M2.983 Aprenentatge per reforç

Pràctica:

Implementació d'un agent per a la robòtica espacial

Continguts

1. Presentació

Al llarg de les tres parts de l'assignatura hem entrat en contacte amb diferents classes d'algoritmes d'aprenentatge per reforç que permeten resoldre problemes de control en una gran varietat d'entorns.

Aquesta pràctica, que s'estendrà al llarg d'un mes aproximadament, dóna la possibilitat d'enfrontar-se al disseny d'un agent per solucionar un cas específic de robotica.

Atacarem el problema a partir de l'exploració de l'entorn i les observacions. Després passarem a la selecció de l'algorisme més oportú per solucionar l'entorn en qüestió amb les observacions seleccionades. Finalment, passarem per l'entrenament i la prova de l'agent fins a arribar a l'anàlisi del rendiment.

Per fer-ho, es presentarà abans l'entorn de referència. Posteriorment, es passarà a la implementació d'un agent Deep Q-Network (DQN) que el solucioni. Després d'aquestes dues primeres fases de presa de contacte amb el problema, es cercarà un altre agent que pugui millorar el rendiment de l'agent DQN implementat anteriorment.

2. Competències

En aquesta activitat es treballen les següents competències:

- Capacitat per analitzar un problema des del punt de vista de l'aprenentatge per reforç.
- Capacitat per analitzar un problema en el nivell d'abstracció adequat a cada situació i aplicar les habilitats i coneixements adquirits per resoldre'ls.

3. Objectius

Els objectius concrets d'aquesta activitat són:

- Conèixer i aprofundir en el desenvolupament d'un entorn real que es pugui resoldre mitjançant tècniques d'aprenentatge per reforç.
- Aprendre a aplicar i comparar diferents mètodes d'aprenentatge per reforç per poder seleccionar el més adequat a un entorn i problemàtica concreta.
- Saber implementar els diferents mètodes, basats en solucions tabulars i solucions aproximades, per a resoldre un problema concret.
- Extreure conclusions a partir dels resultats obtinguts.

4. Entorn

Estem treballant sobre el problema de guia autònoma i en particular volem solucionar el cas de l'aterratge propi, per exemple, dels drons autònoms.

Per això, s'escull **lunar-lander** com a entorn simplificat. L'entorn es pot trobar al següent enllaç:
https://github.com/openai/gym/blob/master/gym/envs/box2d/lunar_lander.py

Lunar Lander consisteix en una nau espacial que ha d'aterrar a un lloc determinat del camp d'observació. L'agent condueix la nau i el seu objectiu és aconseguir aterrar a la pista d'aterratge, coordenades (0,0), i arribar amb velocitat 0.

La nau consta de tres motors (esquerra, dreta i el principal que té a sota) que li permeten anar corregint el rumb fins a arribar a la destinació.

Les accions que pot fer la nau (espai d'accions) són discretes.

Les recompenses obtingudes al llarg del procés d'aterratge depenen de les accions que es prenen i del resultat que se'n deriva.

- Desplaçar-vos de dalt a baix, fins a la zona d'aterratge pot resultar en [+100,+140] punts
- Si s'estrella a terra, perd 100 punts (recompensa -100 punts)
- Si aconsegueix aterrar a la zona d'aterratge (velocitat 0), guanya +100 punts
- Si aterra, però no a la zona d'aterratge (fora de les banderes grogues) es perden punts
- El contacte d'una pota amb el terra rep +10 punts (si es perd contacte després d'aterrar, es perden punts)
- Cada cop que encén el motor principal perd 0.3 punts (recompensa -0.3 punts)
- Cada cop que encén un dels motors d'esquerra o dreta, perd 0,03 punts (recompensa -0.3 punts)

La solució òptima és aquella en què l'agent, amb un desplaçament eficient, aconsegueix aterrar a la zona d'aterratge (0,0), tocant amb les dues potes a terra i amb velocitat nul·la. Es considera que l'agent ha après a fer la tasca (i.e. el "joc" acaba) quan obté una mitjana d'almenys 200 punts durant 100 episodis consecutius.

Exercici 1.1 (0.5 punts)
Es demana explorar l'entorn i representar una execució aleatòria.

Exercici 1.2 (0.5 punts)
Explicar els espais d'observacions i d'accions possibles (informe escrit).

Nota**:** l'entorn Lunar Lander requereix la llibreria `box2d-py`. Si s'usa Google Colab[1], començar sempre el notebook amb: `!pip install box2d-py`

1 https://colab.research.google.com/

5. Agent de referència

A la tercera part de l'assignatura hem introduït l'agent DQN amb *replay buffer* i *target network*, que és un bon candidat per a la solució del problema de robòtica que estem analitzant, donat que permet controlar entorns amb un nombre elevat d'estats i accions de forma eficient.

Es demana resoldre els 3 exercicis següents.

Exercici 2.1 (1.5 punts)
Implementar un agent DQN per a l'entorn **lunar-lander**.

Exercici 2.2 (1 punt)
Entreneu l'agent DQN i busqueu els valors dels hiperparàmetres que obtinguin un alt rendiment de l'agent. Per fer-ho, cal llistar els hiperparàmetres sota estudi i presentar les gràfiques de les mètriques que descriuen l'aprenentatge.

Exercici 2.3 (0.5 punts)
Provar l'agent entrenat a l'entorn de prova. Visualitzar-ne el comportament (a través de gràfiques de les mètriques més oportunes).

6. Proposta de millora

En aquesta part es demana proposar una solució alternativa al problema de robòtica espacial que pugui ser més eficient respecte a allò que s'ha implementat anteriorment. Per assolir aquest objectiu, cal implementar un nou agent, basat en els algoritmes que hem vist al llarg de l'assignatura.

En particular, es demana solucionar els 3 punts següents.

Exercici 3.1 (2 punts)
Implementar l'agent identificat a l'entorn **lunar-lander**.

Justifiqueu les raons que han portat a provar aquest tipus d'observació entre les disponibles i perquè s'ha triat aquest tipus d'agent. Detalleu quins tipus de problemes s'espera que es puguin solucionar respecte a la implementació anterior.

Exercici 3.2 (2 punts)
Entrenar l'agent identificat i cercar els valors dels hiperparàmetres que obtinguin el rendiment "òptim" de l'agent.

Exercici 3.3 (2 punts)
Analitzar el comportament de l'agent entrenat a l'entorn de prova i comparar-lo amb l'agent implementat en el punt 2 (a través de gràfiques de les mètriques més oportunes).

7. Entrega

El lliurable serà un fitxer comprimit en format ZIP amb els següents dos documents:

- **Informe en format PDF** d'entre 10 i 15 pàgines de longitud, aproximadament;
- **Codi** utilitzat, ja sigui en fitxers Jupyter notebook (.ipynb) o Python (.py)

Per l'informe es pot usar la següent guia:

- Tamany de lletra 11 o 12.
- Font: Arial o similar.
- Interlineat senzill.
- Tres apartats definits segons el guió.
    - Especificar l'exercici corresponent com subapartat.
    - Per exemple, Apartat 2 Agent de referència, apartat 2.1 Implementació agent de referència, apartat 2.2 Entrenament agent de referència, apartat 2.3 Prova agent de referència.
- Les captures de pantalla (per exemple, les gràfiques de rendiment) o els fragments de codi (si es consideren rellevants) han d'estar pensats per il·lustrar i no per ser protagonistes.

El **codi font** emprat per a totes les etapes de la pràctica ha d'estar correctament comentat per facilitar la seva comprensió. Podeu fer servir fitxers Python nadius (.py) o basats en Jupyter Notebook (en aquest cas s'ha de lliurar la versió .ipynb i l'exportació en format .html).