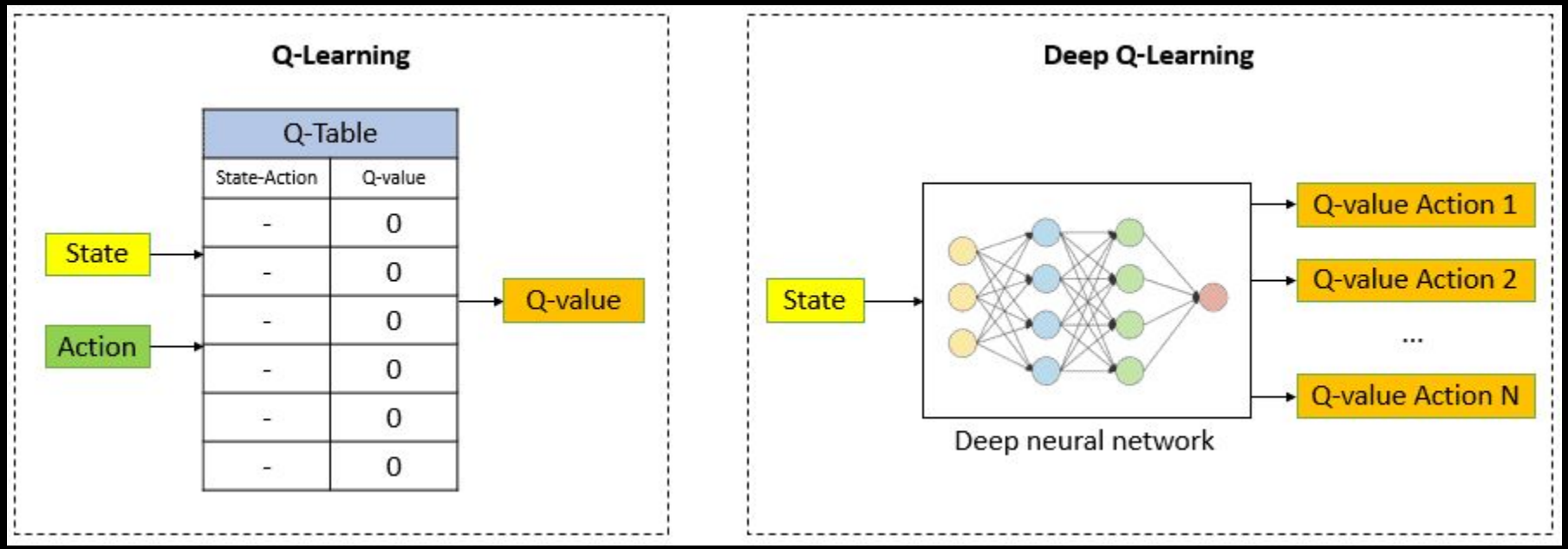
- DP: requer modelo, computacionalmente caro
- Monte Carlo: sem modelo, usa episódios completos
- TD: sem modelo, atualiza por passo
- Q-Learning: off-policy, converge para política ótima
- SARSA: on-policy, mais conservador

Exploração vs. Exploitação (rs)

- Exploração: experimentar ações novas para aprender
- Exploitação: escolher a melhor ação conhecida
- ε decrescente é muito usado em SARSA e Q-Learning

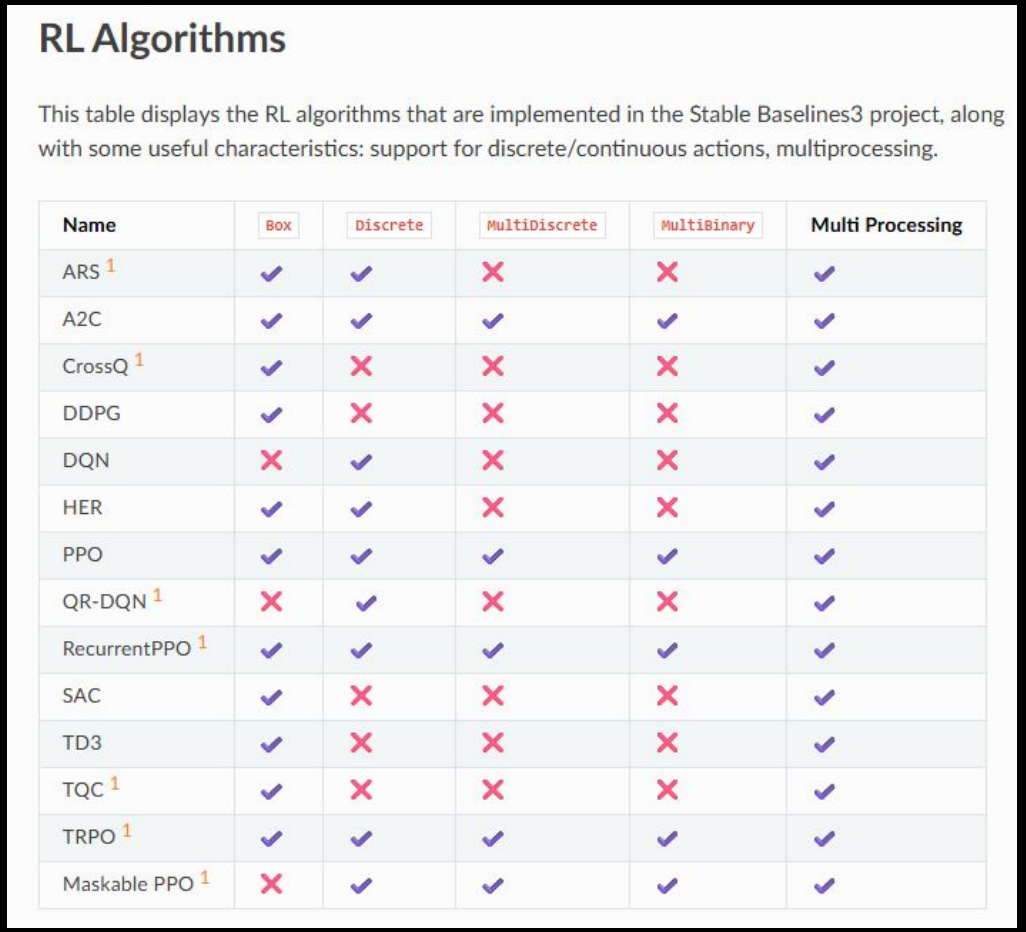
Deep Q-Network (DQN)

- Substitui a tabela Q por uma rede neural (Q-table inviável com muitos estados)
- Replay buffer: armazena e reutiliza experiências
- Target network: reduz instabilidade no treinamento
- Aprendizado eficiente mesmo com estados complexos



Encerramento e Revisão

- RL: interação, recompensa, aprendizado com tentativa e erro
- Evolução:
  DP → MC/TD → Q-Learning → DQN
- Depois: policy gradient...
- Conceitos-chave: política, valor, exploração, estabilidade
- Aplicações reais



RL Algorithms

This table displays the RL algorithms that are implemented in the Stable Baselines3 project, along with some useful characteristics: support for discrete/continuous actions, multiprocessing.

Name	Box	Discrete	MultiDiscrete	MultiBinary	Multi Processing
ARS [1]	✔	✔	✘	✘	✔
A2C	✔	✔	✔	✔	✔
CrossQ [1]	✔	✘	✘	✘	✔
DDPG	✔	✘	✘	✘	✔
DQN	✘	✔	✘	✘	✔
HER	✔	✔	✘	✘	✔
PPO	✔	✔	✔	✔	✔
QR-DQN [1]	✘	✔	✘	✘	✔
RecurrentPPO [1]	✔	✔	✔	✔	✔
SAC	✔	✘	✘	✘	✔
TD3	✔	✘	✘	✘	✔
TQC [1]	✔	✘	✘	✘	✔
TRPO [1]	✔	✔	✔	✔	✔
Maskable PPO [1]	✘	✔	✔	✔	✔