TRABALHO PRÁTICO

Pac-Man

Neste projeto, o objetivo é implementar o algorítmo Q-Learning para fazer o Pac-Man aprender a melhor ação entre correr ou comer.

Cada estado *S* será modelado com um vetor de caractrísticas que contextualizam o estado atual do agente. Cada aluno deverá pensar em um conjunto de características que melhor contextualize o estado atual do agente.

Exemplo de algumas caracteríticas do estado S:
[distancia do fantasma mais próximo,
distancia da pastilha mais próxima,
número de fantasmas
concentração de pastilhas próximas]

A função de utilidade de um agente a executar uma ação pode ser definida como:
Q(s,a) = w1*f1(s,a) + w2*f2(s,a) +w3*f3(s,a) …
onde:
Wn=peso de cada característica.
Fn= Função característica que retorna um valor.

O problema é achar os melhores pesos para W*
Ajuste do peso de cada característica:
wi= wi+ a [correcao] *fi(s,a)
correcao = (r(s,a) + yV(s')) - Q(s,a)
a= taxa de aprendizado
r=recompensa
y=deconto temporal

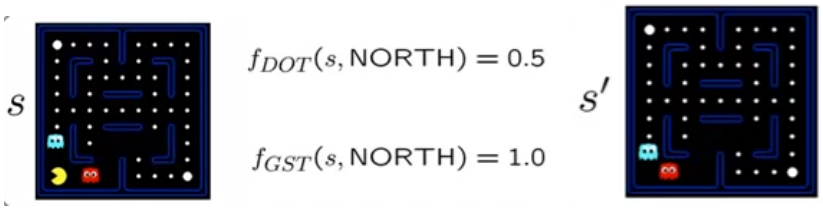
Melhor Politica
A cada movimento o pacman deve guiar seu movimento pelo menor custo, isto é deve escolher a melhor política para atingir seu objetivo. Como o ambiente é observável a busca pelo melhor caminho algorítmo pode ser implementado por um algorítmo de busca em árvore. Como sugestão, pode ser implementado um

algorítmo **busca A***. Uma versão do algoritmo para o ambiente de desenvolvimento sugerido será entregue com o mesmo na semana de 12/09 para auxiliar no desenvolvimento do projeto.

EXEMPLO:

1) Calcular Q(s,a)

$Q(s,a)=4*f_{Dot}(s,a) - 1*f_{Ghost}(s,a)$



Q(s,NORTH) = 4*0,5 - 1*1= **1**
r= -500 (acabou perdendo)
V(s')=0
a=0.004

2) Atualizar pesos: **wi= wi+ a [correcao] *fi(s,a) ; correcao = (r(s,a) + yV(s')) - Q(s,a)**
WDOT
correcao = -500 + 0 - 1 = -501
Wdot = 4 + 0.004 * - 501 * 0.5
Wdot= 3

WGHOST
correcao = -500 + 0 - 1 = -501
Wdot = - 1 + 0.004 * - 501 * 1
Wdot= -3

Novos pesos
Q(s,a)=3*fDot(s,a) - 3*fGhost(s,a)

Material Complementar:

Lecture MIT
http://www.youtube.com/watch?feature=player_embedded&v=Si1_YTw960c#t=3116

PacMan Project MIT
http://inst.eecs.berkeley.edu/~cs188/fa09/projects/reinforcement/reinforcement.html

Capítulo 4 - A* - Inteligência Artificial - Peter Norvig

Capítulo 21 - Aprendizagem Reforço - Inteligência Artificial - Peter Norvig

Reinforcement Learning - Sutton and Barto