Deep Reinforcement Learning

Sheet 02

Sven Ullmann, Valentin Adam

Exercise 1

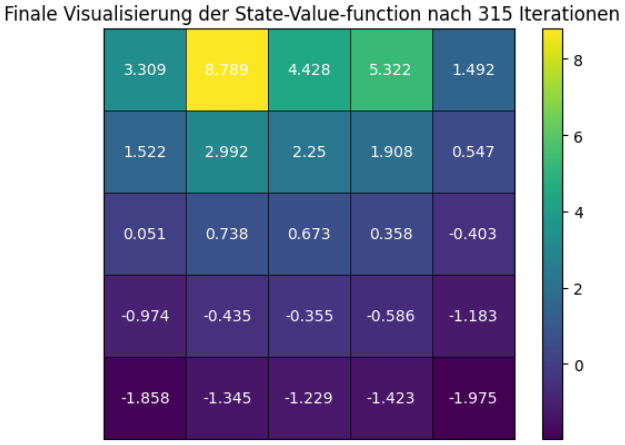
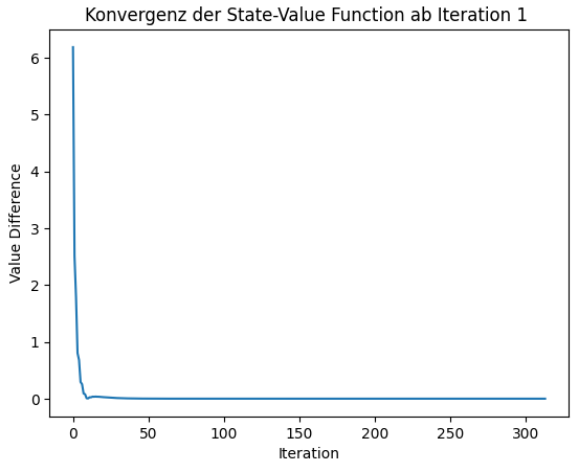
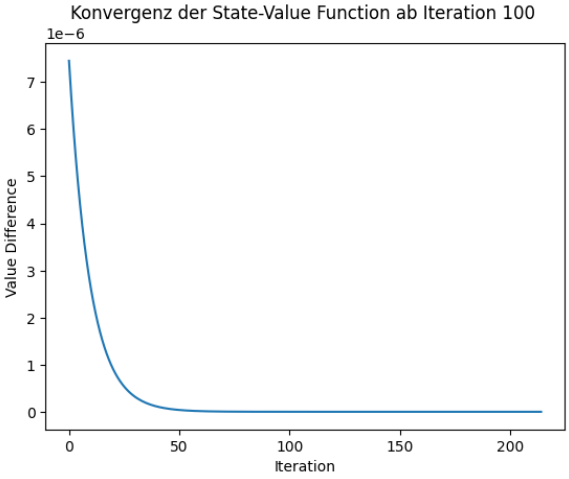


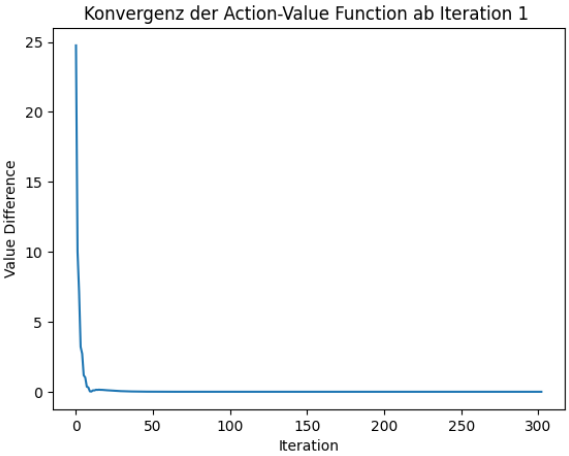
Figure 1: State Value nach Konvergenz

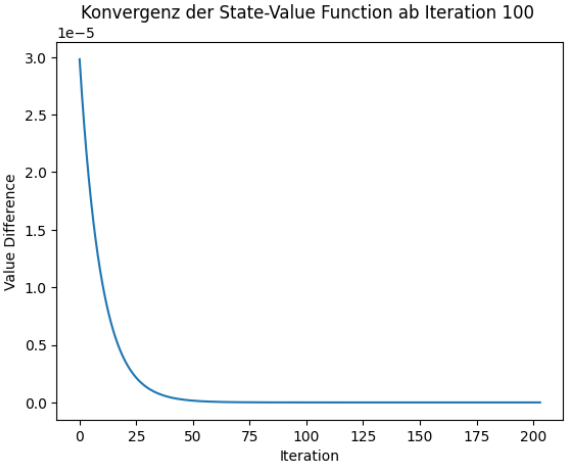
In Figure 1 erkennt man, dass der State value für das Feld A am höchsten ist. Der zweitgrößte Wert findet sich auf Feld B. Dies ist wenig verwunderlich, da dort der größte Reward ausgeschüttet wird. Dies erklärt auch, warum Felder rund um A und B ebenfalls recht große State values besitzen, denn man kommt schnell zu A und B. Die Felder in den Ecken links und rechts unten haben die geringsten State values, denn hier kann läuft man in 50 % der Fällen (Ost und Süd, bzw West und Süd) aus dem Gebiet hinaus und erhält einen negativen Reward von -1. Der Grund warum der State Value dort nicht bei -2 sondern leicht darüber liegt ist der Discountfaktor $\gamma = 0.9$, der relevant wird, wenn man von einem dieser Felder ein Feld mit positivem Reward wie A oder B erreicht.

Man erkennt in Figure 2, dass sowohl für die State-Value function als auch für die action-value function eine sehr schnelle Konvegenz vorherrscht. Die beiden functionen konvergieren nahezu identisch, da in dem Beispiel eine action immer zu einem festen State führt und damit keine zusätzliche Wahrscheinlichkeitsverteilung zugrunde liegt, weshalb die beiden Bellman Formeln nahezu identisch werden.

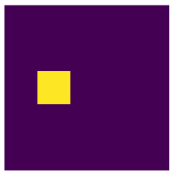

Konvergenz der State-Value Function ab Iteration 1
Value Difference
Iteration
0
1
2
3
4
5
6
0
50
100
150
200
250
300


Konvergenz der State-Value Function ab Iteration 100
1e−6
Value Difference
Iteration
0
1
2
3
4
5
6
7
0
50
100
150
200


Konvergenz der Action-Value Function ab Iteration 1
Value Difference
Iteration
0
5
10
15
20
25
0
50
100
150
200
250
300


Konvergenz der State-Value Function ab Iteration 100
1e−5
Value Difference
Iteration
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
25
50
75
100
125
150
175
200

Exercise 2



(a) Schritt 1

(b) Schritt 2

(c) Schritt 3 – > Special State A

(d) Schritt 4 – > Special State A'

Exercise 3

b) Intuitve approach:

Wir haben zwei intuitive Approaches entwickelt (welche wir mit S (simple) und A (advanced) abkürzen). In S: Bewegen nach links, wenn der Neigungswinkel < 0 ist, bewegen nach rechts, wenn der Neigungswinkel > 0. Diese Idee hat im Schnitt einen Reward von 42 erhalten (dh 42 Schritte). Danach war der Winkel zu groß und das System ist terminiert.
Für A: Dort haben wir zusätzlich zum Neigungswinkel auch noch die Geschwindigkeit des Pendels berücksichtigt, in der Entscheidung ob wir uns nach links oder rechts bewegen sollen. Diese einfach Verbesserung hat bereits zu wesentlich besseren Ergebnissen geführt. Es wurde ein durchschnittlicher Reward von 499.5 erreicht, dann wurde truncated (dh das iteration limit wurde fast immer erreicht, denn der maximal zu erreichende Reward wäre 500 gewesen).