UNIVERSITY OF LIÈGE

MASTER IN ENGINEERING

February 24, 2022

INFO8003-1
Optimal decision making for complex problems

Assignment 1

Maxime Amodei - **s171830**
Cédric Siboyabasore - **s175202**

1 Implementation of the domain

The policy we implemented is an always-go-right policy $\mu(x,y)$: whatever the state (x,y), the agent should choose action $u = (0,1) \in U$.

$$\mu(x,y) = (0,1) \text{ for all } (x,y) \in X$$

In the case of the deterministic domain, the agent always takes the action $(0,1)$ returned by the policy. Since the initial state is $x_0 = (3,0)$, the agent will eventually reach $(3,4)$ in x_4 and stay there for the other steps, as this is the leftmost cell of this row.

In the case of the stochastic domain, this action is only taken if noise $w \leq \frac{1}{2}$. Otherwise, the next state returned by the dynamics f is $(0,0)$. In other words, the agent has equal probabilities of following our policy and of going to state $(0,0)$.

The deterministic domain can be seen as a stochastic domain where $w \leq \frac{1}{2}$. In our implementation, we consider a noise of $w = 0 (\leq \frac{1}{2})$ for the deterministic domain.

The simulation of our policy $\mu(x,y)$ in the two domains through a single trajectory of 10 steps, starting by the initial state $x_0 = (3,0)$, is shown here below.

```
Deterministic domain:
        ((3, 0), (0, 1), -9, (3, 1)),
        ((3, 1), (0, 1), 4, (3, 2)),
        ((3, 2), (0, 1), 19, (3, 3)),
        ((3, 3), (0, 1), -5, (3, 4)),
        ((3, 4), (0, 1), -5, (3, 4)),
        ((3, 4), (0, 1), -5, (3, 4)),
        ((3, 4), (0, 1), -5, (3, 4)),
        ((3, 4), (0, 1), -5, (3, 4)),
        ((3, 4), (0, 1), -5, (3, 4)),
        ((3, 4), (0, 1), -5, (3, 4))

Stochastic domain:
        ((3, 0), (0, 1), -9, (3, 1)),
        ((3, 1), (0, 1), -3, (0, 0)),
        ((0, 0), (0, 1), 1, (0, 1)),
        ((0, 1), (0, 1), -5, (0, 2)),
        ((0, 2), (0, 1), -3, (0, 0)),
        ((0, 0), (0, 1), -3, (0, 0)),
        ((0, 0), (0, 1), 1, (0, 1)),
        ((0, 1), (0, 1), -3, (0, 0)),
        ((0, 0), (0, 1), -3, (0, 0)),
        ((0, 0), (0, 1), -3, (0, 0))
```

2 Expected return of a policy

We know from the theory that the distance in infinity norm between J_N^μ and J^μ after N iterations is bounded by a term that decreases exponentially with the number of iterations, such that

$$\|J^\mu - J_N^\mu\|_\infty \leq \frac{\gamma^N}{1-\gamma} B_r$$

We can exploit this inequation and use J_N^μ to estimate J^μ. Indeed, we can compute J_N^μ using recurrence equation

$$J_N^\mu(x,y) = \underset{w \sim P_w(\cdot|(x,y),u)}{E} \left[r((x,y),\mu(x,y),w) + \gamma J_{N-1}^\mu(f((x,y),\mu(x,y),w)) \right], \quad \forall N \geq 1$$

with $J_0^\mu(x,y) \equiv 0$. We know the discount factor $\gamma = 0.99$.
To choose the number of iterations N, we have to bound the distance in infinity norm between J_N^μ and J^μ. We choose a bound of $\frac{\gamma^N}{1-\gamma} B_r = 0.1$.
Since we know the maximum value of the reward $B_r = B = \max_{(x,y)} R(g,(x,y)) = 19$, we can derive N :

$$\begin{aligned}
&\frac{\gamma^N}{1-\gamma} B_r = 0.1 \\
&\iff \gamma^N = 0.1 \frac{1-\gamma}{B_r} \\
&\iff N = \log_\gamma \left\{ 0.1 \frac{1-\gamma}{B_r} \right\} \\
&\iff N = \frac{\log \left\{ 0.1 \frac{1-\gamma}{B_r} \right\}}{\log \gamma} = 980
\end{aligned}$$

We can now implement the routine estimating J^μ by computing $J_N^\mu(x,y)$ for all $(x,y) \in X$ for $N = 980$ iterations in both domains. The results are displayed in Table 1 and 2.
For the stochastic domain, to compute the expectation appearing in the recurrence equation, we considered in our implementation the case in which there is no noise by picking $w = 0$ ($\leq \frac{1}{2}$) which has a probability of $\frac{1}{2}$ of occurring and the equally likely case in which there is noise by picking $w = 1 (\geq \frac{1}{2})$ and we averaged over them.

	y				
x	0	1	2	3	4
0	1839.51	1857.08	1880.89	1899.89	1899.89
1	989.98	996.95	998.94	999.94	999.94
2	-779.25	-779.04	-790.95	-799.95	-799.95
3	-471.54	-467.21	-475.97	-499.97	-499.97
4	849.32	875.07	887.95	899.95	899.95

Table 1: $J_N^{\mu}(x,y)$ of our $\mu(x,y)$ policy for every (x,y) $\in$ X in the deterministic domain

	y				
x	0	1	2	3	4
0	-72.01	-71.45	-64.24	-54.74	-54.74
1	-67.77	-64.90	-64.16	-63.66	-63.66
2	-77.40	-73.25	-76.98	-81.48	-81.48
3	-75.34	-68.07	-66.51	-78.51	-78.51
4	-82.33	-74.12	-70.65	-64.65	-64.65

Table 2: $J_N^{\mu}(x,y)$ of our $\mu(x,y)$ policy for every (x,y) $\in$ X in the stochastic domain

3 Optimal policy

The reward function $r((x,y),u)$ and $p\left((x',y') \mid (x,y),u\right)$ of the equivalent Markov Decision Process of the domain are given by

$$r((x,y),u) = \underset{w\sim P_w(\cdot|(x,y),u)}{E}\left[r((x,y),u,w)\right] \quad \forall (x,y) \in X, u \in U$$

$$p\left((x',y') \mid (x,y),u\right) = \underset{w\sim P_w(\cdot|(x,y),u)}{E}\left[I_{\{x'=f((x,y),u,w)\}}\right] \quad \forall (x,y),(x',y') \in X, u \in U$$

From them, we can compute the sequence of Q_N-functions using recurrence equation

$$Q_N((x,y),u) = r((x,y),u) + \gamma \sum_{(x',y')\in X} p\left((x',y') \mid (x,y),u\right) \max_{u'\in U} Q_{N-1}\left((x',y'),u'\right), \quad \forall N \geq 1 \text{ with}$$

$$Q_0(x,u) \equiv 0$$

In our implementation, the function that computes Q_N takes as arguments the reward function and the transitions probabilities, both computed by other functions.
To derive the optimal policy $\mu_N^*(x,y)$, we can take leverage of the fact that

$$\mu_N^*(x,y) \in \arg\max_{u\in U} Q_N((x,y),u)$$

This result depends on the value of N.
We know from the theory that the expected return of μ_N^* converges in infinity norm to J^{μ^*} and following bound on the suboptimality of μ_N^* holds:

$$\left\| J^{\mu^*} - J^{\mu_N^*} \right\|_\infty \leq \frac{2\gamma^N B_r}{(1-\gamma)^2}$$

If we choose a bound of $\frac{2\gamma^N B_r}{(1-\gamma)^2} = 0.1$, we can derive $N = \frac{\log\left\{0.1\frac{(1-\gamma)^2}{2B_r}\right\}}{\log\gamma} = 1507$.
Knowing N, we can now compute the sequence of Q_N-functions and derive the optimal policy $\mu_N^*(x)$

from the last Q-function of the sequence by selecting the actions at which the maximum of this last Q-function is obtained. The optimal policies for the deterministic and stochastic domains are respectively displayed in Tables 3 and 4.

	y				
x	0	1	2	3	4
0	(1, 0)	(0, 1)	(0, 1)	(0, 1)	(-1, 0)
1	(0, 1)	(0, 1)	(0, 1)	(0, 1)	(-1, 0)
2	(-1, 0)	(0, 1)	(-1, 0)	(-1, 0)	(-1, 0)
3	(-1, 0)	(0, 1)	(0, 1)	(-1, 0)	(-1, 0)
4	(-1, 0)	(0, 1)	(-1, 0)	(-1, 0)	(0, -1)

Table 3: Optimal policy $\mu_{N=1507}(x,y)$ for every (x,y) $\in$ X in the deterministic domain

	y				
x	0	1	2	3	4
0	(1, 0)	(1, 0)	(1, 0)	(0, 1)	(-1, 0)
1	(0, 1)	(0, 1)	(0, 1)	(0, 1)	(-1, 0)
2	(-1, 0)	(0, 1)	(-1, 0)	(1, 0)	(-1, 0)
3	(0, -1)	(0, 1)	(0, 1)	(0, -1)	(0, -1)
4	(-1, 0)	(0, 1)	(-1, 0)	(-1, 0)	(1, 0)

Table 4: Optimal policy $\mu_{N=1507}(x,y)$ for every (x,y) $\in$ X in the stochastic domain

We notice that greater values of N, for example a value of $N = 2 * 10^6$, do not change the inferred optimal policy in both domains. We can deduce that $N = 1507$ approximates well the infinite time horizon of the policy.

The last Q-function $Q_N((x,y),u)$ of the sequence also allows us to derive the value function $J_N^{\mu*}$, as, since our N approximates well the infinite time horizon of the policy, we have

$$J^{\mu_N^*}(x,y) = \max_{u \in U} Q_N((x,y),u)$$

The value function $J^{\mu_N^*}(x,y)$ for each state $(x,y) \in X$ for both domains is displayed in Tables 5 and 6.

	y				
x	0	1	2	3	4
0	1842.03	1857.18	1880.99	1899.99	1899.99
1	1854.57	1870.27	1881.08	1890.99	1899.99
2	1842.03	1855.57	1870.27	1881.08	1890.99
3	1828.60	1849.00	1863.64	1863.27	1864.08
4	1816.32	1826.51	1849.00	1863.64	1842.00

Table 5: Value function $J^{\mu_N^*}(x,y)$ for every (x,y) $\in$ X in the deterministic domain

	y				
x	0	1	2	3	4
0	159.44	159.63	163.05	172.12	172.12
1	159.63	163.05	164.90	167.62	172.12
2	159.44	160.13	163.05	167.21	167.62
3	159.25	162.19	167.21	162.19	167.21
4	159.25	155.71	162.19	167.21	162.22

Table 6: Value function $J^{\mu_N^*}(x,y)$ for every (x,y) $\in$ X in the stochastic domain

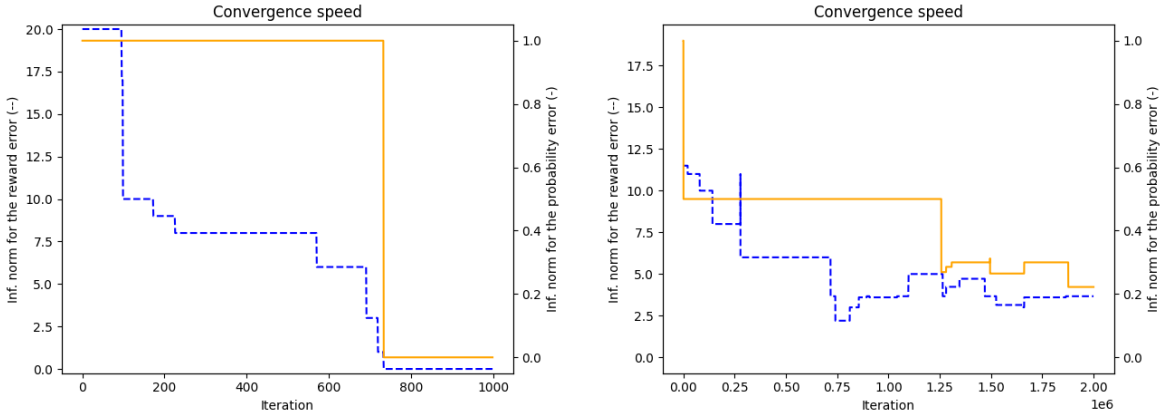
4 System Identification

The goal is to infer the values of $r(x,u)$ and $p(x'|x,u)$ from one trajectory of N transitions and study how fast the empirical estimations $\hat{r}$ and $\hat{p}$ converge to the true values r and p. We obviously need $N > |X|\times|U|$ to observe each value of the reward signal, but the analogous condition for transition probabilities is too restrictive. Indeed, most $|X|^2 \times |U|$ values for $p(x'|x,u)$ are actually zero because the system's dynamics do not allow for teleportation between any two cells, and since the probabilities are initialized at zero, it is not necessary to evaluate every combination in the Cartesian product.
Since the algorithm described in the theoretical course computes the probabilities from a ratio of integers increasing by one, we would need to visit all likely states multiple times to have a good approximation for the ratio. Counting 100 non-zero values for the true transition probabilities from the last section, we suppose that 1000 iterations could give a decent approximation for both targets $r(x,u)$ and $p(x'|x,u)$ in the deterministic system.
In the stochastic system, some states are much harder to evaluate because of how far they are from the absorbing state $(0,0)$. At any state, the probability of going to $(0,0)$ is $\frac{1}{2}$, regardless of the action. If a state is reachable in exactly K steps (in the deterministic world), the probability of reaching it is 2^{-K} in the stochastic world, which means that the probability will be much less precise in these cases. Since now $p((0,0)|x,u) \geq \frac{1}{2}$ for all x,u, this means that the probability of transition to $(0,0)$ from a state that is almost never visited will remain close to 0 even though it should be $\frac{1}{2}$. Therefore, we

can't reasonably expect the inference to be as fast as in the other case, especially if certain states are hard to reach. We experimentally found $N = 2\,10^6$ to give a decreasing curve for both errors.
The implementation differs slightly from the statement with regard to the form in which the input is expected. For the sake of simplicity, the history is viewed as a tuple of (current state, action take, reward perceived, next state) similarly to the first section instead of the full list. If the trajectory was converted to the form described in the statement, it would have to be converted back to this form to be usable.

4.1 Convergence of the reward signal and the transition probabilities estimations

The results for the deterministic world over a trajectory of length 1000 is given in Figure 1a. The errors are reported as the infinity-norm $||\cdot||_\infty$ of the difference of the estimation ($\hat{r}$ or $\hat{p}$) and the true value from section 3 (r or p, respectively). This figure shows the reward error decreasing step by step while the transition error drops at once, both converging to 0. In these settings, the probabilities are always 0 or 1, which explains the sudden drop.



(a) Deterministic System

(b) Stochastic System

Figure 1: Convergence for $\hat{r}$ (blue) and $\hat{p}$ (orange) for both type of worlds.

Figure 1b shows the results corresponding to the stochastic system. As discussed earlier, the stochastic settings restrict the exploration of the world, which is illustrated by the slow convergence : the error in the transition probabilities is 0.5, which most likely mean that some (state, action, next state) combinations were not observed enough or possibly at all. While it was obvious that the errors would decrease monotonically in the deterministic case, the values correspond to expectation over the noise distribution and the iterative computation may under- or over-estimate the said expectations, which does not guarantee a monotonic decrease.
Similarly to section 3, we need to estimate a value N for the iteration of the state-action value function. We will suppose that the trajectory is so long that the MDP learned by the model is close enough to the true MDP (and therefore so is the state-action value function), even though it may not be the case for the non-deterministic problem. With this supposition at hand, we can use the formula (8) from the theoretical lectures

$$\left|\left|J^{\mu_N^*} - J^{\mu^*}\right|\right|_\infty \leq \frac{2\gamma^N Br}{(1-\gamma)^2}$$

and choose a tolerance on J. We start with 0.1 and find using a method similar to section 3

$$N = \frac{\log 0.1(1-\gamma)^2}{\log \gamma} - \frac{2Br}{\log \gamma} \approx 1507.4 \,.$$

The value $N = 1507$ is not computationally prohibitive as it takes only a few seconds to compute, so the tolerance is not too strict (i.e., increasing the tolerance is unnecessary).
To estimate that the bound is strict enough (i.e., we don't need to lower the tolerance), we will look at the variance between the different values of the state-action value function for any state. If two different entries corresponding to the same state but different actions are closer than twice the tolerance (2×0.1), then a statistical error could change which state the policy returns, which is not desirable.

4.2 Convergence of the state-action value function estimation

The state value function $\hat{Q}_N(x,y)$ can be estimated from $\hat{r}(x,y$ and $\hat{p}(x',x,u)$ using the formula given in the theoretical lecture. We can compute $||\hat{Q}_N - Q_N||_\infty$ to estimate how fast the state-action value function converges. Unfortunately, Tables 7 and 8 show that the infinity norm doesn't decrease by a lot, if any at all. To make sure that the convergence that we saw previously wasn't a fluke, Tables 7 and 8 also report the convergence of the expected cumulative reward, $||\hat{J}_N^{\mu^\star} - J^{\mu^\star}||_\infty$ which fortunately converges. This shows that even though some values of $\hat{Q}_N$ don't converge fast to their value, $\max_{u \in \mathscr{U}} \hat{Q}_N(x,u)$ does converge fast enough. Since it is base on the latter value that we extract the policy, we can conclude that the convergence is fast enough in our case.

history length	$\|\|\hat{Q}_N - Q_N\|\|_\infty$	$\|\|\hat{J}_N^{\mu^\star} - J^{\mu^\star}\|\|_\infty$
1	1899.92	1899.92
2	1899.92	1899.92
3	1899.92	1899.92
5	1899.92	1899.92
9	1899.92	1899.92
17	1899.92	1899.92
33	1899.92	1899.92
65	1899.92	1899.92
129	1899.92	1899.92
257	1899.92	10.07
513	1899.92	0.08
1000	1899.92	0.08

Table 7: Convergence of the state-action value functions and the estimated cumulative reward for different trajectory lengths for a deterministic environment.

history length	$\|\|\hat{Q}_N - Q_N\|\|_\infty$	$\|\|\hat{J}_N^{\mu^\star} - J^{\mu^\star}\|\|_\infty$
1	172.12	172.12
2	172.12	172.12
3	172.12	172.12
5	172.12	172.12
9	172.12	172.12
17	440.56	440.37
33	440.56	440.37
65	172.12	172.12
129	172.12	172.12
257	391.00	390.81
513	391.00	390.81
1025	391.00	390.81
2049	287.37	280.14
4097	287.37	280.14
8193	172.12	172.12
16385	167.62	167.21
32769	167.21	167.21
65537	167.21	167.21
131073	167.21	167.21
262145	162.22	162.22
524289	162.22	9.20
1048577	162.22	11.56
2000000	162.22	11.56

Table 8: Convergence of the state-action value functions and the estimated cumulative reward for different trajectory lengths for a stochastic environment.

4.3 Empirical optimal stationary policy

The optimal stationary policy $\hat{\mu}_N^\star$ can be extracted from $\hat{Q}_N$ using the formula from the theoretical lecture $\hat{\mu}_N^\star(x) = \arg\max_{u \in \mathscr{A}} \hat{Q}_N(x,u)$, which gives the action that maximizes the Q function. The said policy is show in Tables 9 and 10 which are quite similar on most states. Note that since the dynamics of the systems are different, we shouldn't expect the policies to be the same. Rather, we could re-run the experiment for the stochastic settings and verify that the system converges to the same value most of the time. Since this depends on random events, we cannot make any strong assumptions and the function does look the same from run to run, even if this is not shown here to keep the report lighter.

	y				
x	0	1	2	3	4
0	(1, 0)	(0, 1)	(0, 1)	(0, 1)	(-1, 0)
1	(0, 1)	(0, 1)	(0, 1)	(0, 1)	(-1, 0)
2	(-1, 0)	(0, 1)	(-1, 0)	(-1, 0)	(-1, 0)
3	(-1, 0)	(0, 1)	(0, 1)	(-1, 0)	(-1, 0)
4	(-1, 0)	(0, 1)	(-1, 0)	(-1, 0)	(0, -1)

Table 9: $\hat{\mu}^*$ extracted from the $\hat{Q}_{1508}$ function for the deterministic settings using a trajectory of 1000 transitions.

	y				
x	0	1	2	3	4
0	(1, 0)	(1, 0)	(0, 1)	(0, 1)	(-1, 0)
1	(0, 1)	(0, 1)	(0, 1)	(0, 1)	(-1, 0)
2	(-1, 0)	(0, 1)	(1, 0)	(1, 0)	(-1, 0)
3	(0, -1)	(0, 1)	(0, 1)	(0, -1)	(0, -1)
4	(-1, 0)	(0, 1)	(-1, 0)	(-1, 0)	(1, 0)

Table 10: $\hat{\mu}^*$ extracted from the $\hat{Q}_{1508}$ function for the stochastic settings using a trajectory of $2 \cdot 10^6$ transitions.

4.4 Convergence of the cumulated reward function estimation

While tables 9 and 10 reported the infinite norm for the value function, this does not tell the whole story, as the function may already be a good approximation for most states at the end of the convergence ($N = 1508$). The tables 11 to 14 show the empirical value functions $\hat{J}_N$ in both settings for all states, and the true value function J for comparison. We can see that even though the infinity norm on $\hat{Q}_N - Q$ was non-negligible, the results for the expected cumulative rewards are actually pretty close to the true values. This shows the convergence is not as bad as the previous section could make it appear.

	y				
x	0	1	2	3	4
0	1841.93	1857.09	1880.90	1899.90	1899.90
1	1854.48	1870.18	1880.99	1890.90	1899.90
2	1841.93	1855.48	1870.18	1880.99	1890.90
3	1828.51	1848.91	1863.55	1863.18	1863.99
4	1816.22	1826.42	1848.91	1863.55	1841.91

Table 11: J extracted from the true reward and transition probabilities for some deterministic settings

	y				
x	0	1	2	3	4
0	1842.03	1857.19	1881.00	1900.00	1900.00
1	1854.58	1870.28	1881.09	1891.00	1900.00
2	1842.03	1855.58	1870.28	1881.09	1891.00
3	1828.61	1849.01	1863.65	1863.28	1864.09
4	1816.32	1826.52	1849.01	1863.65	1842.01

Table 12: $\hat{J}_{1508}$ extracted from $\hat{Q}_{1508}$ over a trajectory of 1000 transition for some deterministic settings

	y				
x	0	1	2	3	4
0	159.44	159.63	162.62	172.12	172.12
1	159.63	163.04	164.89	167.62	172.12
2	159.44	159.70	162.19	167.20	167.62
3	159.25	162.19	167.20	162.19	167.20
4	159.25	155.70	162.19	167.20	162.22

Table 13: J extracted from the true reward and transition probabilities for some deterministic settings

	y				
x	0	1	2	3	4
0	157.64	157.76	161.66	172.20	173.18
1	157.79	161.08	162.67	165.08	169.21
2	157.65	158.63	161.60	165.91	164.51
3	157.19	160.66	166.92	160.45	169.58
4	157.42	153.24	159.03	163.91	167.71

Table 14: $\hat{J}_{1508}$ extracted from $\hat{Q}_{1508}$ over a trajectory of 1000 transition for some deterministic settings

4.5 Influence of the length of the trajectory

As previously discussed, it is mandatory to have a relatively long trajectory to make sure all states get observed a few times : using a random uniform policy make sure no action is preferred, but we could base the policy on what we have already observed and always seek for exploration rather than exploitation if we only care about the convergence, which would reduce the required trajectory length. Again, we already discussed how in the stochastic environment the length of the trajectory is even more important : since there is a 50% chance of going to the upper left corner, the lower right corner will rarely be observed and as such, the infinity norm will not converge very fast. We found the value $2\,10^6$ empirically without giving it proper attention : lower values produced larger variation and the computation time became non-negligible, so we felt like it was a good threshold.