Künstliche Intelligenz

Übungsblatt #6
Lernen
Version 1.2

Prof. Dr. J. Fürnkranz, Dr. G. Grieser

Aufgabe 6.1

Es sei das folgende Neuronale Netz gegeben:

- das Neuron 0 ist ein Biasneuron, das immer den Wert $+1$ liefert.
- die Neuronen 1,2,3,4 sind die Eingabeneuronen, das Neuron 8 ist das Ausgabeneuron
- das Neuron 8 ist durch die Aktivierungsfunktion $g_8(in_8) = \frac{1}{1+e^{-in_8}}$ beschrieben.
- die Neuronen 5, 6 und 7 sind durch die Aktivierungsfunktion $g_i(in_i) = 0,1 \cdot in_i$ beschrieben

Die Gewichte sind durch die folgende Matrix $[w_{i,j}]_{i,j}$ gegeben:

$$
\begin{array}{c} \\ i \\ \downarrow \end{array}
\begin{array}{c}
\begin{array}{ccccccccc} & & & & j \rightarrow & & & & \end{array} \\
\left[\begin{array}{ccccccccc}
0 & 0 & 0 & 0 & 0 & 1 & \text{-0,5} & \text{1,2} & 3 \\
0 & 0 & 0 & 0 & 0 & 2 & \text{0,5} & \text{0,3} & \text{-0,8} \\
0 & 0 & 0 & 0 & 0 & \text{-1,2} & \text{1,5} & 3 & 7 \\
0 & 0 & 0 & 0 & 0 & 3 & \text{0,4} & \text{-3} & \text{-4} \\
0 & 0 & 0 & 0 & 0 & \text{-2} & 4 & \text{0,1} & 2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \text{-2} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right]
\end{array}
$$

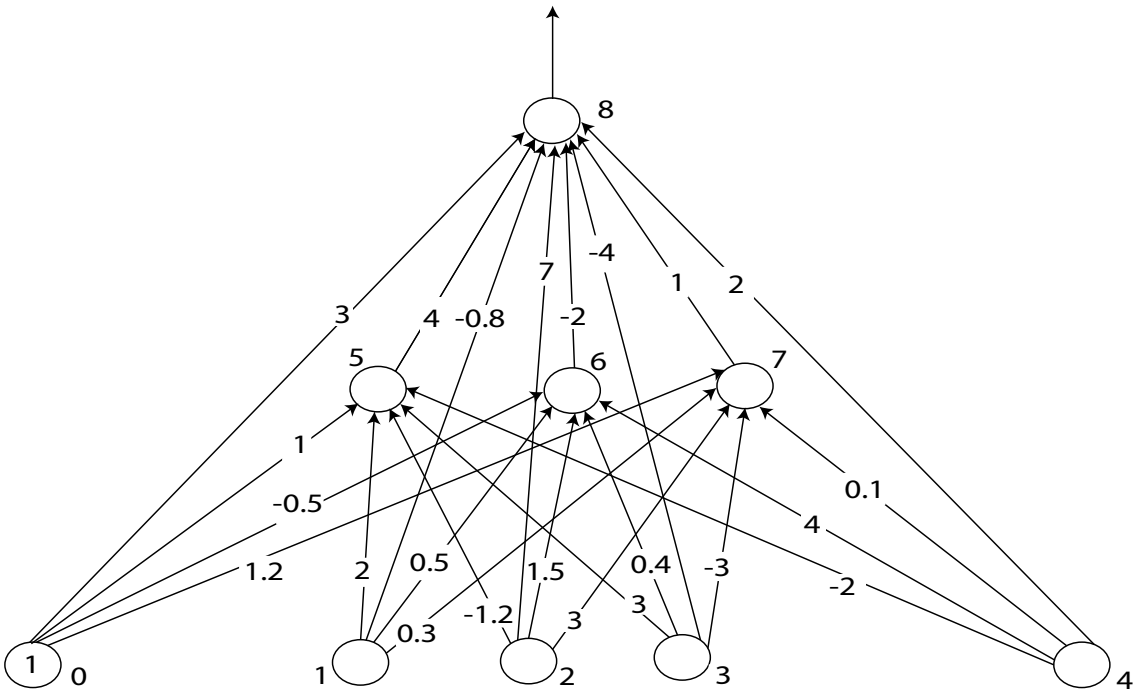
a) In Russel/Norvig ist das Bias-Neuron so definiert, daß es immer den Wert -1 liefert. Diskutieren Sie den Unterschied in der Leistungsfähigkeit/Repräsentationsfähigkeit von Netzen mit Biasneuron +1 bzw. -1.

Lösungsvorschlag:

Es gibt keinen Unterschied, da sich das eine Netz in das andere übersetzen läßt, indem einfach alle vom Biasneuron ausgehenden Gewichte mit -1 multipliziert werden.

b) Visualisieren Sie die Netzwerkstruktur einschl. der Gewichte.

Lösungsvorschlag:



c) Was ist die Ausgabe des Netzwerkes für die Eingabe $\langle 0, 1, 2, 3\rangle$?

Lösungsvorschlag:

Neuron 5 $in_5 = a_0 \cdot w_{05} + a_1 \cdot w_{15} + a_2 \cdot w_{25} + a_3 \cdot w_{35} + a_4 \cdot w_{45}$
$= 1 \cdot 1 + 0 \cdot 2 + 1 \cdot (-1,2) + 2 \cdot 3 + 3 \cdot (-2)$
$= -0,2$
$\Rightarrow a_5 = 0,1 \cdot (-0,2)$
$\Rightarrow a_5 = -0,02$

Neuron 6 $in_6 = a_0 \cdot w_{06} + a_1 \cdot w_{16} + a_2 \cdot w_{26} + a_3 \cdot w_{36} + a_4 \cdot w_{46}$
$= 1 \cdot (-0,5) + 0 \cdot 0,5 + 1 \cdot 1,5 + 2 \cdot 0,4 + 3 \cdot 4$
$= 13,8$
$\Rightarrow a_6 = 0,1 \cdot 13,8$
$\Rightarrow a_6 = 1,38$

Neuron 7 $in_7 = a_0 \cdot w_{07} + a_1 \cdot w_{17} + a_2 \cdot w_{27} + a_3 \cdot w_{37} + a_4 \cdot w_{47}$
$= 1 \cdot 1,2 + 0 \cdot 0,3 + 1 \cdot 3 + 2 \cdot (-3) + 3 \cdot 0,1$
$= -1,5$
$\Rightarrow a_7 = 0,1 \cdot (-1,5) \leq +1$
$\Rightarrow a_7 = -0,15$

Neuron 8 $in_8 = a_0 \cdot w_{08} + a_1 \cdot w_{18} + a_2 \cdot w_{28} + a_3 \cdot w_{38} + a_4 \cdot w_{48} + a_5 \cdot w_{58} + a_6 \cdot w_{68} + a_7 \cdot w_{78}$
$= 1 \cdot 3 + 0 \cdot (-0,8) + 1 \cdot 7 + 2 \cdot (-4) + 3 \cdot 2 + (-0,02) \cdot 4 + 1,38 \cdot (-2) + (-0,15) \cdot 1$
$= 5,01$
$\Rightarrow a_8 = \frac{1}{1+e^{-5,01}} = 0,9934$

Die Ausgabe des Netzes ist die Ausgabe des Neurons 8, d,h, 0,9934.

d) Sei nun die Zielklassifikation für die Instanz $\langle 0, 1, 2, 3\rangle$ der Wert 0. Passen Sie die Gewichte aller Neuronen mittels des Backpropagation-Algorithmus an, Als Lernrate benutzen Sie $\alpha = 0, 1$.

Lösungsvorschlag:

Zunächst wird der Fehler des obersten Neurons, d.h. Neuron 8, berechnet:

$\mathit{Err}_8 = 0 - 0, 9934 = -0, 9934.$

Eine weitere wichtige Größe ist $\Delta_8 = \mathit{Err}_8 \cdot g'_8(in_8)$.

Da $g_8(x) = \frac{1}{1+e^{-x}}$ ist ergibt sich als erste Ableitung $g'_8(x) = g(x)(1 - g(x))$ und wir erhalten:

$\Delta_8 = \mathit{Err}_8 * \cdot g(in_8)(1 - g(in_8)) = (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 = -0, 0065.$

Nun können wir die Gewichte für eingehende Kante nach Neuron 8 anpassen. Im allgemeinen gilt folgende Formel:

$$w_{i8} = w_{i8} + \alpha \cdot \Delta_8 \cdot a_i.$$
$$w_{i8} = w_{i8} + \alpha \cdot \mathit{Err}_8 \cdot g(in_8)(1 - g(in_8)) \cdot a_i.$$

und damit

$$w_{i8} = w_{i8} + \alpha \cdot \mathit{Err}_8 \cdot a_8 \cdot (1 - a_8) \cdot a_i.$$

und eingesetzt schließlich

$$w_{i8} = w_{i8} + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot a_i.$$

Damit erhalten wir folgende aktualisierten Gewichte:

$$\begin{aligned}
w_{08} &= w_{08} + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot a_0 \\
&= 3 + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot 1 \\
&= \underline{2, 9993}
\end{aligned}$$

$$\begin{aligned}
w_{18} &= w_{18} + \alpha \cdot a_1 \cdot \mathit{Err}_8 \\
&= -0, 8 + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot 0 \\
&= \underline{-0, 8}
\end{aligned}$$

$$\begin{aligned}
w_{28} &= w_{28} + \alpha \cdot a_2 \cdot \mathit{Err}_8 \\
&= 7 + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot 1 \\
&= \underline{6, 9993}
\end{aligned}$$

$$\begin{aligned}
w_{38} &= w_{38} + \alpha \cdot a_3 \cdot \mathit{Err}_8 \\
&= -4 + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot 2 \\
&= \underline{-4, 0013}
\end{aligned}$$

$$\begin{aligned}
w_{48} &= w_{48} + \alpha \cdot a_4 \cdot \mathit{Err}_8 \\
&= 2 + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot 3 \\
&= \underline{2, 0020}
\end{aligned}$$

$$\begin{aligned}
w_{58} &= w_{58} + \alpha \cdot a_5 \cdot \mathit{Err}_8 \\
&= 4 + 0, 1 \cdot (-0, 9934) \cdot 0, 9934 \cdot 0, 0066 \cdot (-0, 02) \\
&= \underline{4, 0000}
\end{aligned}$$

$$
\begin{aligned}
w_{68} &= w_{68} + \alpha \cdot a_6 \cdot \mathit{Err}_8 \\
&= -2 + 0,1 \cdot (-0,9934) \cdot 0,9934 \cdot 0,0066 \cdot 1,38 \\
&= \underline{-2,0010}
\end{aligned}
$$

$$
\begin{aligned}
w_{78} &= w_{78} + \alpha \cdot a_7 \cdot \mathit{Err}_8 \\
&= 1 + 0,1 \cdot (-0,9934) \cdot 0,9934 \cdot 0,0066 \cdot (-0,15) \\
&= \underline{1,0001}
\end{aligned}
$$

Als nächstes bestimmen wir den Fehler Δ_j für die Neuronen im Zwischenlayer, d.h. $j = 5, 6, 7$. Im allgemeinen gilt:

$$\Delta_i = g_i'(\mathit{in}_i) \cdot \textstyle\sum_j w_{ij} \cdot \Delta_j$$

Da wir für alle Neuronen 5,6,7 jeweils nur eine ausgehende Kante haben, reduziert sich die Berechnung auf

$$\Delta_i = g_i'(\mathit{in}_i) \cdot w_{i8} \cdot \Delta_8$$

Die Ableitung der Aktivierungsfunktion $g_i(\mathit{in}_i) = 0,1 \cdot \mathit{in}_i$ der mittleren Neuronen ist $g_i'(x) = 0,1$, somit ergibt sich

$$\Delta_i = 0,1 \cdot w_{i8} \cdot -0,0065$$

Für die einzelnen Neuronen ergibt dies:

$$\Delta_5 = 0,1 \cdot 4 \cdot -0,0065 = -0,0026$$

$$\Delta_6 = 0,1 \cdot (-2) \cdot -0,0065 = 0,0013$$

$$\Delta_7 = 0,1 \cdot 1 \cdot -0,0065 = -0,0007$$

Dies erlaubt es uns nun, die restlichen Gewichte zwischen dem Input- und dem Hidden-Layer zu adaptieren:

$$w_{ij} = w_{ij} + \alpha \cdot \Delta_j \cdot a_i$$

$$
\begin{aligned}
w_{05} &= w_{05} + \alpha \cdot \Delta_5 \cdot a_0 \\
&= 1 + 0,1 \cdot (-0,0026) \cdot 1 \\
&= \underline{0,9997}
\end{aligned}
$$

$$
\begin{aligned}
w_{15} &= w_{15} + \alpha \cdot \Delta_5 \cdot a_1 \\
&= 2 + 0,1 \cdot (-0,0026) \cdot 0 \\
&= \underline{2}
\end{aligned}
$$

$$
\begin{aligned}
w_{25} &= w_{25} + \alpha \cdot \Delta_5 \cdot a_2 \\
&= -1,2 + 0,1 \cdot (-0,0026) \cdot 1 \\
&= \underline{-1,2003}
\end{aligned}
$$

$$
\begin{aligned}
w_{35} &= w_{35} + \alpha \cdot \Delta_5 \cdot a_3 \\
&= 3 + 0,1 \cdot (-0,0026) \cdot 2 \\
&= \underline{2,9995}
\end{aligned}
$$

$$
\begin{aligned}
w_{45} &= w_{45} + \alpha \cdot \Delta_5 \cdot a_4 \\
&= -2 + 0,1 \cdot (-0,0026) \cdot 3 \\
&= \underline{-2,0008}
\end{aligned}
$$

Analog ergibt sich für Neuron 6:

$$
\begin{aligned}
w_{06} &= -0,5 + 0,1 \cdot 0,0013 \cdot 1 \\
&= \underline{-0,4999}
\end{aligned}
$$

$$
\begin{aligned}
w_{16} &= 0,5 + 0,1 \cdot 0,0013 \cdot 0 \\
&= \underline{0,5} \\
w_{26} &= 1,5 + 0,1 \cdot 0,0013 \cdot 1 \\
&= \underline{1,5001} \\
w_{36} &= 0,4 + 0,1 \cdot 0,0013 \cdot 2 \\
&= \underline{0,4003} \\
w_{46} &= 4 + 0,1 \cdot 0,0013 \cdot 3 \\
&= \underline{4,0004}
\end{aligned}
$$

und schließlich für Neuron 7:

$$
\begin{aligned}
w_{07} &= 1,2 + 0,1 \cdot (-0,007) \cdot 1 \\
&= \underline{1,1993} \\
w_{17} &= 0,3 + 0,1 \cdot (-0,007) \cdot 0 \\
&= \underline{0,3} \\
w_{27} &= 3 + 0,1 \cdot (-0,007) \cdot 1 \\
&= \underline{2,9993} \\
w_{37} &= -3 + 0,1 \cdot (-0,007) \cdot 2 \\
&= \underline{-3,0014} \\
w_{47} &= 0,1 + 0,1 \cdot (-0,007) \cdot 3 \\
&= \underline{0,0979}
\end{aligned}
$$

e) Berechnen Sie den Ausgabewert des neuen Netzes und vergleichen Sie ihn mit der ursprünglichen Berechnung.

Lösungsvorschlag:

Neuron 5 $in_5 = 1 \cdot 0,9997 + 0 \cdot 2 + 1 \cdot (-1,2003) + 2 \cdot 2,9995 + 3 \cdot (-2,0008)$
$= -0,2040$ (war: $-0,2$)
$\Rightarrow a_6 = 0,1 \cdot -0,204 = -0,0204$ (war: $-0,02$)

Neuron 6 $in_6 = 1 \cdot 0,4999 + 0 \cdot 0,5 + 1 \cdot 1,5001 + 2 \cdot 0,4003 + 3 \cdot 4,004$
$= 14,8126$ (war: $13,8$)
$\Rightarrow a_6 = 0,1 \cdot 14,8126 = 1,4813$ (war: $1,38$)

Neuron 7 $in_7 = 1 \cdot 1,1993 + 0 \cdot 0,3 + 1 \cdot 2,9993 + 2 \cdot (-3,0014) + 3 \cdot 0,0979$
$= -1,5105$ (war: $-1,5$)
$\Rightarrow a_7 = 0,1 \cdot (-1,5105) = -0,1511$ (war: $-0,15$)

Neuron 8 $in_8 = 1 \cdot 2,9993 + 0 \cdot (-0,8) + 1 \cdot 6,9993 + 2 \cdot (-4,0013) + 3 \cdot 2,0020 + (-0,0204) \cdot 4 + 1,4813 \cdot (-2,0010) + (-0,1511) \cdot 1,0001$
$= 4,8052$ (war: $5,01$)
$\Rightarrow a_8 = \frac{1}{1+e^{-4,8052}} = 0,9919$ (war: $0,9934$)

Das Ergebnis des Netzes hat sich also (wenn auch nur ein wenig) in Richtung des Zielwertes (d.h. 0) verändert.

Aufgabe 6.2

Ein Agent bewegt sich in einer einfachen Welt, die wie folgt angeordnet ist:

a	b	c
d	e	f
g	h	i

Der Agent kann sich jeweils ein Feld nach unten, oben, links oder rechts bewegen, falls dort ein Feld ist. Jeder Schritt kostet 0,1 Punkt. Wenn der Agent im Feld f landet, erhält er einen Reward von 1 Punkt und kann sich von dort nicht mehr wegbewegen, auf allen anderen Feldern erhält er einen Reward von 0 Punkten.

Als Discountfaktor setzen wir $\gamma = 0,9$.

a) Formulieren Sie die Reward-Funktion.

Lösungsvorschlag:

$r(a,u) = -0,1$			$r(a,r) = -0,1$
$r(b,u) = -0,1$		$r(b,l) = -0,1$	$r(b,r) = 0,9$
$r(c,u) = 0,9$		$r(c,l) = -0,1$	
$r(d,u) = -0,1$	$r(d,o) = -0,1$		$r(d,r) = -0,1$
$r(e,u) = -0,1$	$r(e,o) = -0,1$	$r(e,l) = -0,1$	$r(e,r) = 0,9$
	$r(g,o) = -0,1$		$r(g,r) = -0,1$
	$r(h,o) = -0,1$	$r(h,l) = -0,1$	$r(h,r) = -0,1$
	$r(i,o) = 0,9$	$r(i,l) = -0,1$	

b) Berechnen Sie die Bewertungsfunktion $V^\pi(s)$ für die Strategie π:
- wenn dies möglich ist, gehe nach oben; ansonsten:
- wenn dies möglich ist, gehe nach rechts; ansonsten:
- wenn dies möglich ist, gehe nach unten; ansonsten:
- gehe nach links

Lösungsvorschlag:

Überlegen wir uns beispielhaft die Bewertung des Feldes g. Im allgemeinen wird der Reward berechnet als $V^\pi = \sum_{k=0}^{\infty} \gamma^k \cdot r_k$. Laut Policy π bewegt sich der Agent ausgehend von g wie folgt: $\rightarrow d \rightarrow a \rightarrow b \rightarrow c \rightarrow f$, im Feld f ist keine Aktion mehr möglich.

Die Bewertung $V^\pi(g)$ ergibt sich also als

$$r(g,o) + \gamma \cdot r(d,o) + \gamma^2 \cdot r(a,r) + \gamma^3 \cdot r(b,r) + \gamma^4 \cdot r(c,u)$$

$$= -0,1 + 0,9 \cdot -0,1 + 0,9^2 \cdot -0,1 + 0,9^3 \cdot -0,1 + 0,9^4 \cdot 0,9$$

$$= 0,25$$

Insgesamt erhalten wir die folgenden Werte:

$V^\pi(a) = 0,54$	$V^\pi(b) = 0,71$	$V^\pi(c) = 0,9$
$V^\pi(d) = 0,39$	$V^\pi(e) = 0,54$	$V^\pi(f) = 0$
$V^\pi(g) = 0,25$	$V^\pi(h) = 0,39$	$V^\pi(i) = 0,9$

c) Berechnen Sie die optimale Bewertungsfunktion $V^*(s)$.

Lösungsvorschlag:

Wir überlegen uns für jedes Feld, welches ein optimaler Weg wäre.

Für das Feld g beispielsweise würde der Weg $\rightarrow h \rightarrow i \rightarrow f$ optimalen Reward bringen, genauso wie $\rightarrow d \rightarrow e \rightarrow f$, nämlich

$$= -0,1 + 0,9 \cdot -0,1 + 0,9^2 \cdot -0,1 + 0,9^3 \cdot 0,9 = 0,39$$

Somit erhalten wir die folgenden Werte:

$V^*(a) = 0,54$	$V^*(b) = 0,71$	$V^*(c) = 0,9$
$V^*(d) = 0,71$	$V^*(e) = 0,9$	$V^*(f) = 0$
$V^*(g) = 0,54$	$V^*(h) = 0,71$	$V^*(i) = 0,9$

d) Bestimmen Sie die Q-Funktion.

Lösungsvorschlag:

Die Q-Funktion ist der kumulierte Reward für das Anwenden einer Aktion a im Feld s und dem darauffolgenden Anwenden der optimalen Strategie, d.h.

$$Q(s,a) = r(s,a) + \gamma v^*(s')$$

Im Feld d beispielsweise erhalten wir

- $Q(d,u) = r(d,u) + \gamma \cdot V^*(g) = -0,1 + 0,9 \cdot 0,54 = 0,39$
- $Q(d,o) = r(d,o) + \gamma \cdot V^*(a) = -0,1 + 0,9 \cdot 0,54 = 0,39$
- $Q(d,r) = r(d,r) + \gamma \cdot V^*(e) = -0,1 + 0,9 \cdot 0,9 = 0,71$

Insgesamt ergibt dies:

$Q(a,u) = 0,54$			$Q(a,r) = 0,54$
$Q(b,u) = 0,71$		$Q(b,l) = 0,54$	$Q(b,r) = 0,71$
$Q(c,u) = 0,9$		$Q(c,l) = 0,54$	
$Q(d,u) = 0,39$	$Q(d,o) = 0,39$		$Q(d,r) = 0,71$
$Q(e,u) = 0,54$	$Q(e,o) = 0,54$	$Q(e,l) = 0,54$	$Q(e,r) = 0,9$
	$Q(g,o) = 0,54$		$Q(g,r) = 0,54$
	$Q(h,o) = 0,71$	$Q(h,l) = 0,39$	$Q(h,r) = 0,71$
	$Q(i,o) = 0,9$	$Q(i,l) = 0,54$	

e) Geben Sie eine optimale Policy an.

Lösungsvorschlag:

Die optimale Policy wählt in jedem Feld diejenige Aktion aus, die den höchsten diskontierten Gewinn verspricht:

$$\pi^*(s) = \operatorname{argmax}_a \left(r(s,a) + \gamma V^*\left(\delta(s,a)\right)\right)$$

Alternativ kann man auch die Q-Funktion benutzen und jeweils die Aktion mit dem höchsten Q-Wert auswählen.

Insgesamt:

$\downarrow\rightarrow$	$\downarrow\rightarrow$	$\downarrow$
$\rightarrow$	$\rightarrow$	%
$\uparrow\rightarrow$	$\uparrow\rightarrow$	$\uparrow$

f) Versuchen Sie, mittels Q-Learning die Q-Funktion zu lernen, indem Sie den Agenten auf ein zufällig gewähltes Anfangsfeld stellen und die jeweils beste Aktion nach der momentanen Q-Funktion ausführen (bei Gleichheit zufällige Auswahl), bis der Agent am Ziel angekommen ist und das ganze bis zur Konvergenz wiederholen. Als Lernrate können Sie 1 annehmen.

Lösungsvorschlag:

Wir intialisieren zunächst alle Werte $\hat{Q}(s,a)$ mit 0:

a 0 0	0 *b* 0 0	0 *c* 0
0 *d* 0 0	0 0 *e* 0 0	*f*
0 *g* 0	0 0 *h* 0	0 *i*

Wir wählen zufällig ein Feld aus, sagen wir g. Da die beiden Aktionen o und r gleich bewertet sind, wählen zufällig eine aus, sagen, wir r.

Nun ergibt sich der neue Wert $\hat{Q}(g,r) = \hat{Q}(g,r) + \alpha(r(g,r) + \gamma \cdot \max_a \hat{Q}(h,a) - \hat{Q}(g,r))$. Da wir α der Einfachheit halber auf auf 1 gesetzt haben (d.h. der alte Wert wird nicht berücksichtigt) ergibt sich $\hat{Q}(g,r) = r(g,r) + \gamma \cdot \max_a \hat{Q}(h,a)$. Alle von h ausgehenden Aktionen haben den Wert 0, deshalb erhalten wir $\hat{Q}(g,r) = -0,1 + 0,9 \cdot 0 = -0,1$:

a 0 0	0 *b* 0 0	0 *c* 0
0 *d* 0 0	0 0 *e* 0 0	*f*
0 *g* -0,1	0 0 *h* 0	0 0 *i*

Im Feld h wählen wir zufällig die Aktion o und erhalten analog:

a 0 / 0	0 b 0 / 0	0 c / 0
0 / d 0 / 0	0 / e 0 / 0	f
0 / g -0,1	-0,1 / 0 h 0	0 i

Im Feld e wählen wir zufällig die Aktion r und erhalten:

a 0 / 0	0 b 0 / 0	0 c / 0
0 / d 0 / 0	0 / e 0,9 / 0	f
0 / g -0,1	-0,1 / 0 h 0	0 i

Damit ist die erste Runde beendet.

Als nächstes starten wir z.B. in d und wählen die Aktion o, dies ergibt:

a 0 / 0	0 b 0 / 0	0 c / 0
-0,1 / d 0 / 0	0 / e 0,9 / 0	f
0 / g -0,1	-0,1 / 0 h 0	0 i

Im Feld a wird zufällig die Aktion u ausgewählt, und wir sind wieder im Feld d:

a 0 / -0,1	0 b 0 / 0	0 c / 0
-0,1 / d 0 / 0	0 / e 0,9 / 0	f
0 / g -0,1	-0,1 / 0 h 0	0 i

Nun sind laut unserer Methode nur noch 2 Aktionen möglich, da wir stets eine maximal auswählen: u und r. Wir wählen r. Hiermit ergibt sich nun $\hat{Q}(d, r) = r(d, r) + \gamma \cdot$

$\max_a \hat{Q}(e, a)$. Laut unserer aktuellen Q-Funktion ist im Feld e die optimale Aktion mit 0,9 bewertet, deshalb erhalten wir $\hat{Q}(d, r) = -0,1 + 0,9 \cdot 0,9 = 0,71$:

a 0 -0,1	0 *b* 0 0	0 *c* 0
-0,1 *d* 0,71 0	0 *e* 0,9 0	*f*
0 *g* -0,1	-0,1 0 *h* 0	0 *i*

Im Feld e wird nun die Aktion r ausgewählt, die Matrix ändert sich dabei nicht.

Starten wir nun im Feld i und wählen folgenden Weg:

→	→	
↑	←	←

dann ergibt sich hinterher:

a 0 -0,1	0 *b* 0 0	0 *c* 0
-0,1 *d* 0,71 0	0 *e* 0,9 0	*f*
0,54 *g* -0,1	-0,1 -0,1 *h* 0	0 -0,1 *i*

Ein Start in h führt nun zwangsläufig zu

	→	↑

und wir erhalten:

a 0 -0,1	0 *b* 0 0	0 *c* 0
-0,1 *d* 0,71 0	0 *e* 0,9 0	*f*
0,54 *g* -0,1	-0,1 -0,1 *h* -0,1	0,9 -0,1 *i*

Startet man nun wieder in h und wählt als erste Aktion l, so ergibt sich

→	→	
↑	←	

a 0 -0,1	0 b 0 0	0 c 0
-0,1 d 0	0 0,71 0 e 0 0,9	f
0,54 g	-0,1 0,39 h -0,1	0,9 -0,1 i

Aufgrund unserer Strategie (den Agenten auf ein zufällig gewähltes Anfangsfeld stellen und die jeweils beste Aktion nach der momentanen Q-Funktion ausführen) werden sich die Werte in den Feldern $d \ldots i$ nicht mehr ändern, da bestimmte Wege nicht mehr ausprobiert werden. Auf diesen Feldern hat also bereits eine Konvergenz stattgefunden.

Auf das Berechnen der Werte für die Felder $a \ldots c$ verzichten wir.