Análise da capacidade de generalização de algoritmos de aprendizado por reforço

Luana G. B. Martins
Orientadora: Profa. Dra. Telma W. L. Soares

2019

INSTITUTO DE INFORMÁTICA
UFG

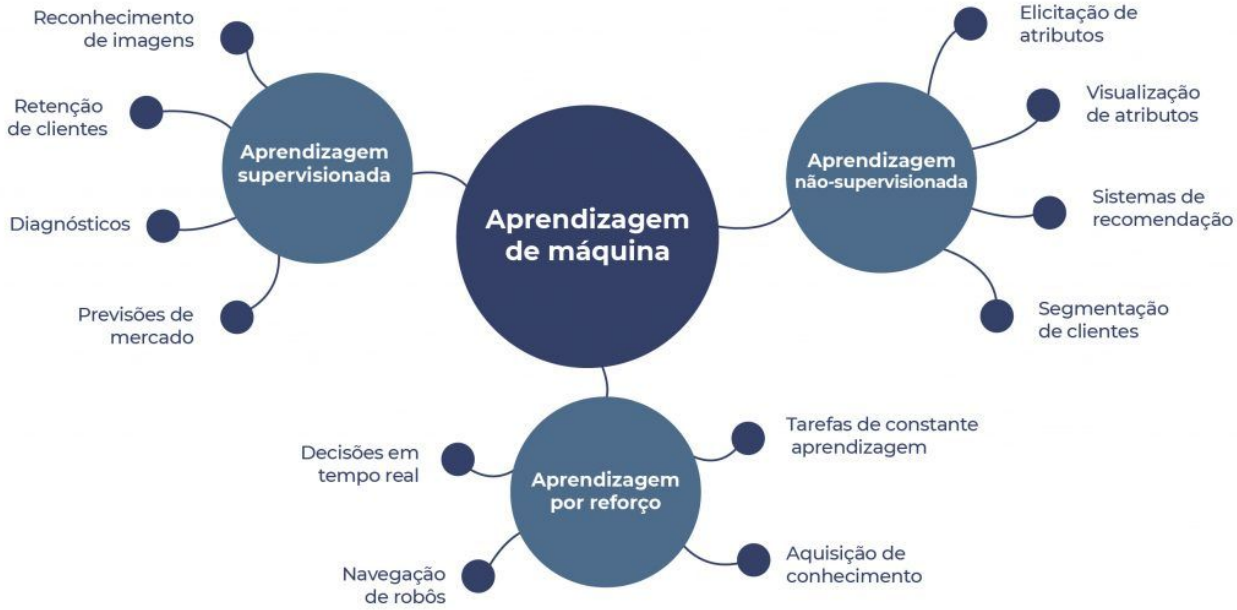
UFG
UNIVERSIDADE FEDERAL DE GOIÁS

Agenda

- Introdução
- Fundamentos
- Metodologia
- Resultados
- Conclusão

Introdução

Introdução



Fonte: https://isitics.com/2018/05/10/principais-sub-divisoes-e-aplicabilidade-da-aprendizagem-de-maquina/

Introdução

Arcade Learning Environment

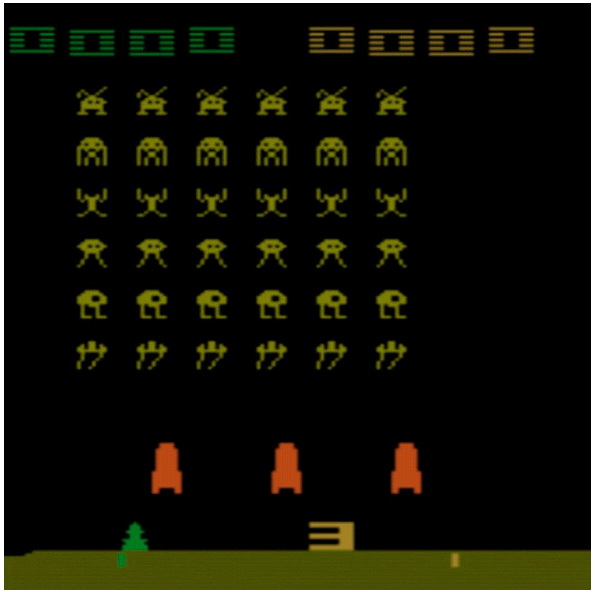


Figura 1: Agente de aprendizado por reforço atuando no jogo *Space Invaders*.

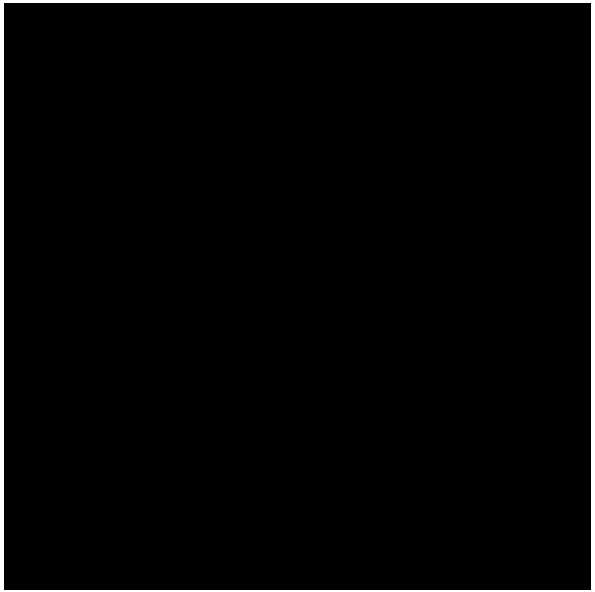


Figura 2: Agente de aprendizado por reforço atuando no jogo *Breakout*.

Introdução

General Video Game Artificial Intelligence

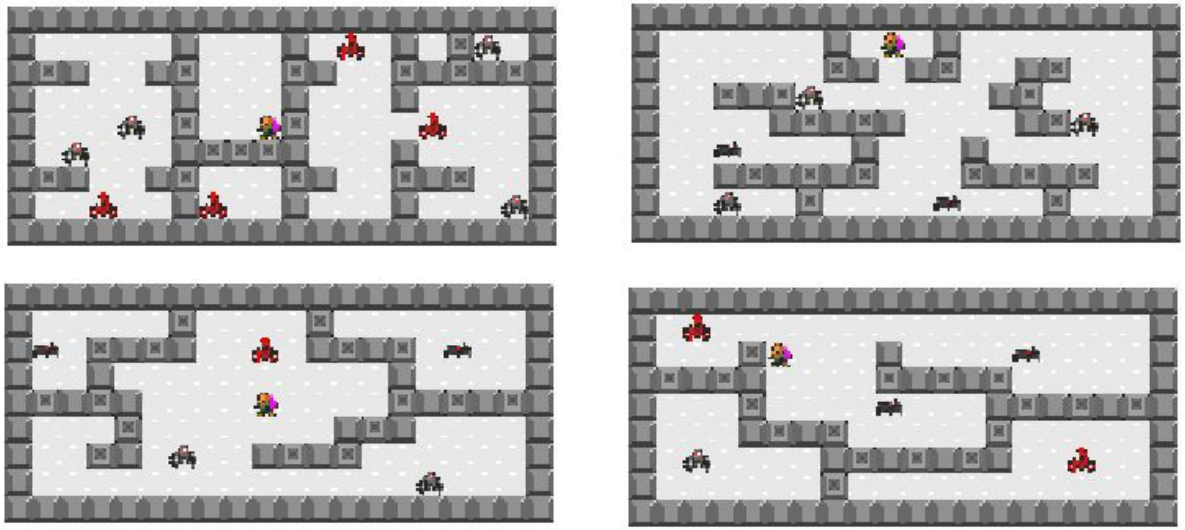


Figura 3: Exemplo da distribuição de um jogo no ambiente de avaliação GVGAI.

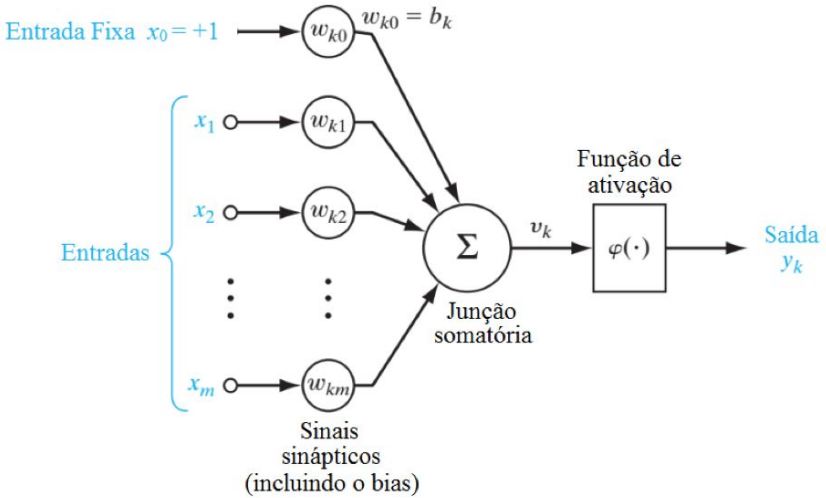
Objetivo

- Análise da capacidade de generalização do algoritmo de aprendizado por reforço *Proximal Policy Optimization*
  - separando explicitamente os ambientes de treinamento e teste

Fundamentos

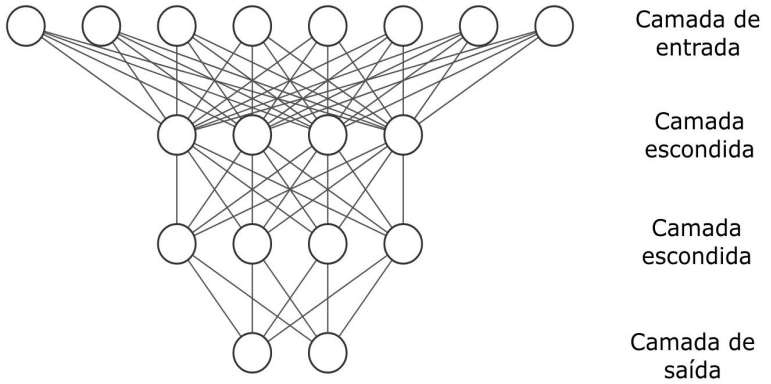
Redes Neurais

Figura 4: Modelo de neurônio artificial.



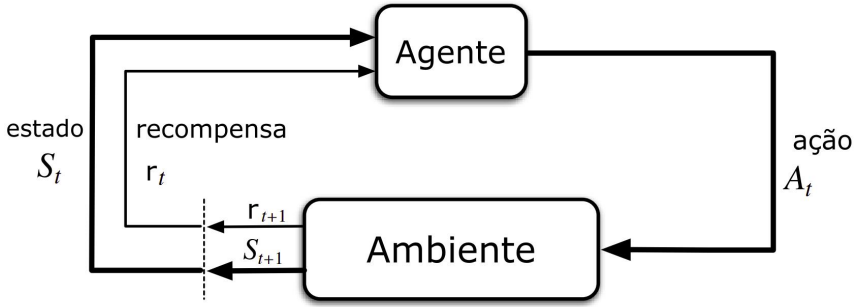
Fonte: Haykin, 1999

Figura 5: Rede Neural com duas camadas intermediárias.



Aprendizado por Reforço

Figura 6: Interação agente-ambiente.



Fonte: Sutton e Barto, 1998

- **Valor** de um estado

$$V(s) = \mathbb{E}\left[\sum_{t=1}^{T} \gamma^{t-1} r_i\right] \forall s \in S$$

- **Política** de tomada de decisões

$$\pi(a|s) = P(A_t = a|S_t = s)$$

Estimativa de Vantagem Generalizada

Determina quanto uma determinada **ação** é uma decisão **boa** ou **ruim**.

- Se **A > 0** , a ação escolhida é **melhor** que o valor médio.
- Se **A < 0**, a ação escolhida é **pior** que o valor médio.

$$\delta_t^V = r_t + \gamma V(s_{t+1}) - V(s_t)$$

$$A_t^{GAE(\gamma,\lambda)} = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_t^V$$

Fonte: Schulman et.tal., 2016

O aprendizado profundo é sobre reduzir a um problema de otimização numérica e, em seguida, usar algum tipo de algoritmo de descida de gradiente para aproximar a solução.

John Schulman

Otimização

Iteração de Valor

- Q-learning
  - Consegue incluir todas as transições vistas até então
  - Otimiza o objetivo errado

$$Q(s,a) = V(s,a) + \gamma \max_{a} Q(s',a)$$

Fonte: Watkins e Dayan, 1992

Otimização

Iteração de Política

- Métodos de gradiente de política
  - Aprender uma política que dará uma recompensa máxima
  - Otimiza o objetivo certo

$$\nabla_\theta J(\theta) = \mathbb{E}_{t \sim \pi_\theta} \left[\sum_{t=0}^{T} \nabla_\theta log \pi_\theta (a_t | s_t) A_t \right]$$

Fonte: Sutton e MacAllester, 1999

Otimização

Proximal Policy Optimization

$$L^{CLIP}(\theta) = \mathbb{E}_t\left[min(r_t(\theta)A_t, clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)A_t\right] \qquad r_t = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$$

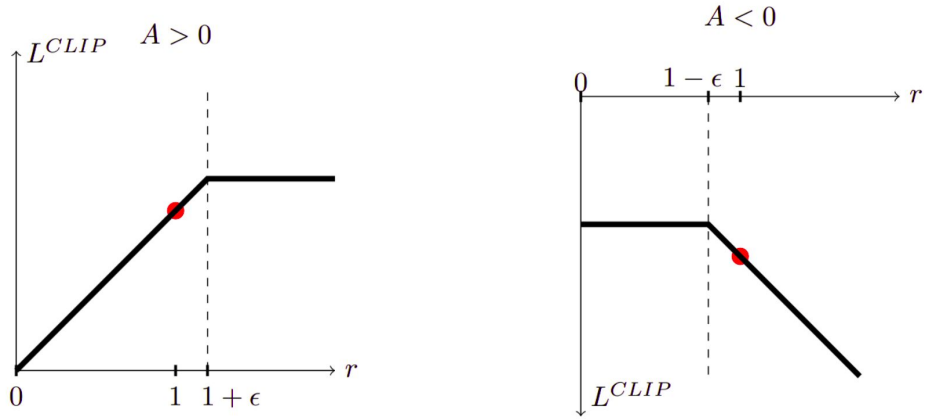


Figura 7: Gráficos da função Lclip.

Fonte: Schulman et.tal., 2017

Proximal Policy Optimization

Entrada: parâmetros da política inicial θ_0, limiar de corte ε

para $k = 0, 1, 2, \ldots$ **faça**

Colete um conjunto de trajetórias D_k com política $\pi_k = \pi(\theta_k)$

Estime a função de vantagem $A_t^{GAE(\gamma,\lambda)}$

Calcule a atualização da política

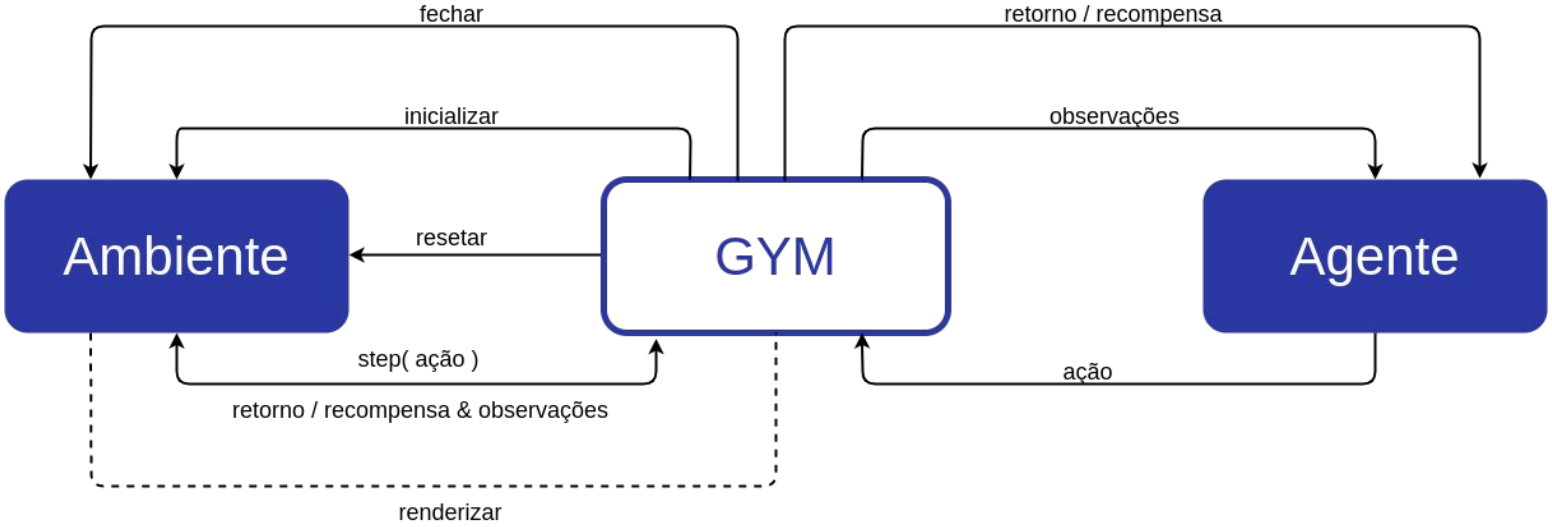
$$\theta_{k+1} = arg\max_\theta L_{\theta_k}^{CLIP}(\theta)$$

executando N etapas do gradiente ascendente, onde

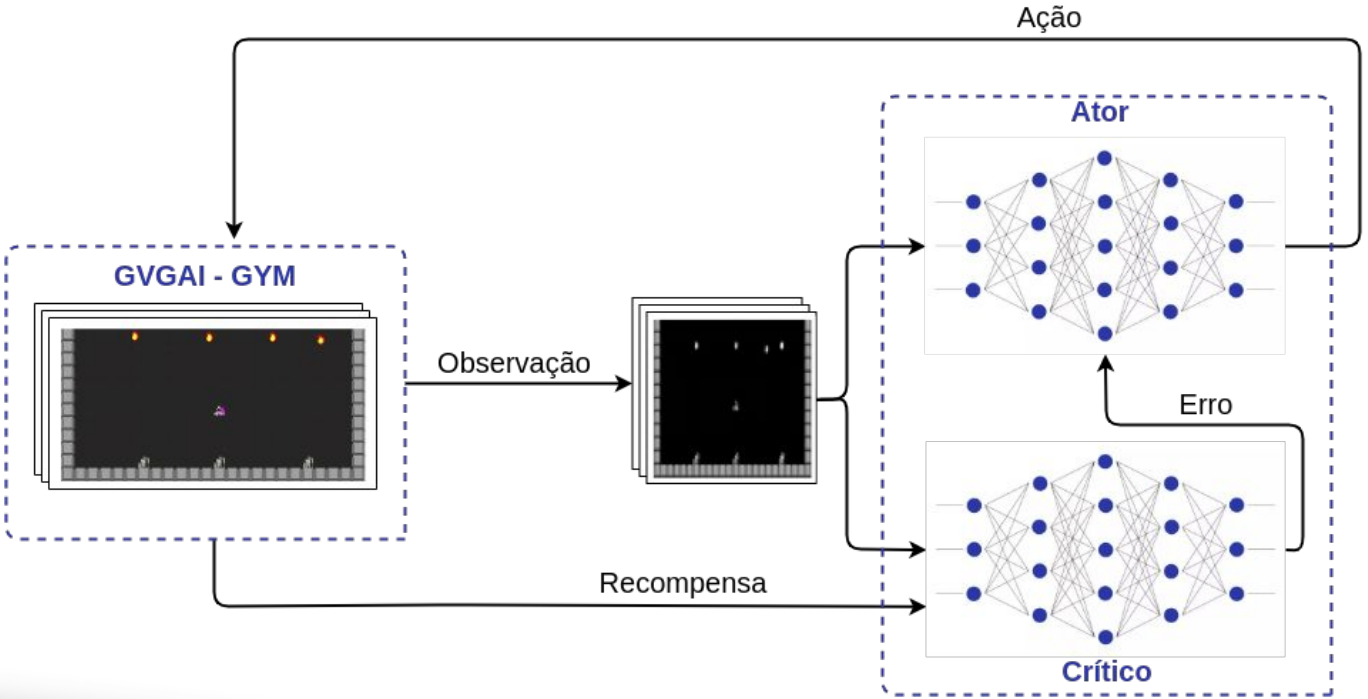
$$L_{\theta_k}^{CLIP}(\theta) = \mathbb{E}_t\left[min(r_t(\theta)A_t, clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)A_t\right]$$

Metodologia

GVGAI GYM


fechar
retorno / recompensa
inicializar
observações
Ambiente
resetar
GYM
Agente
step(ação)
ação
retorno / recompensa & observações
renderizar

Algoritmo de Treinamento



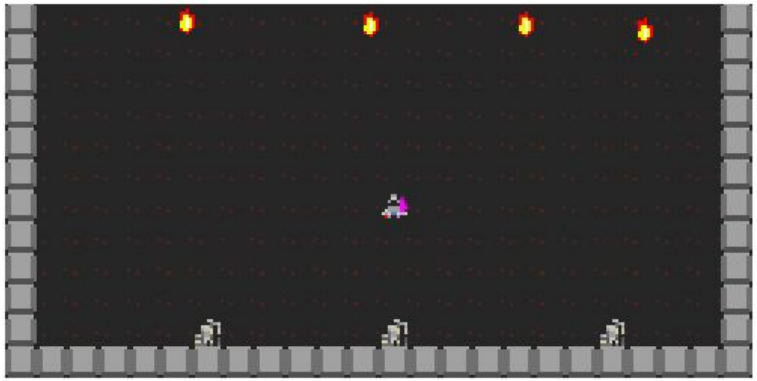
Aliens

O agente deve evitar projéteis inimigos recebidos e disparar no momento certo para atingir o inimigo.

Boulder Dash

O agente deve explorar cavernas, coletando diamantes e chegando a uma saída dentro de um prazo, evitando criaturas perigosas e obstáculos

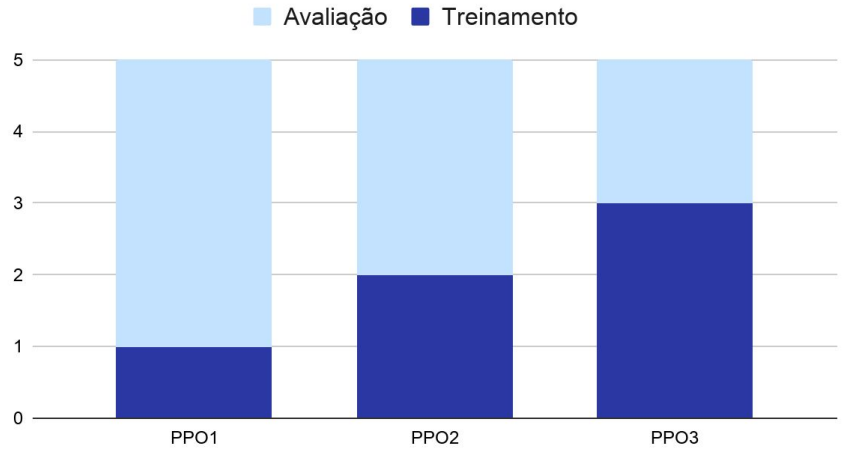
Missile Command



Cidades estão sendo atacadas por mísseis e o agente deve destruí-los em um tempo hábil utilizando os canhões disponíveis.

Testes

Figura 8: Conjuntos de treinamento.



As fases de um jogo são obtidas da mesma **distribuição**, portanto a **diferença** entre o desempenho do conjunto de **treinamento** e de **teste** determina o quão super-ajustado o agente ficou ao conjunto de treinamento.

Resultados

Aliens

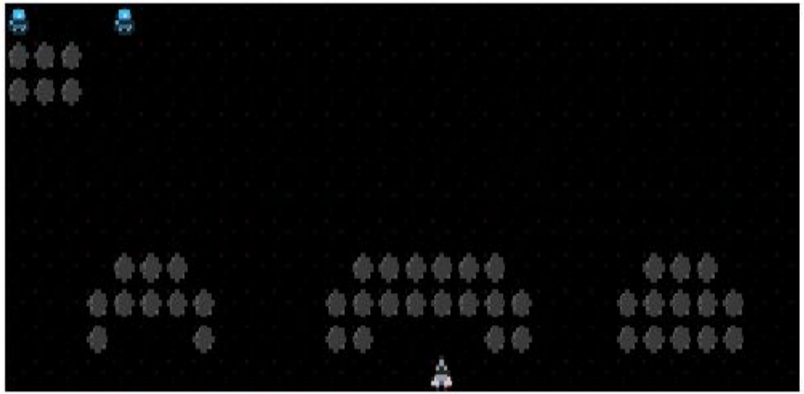


Figura 9: Fase de treinamento.

Figura 10: Fase de avaliação.

Aliens

Figura 11: Gráfico de relação treinamento-avaliação.

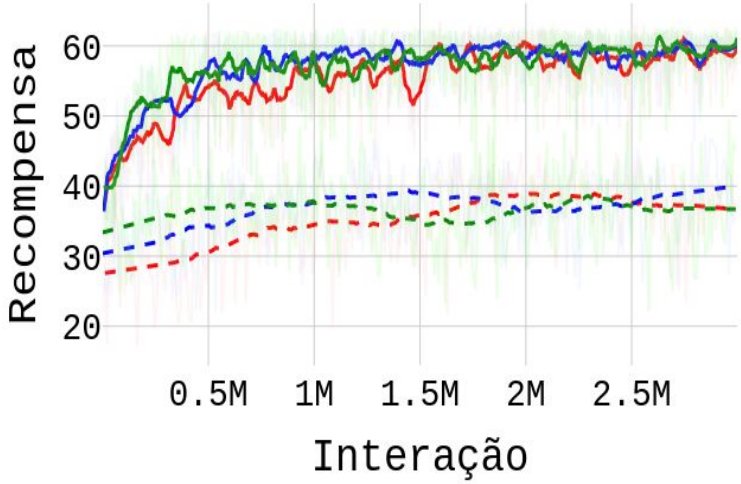
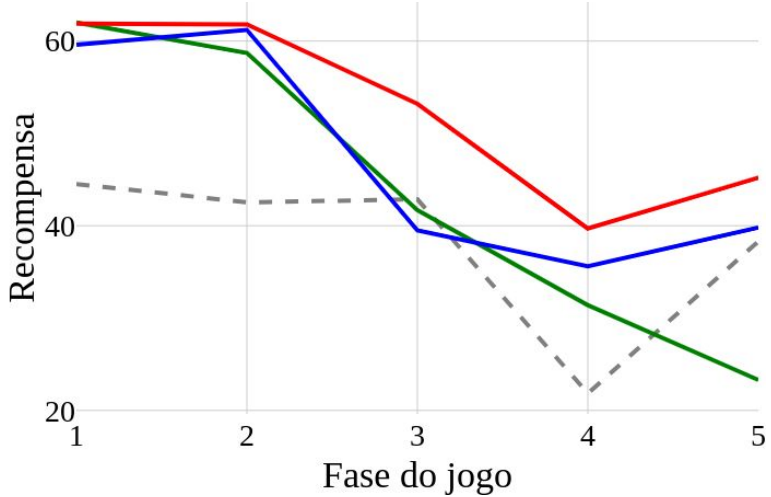


Figura 12: Gráfico de recompensa média dos modelos treinados.



Boulder Dash

Figura 13: Fase de treinamento.

Figura 14: Fase de avaliação.

Boulder Dash

Figura 15: Gráfico de relação treinamento-avaliação.

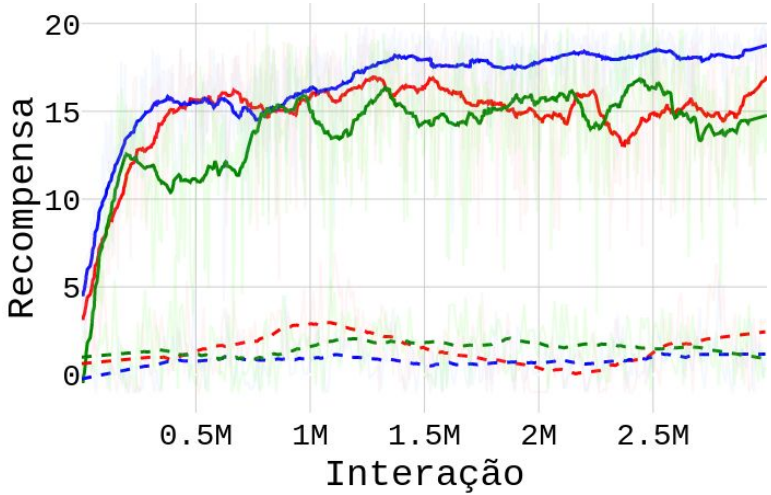
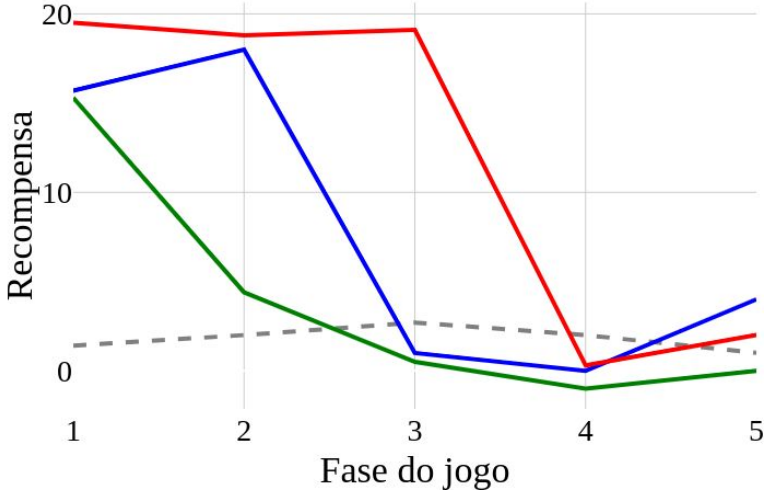


Figura 16: Gráfico de recompensa média dos modelos treinados.



Missile Command

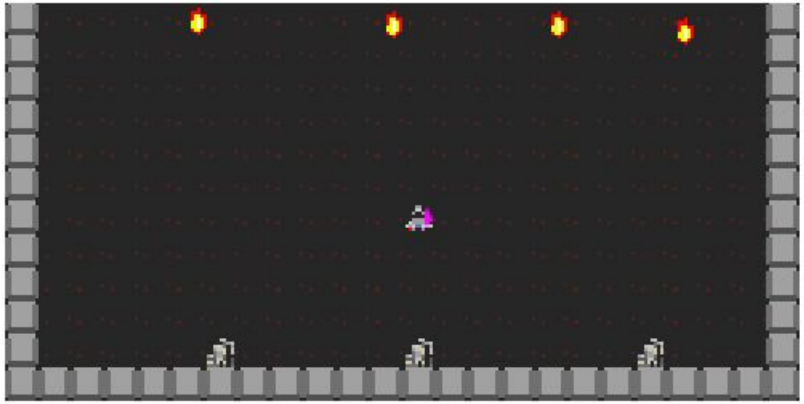


Figura 17: Fase de treinamento.

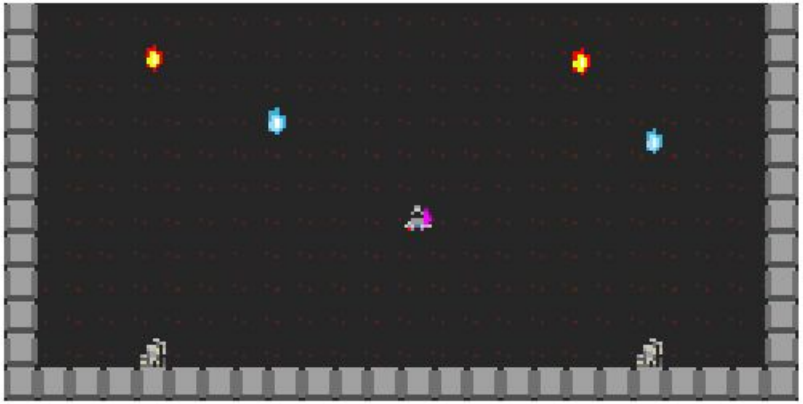


Figura 18: Fase de treinamento.

Missile Command

Figura 19: Gráfico de relação treinamento-avaliação.

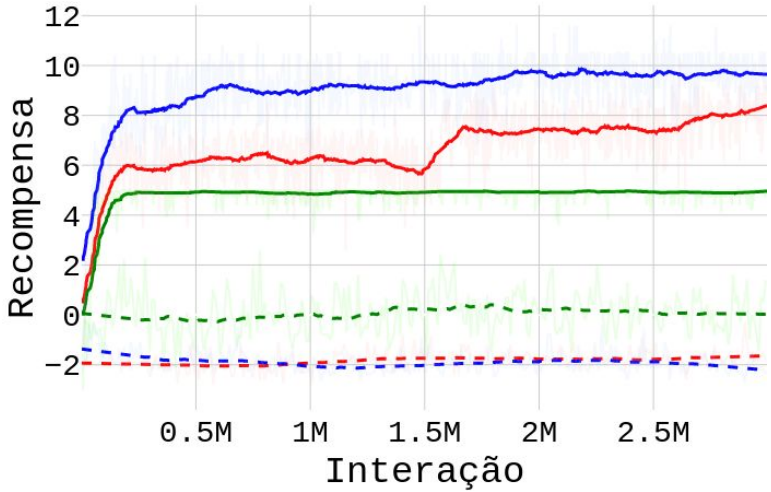
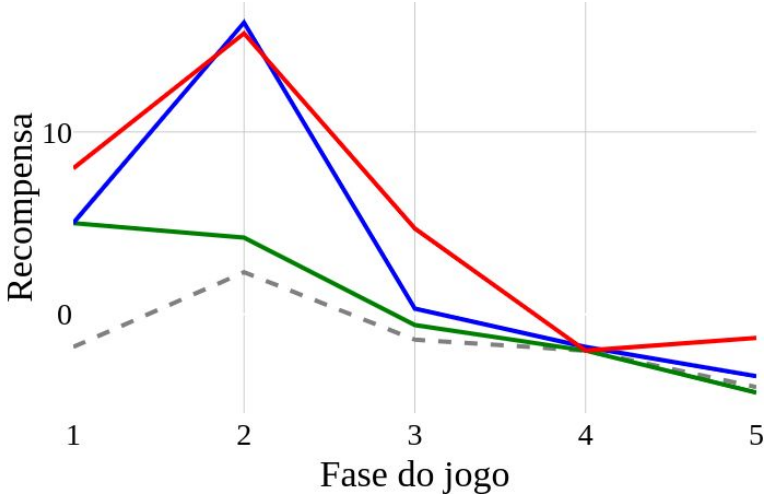


Figura 20: Gráfico de recompensa média dos modelos treinados.



Conclusão

Conclusão

- Apesar dos **avanços** durante os últimos anos, os agentes **falham** em generalizar entre tarefas;
- Agentes tendem a explorar o **determinismo** do ambiente, ignorando os estados e **memorizando** sequências de ação;
- Equivalente ao aprendizado supervisionado, a generalização é obtida com um **grande** conjunto de treinamento.

Trabalhos Futuros

- Foco em uma melhor **eficiência de amostragem**;
- Impacto de diferentes **arquiteturas** e formas de **regularização**;
- Formas de melhorar a **extração de características** da observação.

Obrigada

Dúvidas ou sugestões?

FUNAPE
Fundação de Apoio à Pesquisa - UFG

INSTITUTO DE INFORMÁTICA
UFG

UFG
UNIVERSIDADE FEDERAL DE GOIÁS