UNIDADE 5 - Aprendizado por reforço e aprendizado por imitação

Marta Noronha

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
CURSO: CIÊNCIA DE DADOS
APRENDIZADO DE MÁQUINA II

SUMÁRIO

Conceitos básicos do aprendizado por reforço

AlphaGo (Fonte: DeepMind)

- AlphaGo é um programa de computador que joga o jogo de tabuleiro chinês Go.
- Jogar Go envolve estratégia, criatividade e engenhosidade.
- Go é jogado em uma grade (geralmente 19×19).
- O objetivo do jogo é cercar uma parte do tabuleiro maior que o adversário.

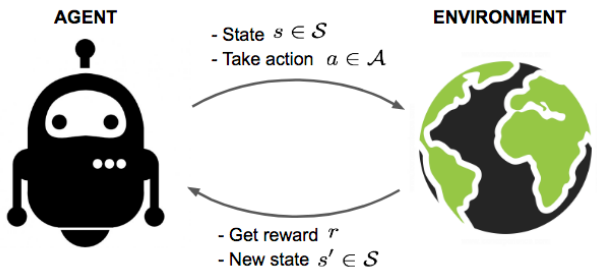
Conceitos básicos do aprendizado por reforço

AlphaGo (Fonte: DeepMind)

- Primeira partida do AlphaGo contra adversário humano em 2015. O adversário foi o tricampeão europeu, Fan Hui. Placar final 5-0 com vitória do AlphaGo.
- Em 2016, AlphaGo venceu o lendário jogador Lee Sedol, vencedor de 18 títulos mundiais e o maior jogador da época. Placar final 4-1 com vitória do AlphaGo, com público de mais de 200 milhões de pessoas em todo o mundo.
- A vitória contra Lee Sedol aconteceu uma década antes do que era previsto por especialistas.
- AlphaGo é o jogador mais forte da história, mostrando que sistemas baseados em IA podem resolver problemas em domínios altamente complexos.
- Link para assistir o documentário AlphaGo - The Movie.

Conceitos básicos do aprendizado por reforço

- Aprendizagem Por Reforço é o treinamento de modelos de aprendizado de máquina para tomar uma sequência de decisões.
- O agente está em um ambiente potencialmente complexo e incerto, no qual ele toma decisões baseadas em tentativa e erro para encontrar a solução para o problema.
- A inteligência artificial (IA) deve recompensar ou penalizar as ações executadas pelo agente de forma a maximizar a recompensa total.
- Não é oferecida nenhuma dica ou sugestão para que o agente possa resolver o jogo.



Fonte da imagem: A (Long) Peek into Reinforcement Learning
Fonte do texto: O Que é Aprendizagem Por Reforço?

Conceitos básicos do aprendizado por reforço

- O modelo deve aprender como executar a tarefa para que a recompensa seja maximizada, onde testes iniciais são aleatórios e, ao final, as táticas para atingir o objetivo (maximização da recompensa) são mais sofisticadas.
- Esta forma de aprendizado tende a sugerir uma criatividade para a máquina, porém a IA é baseada na experiência conhecida anteriormente a respeito das ações de agentes de forma paralela.

Desafios

- Transferir o modelo do ambiente de treinamento para o mundo real.
- Escalar e ajustar a rede neural que controla o agente devido à única comunicação ser baseada no sistema de recompensas e penalidades, sendo necessário guardar o aprendizado na "memória"do agente.
- Descoberta do ótimo local porque o agente executa a tarefa como está, mas não da maneira ideal ou necessária. Deve-se cuidar para que o agente não maximize a recompensa sem executar a tarefa.

Fonte: O Que é Aprendizagem Por Reforço?

Algumas diferenças entre os tipos de aprendizado

Aprendizado de máquina

"Aprendizado de Máquina é uma forma de IA na qual os computadores têm a capacidade de melhorar progressivamente o desempenho de uma tarefa específica com dados, sem serem diretamente programados."

- Aprendizado de máquina supervisionado
- Aprendizado de máquina não supervisionado

Aprendizado profundo

- Inspirada no funcionamento dos neurônios cerebrais, onde várias camadas neurais são projetadas para executar tarefas sofisticadas por meio do aprendizado gradual.
- Cada camada usa o resultado da camada anterior para treinar toda a rede.
- Estruturas como TensorFlow, Keras e PyTorch tornam a construção de modelos mais conveniente.

Algumas diferenças entre os tipos de aprendizado

Aprendizado por reforço

- Aplicação especializada de técnicas de Deep Learning e Machine Learning, projetada para resolver problemas de uma maneira específica.
- Sistemas de recompensas e penalidades para obrigar o sistema a resolver o problema sozinho, com no máximo envolvimento humano para mudança do ambiente e ajuste do sistema de recompensas e penalidades.
- Útil para fazer tarefas quando não existe uma forma adequada, mas existem regras que podem ser seguidas para que a tarefa seja executada.

Algoritmo Q-Learning

- Discutido em uma tese de 1989 na Universidade de Cambridge por Chris Watkins. Publicado em 1992.
- Não é fornecido o modelo do ambiente para orientar o processo de aprendizagem.
- O modelo aprende de forma iterativa, melhorando ao longo do tempo, a escolha de ações corretas.
- O agente não conhece o ambiente, porém toma cada ação com base no estado atual que o auxilia a aprender iterativamente e fazer previsões sobre o ambiente por conta própria.
- É um algoritmo "*off-policy*"porque é capaz de desenvolver o seu próprio conjunto de regras ou, quando achar melhor, desviar da política prescrita por meio do uso dos valores Q (valores de ação ou qualidade) armazenados na tabela Q.
- Um Q-learning que utiliza redes neurais é o Q-Learning Profundo.

Algoritmo Q-Learning

- Exploração do ambiente de forma aleatório (*exploration*) ou base-ada na informação disponível (*exploitation*).
- O modelo é atualizado iterativamente na medida em que o agente explora o ambiente.
- O algoritmo define (Q-learning):
  - Agentes: Entidade de IA que atua e opera no ambiente.
  - Estados: Variável que identifica a posição atual do agente no ambiente.
  - Ações: Operação do agente quando ele está em um estado específico.
  - Recompensa: Pode ser negativa ou positiva e depende da ação tomada pelo agente.
  - Episódio: O agente não consegue mais realizar uma ação e encerra.
  - Valores Q: Métrica baseada na Equação de Bellman ou na diferença temporal para medir a ação em um estado específico.

Algoritmo Q-Learning

Tabela Q : Tabela com listas de recompensas para as ações em cada estado em um determinado ambiente.

Fonte: Aprendizagem por reforço e Q-Learning

Algoritmo Q-Learning

Tabela Q : Tabela com listas de recompensas para as ações em cada estado em um determinado ambiente.

- Auxilia o agente a tomar decisões sobre as ações que podem levar a resultados positivos em diferentes situações.
- A tabela é atualizada de acordo com as recompensas ou penalidades recebidas pelo agente, refletindo o aprendizado do modelo.
- Atualização baseada na métrica Q escolhida.

Equação de Bellman

$$q^{new}(s,a) = (1-\alpha)\underbrace{q(s,a)}_{\text{old value}} + \alpha\overbrace{\left(R_{t+1} + \gamma \max_{a'} q(s',a')\right)}^{\text{learned value}}$$

onde α é a taxa de aprendizado e γ é o fator de desconto que determina a importância das recompensas futuras para o agente.

Algoritmo Q-Learning

Q-learning: Learn function $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$

Require:
Sates $\mathcal{X} = \{1, \ldots, n_x\}$
Actions $\mathcal{A} = \{1, \ldots, n_a\}$, $A : \mathcal{X} \Rightarrow \mathcal{A}$
Reward function $R : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$
Black-box (probabilistic) transition function $T : \mathcal{X} \times \mathcal{A} \to \mathcal{X}$
Learning rate $\alpha \in [0, 1]$, typically $\alpha = 0.1$
Discounting factor $\gamma \in [0, 1]$
procedure QLEARNING($\mathcal{X}$, A, R, T, α, γ)
  Initialize $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ arbitrarily
  while Q is not converged **do**
    Start in state $s \in \mathcal{X}$
    while s is not terminal **do**
      Calculate π according to Q and exploration strategy (e.g. $\pi(x) \leftarrow \arg\max_a Q(x, a)$)
      $a \leftarrow \pi(s)$
      $r \leftarrow R(s, a)$ ▷ Receive the reward
      $s' \leftarrow T(s, a)$ ▷ Receive the new state
      $Q(s', a) \leftarrow (1 - \alpha) \cdot Q(s, a) + \alpha \cdot (r + \gamma \cdot \max_{a'} Q(s', a'))$
      $s \leftarrow s'$
  return Q

Fonte: Capítulo 68 – Algoritmo de Agente Baseado em IA com Reinforcement Learning – Q-Learning

Algoritmo Q-Learning

Vantagens:

- Sem necessidade de conhecer previamente o ambiente, sendo de bom uso quando é difícil de modelar ou conhecer a dinâmica do ambiente.
- Otimização não vincluada estritamente a uma política.
- Treinamento off-line.

Desvantagens:

- A decisão do agente entre *Exploration* vs. *exploitation* pode prejudicar na tomada de novas ações ou manter o que já se conhece.
- Maldição da dimensionalidade.
- Superestimação da qualidade de uma determinada ação ou estratégia.
- Desempenho pode ser prejudicado se houver várias formas de resolver um problema.

Algoritmo SARSA (State-action-reward-state-action))

- Desenvolvido por Rummery e Niranjan em seu artigo de 1994 “On-Line Q-Learning Usando Sistemas Conexionistas”.
- *Online policy* para treinar modelos de processo de decisão de Markov, com o aprendizado do modelo sendo baseado no conjunto atual de ações realizadas pelo agente.
- A principal diferença entre SARSA e Q-learning é que o primeiro não maximiza a recompensa para a próxima etapa da ação a ser executada e atualiza o valor Q para os estados correspondentes.

Equação de Bellman no Algoritmo SARSA

$$Q(S, A) \leftarrow Q(S, A) + \alpha(R + \gamma Q(S', A') - Q(S, A))$$

onde α é a taxa de aprendizado e γ é o fator de desconto que determina a importância das recompensas futuras para o agente.

Algoritmo SARSA

- Realiza ações baseadas em recompensas recebidas em ações anteriores, onde o valor de estado (S) - ação (A) é armazenado na Tabela Q, com pares ordenados Q(S,A).
- Uso da política ϵ para equilibrar a decisão sobre *exploration* e *exploitation* no processo de aprendizagem para selecionar a ação com a recompensa estimada mais alta.
- Inicialização da Tabela Q com valores arbitrários.
- Estado inicial e ação inicial são escolhidos baseado na política gulosa ϵ que se baseia nos valores Q atuais.

Fonte: SARSA Reinforcement Learning Algorithm: A Guide

Algoritmo SARSA

Algoritmo 1: SARSA.

Definir os parâmetros: α, γ e ϵ

Para cada par ***s,a*** inicialize a matriz ***Q(s,a)***=0

Observe o estado ***s***

Selecione a ação ***a*** usando a política $\epsilon - greedy$

repita

- Execute a ação ***a***
- Receba a recompensa imediata ***r(s,a)***
- Observe o novo estado ***s'***
- Selecione a nova ação ***a'*** usando a política ε-gulosa
- $Q_{t+1}(s,a) = Q_t(s,a) + \alpha[r_t + \gamma Q_t(s',a') - Q_t(s',a')]$
- ***s = s'***
- ***a = a'***

até *o critério de parada ser satisfeito*;

Fonte: Ottoni, Nepomuceno e Oliveira

Diferenças entre SARSA e Q-Learning

SARSA

- O agente usa a política On para aprendizagem, onde o agente aprende com o conjunto atual de ações no estado atual e a política alvo ou a ação a ser executada.
- A aprendizagem do agente é melhorada usando o conjunto atual de ações executadas no estado atual.
- Estados anteriores e recompensas anteriores não são considerados para estados de operação mais recentes.

Q-Learning

- O agente utiliza a técnica de aprendizagem fora da política, onde o agente aprende as ações a serem executadas nos estados anteriores e os prêmios recebidos do conjunto anterior de ações.
- A aprendizagem do agente é melhorada através da realização de uma busca gulosa onde apenas a recompensa máxima recebida para o conjunto particular de ações naquele estado particular é considerada.
- Os estados anteriores e as recompensas anteriores são considerados para os estados de operações mais recentes.

Traduzido de :All you need to know about SARSA in Reinforcement Learning

REFERÊNCIAS

Q-Learning

- Q-learning
- Capítulo 68 – Algoritmo de Agente Baseado em IA com Reinforcement Learning – Q-Learning
- Q-Learning Algorithm: From Explanation to Implementation
- Aprendizagem por reforço e Q-Learning

Sarsa

- SARSA Reinforcement Learning Algorithm: A Guide
- All you need to know about SARSA in Reinforcement Learning

OTTONI, A. L. C.; NEPOMUCENO, E. G.; OLIVEIRA, M. S. de. Aprendizado por reforço na solução do problema do caixeiro viajante assimétrico: Uma comparação entre os algoritmos q-learning e sarsa. *XII Simpósio de Mecânica Computacional*.