

Aprendizado por Reforço

Prof. Domingos Napolitano

Aula 0: Apresentação do Curso



Aprendizado por Reforço

Disciplina: IBM8919 - 6 Semestre 2026.1

Graduação em Ciência de Dados e Inteligência Artificial

Prof. Dr. Domingos M R Napolitano

Quartas Feiras : 9:40 – 11:40

Sextas Feiras : 7:40 – 9:40 h

DOMINGOS.NAPOLITANO@professores.ibmec.edu.br





Bem-vindos à Disciplina

Nesta disciplina, exploraremos o fascinante mundo do Aprendizado por Reforço, uma das áreas mais promissoras e dinâmicas da Inteligência Artificial moderna.

Quem somos

Engenheiro Mecânico com 30 anos de experiência em gestão de projetos em Engenharia, Construção e TI. Mestre e Doutor em Informática.

Nossa jornada

Ao longo do semestre, construiremos uma base sólida desde os conceitos fundamentais até aplicações avançadas, capacitando-os a desenvolver soluções práticas utilizando algoritmos de Aprendizado por Reforço.

Objetivo principal

Capacitar você a compreender e aplicar os fundamentos do Aprendizado por Reforço, modelando problemas sequenciais com agentes autônomos que aprendem a interagir com o ambiente.

O que é Aprendizado por Reforço?

O Aprendizado por Reforço (RL) é um paradigma de aprendizado de máquina onde agentes autônomos aprendem a tomar decisões sequenciais em ambientes incertos, através de interações baseadas em recompensas.

Diferentemente de outros paradigmas de aprendizado de máquina:

- **Aprendizado Supervisionado:** Aprende com exemplos rotulados
- **Aprendizado Não-Supervisionado:** Encontra padrões em dados não rotulados
- **Aprendizado por Reforço:** Aprende por tentativa e erro com feedback do ambiente

O agente desenvolve uma política para maximizar a recompensa cumulativa ao longo do tempo, sem supervisão explícita.



O que é Aprendizado por Reforço?

O Aprendizado por Reforço (RL) é um paradigma de aprendizado de máquina onde agentes autônomos aprendem a tomar decisões sequenciais em ambientes incertos, através de interações baseadas em recompensas.



O agente desenvolve uma política para maximizar a recompensa cumulativa ao longo do tempo, sem supervisão explícita.

Por que estudar Aprendizado por Reforço?



Inspirado na natureza

O RL se inspira em como humanos e animais aprendem naturalmente, tomando decisões baseadas em experiências passadas e adaptando comportamentos para maximizar recompensas.



Autonomia

Capacita sistemas a tomarem decisões independentes em ambientes complexos e dinâmicos, essencial para robótica avançada e sistemas autônomos.



Superando humanos

Algoritmos de RL têm alcançado resultados impressionantes, superando campeões humanos em jogos como xadrez, Go e StarCraft II.

O aprendizado por reforço representa uma fronteira fundamental da IA, permitindo que sistemas aprendam a realizar tarefas complexas que seriam difíceis de programar explicitamente. As habilidades que você desenvolverá nesta disciplina serão aplicáveis em diversos domínios de ponta na indústria e pesquisa.



Aplicações do Aprendizado por Reforço



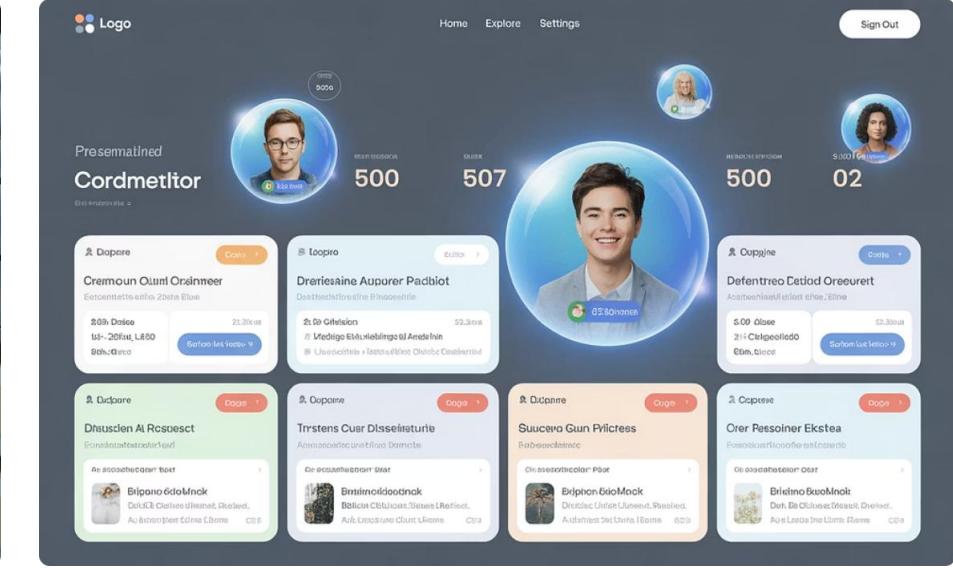
Veículos Autônomos

Desenvolvimento de políticas de navegação, estacionamento e resposta a situações imprevistas em tempo real.



Robótica Industrial

Controle de braços robóticos para manipulação de objetos, montagem de peças e otimização de processos fabris.



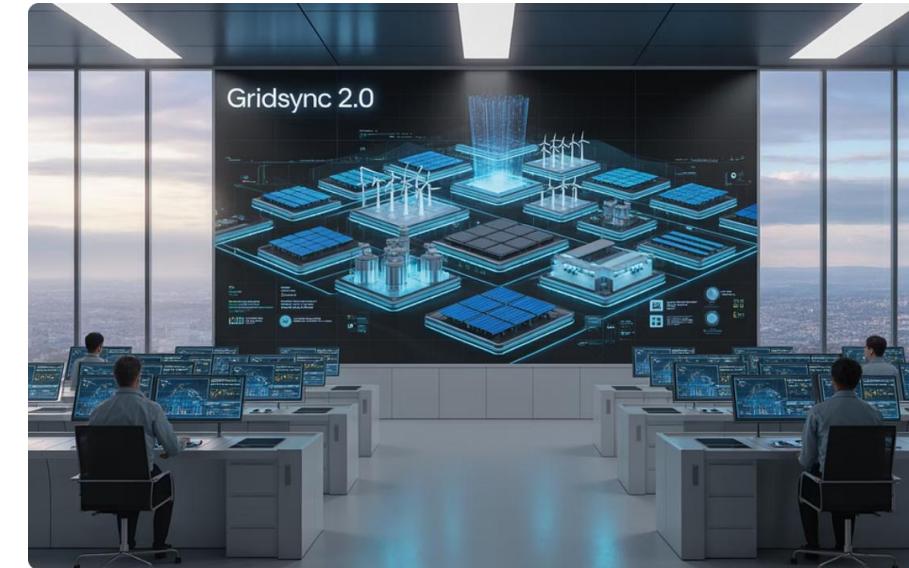
Sistemas de Recomendação

Personalização de conteúdo, otimização de interfaces e estratégias de engajamento adaptativas.



Mercado Financeiro

Desenvolvimento de estratégias de negociação, otimização de portfolios e gerenciamento de riscos.



Gestão Energética

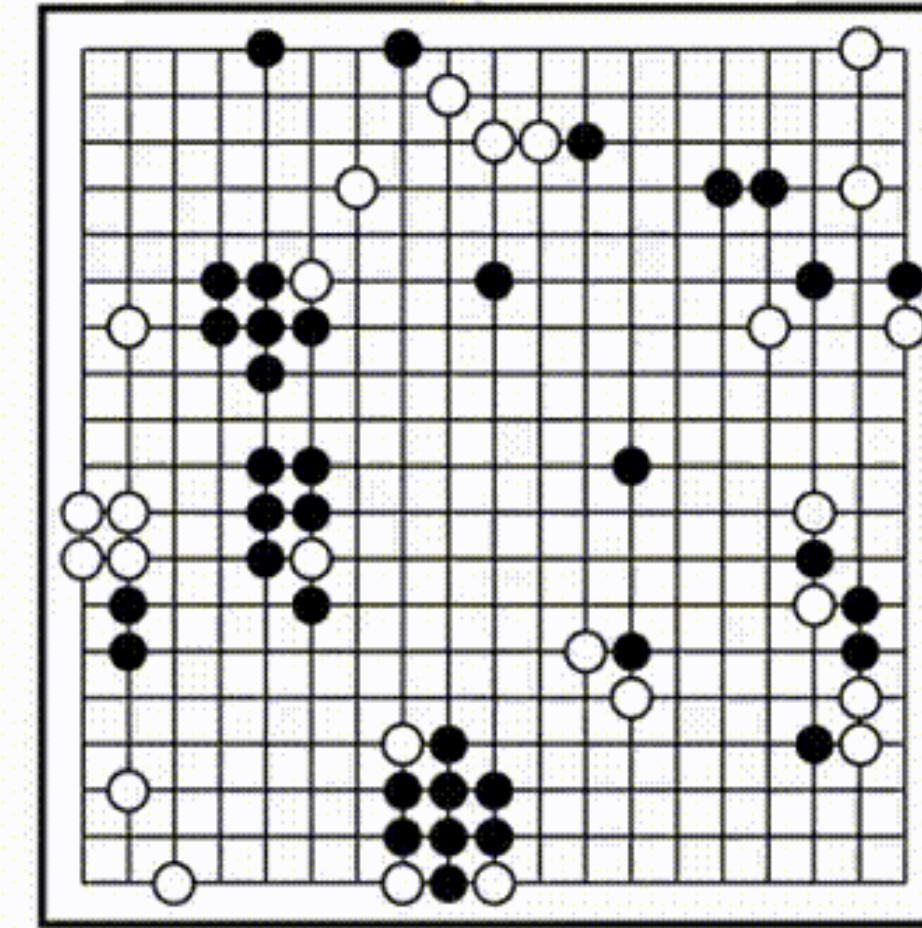
Otimização do consumo de energia em redes inteligentes e controle de sistemas de geração distribuída.



Saúde

Diagnóstico médico, personalização de tratamentos e assistência em procedimentos cirúrgicos.

Aplicações do Aprendizado por Reforço



Objetivos da Disciplina

01

Compreender os fundamentos

Introduzir os conceitos básicos do Aprendizado por Reforço, incluindo a formulação de agentes, recompensas, estados e políticas que formam a base deste paradigma.

02

Dominar os modelos matemáticos

Apresentar os fundamentos matemáticos de Modelos de Decisão de Markov (MDPs) e sua aplicação na modelagem de ambientes estocásticos.

03

Implementar algoritmos clássicos

Explorar e implementar algoritmos tradicionais como Programação Dinâmica, Métodos de Monte Carlo e Diferenças Temporais para solução de problemas sequenciais.

04

Aplicar métodos baseados em valor

Utilizar algoritmos como Q-learning e SARSA para tomada de decisão em ambientes simulados e reais, comparando suas características e desempenho.

05

Explorar tópicos avançados

Discutir métodos baseados em política, ambientes contínuos, aprendizado por reforço profundo e aplicações práticas em diversas áreas.

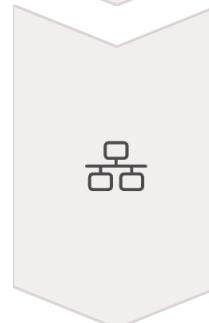
Conteúdo Programático



Fundamentos do RL



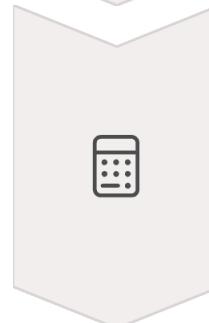
Introdução, conceitos básicos (agente, ambiente, política, função valor, recompensa) e aplicações práticas.



Processos de Decisão



Cadeias de Markov, Modelos de Decisão de Markov (MDPs), definição formal e propriedades do processo de decisão.



Métodos de Solução



Programação Dinâmica, Equações de Bellman, Value Iteration, Policy Iteration, convergência e complexidade.



Métodos de Amostragem



Monte Carlo, estratégias de exploração vs. aproveitamento, Diferenças Temporais (TD Learning), TD(0), TD(λ).



Algoritmos de Valor



Aprendizado off-policy e on-policy, Q-Learning, SARSA e variações destes algoritmos.



Tópicos Avançados



Métodos baseados em política, espaços contínuos, aprendizado por reforço profundo, múltiplos agentes e aplicações.

A disciplina está estruturada de forma progressiva, começando com os fundamentos teóricos e avançando gradualmente para técnicas mais sofisticadas e aplicações práticas.

Metodologia de Ensino

Aulas Teórico-Práticas

Exposições dialogadas combinadas com resolução de exercícios e implementações práticas, utilizando slides, quadros digitais e notebooks interativos.

Ambientes de Simulação

Uso de plataformas como Gymnasium (antigo OpenAI Gym), RLGlue e Google Colab para experimentação e implementação de algoritmos em ambientes controlados.

Exercícios e Projetos

Listas de exercícios teóricos e práticos, atividades individuais e em grupo, com desenvolvimento incremental de agentes e análise de desempenho.

Discussões e Aprofundamento

Análise de artigos científicos e discussão de tópicos avançados, promovendo o pensamento crítico e a exploração de subáreas emergentes.

O tempo estimado para estudo extraclasse é de no mínimo 3 horas semanais para revisão de conteúdo e realização de tarefas. É fundamental manter a regularidade nas atividades práticas para consolidar os conceitos teóricos.

Pré-requisitos e Conhecimentos Recomendados

Conhecimentos Prévios Importantes



Programação em Python

Estruturas de dados, funções, classes, bibliotecas científicas (NumPy, Pandas) e manipulação de arquivos.



Estatística

Probabilidade, variáveis aleatórias, distribuições, esperança matemática e processos estocásticos.



Inferência Estatística

Estimação, testes de hipóteses, intervalos de confiança e métodos de amostragem.

Disciplinas Relacionadas

Conhecimentos de outras disciplinas do mesmo semestre que compõem o grupo de Aprendizado de Máquina:

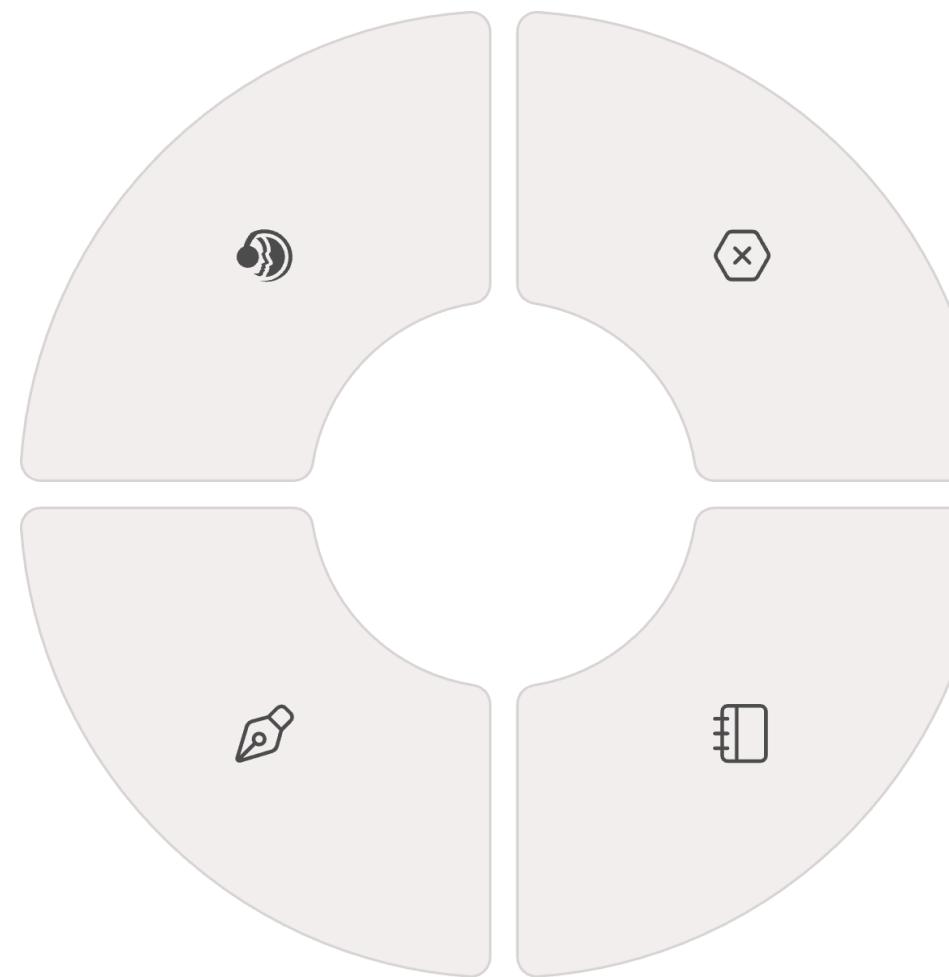
- **Aprendizado de Máquina:** Técnicas supervisionadas e não-supervisionadas que complementam o Aprendizado por Reforço
- **Projeto de Machine Learning:** Metodologias de desenvolvimento que serão aplicadas nos projetos da disciplina
- **Processamento de Linguagem Natural:** Técnicas que podem ser integradas com RL para sistemas mais complexos

Estas disciplinas fornecem uma visão holística do campo de Inteligência Artificial e como o Aprendizado por Reforço se integra ao ecossistema de técnicas de IA.

Sistema de Avaliação

AP1 (40%)

Prova individual teórico-prática abrangendo todo o conteúdo ministrado até a semana anterior à avaliação.



AP2 (40%)

Prova individual teórico-prática abrangendo todo o conteúdo ministrado até a semana anterior à avaliação.

AS (Opcional)

Avaliação suplementar facultativa que substituirá a menor nota entre AP1 e AP2, abrangendo todo o conteúdo do semestre.

AC (20%)

Trabalhos individuais ou em grupo realizados com auxílio de Jupyter Notebooks, envolvendo conceitos teóricos e práticos.

Fórmula para cálculo da Média Final

$$\text{Média Final} = (0,4 \times \text{AP1}) + (0,4 \times \text{AP2}) + (0,2 \times \text{AC})$$

Será considerado aprovado o aluno que obtiver Média Final igual ou superior a 7 (sete) e frequência mínima de 75% nas aulas.

Ferramentas e Ambientes de Aprendizado

Gymnasium

Plataforma para desenvolvimento e avaliação de algoritmos de Aprendizado por Reforço, com ambientes padronizados para testar e comparar diferentes abordagens.



Google Colab

Ambiente de notebook Jupyter baseado em nuvem que permite escrever e executar código Python, ideal para implementação e experimentação com algoritmos de RL.



Bibliografia

Bibliografia Básica

FACELI, Katti et al. Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina. Rio de Janeiro: LTC, 2021.

NETTO, Amilcar; MACIEL, Francisco. Python para Data Science e Machine Learning Descomplicado. Rio de Janeiro: Alta Books, 2021.

SÁ, Y. V. A. Desenvolvimento de aplicações IA: Robótica, Imagem e Visão Computacional. São Paulo: Platos, 2021.

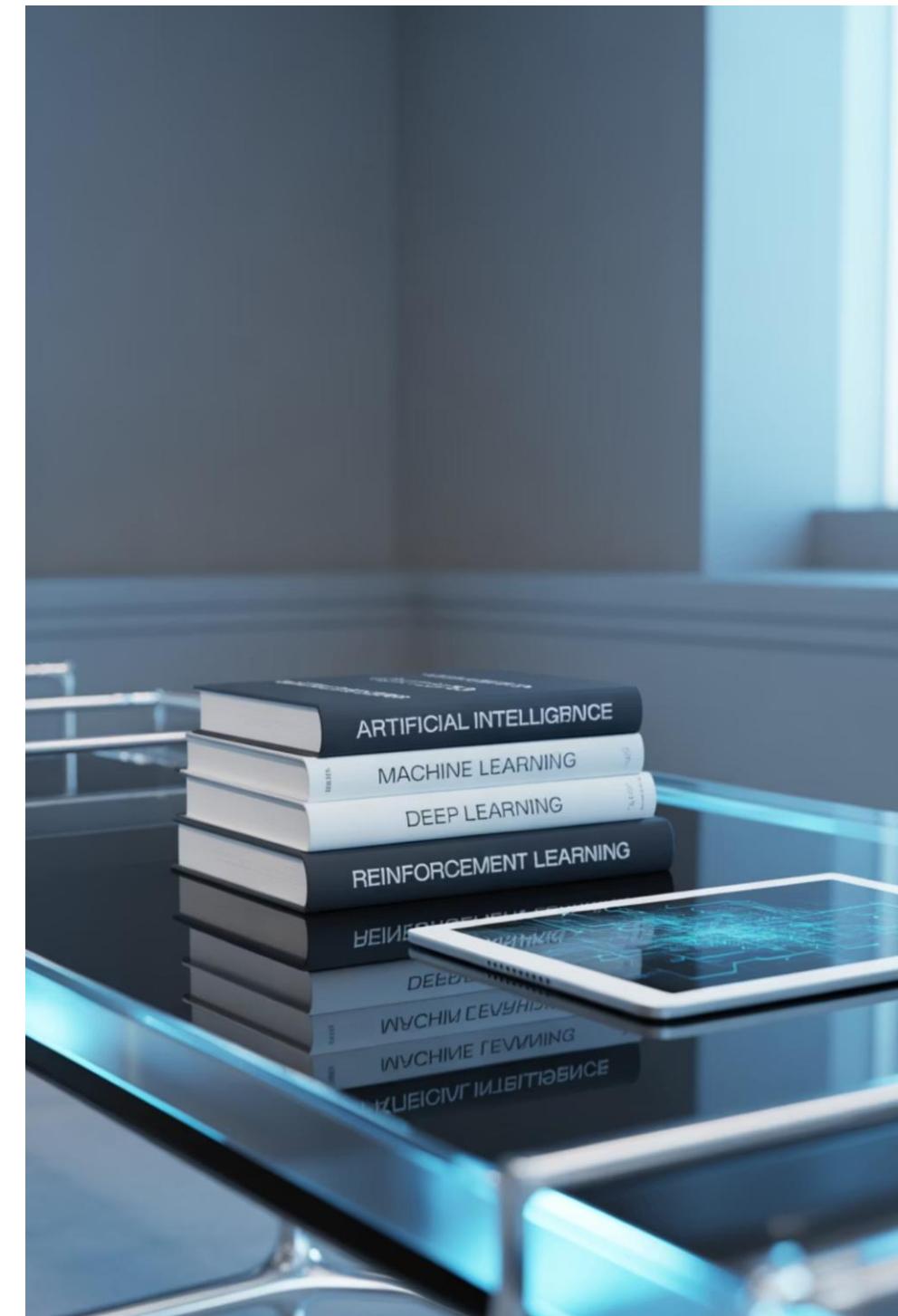
Bibliografia Complementar

SUTTON, Richard; BARTO, Andrew. Reinforcement Learning: An Introduction. Cambridge: MIT Press, 2018.

RUSSELL, Stuart J.; NORVIG, Pete. Inteligência Artificial: Uma Abordagem Moderna. Rio de Janeiro: GEN LTC, 2022.

SILVA, Fabrício M. et al. Inteligência artificial. Porto Alegre: SAGAH, 2018.

Além dos livros listados, compartilharemos ao longo do curso artigos científicos recentes, tutoriais online e recursos complementares para aprofundamento em tópicos específicos.



O Poder do Aprendizado por Reforço Profundo



Atari Games (2013)

A DeepMind revolucionou o campo com o DQN (Deep Q-Network), que aprendeu a jogar jogos Atari apenas a partir dos pixels da tela, superando humanos em vários jogos.



AlphaGo (2016)

Derrotou o campeão mundial de Go, Lee Sedol, um feito considerado impossível na época devido à complexidade do jogo (mais configurações de tabuleiro que átomos no universo).



Dexterous Manipulation (2018)

Sistemas que aprendem a manipular objetos com destreza semelhante à humana, desenvolvidos pela OpenAI e outros grupos de pesquisa.



AlphaFold (2020)

Revolucionou a biologia estrutural, prevendo com precisão a estrutura 3D de proteínas a partir da sequência de aminoácidos, um problema fundamental da ciência.

Estes avanços exemplificam como o Aprendizado por Reforço, especialmente quando combinado com redes neurais profundas, tem transformado diferentes áreas e resolvido problemas anteriormente considerados intratáveis.

Desafios Atuais no Aprendizado por Reforço

D

Eficiência de Amostra

Algoritmos tradicionais de RL necessitam de muitas interações com o ambiente para aprender políticas efetivas, tornando o treinamento demorado e computacionalmente custoso.

E

Exploração vs. Aproveitamento

Encontrar o equilíbrio ideal entre explorar novos estados e explorar conhecimento adquirido continua sendo um desafio fundamental.

T

Transferência de Conhecimento

Desenvolver agentes capazes de transferir aprendizado entre tarefas relacionadas, reduzindo a necessidade de treinamento do zero para cada novo problema.

A

Aplicações no Mundo Real

Transpor o sucesso dos algoritmos em ambientes simulados para aplicações práticas em sistemas reais, com suas incertezas e restrições físicas.

I

Interpretabilidade

Compreender e explicar as decisões tomadas pelos agentes, especialmente quando utilizam modelos complexos como redes neurais profundas.

S

Segurança e Robustez

Garantir que os agentes de RL atuem de forma segura e previsível, evitando comportamentos indesejados ou exploração de falhas nos sistemas.

AULAS	PLANO DE AULAS	Disciplina: IBM1957 – Business Lab
Dia/Mês	Conteúdo da Aula	Atividades de Apoio
11/fev	Apresentação da disciplina, plano de ensino, material de aula e modelos avaliativos Revisão Revisão de Probabilidades para Aprendizado por Reforço (Demonstração prática de Códigos em Jupyter Notebook a ser executada pelos alunos como tarefa para casa)	Jupyter Notebooks para Revisão Lista de Exercícios
13/fev	Introdução ao Aprendizado por Reforço aplicações e ferramentas. Conceitos básicos de aprendizado por reforço: agente, ambiente, política, função valor, função de recompensa.	Leituras: Sutton e Bartos Capítulo 1 Russel e Novig Capítulo 23.1 Lista de Exercícios 1
20/fev	O problema do k armed Bandits: Intuição e Conceituação (Dinâmica: de Estudo Médico)	Leituras: Sutton e Bartos Capítulo 2 Russel e Novig Capítulo 16.3 Lista de Exercícios 2
25/fev	O problema do k armed Bandits: Implementação em Python	Jupyter Notebook # 1 Lista de Exercícios 3
27/fev	O problema da Exploração e Aproveitamento O problema do k armed Bandits: ϵ -greedy (Implementação em Python)	Jupyter Notebook # 2 Lista de Exercícios 4
04/mar	Cadeias de markov: conceitos básicos e definição.	Leitura Russel e Novig Capítulo 16.3 Lista de Exercícios 5
11/mar	Cadeias de markov: Implementação em Python	Jupyter Notebook # 3 Lista de Exercícios 6
11/fev	Apresentação da disciplina, plano de ensino, material de aula e modelos avaliativos Revisão Revisão de Probabilidades para Aprendizado por Reforço (Demonstração prática de Códigos em Jupyter Notebook a ser executada pelos alunos como tarefa para casa)	Jupyter Notebooks para Revisão Lista de Exercícios
13/fev	Introdução ao Aprendizado por Reforço aplicações e ferramentas. Conceitos básicos de aprendizado por reforço: agente, ambiente, política, função valor, função de recompensa.	Leituras: Sutton e Bartos Capítulo 1 Russel e Novig Capítulo 23.1 Lista de Exercícios 1
20/fev	O problema do k armed Bandits: Intuição e Conceituação (Dinâmica: de Estudo Médico)	Leituras: Sutton e Bartos Capítulo 2 Russel e Novig Capítulo 16.3 Lista de Exercícios 2
25/fev	O problema do k armed Bandits: Implementação em Python	Jupyter Notebook # 1

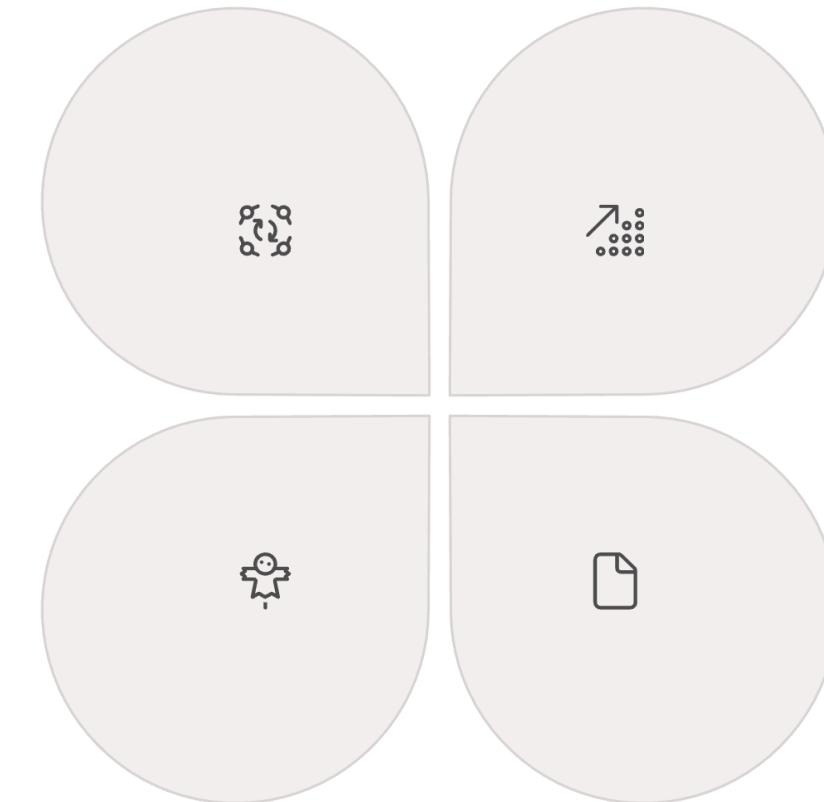
27/fev	O problema da Exploração e Aproveitamento O problema do k armed Bandits: ϵ -greedy (Implementação em Python)	Jupyter Notebook # 2 Lista de Exercicios 4
04/mar	Cadeias de markov: conceitos básicos e definição.	Leitura Russel e Novig Capítulo 16.3 Lista de Exercicios 5
13/mar	Aula Pática de Exercicios	
18/mar	Aula Pática de Exercicios	
20/mar	Processo de Decisão de Markov (MDP) Equações de Bellman :Intuição e Conceituação (Dinâmica Gridworld)	Sutton e Bartos Capitulo 3 Russel e Novig Capítulo 16.1 e 16.3 Lista de Exercicios 7
25/mar	Programação Dinâmica: Intuição e Conceituação	Leituras: Sutton e Bartos Capitulo 4 Russel e Novig Capítulo 16.1 e 16.3 Lista de Exercicios 7
27/mar	Programação Dinâmica: Implementação em Python	Jupyter Notebook # 4 Lista de Exercicios 8
01/abr	Programação Dinâmica: Implementação em Python	Jupyter Notebook # 5 Lista de Exercicios 9
08/abr	Aula Pática de Exercicios	
10/abr	Revisão para AP	
15/abr	Período Avaliação Parcial AP 1	
17/abr	Período Avaliação Parcial AP 1	
22/abr	Devolutiva AP1 (Vista de Prova) Introdução ao Método de Monte Carlo	Leituras: Sutton e Bartos Capitulo 5 Russel e Novig Capítulo 23 Lista de Exercicios 10

		Leituras: Sutton e Bartos Capítulo 5 Russel e Novig Capítulo 23 Lista de Exercícios 11
24/abr	Métodos de Monte Carlo em Aprendizado por Reforço: Predição, Controle, On Policy OFF Policy	Jupyter Notebook # 6 Lista de Exercícios 12
29/abr	Métodos de Monte Carlo em Aprendizado por Reforço: Implementação	
06/mait	Aula Pática de Exercicios	
08/mai	Métodos de Diferença Temporal Introdução e Conceitos	Leituras: Sutton e Bartos Capítulo 6 Russel e Novig Capítulo 23 Lista de Exercícios 13
13/mai	Métodos de Diferença Temporal : Q Learning Implementação	Jupyter Notebook # 7 Lista de Exercícios 14
15/mai	Métodos de Diferença Temporal : SARSA Implementação	Jupyter Notebook # 8 Lista de Exercícios 15
20/mai	Métodos de Diferença Temporal : Expected SARSA Implementação	Jupyter Notebook # 8 Lista de Exercícios 15
22/mai	Aula Pática de Exercicios	
03/jun	Tópicos Avançados: <ul style="list-style-type: none">• Métodos baseados em política (Policy Gradient);• Aprendizado em espaço contínuo;• Aprendizado com múltiplos agentes; Aplicações em jogos, robótica e sistemas autônomos..	Leituras: Sutton e Bartos Capítulo 16 Russel e Novig Capítulo 23.7
10/jun	Revisão para AP2 Correção AC4 e Listas de Exercícios	
12/jun	Período de Aplicação a AP2	
17/jun	Período de Aplicação a AP2	
19/jun	Vista de Prova AP2	
24/jun	Período de Aplicação da AS	

Vamos Começar Nossa Jornada!

Comunidade Ativa

O campo do Aprendizado por Reforço possui uma comunidade vibrante de pesquisadores e praticantes, com conferências, workshops e fóruns dedicados.



Oportunidades Profissionais

Habilidades em Aprendizado por Reforço são altamente valorizadas no mercado, abrindo portas para carreiras em pesquisa, desenvolvimento e inovação.

Crescimento Acelerado

A área está em rápida evolução, com novos algoritmos e aplicações surgindo constantemente, oferecendo inúmeras oportunidades para contribuições.

Interdisciplinaridade

O RL conecta-se com diversas áreas como neurociência, psicologia, economia, controle ótimo e matemática aplicada.

"A inteligência é a capacidade de se adaptar à mudança." - Stephen Hawking

Estamos animados para explorar com vocês os fundamentos e as fronteiras do Aprendizado por Reforço neste semestre! Vamos juntos desenvolver as habilidades necessárias para criar os sistemas inteligentes do futuro.

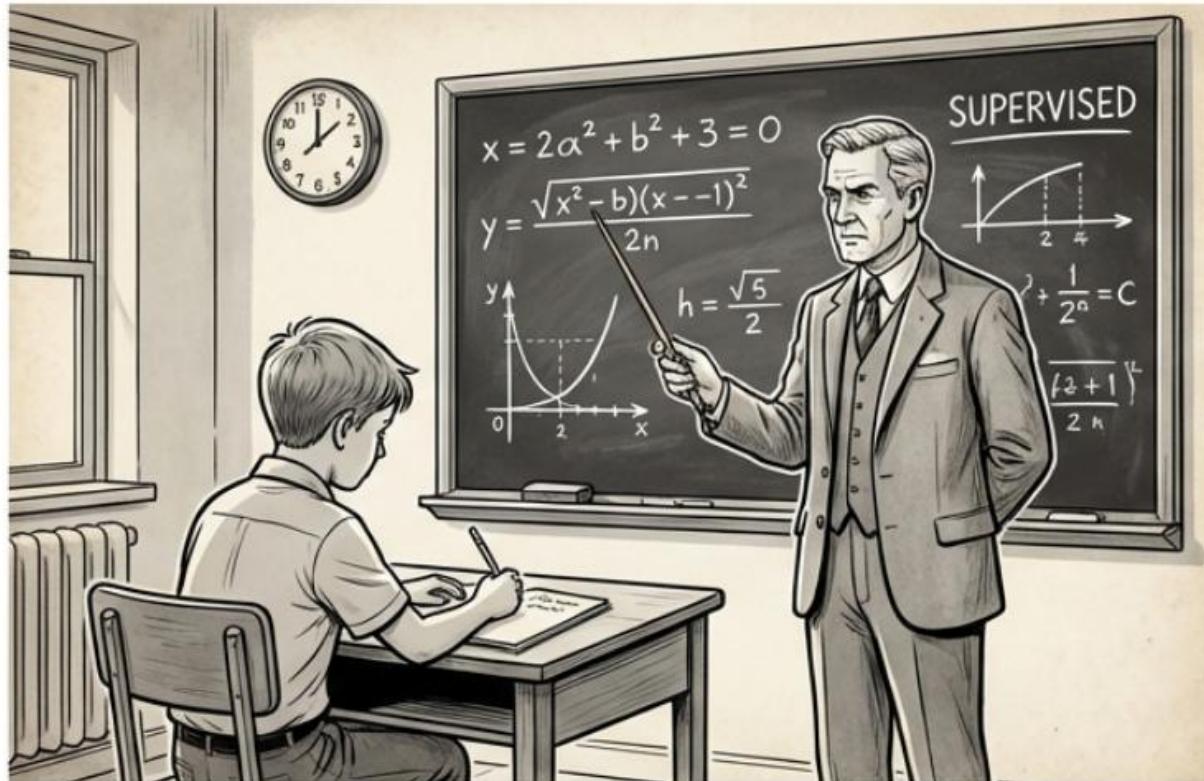
Aprendizado por Reforço: Uma Abordagem Computacional

Do erro à excelência: como agentes aprendem interagindo com o mundo.



BASEADO NA OBRA DE SUTTON & BARTO

A Natureza do Aprendizado



InSTRUÇÃO (Supervisionado)

A ideia fundamental é que aprendemos interagindo com o nosso ambiente.

- Quando um bebê brinca, não há um professor explícito. Existe apenas uma conexão sensório-motora direta.
- Exercitar essa conexão produz informações sobre causa e efeito e as consequências das ações.



Interação (Reforço)

Key Insight: O Aprendizado por Reforço é focado no aprendizado orientado a objetivos através da interação, distinto de outras abordagens que dependem de supervisão externa.

 NotebookLM



Onde o Aprendizado por Reforço se Encaixa



Aprendizado Supervisionado

Aprende a partir de exemplos rotulados (o professor).
Extrapolia o conhecimento.
Falha em territórios desconhecidos onde não há exemplos.



Aprendizado Não Supervisionado

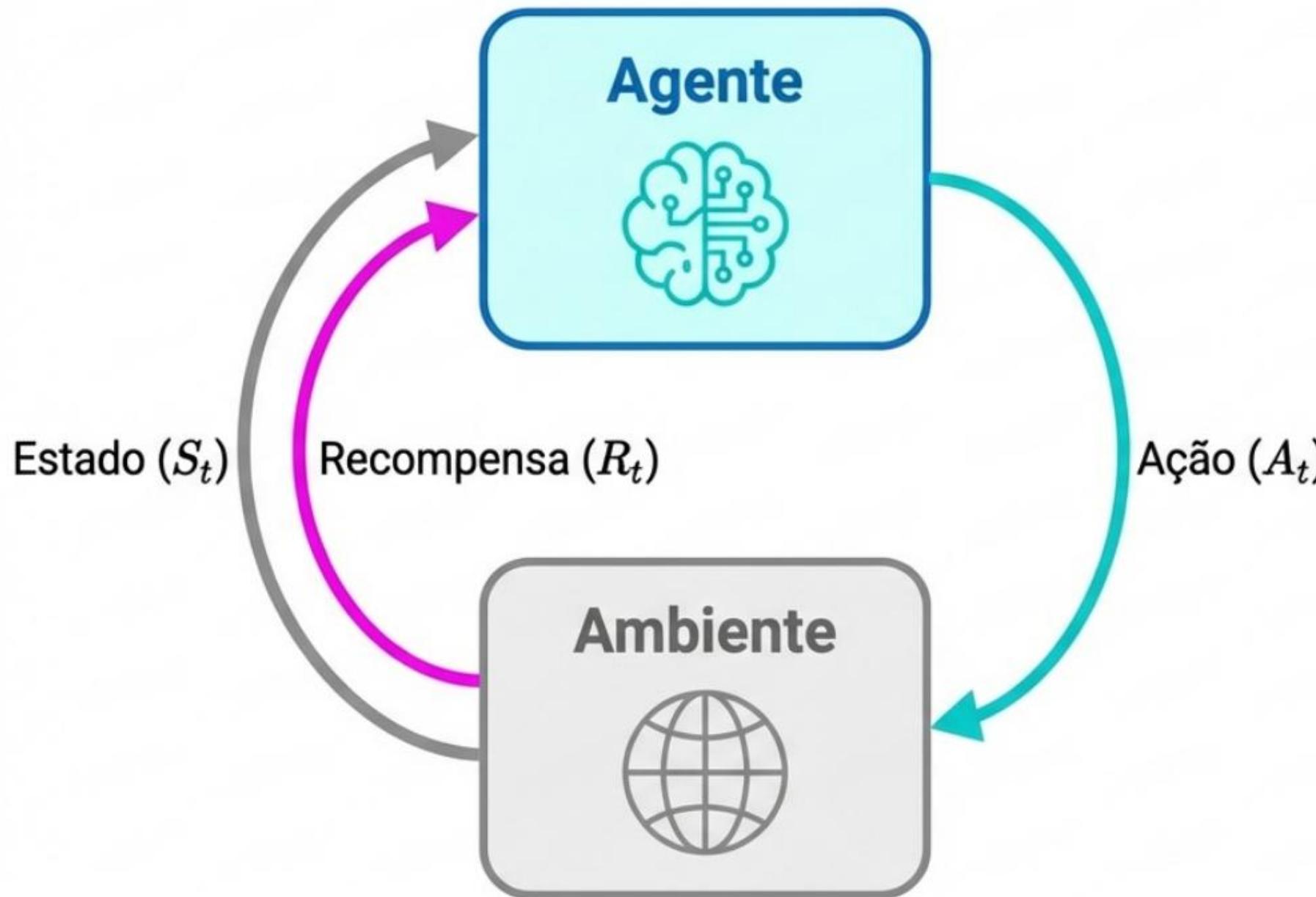
Busca estruturas ocultas em dados não rotulados. Útil para clustering, mas não maximiza recompensas.



Aprendizado por Reforço (RL)

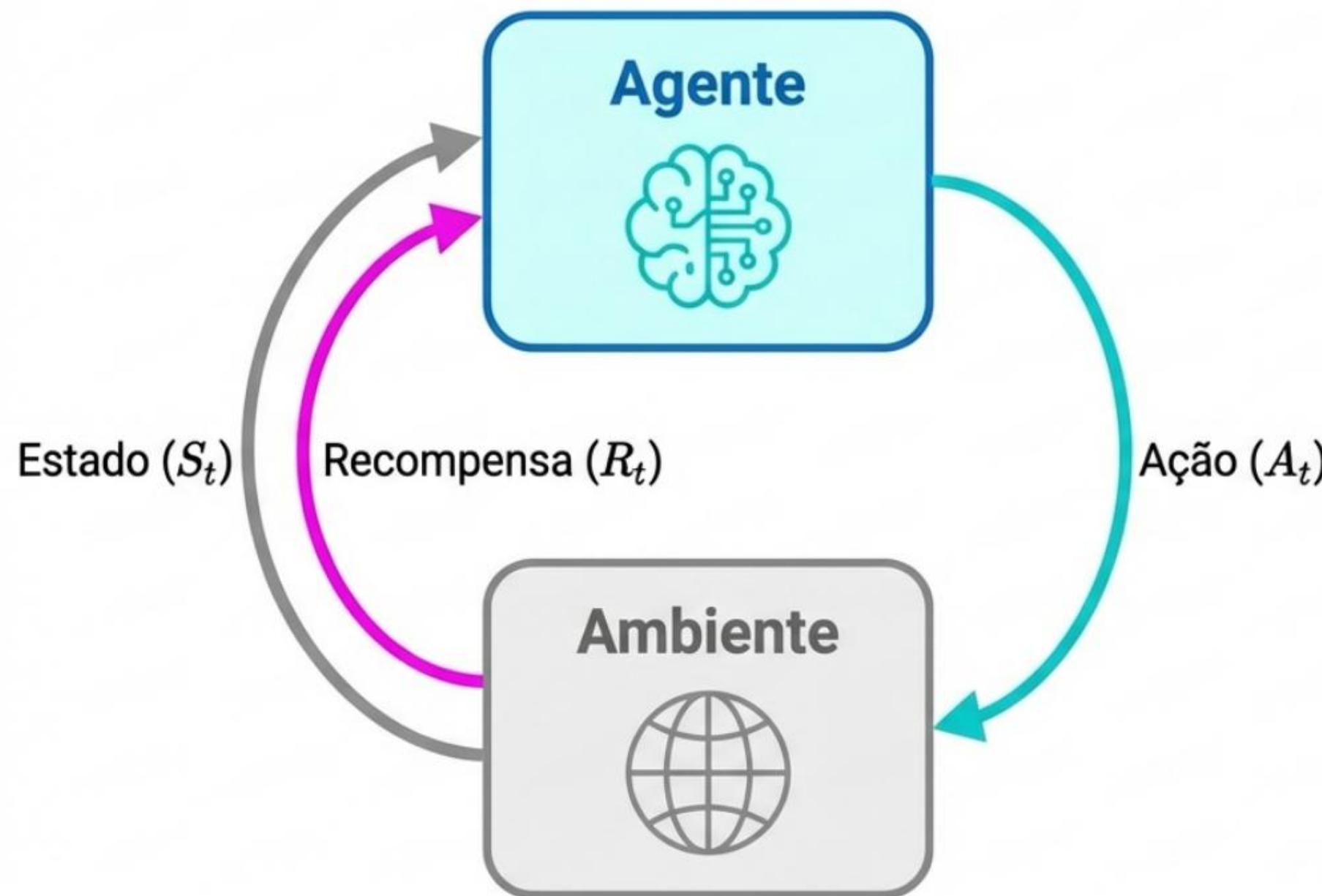
O paradigma da descoberta. O agente não recebe o gabarito; ele descobre quais ações geram recompensa através da tentativa e erro.

O Ciclo de Interação (Agente-Ambiente)



- Utilizamos Processos de Decisão de Markov (MDPs).
- O agente sente o estado (S_t) e toma uma ação (A_t).
- O ambiente reage com um novo estado (S_{t+1}) e uma recompensa (R_{t+1}).
- Objetivo: Maximizar a soma total de recompensas.

O Ciclo de Interação (Agente-Ambiente)



- Utilizamos Processos de Decisão de Markov (MDPs).
- O agente sente o estado (S_t) e toma uma ação (A_t).
- O ambiente reage com um novo estado (S_{t+1}) e uma recompensa (R_{t+1}).
- Objetivo: Maximizar a soma total de recompensas.



Elemento 1: A Política (π)

O Cérebro do Agente

Estado Percebido	Ação Escolhida
<ul style="list-style-type: none">• Obstáculo à frente• Caminho livre• Bateria fraca	<ul style="list-style-type: none">• Virar à esquerda• Acelerar• Buscar carregador

- A política define o comportamento do agente.
- É um mapeamento de situações para ações.
- Pode ser uma simples tabela ou uma rede neural complexa.
- A política sozinha é suficiente para determinar como o agente age.



Elemento 2: O Sinal de Recompensa (R\$)

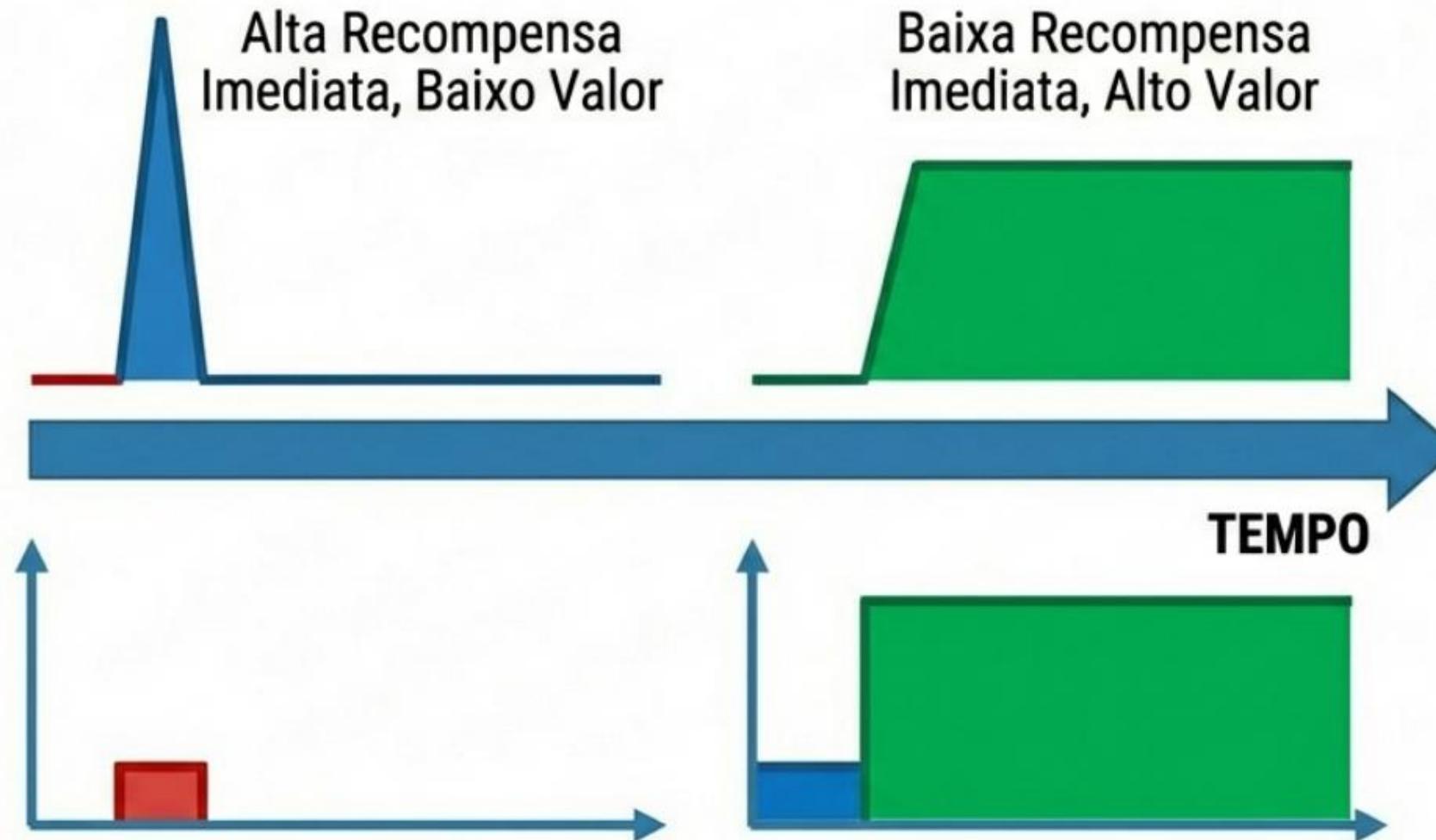


- Define o objetivo imediato. É o feedback de curto prazo.
- Análogo biológico: Prazer (+1) ou Dor (-1).
- O agente altera sua política se as ações resultam em recompensa baixa.
- Atenção: Indica o que é bom *agora*, não necessariamente o que é bom para o futuro.



Elemento 3: A Função de Valor (\$V)

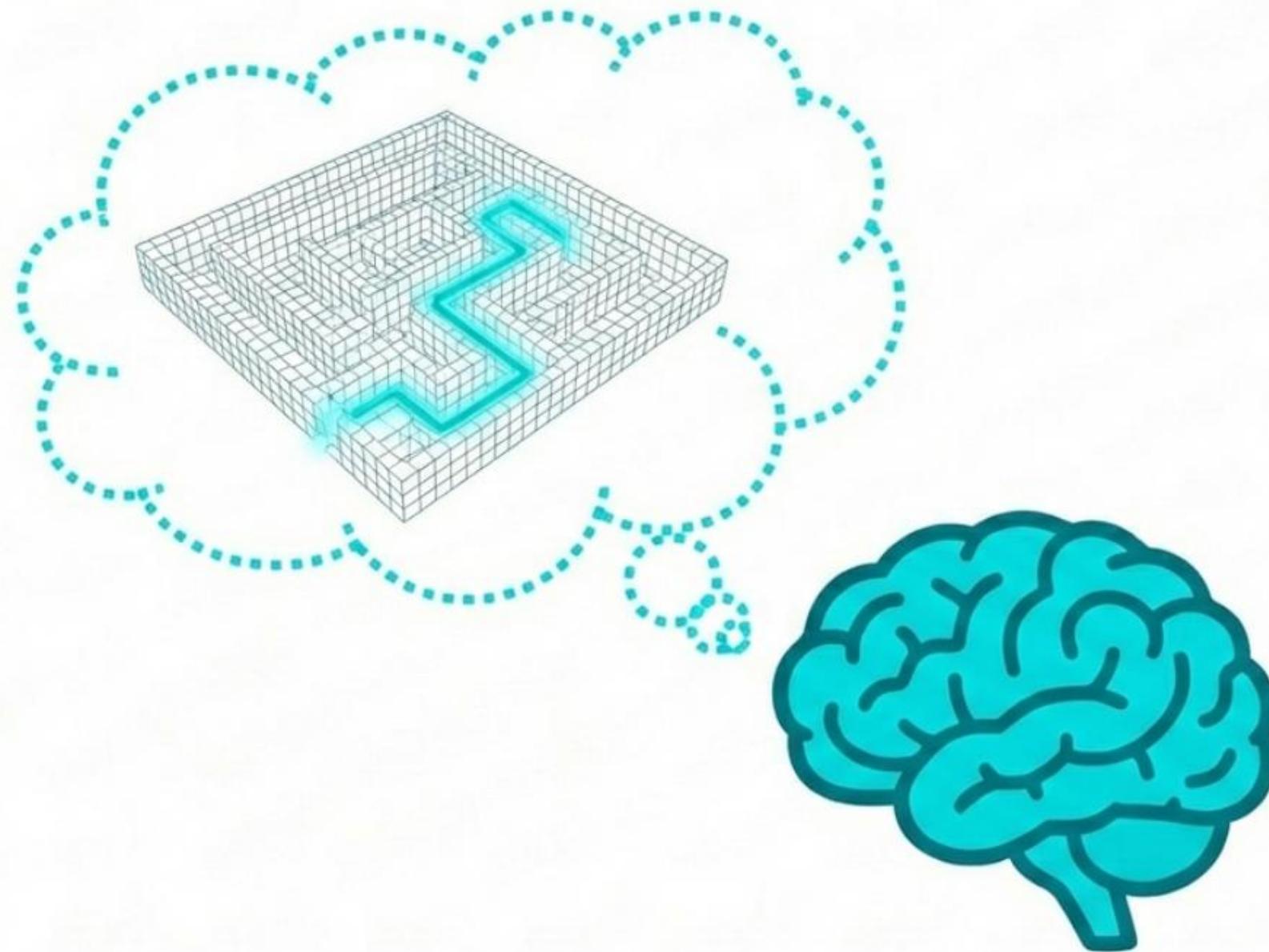
A Visão de Longo Prazo



- O **Valor** de um estado é a quantidade total de recompensa que esperamos acumular no futuro a partir dele.
- Recompensa é primária (imediata). Valor é secundário (predição).
- Buscamos estados de alto Valor, pois eles garantem o sucesso a longo prazo.



Elemento 4: O Modelo do Ambiente



- O modelo mimetiza o comportamento do ambiente.
- Permite **Planejamento**: considerar possíveis cenários futuros antes de agir.
- Model-Based (Com Modelo): Planeja e pensa à frente.
- Model-Free (Sem Modelo): Aprende puramente por tentativa e erro (como o exemplo do Jogo da Velha).

A Matemática da Atualização (Temporal Difference)

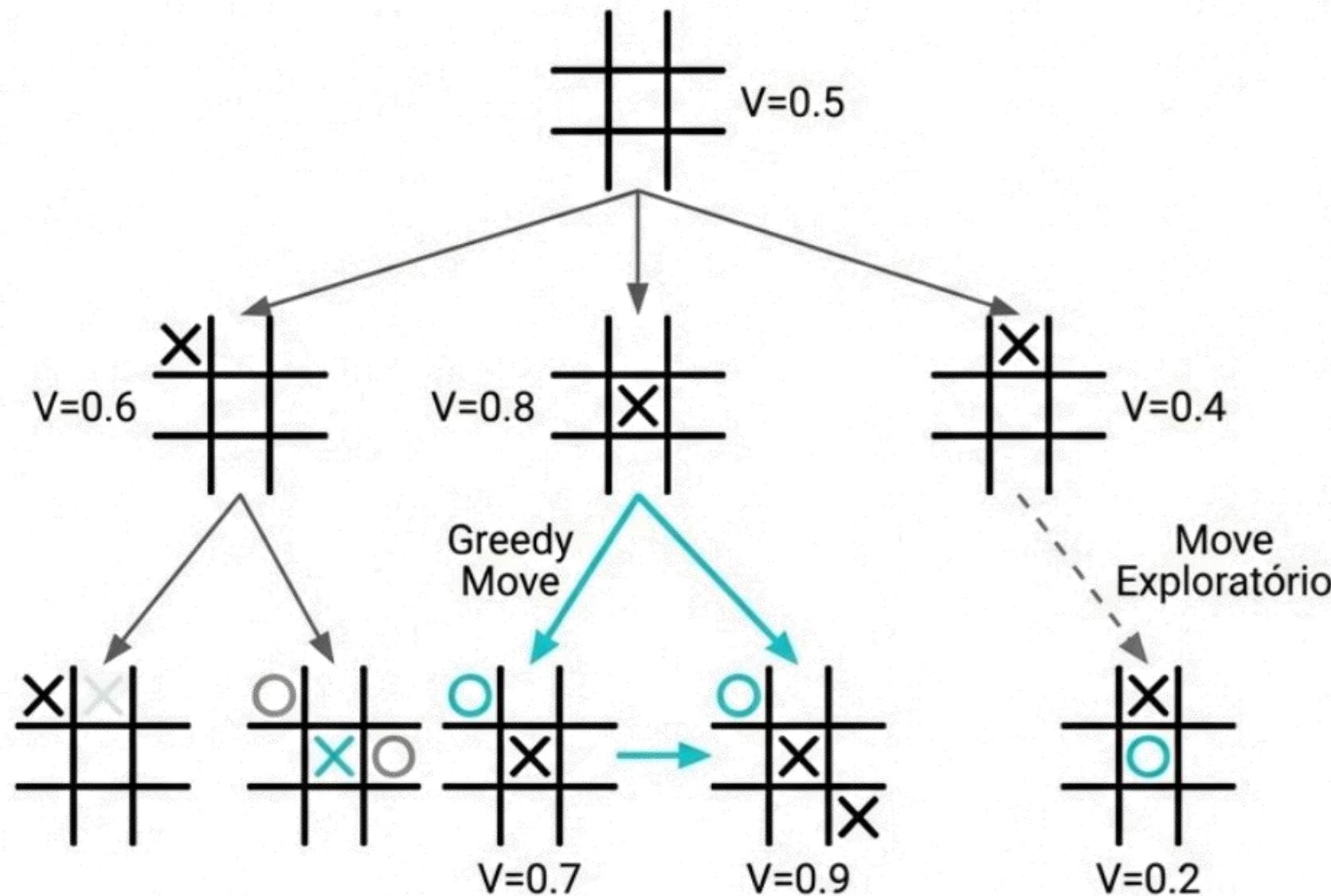
$$V(S_t) \leftarrow V(S_t) + \alpha[V(S_{t+1}) - V(S_t)]$$

Diagram illustrating the components of the Temporal Difference update rule:

- Taxa de Aprendizado (Step-size): α
- Estimativa Atual: $V(S_t)$
- Próximo Estado Real (Melhor estimativa): $V(S_{t+1})$
- Erro de Predição (Surpresa): $V(S_{t+1}) - V(S_t)$

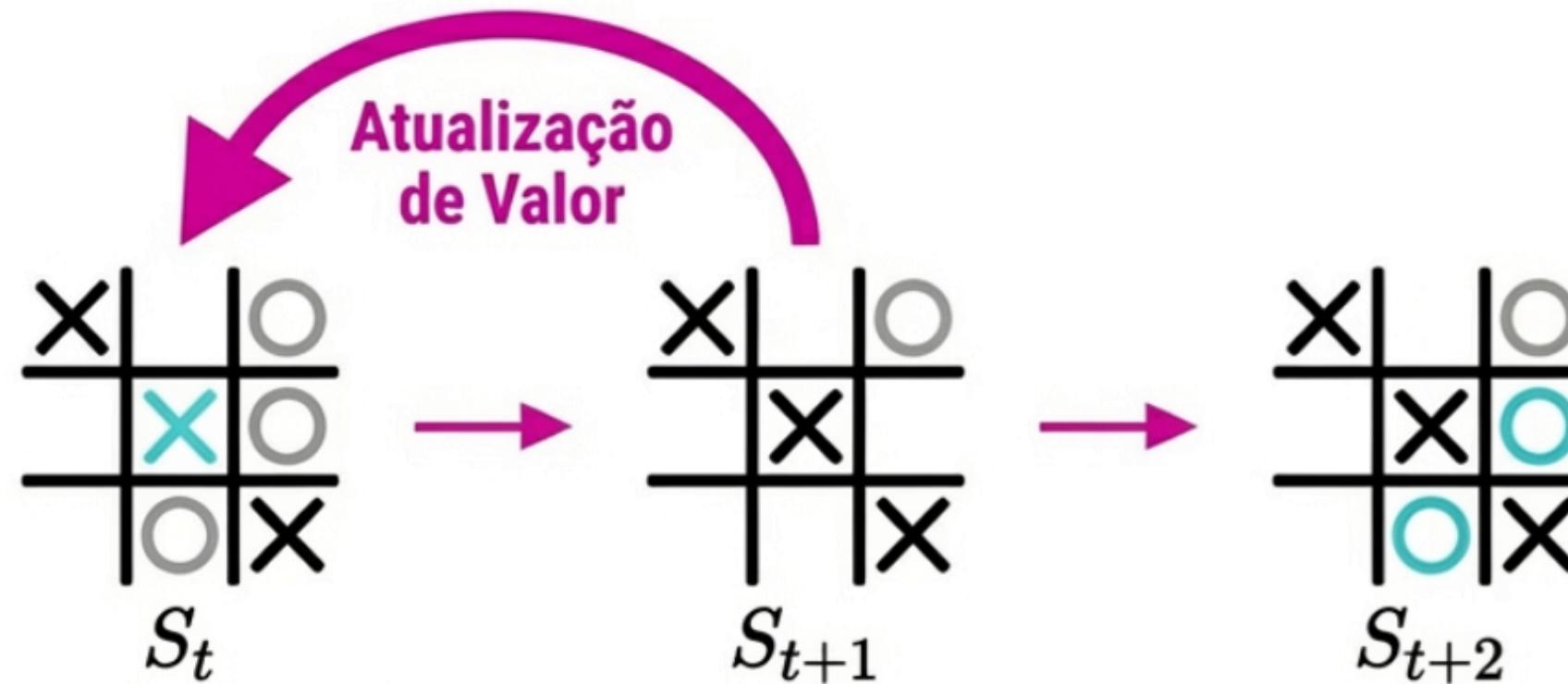
O agente corrige sua estimativa anterior baseando-se na nova informação recebida após a ação.

Exemplo Prático: Jogo da Velha



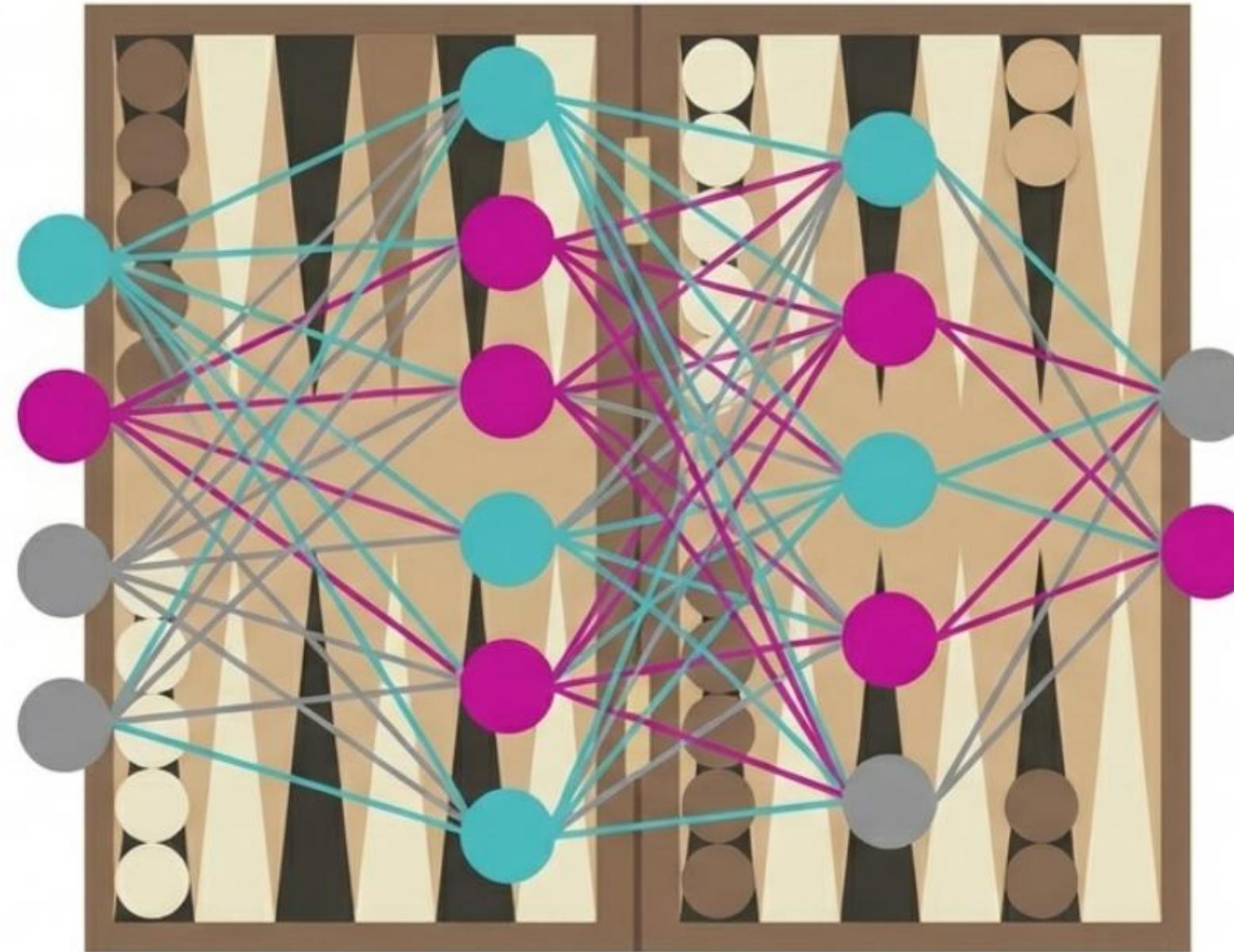
- Neste exemplo, cada estado do tabuleiro tem um Valor (probabilidade de vitória).
- **Movimento Gulosos (Greedy):** Escolhemos o estado com maior valor atual.
- **Exploração:** Ocasionalmente escolhemos aleatoriamente para descobrir novas táticas.

Aprendendo Durante o Jogo (Back-up)



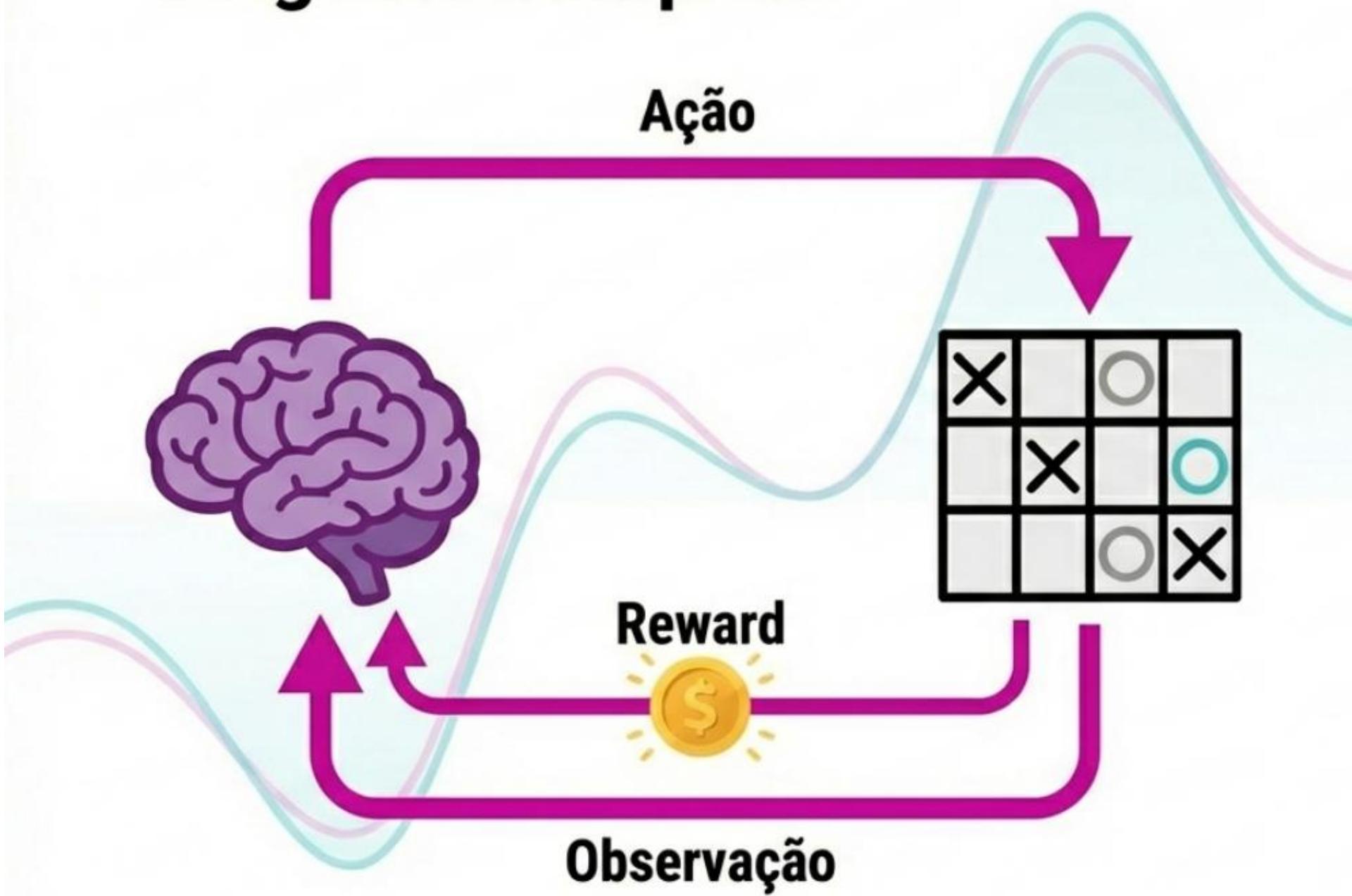
- Ao fazermos um movimento e observarmos o resultado (S_{t+1}), atualizamos nossa estimativa do estado anterior (S_t).
- O valor de S_t é ajustado para ficar mais próximo do valor de S_{t+1} .
- Com o tempo, essa propagação para trás ensina ao agente quais movimentos iniciais levam à vitória.

Além das Tabelas: Generalização



- Tabelas funcionam para jogos pequenos (Tic-Tac-Toe).
- Para problemas reais (Gamão: 10^{20} estados), a tabela não cabe na memória.
- Usamos **Redes Neurais** para generalizar: o agente aprende a reconhecer padrões em estados novos baseando-se em experiências passadas.

O Agente Completo



- O Aprendizado por Reforço une sensação, ação e objetivo em um único framework.
- Não é apenas sobre classificar dados, mas sobre interagir e sobreviver.
- Começa com um agente completo interagindo com um mundo incerto.



IBMEC.BR

/IBMEC

IBMEC

@IBMEC_OFICIAL

@IBMEC

ibmec