UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

Project 3: Reinforcement Learning

Inteligenta Artificiala

Autori: Moșilă Luciana
Grupa: 30236

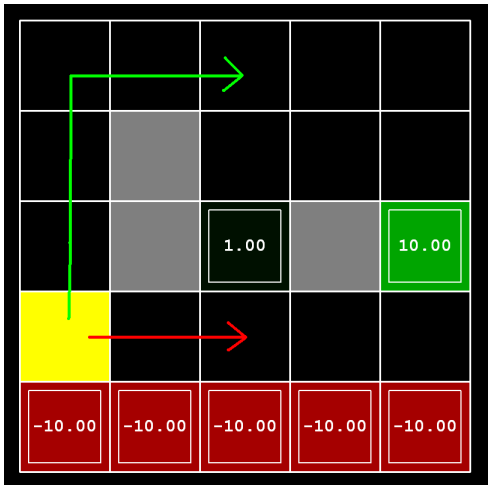
Facultatea de Automatica
si Calculatoare

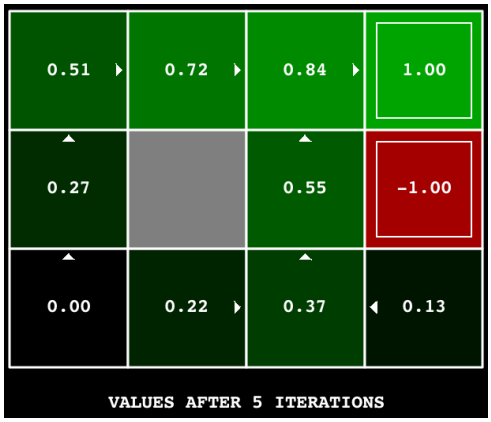
17 Ianuarie 2023

Cuprins

1 Introducere

În acest proiect, vom dezvolta implementări pentru iterația de valori și Q-learning. În prima etapă, veți evalua performanța agenților pe scenariul Gridworld.





2 Value Iteration

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s, a, s') \left[R(s, a, s') + \gamma V_k(s') \right]$$

Iterația Valorilor reprezintă o tehnică folosită în învățarea automată, cu aplicare în rezolvarea problemelor ce implică luarea deciziilor în condiții de incertitudine și recompense. Scopul acestei metode este să identifice politici optime în cadrul unui MDP.

Ecuția cheie care guvernează actualizarea valorilor stărilor, $V_{k+1}(s)$, este exprimată în funcție de valorile stărilor următoare, $V_k(s')$.

Metoda runValueIteration(self) Implementează algoritmul de Iterație de Valori. Aceasta ajustează valorile stărilor conform ecuației de actualizare specifică algoritmului Iterației Valorilor, pe durata unui număr stabilit de iterații.

Metoda getValue(self, state) furnizează valoarea asociată unei stări specificate. Această valoare este determinată în momentul construirii, folosind algoritmul Iterației Valorilor.

Metoda computeActionFromValues(self, state) determină acțiunea optimă pentru o stare dată, având la bază valorile stărilor calculate. În situația în care există egalitate, se poate alege oricare dintre acțiuni.

Metoda getPolicy(self, state) furnizează politica optimală (acțiunea optimă) pentru o stare specificată..

Metoda getAction(self, state) oferă acțiunea optimă pentru o stare dată, fără a efectua explorări în acest context.

Metoda getQValue(self, state, action) returnează valoarea Q pentru o pereche specificată de stare și acțiune..

3 Policies

Expresia "Policies" descrie strategii sau ansambluri de reguli pe care un agent le adoptă în vederea luării deciziilor în cadrul unui MDP.

3a Configurația indică un factor de reducere scăzut (0.1), absența zgomotului (0.0) și o recompensă de -1.0. Alegerea unui factor de reducere mic sugerează o atenție deosebită acordată recompenselor imediate, în timp ce absența zgomotului indică un model de acțiuni deterministe. Recompensa negativă de -1.0 încurajează agentul să își finalizeze sarcina rapid.

3b Configurația are un factor de reducere scăzut (0.1), un nivel moderat de zgomot (0.1), și o recompensă de -1.0. Alegerea unui factor de reducere mic sugerează o concentrare asupra recompenselor imediate, iar nivelul moderat de zgomot introduce o anumită incertitudine în acțiuni. Recompensa negativă de -1.0 continuă să încurajeze agentul să își termine sarcina rapid.

3c Factorul de reducere este mai mare (0.9), zgomotul este la un nivel moderat (0.1), și recompensa este de -1.0. Un factor de reducere mai mare sugerează că agentul se concentrează asupra recompenselor viitoare, iar nivelul moderat de zgomot adaugă incertitudine în acțiuni.

3d Configurația are un factor de reducere mai mare (0.9), un nivel mai ridicat de zgomot (0.2), și o recompensă de -1.0. Factorul de reducere mai mare indică o atenție sporită asupra recompenselor viitoare, iar nivelul mai ridicat de zgomot introduce o incertitudine mai mare în acțiuni.

3e Configurația are un factor de reducere scăzut (0.1), un nivel mai ridicat de zgomot (0.2), și o recompensă de -1.0. Alegerea unui factor de reducere mic sugerează o atenție deosebită acordată recompenselor imediate, iar nivelul mai ridicat de zgomot adaugă mai multă incertitudine în acțiuni. Recompensa negativă de -1.0 încurajează agentul să își finalizeze sarcina rapid.