Prof. Dr. Leif Döring
André Ferdinand, Sara Klein

Reinforcement Learning

7. Solution Sheet

1. Proof Theorem 4.2.9.

Close the gap in the proof of Theorem 4.2.9. from the lecture. Therefore go through the one dimensional proof of Theorem 4.2.8. and check that also in the d-dimensional case there exits a sequence $t_k \to \infty$ such that $\sup_{t \geq t_k} |x(t)| \leq D_k$ almost surely and $\lim_{k\to\infty} D_k = 0$.
Solution:
From the lecture we already know that $\sup_{t\geq 0}\|x(t)\|_\infty < \infty$ *almost surely. Thus define* $D_0 = \sup_{t\geq 0}\|x(t)\|_\infty$. *Exactly as in the one dimensional case, we set* $D_{k+1} = \beta(1+3\epsilon)D_k$ *for some* $\epsilon > 0$ *such that* $(1+2\epsilon)\beta < 1$, *i.e.* $D_k \to 0$ *for* $k \to \infty$. *Now we inductively show that there exists a random sequence* $(t_k)_{k\geq 0}$ *such that* $t_k \to \infty$ *for* $k \to \infty$. *and* $\sup_{t\geq t_k}\|x(t)\|_\infty \leq D_k$ *for all* $k \geq 0$.
Induction beginning: *We set* $t_0 = 0$, *then the induction beginning follows from the choice of* D_0.
Now suppose that t_k *is given such that* $\sup_{t\geq t_k}\|x(t)\|_\infty \leq D_k$ *almost surely.*
Induction conclusion: *First recall the process* W *from Lemma 4.2.5 and denote for all* $i = 1,\dots,d$ *by* $W_i(\cdot:\cdot)$ *this process defined by the error sequence* ϵ_i *from the theorem. Next we define*

$$\tau = \min\{s \geq t_k \,:\, W_i(t:s) < \beta\epsilon D_k \,\forall i = 1,\dots,d, t \geq s\},$$

the $\tau < \infty$ *almost surely because for all* $i = 1,\dots,d$ *we have that* $W_i(t:s) \to \infty$ *for* $t \to \infty$ *almost surely. Define the process*

$$Y_i(t+1:\tau) + (1+\alpha_i(t))Y_i(t:\tau) + \alpha_i(t)A,$$

for $A = \beta D_k$ *started at time* τ *in* D_k, *i.e.* $Y_i(\tau:\tau) = D_k$. *Then we will show that*

$$|x_i(t) - W_i(t:\tau)| \leq Y_i(t:\tau) \tag{1}$$

for all $t \geq \tau$ *and* $i = 1,\dots,d$. *We beginn with* $t = \tau$, *then*

$$|x_i(t) - W_i(t:\tau)| = |x_i(\tau)| \leq \|x(\tau)\|_\infty \leq D_k$$

for all $i = 1,\dots,d$ *by the induction hypothesis and* $\tau \geq t_k$ *by definition. Suppose the claim (1) holds for fixed* $t \geq \tau$ *and all* $i = 1,\dots,d$, *then for* $t+1$ *we follow*

$$\begin{aligned}
x_i(t+1) &= (1-\alpha_i(t))x_i(t) + \alpha_i F_i(x(t)) + \alpha_i\epsilon_i(t)\\
&\leq (1-\alpha_i(t))(Y_i(t:\tau) + W_i(t:\tau)) + \alpha_i(t)\beta\|x(t)\|_\infty + \alpha_i\epsilon_i(t)\\
&\leq (1-\alpha_i(t))(Y_i(t:\tau) + W_i(t:\tau)) + \alpha_i(t)\beta D_k + \alpha_i\epsilon_i(t)\\
&= Y_i(t+1:\tau) + (1-\alpha_i(t))W_i(t:\tau) + \alpha_i\epsilon_i(t)\\
&= Y_i(t+1:\tau) + W_i(t+1:\tau),
\end{aligned}$$

where we used in the first inequality that F_i is a β-contraction and (1) for fixed $t \geq \tau$. In the second inequality we used the induction hypothesis $\sup_{t \geq t_k} \|x(t)\|_\infty \leq D_k$, because $t \geq \tau \geq t_k$. The two equations follow from the recursive definition of Y and W_i. To close the second induction and prove (1) it remains to show $x_i(t+1) \geq -Y(t:\tau) + W_i(t:\tau)$.

$$\begin{aligned}
-Y_i(t+1:\tau) + W_i(t+1:\tau) &= (1-\alpha_i(t))(-Y_i(t:\tau) + W_i(t:\tau)) - \alpha_i(t)\beta\|x(t)\|_\infty + \alpha_i\epsilon_i(t) \\
&\leq (1-\alpha_i(t))x_i(t) + \alpha_i(t)F_i(x(t)) + \alpha_i\epsilon_i(t) \\
&= x_i(t+1),
\end{aligned}$$

for every $i = 1, \ldots, d$. This condludes the second induction and proves that (1) is true for ever $t \geq \tau$ and $i = 1, \ldots, d$.
Next we used that $|a| - |b| \leq |a-b|$ to follow from (1) that

$$|x_i(t)| \leq Y_i(t:\tau) + |W_i(t:\tau)|, \quad \forall t \geq \tau,\, i = 1, \ldots, d.$$

By the definition of τ it holds that $|W_i(t:\tau)| \leq \beta\epsilon D_k$ for all $t \geq \tau$, $i = 1, \ldots, d$. As $Y_i(t:\tau) \to D_k\beta$ for $t \to \infty$ there exits a $t_{k+1} \geq \tau > t_k$ s.t.

$$|x_i(t)| \leq \beta\epsilon D_k + (1+\epsilon)\beta D_k = \beta D_k(1+2\epsilon) = D_{k+1},$$

for all $t \geq t_{k+1}$ and $i = 1, \ldots, d$. We follow that

$$\|x(t)\|_\infty \leq D_{k+1} \quad , \forall t \geq t_{k+1}.$$

Thus,

$$\sup_{t \geq t_{k+1}} \|x(t)\|_\infty \leq D_{k+1}.$$

This concludes the induction and proves the claim of Theorem 4.2.9.

2. SARSA

Rewrite a k-armed Bandit as a MDP in such a way that SARSA (Algorithm 25 with ϵ_n-greedy policy updates and $\alpha(s,a) = \frac{1}{N(s,a)+1}$) corresponds to the ϵ_n-greedy algorithm introduced in Chapter 1.
Solution:
We define the state space to be $\mathcal{S} = \{1, T\}$ where 1 is the first state, the initial distribution is thus $\mu = \delta_1$, and T is the terminal state.
The action space is defined to be $\mathcal{A}_1 = \{1, \ldots, k\}$ and $\mathcal{A}_T = \{N\}$ and can be interpreted as we play an arm between in $1, \ldots, k$ in the state 1 and we do noting in the terminal state T.
Then we define the transition probabilities to be $p(\{T\}; \{1\}, a) = 1$ for all $a \in \mathcal{A}_1$.
The reward set $\mathcal{R}$ is given by the set of all possible rewards of all k arms united with a terminal reward $\{0\}$ whenever we are in the terminal state T and play action N. I.e. the rewards are defined to be independent of the states and whenever we play action $A_t = a \in \mathcal{A}_1$ the reward is distributed as the rewards of arm a of the bandit, $R_{t+1} = R(a) \sim P_a$ and whenever we play

Algorithm 1: SARSA

Result: Approximation $Q \approx Q^*$

Initialize $Q(s,a) = 0$ and $N(s,a) = 0$ for all $(s,a) \in S \times A$

Choose initial policy π.

while *not converged* **do**

  Initialize s

  Choose $a \sim \pi(\cdot\,;\, s)$

  while s *not terminal* **do**

    Take action a, sample reward $R(s,a)$ and next state s'.

    Choose $a' \sim \pi(\cdot \mid s')$.

    Determine step size α.

    $Q(s,a) = Q(s,a) + \alpha(R(s,a) + \gamma Q(s',a') - Q(s,a))$

    $N(s,a) = N(s,a) + 1$

    $s = s', a = a'$

    Choose policy π derived from updated Q-values.

  end

end

action $A_t = N$ the reward is defined to be $R_{t+1} = R(N) = 0$.
We choose $\gamma \in (0,1)$ arbitrarily, as γ will be irrelevant in the algorithm. Now recall the SARSA Algorithm 1 stated below. For the initialisation of Q and N changes nothing. As we consider ϵ_n-greedy policies, consider a fixed sequence $(\epsilon_n)_{n\in\mathbb{N}_0}$ and initialise π with any ϵ_0-greedy policy, where we only have to consider state 1 as the action in state T is always N with probability one. As $Q \equiv 0$ we choose an arbitrary action (wlog action $a = 1$) with probability $1 - \frac{\epsilon_0(k-1)}{k}$ and all other actions $a' \in \mathcal{A}_1$ with probability $\frac{\epsilon_0}{k}$.
Next we enter the 'while not convergend'-loop and see that we initialise s always with 1, as we choose $\mu = \delta_1$. Then we choose $a \sim \pi(\cdot|1)$ after the ϵ_0-greedy policy defined above. As 1 is not a terminal state we take the action we sampled and recive a reward $R(1,a)$. Then we transit in the terminal state $s' = T$ almost surely and choose action $a' = N$ almost surely and update $Q(1,a)$, using $\alpha = \frac{1}{N(1,a)+1}$, and $N(1,a)$. As $s' = T$ is a terminal state we update the policy π as ϵ_1-greedy policy and continue again with initialising $s \sim \mu$ in the outer loop.
Observations:

- *We only fulfill the 'while s not terminal' condition once, i.e. this is not a real loop. Moreover we choose always $s' = T$ and $a' = N$.*
- *$Q(T,N)$ is never updated and stayes 0 forever, i.e. together with the first overvation we note that the term $\gamma Q(s',a')$ is zero forever.*
- *We only need to consider $Q(s,a)$ and $N(s,a)$ for $s = 1$, i.e. we can drop the dependence on s.*
- *We only need a policy in the state $s = 1$, i.e. we will only wrtie $\pi(\cdot)$ as a probability distribution of the possible arms.*

- *Sampling an action a after a ϵ-greedy policy is equivalent to sampling a uniform random variable $U \sim \mathcal{U}[0,1]$ and play the greedy action whenever $U > \epsilon$ or a uniformly choosen random action whenever $U \leq \epsilon$.*

All in all the algorithm simplifies to Algorithm 2. Finally we observe that this algorithm equals

Algorithm 2: Bandit-SARSA

Result: Approximation $Q \approx Q^*$
Initialize $Q(a) = 0$ and $N(a) = 0$ for all $a \in \{1, \dots, k\}$
Set $n = 0$
Set $\pi(\cdot) = \delta_1$ (choose any arm)
while *not converged* **do**
  Sample $U \sim \mathcal{U}[0,1]$
  if $U \leq \epsilon_n$ **then**
    Choose a_n uniformly in $\{1, \dots, k\}$
  else
    Choose $a_n \sim \pi(\cdot)$
  end
  Play arm a_n, observe reward $R(a_n)$.
  Determine stepsize $\alpha = \frac{1}{N(a_n)+1}$.
  $Q(a_n) = Q(a_n) + \alpha(R(a_n) - Q(a_n))$
  $N(a_n) = N(a_n) + 1$
  Set policy $\pi(\cdot)$ as ϵ_n greedy policy over the Q-values.
  $n = n + 1$
end

the ϵ_n-greedy algorithm from Chapter 1 of the lecture, because for action a_n

$$\begin{aligned} Q^{new}(a_n) &= Q(a_n) + \frac{1}{N(a_n)+1}(R(a_n) - Q(a_n)) \\ &= \frac{1}{N(a)+1} \sum_{i=0}^{n} R(a_n) \mathbf{1}_{\{a_n = a\}}. \end{aligned}$$

is the memory trick and equals the estimator of $\hat{Q}_a$ of arm a in the ϵ_n-greedy algorithm.

3. Convergence of Q-Learning

The assumptions and definitions of Theorem 4.3.4 (Convergence of Q-Learning) are given. Moreover let

$$F(Q)(s,a) := \mathbb{E}_s^{\pi^a}\big[R(s,a) + \gamma \max_{a' \in \mathcal{A}_{S_1}} Q(S_1, a')\big]$$

and

$$\varepsilon_n(s,a) := R(s,a) + \gamma \max_{a' \in \mathcal{A}_{s'}} Q_n(s', a') - F(Q_n)(s,a)$$

for all $(s,a) \in \mathcal{S} \times \mathcal{A}$ and $n \in \mathbb{N}$. Show that the sequence

$$Q_{n+1}(s,a) := Q_n(s,a) + \alpha_n(s,a)\big(F(Q_n)(s,a) - Q_n(s,a) + \varepsilon_n(s,a)\big), n \in \mathbb{N}$$

almost surely converges to Q^π.

Solution:

We aim to apply Theorem 4.2.9.. Therefore we have to show that

a) $F : \mathbb{R}^{|\mathcal{S}||\mathcal{A}|} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ *is a contraction with respect to the* $||\cdot||_\infty$*-norm, and*

b) $\varepsilon_n(s,a)$ *is* $\mathcal{F}_{n+1}$*-measurable,* $\mathbb{E}[\varepsilon_n(s,a)|\mathcal{F}_n] = 0$ *and there is some* $C > 0$ *such that* $\sup_{n,s,a} \mathbb{E}[\varepsilon_n^2(s,a)|\mathcal{F}_n] \leq C$.

We show a) by checking the definition of a contraction:

$$
\begin{aligned}
&\|F(Q_1) - F(Q_2)\|_\infty \\
&= \max_{s,a} \Big\{ \big|\mathbb{E}_s^{\pi^a}\big[R(s,a) + \gamma \max_{a' \in \mathcal{A}_{S_1}} Q_1(S_1,a')\big] - \mathbb{E}_s^{\pi^a}\big[R(s,a) + \gamma \max_{a' \in \mathcal{A}_{S_1}} Q_2(S_1,a')\big]\big| \Big\} \\
&= \gamma \max_{s,a} \Big\{ \big|\mathbb{E}_s^{\pi^a}\big[\max_{a' \in \mathcal{A}_{S_1}} Q_1(S_1,a') - \max_{a' \in \mathcal{A}_{S_1}} Q_2(S_1,a')\big]\big| \Big\} \\
&\leq \gamma \max_{s,a} \Big\{ \big|\mathbb{E}_s^{\pi^a}\big[\max_{a' \in \mathcal{A}_{S_1}} (Q_1(S_1,a') - Q_2(S_1,a'))\big]\big| \Big\} \\
&\leq \gamma \max_{s,a} \Big\{ \mathbb{E}_s^{\pi^a}\big[\max_{s' \in \mathcal{S}, a' \in \mathcal{A}_{S_1}} \big|Q_1(s',a') - Q_2(s',a')\big|\big] \Big\} \\
&= \gamma \max_{s,a} \Big\{ \mathbb{E}_s^{\pi^a}\big[\|Q_1 - Q_2\|_\infty\big] \Big\} \\
&= \gamma \|Q_1 - Q_2\|_\infty.
\end{aligned}
$$

We move on to claim b). The errors are $\mathcal{F}_n$*-measurable by definition and so also* $\mathcal{F}_{n+1}$*-mesurable. For the expectation we see directly by definition*

$$
\begin{aligned}
\mathbb{E}[\varepsilon_n(s,a)|\mathcal{F}_n] &= \mathbb{E}[R(s,a) + \gamma \max_{a' \in \mathcal{A}_{s'}} Q_n(s',a') - F(Q_n)(s,a)|\mathcal{F}_n] \\
&= \mathbb{E}\Big[R(s,a) + \gamma \max_{a' \in \mathcal{A}_{s'}} Q_n(s',a') - \mathbb{E}_s^{\pi^a}\big[R(s,a) + \gamma \max_{a' \in \mathcal{A}_{S_1}} Q_n(S_1,a')\big]\Big] \\
&= 0,
\end{aligned}
$$

because the state state s' *in the algorithm is sampled from* $p(\cdot\,; s,a)$*. The last claim follows directly from the assumption on bounded rewards as in 4.3.2.*