

# EL7021: Tarea 1

Profesor: Javier Ruiz del Solar  
Auxiliar: Francisco Leiva  
Ayudante: Rodrigo Salas

Marzo, 2023

## Requisitos

- Python  $\geq 3.x$
- NumPy
- Matplotlib

## Fechas de entrega

Parte I (Avance) : 3 de abril, hasta las 23:59  
Parte II (Final) : 10 de abril, hasta las 23:59

## Descripción

Considere un mundo discreto que posee regiones navegables, no navegables (paredes exteriores e interiores), y una región objetivo. Dentro de este mundo, un agente puede intentar moverse en cuatro direcciones: arriba, abajo, a la izquierda, o a la derecha. Si el agente se mueve en dirección a una región no navegable, entonces se mantiene en la misma posición en la que se encontraba.

Por razones desconocidas, el mundo confunde a los agentes que lo visitan. Si un agente intenta moverse en una dirección dada, existe una probabilidad  $1 - p$  de que se mueva en una de las direcciones perpendiculares a la dirección en que él quería moverse originalmente.

Se pide que usted derive una política tal que un agente interactuando con este mundo logre llegar a la región objetivo utilizando la menor cantidad de acciones posibles.

## Parte I

### Instrucciones

1. Describa el MDP que define al problema (espacio de estados, espacio de acciones, función de recompensa, y función de transición de estados).
2. Programe el algoritmo “policy iteration” para resolver el problema propuesto. Para ello complete los métodos `_policy_evaluation` y `_policy_improvement` que se encuentran en el archivo `policy_iteration.py`

3. Sin modificar las funciones auxiliares proporcionadas en el código base, muestre la función de valor encontrada, junto a la política aprendida, y el número de iteraciones sobre la función de valor que fue necesario realizar.

## Parte II

### Instrucciones

1. Programe el algoritmo “value iteration” para resolver el problema propuesto. Esta vez complete la función `run_value_iteration` del archivo `value_iteration.py`.
2. Muestre la función de valor encontrada, junto a la política aprendida, y el número de iteraciones sobre la función de valor que fue necesario realizar.
3. Por defecto, la probabilidad de transicionar a un lugar diferente al deseado (exceptuando colisiones con paredes) es  $1 - p = 0.2$ . Cambie esta probabilidad a cero y analice los cambios presentados en la función de valor y la política obtenida empleando tanto *policy iteration* como *value iteration*. Comente.
4. Fije  $1 - p = 0.4$ , y varíe el factor de descuento  $\gamma$  a 0.2 y luego a 1.0. Muestre las funciones de valor y las políticas correspondientes a cada caso, esta vez solo empleando *value iteration*. Interprete sus diferencias.
5. Dada la recompensa fijada en el problema, ¿qué representa la función de valor obtenida cuando  $\gamma = 1$ ?

## Reglas de formato

Las entregas deben cumplir con los siguientes requerimientos:

- Reporte en formato PDF.
- Incluya un archivo README.txt junto al código, donde indique las versiones de las dependencias que utilizó, y las instrucciones de ejecución de su código.
- Entregas parciales y finales en formato zip (reporte y código en un único archivo).
- Figuras legibles, de preferencia vectorizadas.
- Enumerar las respuestas de la misma forma en que se enumeran las preguntas.
- Respuestas concisas. No es necesario describir gráficos, es suficiente con mostrarlos en el reporte, y responder las preguntas que se realizan textualmente.
- No es necesario crear una portada, no obstante, la primera página debe contar con:
  - Título con formato “Avance Tarea X o Entrega Tarea X”, según corresponda.
  - Código del curso.
  - Nombre del estudiante.