Machine Learning
CPS 863
Terceiro Trimestre de 2024

Professor: Edmundo de Souza e Silva

Lista de Exercícios 6

ATENÇÃO!

- Faça as listas de forma que TODAS AS RESPOSTAS sejam DEVIDAMENTE COMENTADAS (passos para se chegar a resposta).
- A entrega da lista deve ser feita em UM ÚNICO arquivo PDF. Não envie vários pedaços separadamente!
- ATENÇÃO! Faça as listas de forma que TODAS AS RESPOSTAS sejam DEVIDAMENTE COMENTADAS (passos para se chegar a resposta).
  Não procure a solução na Internet ou em livros ou no chatGPT, pois o objetivo é que você mesmo avalie o que sabe. Obviamente, caso você já tenha conhecimento do problema, não leia a resposta (mesmo que já conheça o resultado final) e tente fazer sozinho. Só assim você poderá ter uma ideia melhor dos tópicos que você ainda não domina com desenvoltura.
- Anote as dúvidas encontradas para resolver **sozinho**. Em classe gostaria de saber quais as dúvidas que cada um teve para resolver o problema sem olhar a resposta.
- Qualquer referência a código é MUITO menos importante do que a EXPLICAÇÃO DOS PASSOS que foram realizados. O que mais importa é a explicação de como se chegou na solução.
- Para facilitar escrever a lista de forma clara, é possível traduzir equações a mão para LaTex: https://mathpix.com/, ver também https://www.overleaf.com/learn/latex/Questions/Are_there_any_tools_to_help_transcribe_mathematical_formulae_into_LaTeX%3F

O objetivo da lista é treinar os conceitos de *reinforcement learning*, de acordo com as últimas aulas. A lista ilustra conceitos relacionados ao tópico. A lista é simples.

Questão 1 *Reinforcement Learning, Conceitos*

1. Explique a diferença entre *Value iteration*, *Policy iteration* e *Q-learning*. Para facilitar a explicação, mostre as equações usadas para cada caso.

Questão 2 *Reinforcement Learning*

Otimização de um Sistema de Fila.
Considere um sistema de fila onde clientes chegam e são atendidos em intervalos de tempo discretos (a cada *slot*) de tempo. O sistema é definido a seguir:

1. **Chegada de Clientes:**
   - No final de cada intervalo de tempo:
     - 0, 2 ou 4 clientes chegam com probabilidades p_0, p_2 e p_4, respectivamente ($p_0+p_2+p_4 = 1$).
2. **Atendimento:**
   - No início de cada intervalo de tempo:
     - Se houver S servidores e C clientes, o sistema atende $\min(S, C)$ clientes.

  - Clientes restantes permanecem no sistemas (ficam para o próximo intervalo de tempo).
3. **Restrições do Sistema (para facilitar):**
   - Um máximo de 8 clientes pode estar no sistema em qualquer momento.
   - O sistema inicia no estado inicial $(C, S) = (0, 0)$, isto é, sem usuários e sem nenhum servidor.
4. **Ações:**
   - No início de cada intervalo de tempo, pode-se:
     - -1: Remover 1 servidor.
     - 0: Manter o número atual de servidores.
     - $+1$: Adicionar 1 servidor.
   - O número de servidores é limitado entre 1 e 3.
5. **Recompensas e Custos:**
   - **Ganhos:** T por cliente atendido.
   - **Custo do Servidor:** R_s por servidor por intervalo de tempo.
   - **Penalidade de Fila:** R_q por *slot*, quando houver mais que 4 clientes no sistema.
   - **Penalidade de Ociosidade:** R_0 por cada servidor não utilizado por intervalo de tempo.
6. **Objetivo:**
   - Projetar uma política $\pi(s)$ que maximize a recompensa esperada:
   - Você deve escolher um valor γ para o fator de desconto.
7. Você pode escolher entre *value iteration*, *plicy iteration*, etc.