

Criação de Agente Inteligente usando Aprendizagem por Reforço

Fernando Azevedo Peres

Vitória, Espírito Santo

Abstract

Este trabalho consiste na criação de um agente inteligente que seja capaz de desempenhar o jogo do Dino, jogo simples onde se deve desviar de obstáculos pulando ou abaixando. Para tal, foi usado um método de busca heurística (Algoritmo Genético) e um método de classificação (KNN) para definir o agente. Foi obtido um agente inteligente que faz em média 1300 pontos.

Keywords: Aprendizagem por Reforço, Jogo do Dino, KNN, Algoritmo Genético

1. Introdução

A aprendizagem por reforço (reinforcement learning) é o treinamento de modelos de aprendizado de máquina em função da sequência de decisões que realiza. Um agente aprende a atingir uma meta em um ambiente dinâmico, incerto e potencialmente complexo. No aprendizado por reforço, o sistema de inteligência artificial utiliza tentativa e erro para encontrar uma solução para o problema.

O aprendizado por reforço tem sido muito empregado para criar agentes inteligentes capazes de aprender a jogar jogos virtuais. Este trabalho consiste em usar aprendizado por reforço para criar um agente inteligente que seja capaz de jogar o jogo Dino da Google.

Uma forma comum de implementar aprendizado por reforço consiste em usar um método de busca heurística e um método de classificação. O método de classificação analisa o estado atual do jogo e toma uma decisão sobre qual ação o agente inteligente deve tomar. O método de busca heurística procura os valores de parâmetros de funcionamento do método de classificação que maximizam o desempenho do agente no jogo. Ao final da busca, o agente inteligente será o classificador que apresentou o melhor desempenho no processo.

2. Descrição do Classificador

O classificador escolhido para a criação do agente inteligente foi o K Vizinhos Mais Próximos [1] (KNN).

K vizinhos mais próximos é um algoritmo simples que armazena todos os casos disponíveis e classifica novos casos com base em uma medida de similaridade (por exemplo, funções de distância). O classificador foi aplicado no problema de jogar o jogo dino da seguinte forma:

1. Foi selecionado um número N de pontos na tela (plano XY).
2. Cada um dos N pontos foi relacionado com um label do problema (K_UP, K_NO, K_DOWN).
3. O classificador é treinado com esses N pontos e labels.
4. A cada frame, o classificador identifica qual o ponto está mais próximo da posição atual do obstáculo, e retorna para o dinossauro executar a ação que está relacionada ao label do ponto selecionado.

3. Descrição da Meta Heurística

A meta heurística escolhida para a criação do agente inteligente foi Algoritmo Genético [2] (AG).

Algoritmos Genéticos são implementados como uma simulação de computador em que uma população de representações abstratas de solução é selecionada em busca de soluções melhores. A evolução geralmente se inicia a partir de um conjunto de soluções criado aleatoriamente e é realizada por meio de gerações. A cada geração, a adaptação de cada solução na população é avaliada, alguns indivíduos são selecionados para a próxima geração, e re combinados ou mutados para formar uma nova população. A nova população então é utilizada como entrada para a próxima iteração do algoritmo.

4. Resultados

Experimentos. O algoritmo genético possui parâmetros, tais quais tamanho da população (P), porcentagem de elitismo (PE), taxa de recombinação (TR), taxa de mutação (TM). Para os parâmetros PE, TR e TM foram utilizados 20%, 0.9 e 0.1 respectivamente. O valor de P foi variado entre 100, 1000 e 10000.

O classificador KNN, possui K como valor de hiperparâmetro, porém para este problema ele foi setado em 1. O que foi tratado como um hiperparâmetro foi o número N de pontos, ou seja, os pontos que foram usados para treinar o classificador que variou de 6 a 15, onde o melhor resultado obtido foi utilizando 11 pontos. Os pontos utilizados foram os seguintes: [[529, 300], [575, 325], [161, 300], [829, 325], [554, 260], [162, 325], [562, 300], [624, 325], [451, 260], [833, 300], [553, 325]].

Resultados. Os resultados disponibilizados pelo professor orientador serão expostos sob o nome AI.Flávio e os desenvolvidos pelo autor sob o nome AI.Fernando.

A Tabela 1 mostra os resultados obtidos para cada agente inteligente, levando em conta média, desvio padrão, e intervalo de confiança 95% de significância da acurácia.

Table 1: Resultados dos Agentes Inteligentes

Agente	Média	Desvio Padrão	Limite Inferior	Limite Superior
IA_Fernando	1339.13	129.20	1292.90	1385.37
IA_Flavio	1026.91	308.13	916.65	1137.17

A Tabela 2 mostra os resultados das 30 execuções para cada agente inteligente, ordenadas de forma decrescente.

Table 2: Resultados das 30 execuções

	AI.Fernando	AI_Flavio		AI.Fernando	AI.Flavio
1	1528.25	1625.25	16	1351.75	951.00
2	1518.50	1493.50	17	1346.75	945.50
3	1496.00	1467.25	18	1342.75	911.75
4	1495.25	1413.75	19	1313.75	882.50
5	1494.00	1408.25	20	1287.75	879.00
6	1485.75	1400.25	21	1287.00	814.50
7	1468.00	1341.50	22	1266.25	788.50
8	1407.50	1336.25	23	1260.00	779.00
9	1403.25	1314.00	24	1239.00	776.50
10	1403.25	1248.50	25	1218.00	737.25
11	1401.50	1218.50	26	1215.00	720.00
12	1400.75	1011.25	27	1156.50	715.75
13	1389.25	1005.50	28	1138.75	710.25
14	1377.25	984.75	29	1111.50	506.25
15	1368.00	973.00	30	1002.75	448.00

A Tabela 3 mostra os testes de hipótese entre os agentes. Na matriz triangular superior é apresentado o resultados do teste t pareado (amostras dependentes) e na matriz triangular inferior

é apresentado o resultado do teste não paramétrico de wilcoxon. Os valores da célula da tabela que rejeitarem a hipótese nula para um nível de significância de 95% estão escritos em negrito.

Table 3: p-values pareados

AI_Fernando	0.000066
0.000388	AI_Flavio

A figura 1 mostra um boxplot dos resultados de cada agente nas 30 execuções.

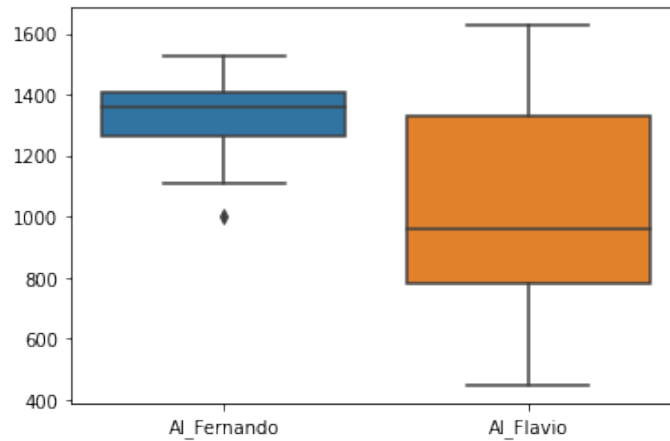


Figure 1: Boxplot comparativo dos agentes

5. Conclusões

Análise Geral dos Resultados. A partir da Tabela 3, podemos observar que os resultados para ambos os testes rejeitaram a hipótese nula. Assim, podemos concluir que existe diferença estatística entre esses agentes inteligentes. O AI_Fernando apresentou resultados no geral melhores e mais consistentes que o AI_Flávio.

Contribuições do Trabalho. Este trabalho contribuiu para a consolidação do conhecimento a respeito do uso de meta-heurísticas, em específico algoritmo genético: seleção, elitismo, recombinação e mutação.

Melhorias e Trabalhos Futuros. Uma proposta de atividade para trabalhos futuros seria a adição de mais uma dimensão (que representa a velocidade, em um espaço 3D), para verificar se o agente

inteligente utilizando o classificador kNN consegue resultados melhores se levar em conta a velocidade do obstaculo.

References

- [1] S. Sayad, K neighbors classifier, https://www.saedsayad.com/k_nearest_neighbors.htm (Jun. 2022).
- [2] A. P. de Leon F. de Carvalho, Algoritmos genéticos, <https://sites.icmc.usp.br/andre/research/genetic/> (Jul. 2022).