Deep Reinforcement Learning - Jogos

Universidade do Minho, Departamento de Informática
Braga, Portugal

Abstract. Deep Reinforcement Learning tem vindo a permitir que sejam feitos imensos avanços tecnológicos que mostram a importância de Aprendizagem Máquina e o quão dependentes estamos dela. Tal como referido por [15], este pode ser considerado o mais importante tipo de Machine Learning por conseguir resolver uma grande variedade de problemas complexos e de tarefas que antes não estavam ao alcance de máquinas treinadas à semelhança da inteligência humana.
O presente trabalho de investigação visa a abordar o funcionamento e vantagens do uso de *Deep Reinforcement Learning* na área de Jogos. Serão analisados vários exemplos de investigação onde estes algoritmos tiveram um grande impacto na evolução e desenvolvimento destes jogos e na eficiência com que são jogados.

Keywords: Deep Reinforcement Learning · Aprendizagem Profunda · Jogos · AlphaGo.

1 Introdução

À medida que Jogos evoluem e se tornam mais complexos, também a necessidade para procurar soluções automatizadas cresce uma vez que testes feitos por pessoas deixam se ser suficientes ou até mesmo possíveis por não serem práticos ou por serem demasiado custosos [3].

Em anos recentes, tecnologias de aprendizagem que processam grandes densidades de dados usando-os para aprender, vieram substituir em algumas áreas os antigos sistemas que eram programados baseando-se em regras. Uma vez que existem jogos que usam milhares de dados em mapas com tamanhos que podem ser medidos em kilómetros quadrados de tamanho tornou-se imperativo aplicar estes métodos de aprendizagem a esta área.

Tal como mencionado por [9], surgiram diversas tecnologias que permitiram fazer avanços em diversas áreas que até então estavam subdesenvolvidas. Por um lado, Deep Learning permitiu fazer grandes avanços em várias áreas da computação, enquanto que Deep Reinforcement Learning (DRL) ajudou exponencialmente agentes a fazerem decisões corretas e assim melhorando a generalização e escalabilidade de paradigmas de aprendizagem por reforço clássicos. Estes modelos de aprendizagem por reforço trouxeram a possibilidade de complementar as atuais soluções de aprendizagem, aprendendo diretamente ao jogar sem a necessidade de intervenção humana [3].

Segundo [14], diversos algoritmos de RL, nomeadamente Q-learning mostraram que, quando combinados com redes neuronais convolucionais, poderiam aprender a jogar diversos jogos a uma capacidade sobre-humana. Como mencionado por [13], há data de escrita do artigo, havia exemplos de desafios não só para o intelecto humano mas também para sistemas inteligentes por requererem estudos precisos e sofisticados (não conseguindo ainda atingir níveis humanos). O caso mencionado seria o do jogo de tabuleiro Go, que apenas em tempos mais recentes foi atingida uma solução de qualidade graças ao aparecimento do AlphaGo.

2 Deep Reinforcement Learning - DRL

Machine Learning (ou aprendizagem máquina) é uma área da inteligência artificial que recorre a diversos modelos matemáticos para tentar fazer previsões.

Esta área de estudo pode ser dividiva em três abordagens: supervisionada, não supervisionada e por reforço. Tendo cada uma destas as suas vantagens e desvantagens mediante o problema em que são inseridas. Segundo [5], pode-se definidir estas abordagens da seguinte maneira:

- **Aprendizagem Supervisionada**: o treino é feito a partir de um conjunto de dados de input e output e o objetivo é conseguir encontrar uma regra geral que permita encontrar a resposta correta para todos os inputs. Pode ser dividida em Classificação e Regressão.
- **Aprendizagem Não Supervisionada**: não é fornecida nenhuma orientação, sendo a aprendizagem realizada pela descoberta de características semelhantes nos dados de entrada. Um exemplo deste tipo de aprendizagem é Clustering.
- **Aprendizagem por Reforço**: ao trabalhar para chegar a um dado objetivo, recebe "recompensas" ou "punições" mediante os resultados que obtém de modo a encorajar a aprendizagem. Um exemplo do uso desta aprendizagem é o AlphaGo. Algoritmos de aprendizagem por reforço que podem ser referidos são Q-learning, temporal-difference learning e **Deep Reinforcement Learning**.

Sabendo onde se insere o caso de estudo deste documento pode-se agora desenvolver este tipo de aprendizagem.

Tal como referido anteriormente, um algoritmo de aprendizagem por reforço aprende a resolver problemas complexos por tentativa-erro. A máquina é treinada baseando-se em vários cenários de modo a conseguir tomar várias decisões acertadas e assim maximizar a quantidade de recompensas que recebe. Assim, por Deep Reinforcement Learning (ou Aprendizagem por Reforço Profunda) entende-se o uso de várias camadas de uma rede neuronal que funcionem à semelhança do cérebro humano e que apliquem o modelo de Reinforcement Learning. Segundo [2], alguns componentes envolvidos na criação de uma rede funcional podem ser definidos como:

- **Agente**: Toma ações que afetam o ambiente em que está inserido.
- **Ação**: Conjunto de todas as operações ou movimentos possíveis. O agente escolhe de um conjunto de ações possíveis, a melhor ação a executar.
- **Ambiente**: Todas as ações feitas pelos agentes afetam diretamente o ambiente em que estão inseridos, sendo este quem lhes devolve as recompensas e um estado novo por cada ação.
- **Estado**: Situação em que o agente se encontra.
- **Recompensa (R)**: O ambiente devolve feedback em cada estado que permite ao agente validar as suas ações.
- **Fator de desconto**: Tem influência sobre a importância das recompensas.
- **Política (π)** : Pelo qual o agente se rege para tomar as várias decisões e maximizar as recompensas.
- **Valor (V)**: Mede a optimalidade de um dado estado.

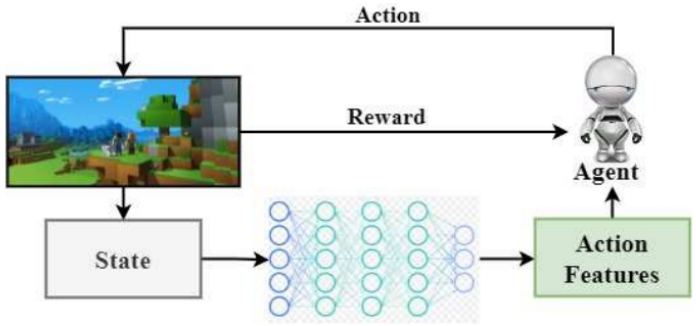

Fig. 1. [9] Componentes envolvidos na criação de uma rede funcional

A aprendizagem por reforço profunda é atualmente usada para vários fins e tem tendência a ser usada em cada vez mais áreas de estudo e trabalho. Tal como enumerado em [2] destacam-se:

- Indústria da manufatura na qual se recorre cada vez mais à robótica e à aprendizagem profunda.
- Carros inteligentes.
- Mercado financeiro, nomeadamente aplicado a stocks.
- Saúde.
- **Jogos e Videojogos**.

O presente trabalho tem assim como foco o último tema mencionado.

3 Análise de Estado de Arte

O objetivo a longo prazo da Inteligência Artifical (IA) pode dizer-se ser a resolução de problemas avançados e desafiantes do mundo real. Os jogos têm servido

como "pequenos degraus" num caminho que já começou a ser percorrido pela IA há algumas décadas. Os modelos de Deep Reinforcement Learning, têm evoluído bastante nos últimos anos e já são capazes de efetuar várias tarefas na área da robótica, no processamento de texto e em diversos jogos e jogos de vídeo.[4]

De modo a melhor compreender a aplicação prática de Deep Reinforcement Learning no domínio em estudo, apresenta-se de seguida uma investigação de potenciais casos e posteriormente de experiências existentes na literatura científica juntamente com uma tentativa de avaliação de cada um deles.

3.1 AlphaGo

AlphaGo é um programa desenvolvido pela empresa DeepMind da Google para jogar um jogo originário da China de há mais de 3000 anos denominado Go.

Segundo [8], o jogo Go é conhecido como um dos jogos clássicos mais desafiantes para a Inteligência Artificial devido à sua elevada complexidade, embora as regras possam parecer simples. Existe um impressionante total de 10^{170} configurações possíveis para disposições de todas as peças no tabuleiro - mais do que o número de átomos que existem no universo.

O AlphaGo tornou-se no primeiro sistema a derrotar o campeão mundial do jogo Go e já passou por diversas versões, de entre as quais se destacam as seguintes:

- **AlphaGo Fan:** A versão original do AlphaGo é também conhecida por AlphaGo Fan uma vez que foi a primeira vez que um programa de computador conseguiu derrotar um jogador profissional de Go, o campeão europeu Fan Hui em 2015. Esta versão do AlphaGo utiliza uma combinação de *Deep Neural Networks* (DNNs) e do algoritmo Monte Carlo Tree Search. Segundo [12], utilizando este algoritmo, o programa AlphaGo conseguiu obter uma percentagem de vitórias de 99.8% em confronto direto com outros programas semelhantes e, como dito anteriormente, derrotou o detentor do título europeu. Esta versão do AlphaGo utiliza uma **combinação de supervised e unsupervised learning**. A componente de supervised learning baseia-se no treino de uma rede neuronal de forma a prever a próxima jogada, através de dados obtidos de jogadores humanos experientes. A componente de unsupervised learning, baseia-se na realização de jogos de Go do programa com ele próprio. Os resultados dos jogos são, no final, utilizados para atualizar a rede neuronal.
- **AlphaGo Lee:** A segunda versão do AlphaGo é também conhecida por AlphaGo Lee, uma vez que, à semelhança da primeira versão do mesmo, também derrotou um jogador profissional de Go, o campeão mundial Lee Sedol. Esta segunda versão do AlphaGo é semelhante à primeira versão em termos de arquitetura. No entanto, foram efetuadas diversas melhorias quer ao nível das redes neuronais, quer ao nível do algoritmo de procura.
- **AlphaGo Zero:** O AlphaGo Zero é a primeira versão do sistema AlphaGo a ser treinado única e exclusivamente atráves de *reinforcement learning* em que o sistema aprende jogando contra ele próprio.

Este último será desenvolvido de seguida.

AlphaGo Zero [13]:

AlphaGo Zero utiliza um nova DNN que recebe como input uma representação do tabuleiro de jogo com a posição e o histórico das mesmas. O output obtido é um tuplo com as probabilidades de selecionar uma jogada (incluindo a jogada de passar a vez). É estimada também a probabilidade de o jogador ganhar o jogo caso escolha uma certa jogada, ou seja, a partir do estado em que o tabuleiro se encontrará após efetuar a mesma.

A rede neuronal deste sistema é treinada a partir de jogos contra o próprio sistema através de um algoritmo de *reinforcement learning*.

A cada estado do tabuleiro é executado o algoritmo Monte Carlo Tree Search (MCTS), utilizando a última rede neuronal. As jogadas são selecionadas de acordo com a probabilidade π calculada pelo algoritmo mencionado.

A atualização dos parâmetros têm também em conta a minimização do erro entre o vencedor esperado e o vencedor real do jogo. Os novos parâmetros são utilizados na próxima iteração do jogo que ele efetua com ele próprio.

Os parâmetros são ajustados com **gradiente descendente** aplicado a uma loss function, que tem um parâmetro de regularização cujo objetivo é diminuir o overfitting.

A cada iteração, a performance do sistema vai aumentando, à semelhança da qualidade dos jogos contra o próprio sistema, o que leva a que as DNNs se tornem mais precisas e que o sistema se torne uma versão ainda mais poderosa dele mesmo [6].

Esta técnica enunciada já não se encontra restringida aos limites do conhecimento humano e possui assim a possibilidade de aprender a jogar o jogo Go tornando-se o "melhor jogador de Go do mundo: o próprio AlphaGo"[6].

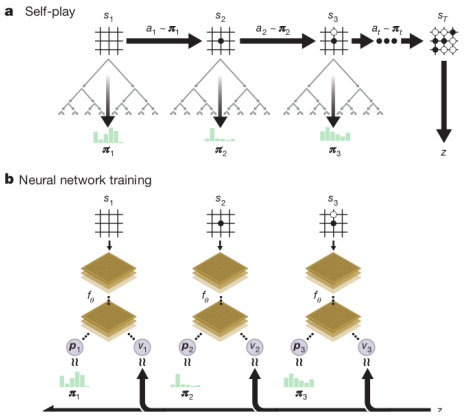

Fig. 2. Esquema de funcionamento do AlphaGo Zero

Acima apresenta-se um esquema de como o AlphaGo Zero funciona, para se compreender melhor os métodos enunciados anteriormente.

Iremos agora enumerar algumas métricas de aprendizagem interessantes e que ajudam a perceber mais facilmente a capacidade que este sistema apresenta:

- **Com apenas 3 horas de treino:** O AlphaGo Zero atingiu o nível de um jogador humano iniciante. Ainda executa muitos movimentos "gulosos" (*greedy*) prejudicando assim uma estratégia mais proveitosa a longo termo.
- **Após 19 horas de treino:** Conseguiu aprender as bases do jogo e algumas estratégias mais avançadas. **Após 36 horas de treino:** O sistema AlphaGo Zero ultrapassa o sistema AlphaGo Lee em termos de performance.
- **Após 70 horas de treino:** Atinge um nível sobre-humano a jogar Go. Apresenta uma estratégia de jogo disciplinada.

Podemos constatar os factos enunciado em cima através do primeiro gráfico da imagem seguinte.

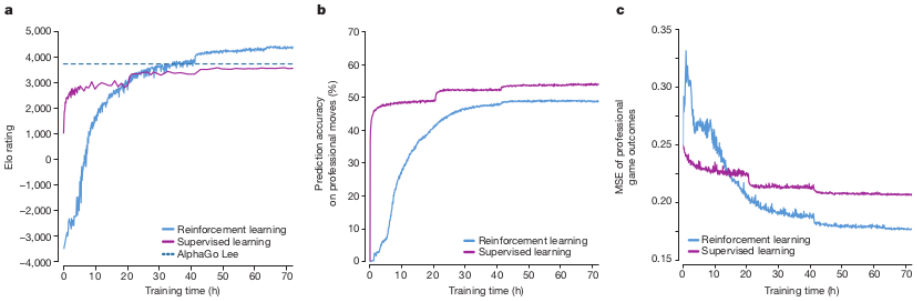

Fig. 3. Avaliação empírica do Sistema AlphaGo Zero

Na imagem 3a, é feita a comparação da performance do sistema AlphaGo Zero com o AlphaGo Lee e um outro modelo treinado usando aprendizagem supervisionada. Podemos constatar que o AlphaGo Zero rapidamente ultrapassou os outros sistemas. No entanto, é de notar que nas primeiras horas de treino o modelo treinado usando aprendizagem supervisionada apresenta uma melhor performance uma vez que o AlphaGo Zero começa a treinar sem qualquer conhecimento do jogo, apenas com as regras do mesmo. Na imagem 3b, é apresentada a percentagem de jogadas previstas. Notamos que o AlphaGo Zero apresenta uma menor percentagem de acerto no que toca a esta métrica, no entanto, apresenta melhor desempenho no que toca a derrotar um jogador treinado por humanos nas primeiras 24 horas. Isto sugere que o sistema AlphaGo Zero pode estar a aprender uma nova estratégia de jogo totalmente diferente das estratégias observadas em humanos até aos dias de hoje, mostrando assim a sua criatividade [12].

Estes resultados deixam-nos a pensar no que sistemas baseados em Deep Reinforcement Learning podem atingir no futuro dada a performance tão alta

que foi atingida num jogo tão complexo como o Go. Sistemas como este fazem agora acreditar que podem impactar a humanidade de forma positiva no que toca aos seus problemas mais desafiantes e impactantes.

3.2 AlphaZero

AlphaZero é um sistema baseado em Deep Reinforcement Learning, à semelhança do sistema AlphaGo Zero analisado anteriormente em mais detalhe. No entanto, difere deste na medida em que aprende a jogar xadrez, shoggi e Go através da realização de jogos com ele próprio começando sem qualquer informação dos mesmos para além das suas respetivas regras. Começa, assim, à semelhança do AlphaGo Zero, o treino de forma aleatória.

Diversos jogadores profissionais olham com bons olhos para sistemas como este pois podem assim estudar novas estratégias e aprender através dos mesmos. Segundo Garry Kasparov, considerado por muitos o melhor jogador de xadrez de todos os tempos, em [7],

> "The implications go far beyond my beloved chessboard... Not only do these self-taught expert machines perform incredibly well, but we can actually learn from the new knowledge they produce."

3.3 Jogos Atari

Os jogos Atari foram desenvolvidos pela empresa *Atari Inc.* Alguns dos jogos mais populares incluem clássicos como Pong, Asteroids, Centipede e Missile Command.

Deep Reinforcement Learning tem sido usado em jogos da Atari para desenvolver agentes inteligentes capazes de jogar esses jogos a um nível por vezes superior ao do humano.

No artigo de uma equipa da *Deepmind* [11], introduz-se o uso de Deep Q-Networks (DQN), mencionado anteriormente, (uma técnica de aprendizagem por reforço baseada em redes neurais profundas) para testar a aprendizagem destes jogos por máquinas inteligentes.

A ideia básica desta equipa, por trás do uso de *Deep Reinforcement Learning* passa pelas seguintes fases:

- **Pré-processamento**: A entrada dos jogos Atari (frames do jogo) é pré-processada para reduzir a dimensionalidade e complexidade dos dados introduzidos. Segundo [11], trabalhar directamente com frames do jogo Atari, que são imagens de 210 $\times$ 160 pixéis com uma paleta de 128 cores, pode ser computacionalmente exigente, pelo que é necessário aplicar uma etapa básica de pré-processamento destinada a reduzir a dimensionalidade da entrada. Os frames são pré-processados facilitando assim o processo de treino do sistema.

- **Arquitetura do modelo**: Uma das arquitecturas mais populares é a Deep Q-Network (DQN), usada também em modelos mencionados anteriormente, que combina o algoritmo Q-learning com redes neurais profundas. No contexto dos jogos da Atari, o agente recebe entradas visuais na forma de *frames* de jogo e usa redes neurais profundas para aprender a prever que ações resultarão na maior recompensa cumulativa ao longo do tempo.
- ***Exploration* vs. *Exploitation***: Explorar o ambiente (*Exploration*) em que se encontra permite ao agente aumentar o conhecimento que tem sobre cada ação e sobre os resultados que cada uma trará em longa data. Explorar o valor do agente (*Exploitation*) faz com que seja escolhida uma abordagem "gulosa" (greedy) de modo a obter o máximo de recompensas [1]. O objetivo do agente deve assim, ser equilibrado entre a optar pela exploração do ambiente de jogo, para encontrar melhores estratégias e a utilização do seu conhecimento actual para maximizar as recompensas.
- **Treino**: Tal como mencionado em [11], a equipa optou por realizar testes nos jogos *Beam Rider*, *Breakout*, *Enduro*, *Pong*, *Q*bert*, *Seaquest*, *Space Invaders*. Considerou tarefas em que um agente interage com um ambiente (o emulador Atari), numa sequência de acções, observações e recompensas. Em cada etapa, o agente seleccionou uma ação a partir do conjunto de ações de jogo. Esta é passada para o emulador e modifica o seu estado interno e a pontuação do jogo. Para além disso, recebe uma recompensa que representa a mudança na pontuação do jogo. O objectivo do agente é interagir com o emulador, seleccionando ações de forma maximizar recompensas futuras.
- **Avaliação**: Após um certo número de iterações de treino, o desempenho do agente é avaliado. Esta avaliação determina se o agente alcançou um desempenho satisfatório ou se é necessário treino adicional.

3.4 OpenAI Five

Em 2019, o OpenAI Five tornou-se o primeiro sistema a derrotar o campeão mundial de um jogo de e-sports. O sistema consiste numa equipa de 5 agentes de inteligência artificial capazes de jogar o jogo Dota2, um jogo que apresenta diversos desafios no que toca a agentes inteligentes tais como informação imperfeita [4]. No entanto, a utilização de Deep Reinforcement Learning mostrou-se incrivelmente eficaz, registando uma percentagem de vitórias de 99.4% em mais de 7000 jogos, assim que o sistema se tornou disponível para o público geral jogar contra o mesmo [4].

À semelhança dos sistemas anteriormente analisados, o OpenAI Five mostra que esta abordagem de DRL se mostra eficaz e consegue atingir uma performance sobre-humana na realização de tarefas difíceis [4].

3.5 Outros casos de estudo

Em [14], são aplicados algoritmos de aprendizagem por reforço profunda (Deep Q-Learning (DQN), Prioritized Dueling DQN e Advantage Actor Critic (A2C)) a diversos exemplos de videojogos. Podemos assim mencionar:

- **Aliens**: É um jogo de sobrevivência passado no universo de Alien. Foi usado o algoritmo de Deep Q-Learning.
- **Frogs**: Inspirado no jogo Frogger (criado pela Atari Games). Foi fornecido apenas um ponto como recompensa o que levou a que nenhum agente conseguisse encontrar uma boa solução.
- **Wait For Breakfast**: O jogador deve dirigir-se a uma mesa e aguardar pelo pequeno-almoço. Mais uma vez, foi fornecido apenas um ponto como recompensa, no entanto, uma vez que apenas tem uma única solução mais simples só tem que a memorizar.
- **Missile Command**: O objetivo é defender uma cidade do ataque de mísseis vindos do céu. Os 3 algoritmos experimentados ficam presos num ótimo local. Para além disso devido à necessidade de precisão nenhum conseguiu manter uma pontuação perfeita.
- **Superman**: O superhomem deve salvar todas as pessoas e eliminar os vilões, ganhando quando todos estiverem a salvo. Os agentes conseguiram, ocasionalmente, encontrar um bom padrão de aprendizagem.
- **Boulder Dash**: O jogador coleciona diamantes e foge de vários perigos. O único algoritmos dos três que teve sucesso foi o A2C.
- **Seaquest**: O objetivo é salvar mergulhadores desviando-se de peixes. Os resultados tiveram imenso ruído suspeitando-se que tenha sido provocado por nenhum dos agentes ter aprendido as regras do jogo.
- **Zelda**: Neste o agente deve encontrar uma chave e evitar inimigos. Neste jogo os algoritmos de deep reinforcement learning mostraram ter bons resultados, aprendendo bem com os diversos eventos.

Tal como apresentado em [14], diferentes dinâmicas de jogos têm diferente impacto na dinâmica e eficiência deste modelo de aprendizagem. Acredita-se, no entanto, que uma mais profunda exploração destes algoritmos de aprendizagem e seu treino mais profundo poderiam ter trazido melhores resultados de aprendizagem, principalmente nos jogos menos complexos. Sublinha-se, apesar de tudo, que talvez esse não fosse o objetivo dos autores, dado terem abordado estes testes numa perspetiva de avaliação do impacto de certos componentes dos algoritmos, por exemplo, recompensas.

4 Conclusões

Apesar de ainda estarmos no início da evolução destes sistemas de aprendizagem, atualmente já se pode dizer que estes irão revolucionar não só a área dos jogos e vídeojogos mas toda a vida e interação humana como a conhecemos.

Tal como analisámos, o AlphaGo Zero e os outros sistemas analisados fazem parte de um avanço crítico em direção a este objetivo. Tal como mencionado por [6], se estas técnicas fossem aplicadas a outras áreas e a outros problemas que afetam a nossa sociedade, os avanços seriam extraordinários.

Os avanços observados em exemplos como o AlphaZero, mostram que a sua capacidade de dominar três jogos complexos diferentes (e possivelmente outros)

é a prova de que um único algoritmo pode aprender conhecimento novo em diversas situações distintas.

Existem, no entanto, exemplos de situações em que estes algoritmos atualmente poderiam trazer problemas. Tal como mencionado em [10], se aplicássemos este modelo ao design de um carro autónomo, fazer testes virtuais poderia não ser suficiente para replicar situações reais que são bem mais complexas e que por vezes requerem noções humanas. Poderia até ser perigoso quando se passasse de um ambiente de treino seguro para o mundo real.

References

1. AI ML Analytics. Reinforcement Learning – Exploration vs Exploitation Tradeoff. https://ai-ml-analytics.com/reinforcement-learning-exploration-vs-exploitation-tradeoff/
   Consultado a 22 março 2023
2. Baheti, P. (2023). The Beginner's Guide to Deep Reinforcement Learning [2023]. Microsoft. v7labs. https://www.v7labs.com/blog/deep-reinforcement-learning-guide#h1
   Consultado a 17 março 2023
3. Bergdahl, J., Gordillo, C., Tollmar, K., Gisslén, L. (2020). Augmenting Automated Game Testing with Deep Reinforcement Learning. Electronic Arts (EA), Stockholm, Sweden.
   Consultado a 17 março 2023
4. Berner, C. et al. (2019). Dota 2 with Large Scale Deep Reinforcement Learning. OpenAI. https://cdn.openai.com/dota-2.pdf
   Consultado a 21 março 2023
5. CQF Blog. What Is Machine Learning? Definition, Types, and Examples. https://www.cqf.com/blog/what-machine-learning-definition-types-and-examples?gclid=CjwKCAjw5dqgBhBNEiwA7PryaLq-uZSbrONcJ6fC09FSh1zV-tTFEr0bxWB2zJE23CXdq3ZFlj-rCBoCycMQAvD_BwE
   Consultado a 18 março 2023
6. DeepMind. (2017). AlphaGo Zero: Starting from scratch. https://www.deepmind.com/blog/alphago-zero-starting-from-scratch
   Consultado a 19 março 2023
7. DeepMind. (2018). AlphaZero: Shedding new light on chess, shogi, and Go. https://www.deepmind.com/blog/alphazero-shedding-new-light-on-chess-shogi-and-go
   Consultado a 19 março 2023
8. DeepMind. AlphaGo. https://www.deepmind.com/research/highlighted-research/alphago
   Consultado a 18 março 2023
9. Farouk, H., ElDahshan, K. A., Mofreh, E. (2022). Deep Reinforcement Learning based Video Games: A Review. 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC).
   Consultado a 17 março 2023
10. Great Learning. (2022). Reinforcement Learning. https://www.mygreatlearning.com/blog/reinforcement-machine-learning/
11. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M. Playing Atari with Deep Reinforcement Learning. DeepMind Technologies. https://www.cs.toronto.edu/ vmnih/docs/dqn.pdf
    Consultado a 19 março 2023
12. Silver, D., Huang, A., Maddison, C. et al. (2016) Mastering the game of Go with deep neural networks and tree search. Nature 529, 484–489. https://doi.org/10.1038/nature16961
    Consultado a 18 março 2023
13. Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., Driessche, G., Graepel, T., Hassabis, D. (2017). Mastering the game of Go without human knowledge. DeepMind, 5 New Street Square, London EC4A 3TW, UK.
    Consultado a 19 março 2023

14. Torrado, R. R., Bontrager, P., Togelius, J., Liu, J., Perez-Liebana, D. (2018). Deep Reinforcement Learning for General Video Game AI.
Consultado a 18 março 2023
15. Torres, J. (2020). Deep Reinforcement Learning Explained — 01: A gentle introduction to Deep Reinforcement Learning - Learning the basics of Reinforcement Learning. https://towardsdatascience.com/drl-01-a-gentle-introduction-to-deep-reinforcement-learning-405b79866bf4