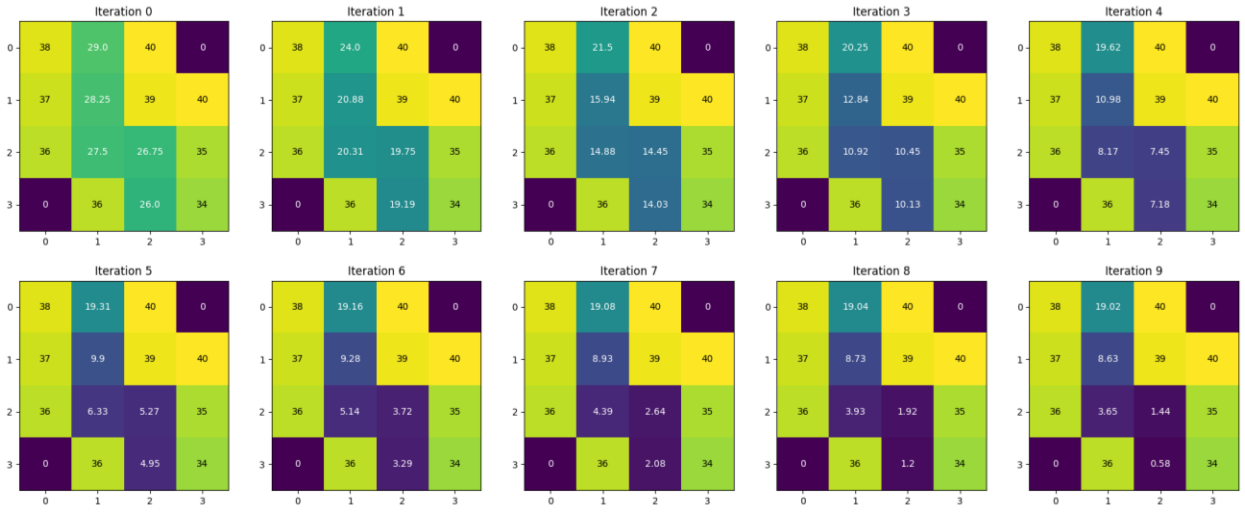
Opdracht 2.2
Adaptive Systems

Student:	Storm Joannes
Studentnummer:	1760581
Opleiding:	HBO-ICT Artificial Intelligence
Instelling:	Hogeschool Utrecht
Code:	2022_TICT_VINNO1-33_3_V
Datum:	09-04-2024

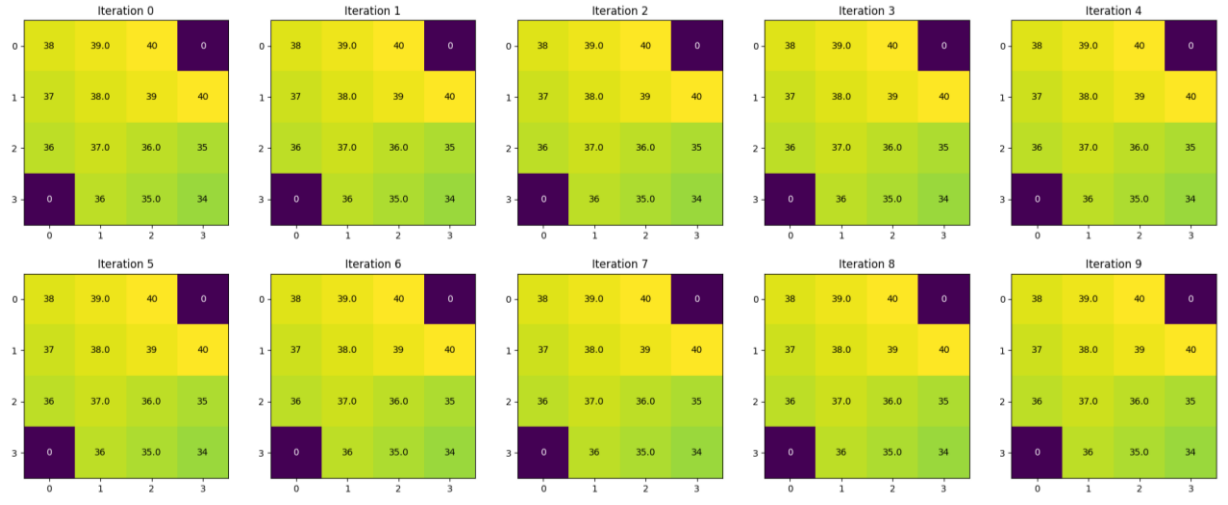
A. Temporal difference learning

De temporal differences worden berekend met 10 epochs, en een learning rate van 0,5.

De eerste figuur zijn de eind waardes van iedere episode temporal difference learning met een discount van 0.5. De episode werd beëindigd als de current position een terminal state was. Dit was in het geval van iedere iteratie positie (0, 3).



In de volgende figuur zijn de iteraties te zien van Temporal difference learning met een discount van 1. Ook hier eindigde iedere iteratie bij de terminal state (0, 3).

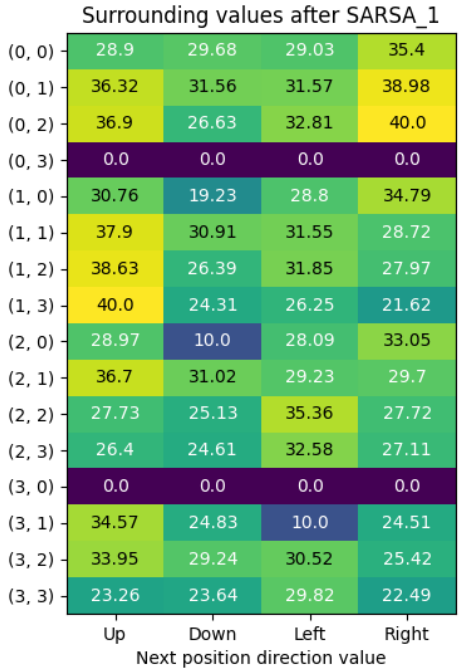


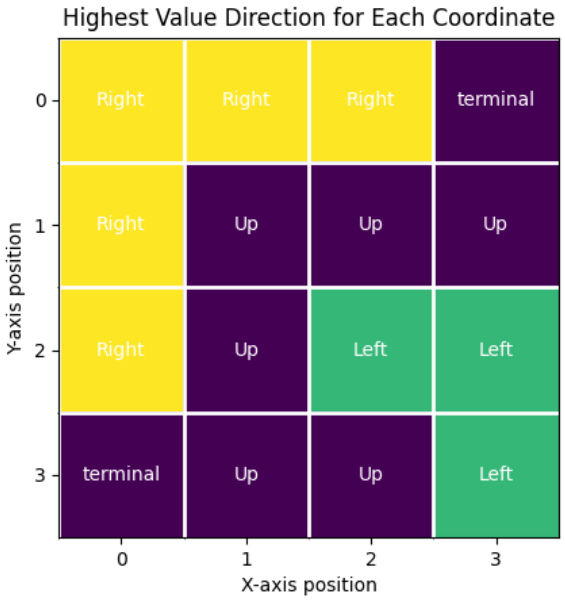
Het verschil tussen deze twee grafieken is duidelijk te zien door het verschil in waarden van de afgelopen posities. Het is duidelijk te zien dat de agent met de hogere discount veel sneller convergeert. Dit is te zien doordat de waardes in de plots vanaf de eerste iteratie al niet meer veranderen. In tegenstelling tot de agent met de lagere discount, neemt deze veel langer de tijd om te convergeren, en zoekt deze meer balans tussen de korte- en langetermijnbeloningen.

B. SARSA

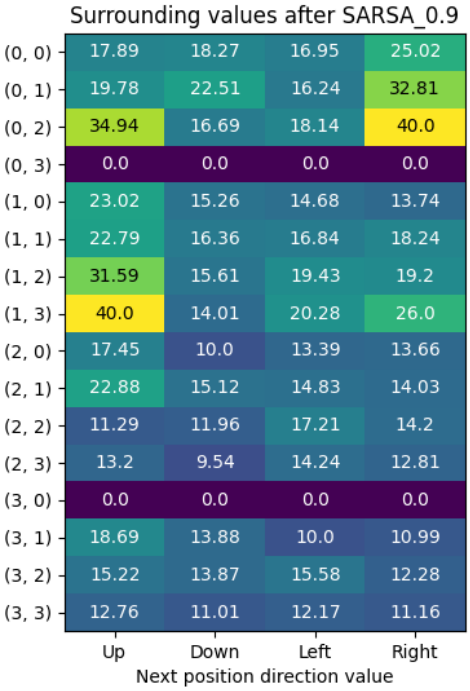
De instellingen waarmee wij dit model runnen is een learning rate van 0.25, een epsilon van 0.1 en het aantal epochs van 20.

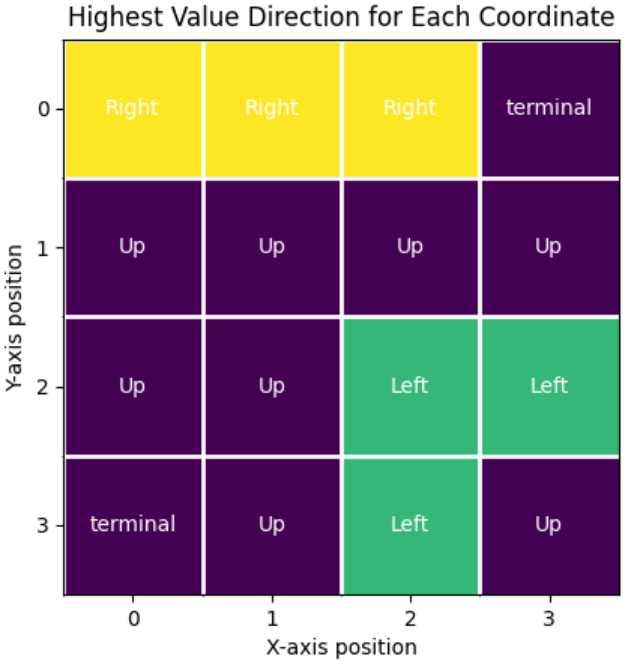
De eerste is SARSA met een discount van 1.0:





De tweede is SARSA met een discount 0.9:





Als we kijken naar de verschillen tussen de twee uitkomsten, zien we bij gekozen acties niet heel veel verschil. We zien dat beide een optimale policy zijn waarbij de agent vanuit de startpositie op een optimale manier je terminal state op (0, 3) weet te bereiken.

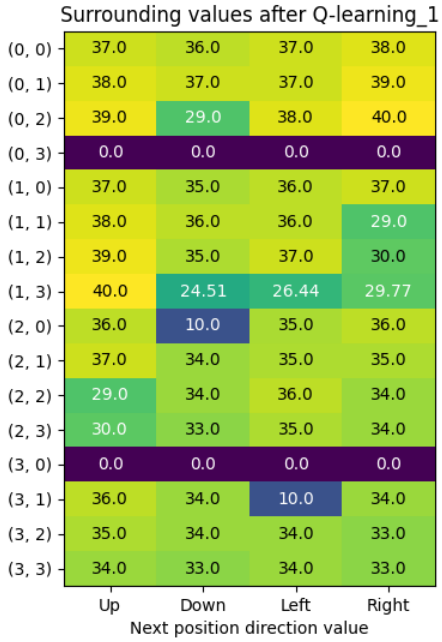
Echter zien we wel een groot verschil als we kijken naar alle waardes van de states. We zien dat met een discount van 0.9 de waardes veel lager zijn. Dit kan komen omdat het bij een discount van 0.9 de agent toekomstige beloningen minder waardeert dan bij een hogere discount. Hierdoor zullen waardes ook lager uitslaan.

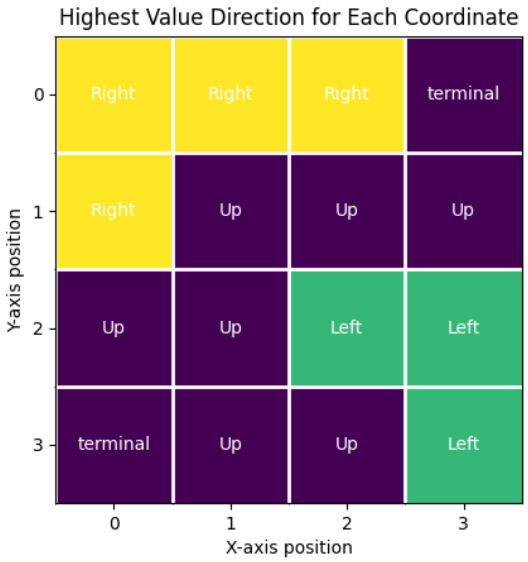
De resultaten die hieruit zijn gekomen zijn naar verwachting, en laten geen aparte uitkomsten zien.

C. Q-learning

We hebben hier Q-learning ookwel bekend als SARSAMAX toegepast met een learning rate van 0.25, en een epsilon van 0.1 en het aantal epochs van 30.000, na dit aantal epochs veranderde de resultaten niet meer. Daarnaast waren de resultaten van een discount van 0.9 en 1 hierbij hetzelfde. We hebben voor iedere positie de waardes van de omliggende posities bekeken waarna we een bepaling konden maken van welke stappen de agent zou nemen in de matrix.

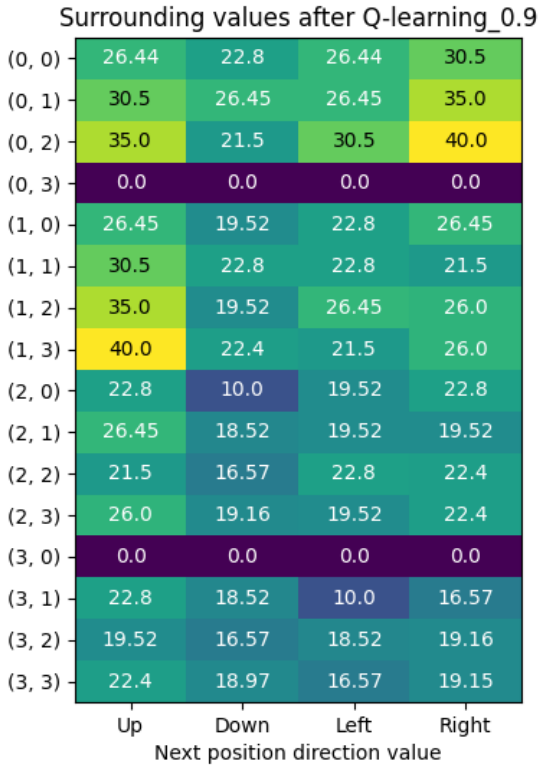
Uit Q-learning met een discount van 1:

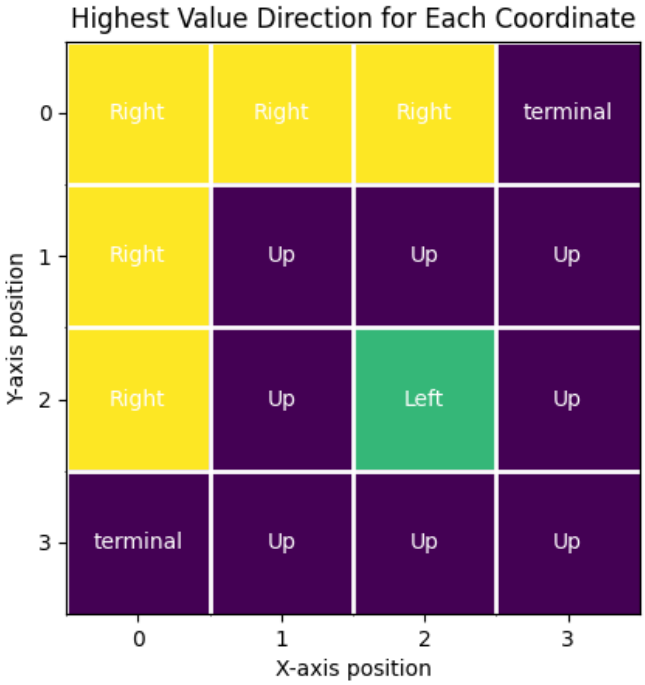




We kunnen hier zien dat de posities weergegeven in de plot acties weergeven om zich naar de terminal state te begeven. Iedere positie leidt tot de optimale policy.

Uit Q-learning met een discount van 0.9:





Ook hier zien we weer dat er een optimale policy is uitgekomen. Daarnaast zien we netals bij SARSA dat door de lagere discount de values van de actie states ook lager zijn. Toch kom je wel weer vanaf je start positie (3, 2) op de optimale manier naar de terminal state op (0, 3)