Práctica 1 – Proyecto de Aprendizaje por refuerzo.

Manipuladores- Grado en Ingeniería Robótica

Curso 25/26

1. Introducción

El objetivo de esta práctica es que el alumnado adquiera un entendimiento práctico y experimental de los algoritmos de Aprendizaje por Refuerzo (RL) estudiados en clase. A través de la implementación y comparación de distintos métodos, se busca:

- Comprender cómo los agentes aprenden en entornos de distinta complejidad.
- Analizar las fortalezas y limitaciones de los algoritmos Monte Carlo, SARSA y Q-Learning en un entorno determinado.
- Observar el efecto de los parámetros y las políticas de acción sobre el aprendizaje, incluyendo exploración y explotación.
- Aprender a interpretar resultados experimentales para justificar decisiones técnicas y de diseño.

Esta práctica se realiza utilizando la librería Gymnasium, que permite simular diferentes entornos de RL con características variadas: determinismo, recompensas escasas, discretización del espacio, entre otros.

2. Entornos propuestos

Cada grupo (formado por máximo 3 personas) deberá seleccionar uno de los siguientes escenarios:

- **Frozen Lake (slippery=True, entorno no determinista)**
- **Taxi**
- **MountainCar (versión NO CONTINUA)**
- **LunarLander**
- **Blackjack**
- **CliffWalking (sliperry=True, entorno no determinista)**

3. Algoritmos y políticas a implementar

Sobre el entorno elegido se deberán de implementar y comparar los algoritmos y políticas vistos en clase en base a **RESULTADOS OBTENIDOS**.

Específicamente sobre los algoritmos se debe de hacer una comparativa sobre como convergen y como de rápido aprenden (estimación de la función de valor), y que dificultades presentan dado el entorno.

En cuanto a las políticas, se debe de justificar la elección de la política y analizar su impacto en la exploración, explotación y rendimiento del agente en el aprendizaje.

Finalmente, tambien se deben de analizar el efecto de los diferentes parámetros que intervienen en el aprendizaje, como la tasa de aprendizaje α, el factor de descuento γ, así como los parámetros de las políticas ε de *ε-greedy o τ en Softmax.*

4. Desarrollo de la práctica

4.1. Fase inicial (Práctica 0)

Con el objetivo de comprender bien la dinámica de aprendizaje de los algoritmos y políticas, se recomienda implementar y depurar todos los algoritmos y políticas en el entorno visto en la práctica 0, FrozenLake 4x4 (slipeery=False). Esto permite que observar la convergencia de los algoritmos en un entorno determinista y de baja complejidad donde realizar pruebas rápidas que permitan analizar y aprender la influencia de los parámetros en el aprendizaje del agente. **Esta parte no es entregable.**

Esta parte es opcional… pero si la ignoras, podrías acabar como en Bihar: ¡persiguiendo recompensas y criando serpientes en casa, sin aprender lo que realmente importa ! 😛

4.2. Fase experimental

Una vez asegurado el correcto funcionamiento de los algoritmos y su comprensión, se ejecutarán sobre el entorno seleccionado.

Específicamente en esta fase se deberá:

- Entrenar un agente en el entorno elegido.
- Registrar métricas que permitan evaluar el aprendizaje del agente (tasa de éxito, episodios hasta convergencia, recompensa acumulada media, estabilidad del aprendizaje, así como los aspectos indicados en la practica 0 (recompensas inmediatas acumuladas, medias, detección de estados terminales y condiciones que lo producen, evolución temporal de las observaciones y variables relevantes del entorno y parámetros de aprendizaje, así como cualquier otro aspecto que creas relevante.
- Realizar comparativas entre algoritmos y políticas.
- Analizar dificultades del entorno (estocasticidad, recompensas escasas, exploración costosa…).

4.3. Aspectos técnicos a justificar

En documento final a entregar deben aparecer justificados (**en base a resultados**):

- La elección del algoritmo principal.
- La política de exploración escogida.
- Los parámetros finales utilizados.

Además, se deberá justificar si ha sido necesario discretizar el entorno (como en el caso de MountainCar o LunarLander), modificar o redefinir recompensas (reward wrapper), o si se ha incluido alguna optimización.

5. Entregables

5.1. Exposición

La exposición se realizará el viernes 19 de diciembre para **todos los grupos** después del examen de teoría. Cada presentación debe durar 10-12 minutos. En la presentación se debe de incluir:

- Explicación breve del entorno seleccionado
- Algoritmos implementados.
- Resultados experimentales.
- Comparativas y conclusiones.
- Demostración del agente funcionando.

5.2. Memoria justificativa

La memoria debe de incluir además de los aspectos técnicos a justificar indicados en la sección 4.3, una introducción breve y descripción del entorno elegido, así como el análisis experimental llevado a cabo especificado en la sección 4.2.

5.3. Código

Se debe de entregar el código implementado en formato .py y .ipynb (tanto desarrollos locales como si se utiliza el entorno GoogleColab). Para este último, se deberá entregar también el enlace de desarrollo.

6. Evaluación y entrega

La práctica se evaluará en dos partes, cada una con un peso del 50% sobre la nota final de prácticas:

- Presentación oral (50%): exposición del trabajo realizado el día 19 de diciembre.

- Memoria explicativa (50%): documento escrito que justifica la implementación, los resultados y las decisiones técnicas.

Ten en cuenta que entre los aspectos principales que se van a evaluar son:

- Claridad y coherencia en la exposición y en la documentación escrita.
- Rigor técnico y experimental (experimentación llevada a cabo, análisis de resultados).
- Capacidad de síntesis y de explicar los conceptos y resultados de manera comprensible, justificando la elección parámetros, algoritmos y políticas).
- Eficiencia en el aprendizaje del agente considerando aspectos como la rapidez de convergencia, estabilidad y calidad de las políticas aprendidas.