Reinforcement Learning: a comprehensive introduction [Part 2]

Jul 11, 2018 12:00 · 2351 words · 12 minute read
REINFORCEMENT LEARNING MACHINE LEARNING AI

Recap

In the previous post we introduced state-value and history-value functions for a policy π which allow us to compute the **expected return** at different starting points in time. They can be used to **compare** the effectiveness of different policies, which plays well with our intent of finding the **optimal** policy for the task at hand.

How do we compute them?
We derived a generalized form of the Bellman equation which, under a set of stronger hypotheses on the agent and on the environment, simplifies to a manageable expression which we feel confident to solve.

It remains to prove that those stronger hypotheses are reasonable and that they do not damage our chances of finding the overall **optimal** policy.
This will be our focus for today.

Decision rules: a breakdown

We introduced decision rules as mappings between $\mathcal{H}_t$, the set of possible histories of our system up to time t, and $\mathcal{P}(\mathcal{A})$, the collection of probability distributions over the set of actions:

$$d_t : \mathcal{H}_t \to \mathcal{P}(\mathcal{A}) \tag{1}$$

This definition provides the greatest amount of generality and flexibility - we usually refer to this class of decision rules as **history-dependent randomized decision rules**: every action and event up to the current time instant is taken into account before specifying the agent preferences for the next action.
This class of decision rules will be denoted by D_t^{HR}.

Everything comes with a price: it is computationally very expensive to keep track of a function whose input dimension grows exponentially in time.
Consider, for example, a state space $\mathcal{S}$ with five elements equipped with an action set $\mathcal{A}$ composed of three choices.
For $t = 1$ we have 5 elements in $\mathcal{H}_1$ (we take $t = 1$ as starting state).
For $t = 2$ we have 5x3x5=75 elements in $\mathcal{H}_2$.
For an arbitrary $t \in \mathbb{N}$ we have

$$|\mathcal{H}_t| = 5^t 3^{t-1} \tag{2}$$

which, you surely agree with me, is going to become problematic soon enough, even if the system is ridiculously small.

A much more convenient class of decision rules is the collection of **randomized markovian decision rules**, which we denote by D^{MR}: the action choice is chosen randomly from a probability distribution depending only on the *current* state of the system (no t subscript this time!).
This can be stated formally saying that every $d \in D^{MR}$ if a function from $\mathcal{S}$ to $\mathcal{P}(\mathcal{A})$.
The computational gain is evident: a markovian randomized decision rule d only requires $|\mathcal{S}||\mathcal{A}|$ memory slots - a much more feasible choice.

Policies: a breakdown

To each class of decision rules we can associate a class of policies:

- π is a **history-dependent randomized policy** if

$$\pi = (d_1,\ d_2,\ \ldots) \qquad \text{where } d_i \in D_i^{HR} \tag{3}$$

We shall refer to this class using Π^{HR};

- π is a **markovian randomized policy** if

$$\pi = (d_1,\ d_2,\ \ldots) \qquad \text{where } d_i \in D^{MR} \tag{4}$$

We shall refer to this class using Π^{MR}.

Markovian randomized decision rules are a subset of history-dependent randomized decision rules, which implies that

$$\Pi^{MR} \subset \Pi^{HR} \tag{5}$$

Before proceeding anything further is time to give a proper definition of **optimal** policy.

Optimal policies

Using the concept of state-value function we can affirm that a policy π is better than a policy π' if

$$v_\pi^1(s) \geq v_{\pi'}^1(s) \quad \forall s \in \mathcal{S} \tag{6}$$

where s is the initial state of our system.

A policy π is said to be **optimal** if

$$v_\pi^1(s) \geq v_{\pi'}^1(s) \quad \forall s \in \mathcal{S} \tag{7}$$

for all $\pi' \in \Pi^{HR}$ [remember that randomized history-dependent policies are the most general class!].

It is important to remark that $v_{(\cdot)}$ is an *expected value* - even if a policy π is better *on average* than a policy π' it might still happen than an experiment using policy π produces a higher return compared to an experiment using policy π': stochasticity baby!

We can now ask the following question: are we compromising our search for an optimal policy if we restrict our attention to markovian randomized policies?
In other words, are we going to achieve a **lower** expected return if we focus on policies which do not use the whole system history before committing to an action?

It depends on the environment - we shall see why.

Return: a different viewpoint

Let's recall the definition of state-value function for a policy π:

$$\begin{aligned} v_\pi^1(s_1) &:= \mathbb{E}[G_1 \mid S_1 = s_1] = \\ &= \sum_{i=1}^{T} \gamma^{i-1}\, \mathbb{E}[R_{i+1} \mid S_1 = s_1] \end{aligned} \tag{8}$$

where $s_1 \in \mathcal{S}$, $\gamma \in [0, 1)$ and $T \in \mathbb{N} \cup \{+\infty\}$.
Using the definition of expectation we get:

$$v_\pi^1(s_1) := \sum_{i=1}^{T} \gamma^{i-1} \sum_r r\, \mathbb{P}(R_{i+1} = r \mid S_1 = s_1) \tag{9}$$

which can be decomposed using the law of total probability

$$v_\pi^1(s_1) := \sum_{i=1}^{T} \gamma^{i-1} \sum_r r \sum_{s \in S} \sum_{a \in \mathcal{A}} \mathbb{P}(R_{i+1} = r \mid A_t = a, S_t = s, S_1 = s_1)\, \mathbb{P}(A_t = a, S_t = s \mid S_1 = s_1)$$

This is were we need to introduce the first additional hypothesis on the **environment**: let's suppose that rewards are **markovian**. Then

$$\mathbb{P}(R_{i+1} = r \mid A_t = a, S_t = s, S_1 = s_1) = \mathbb{P}(R_{i+1} = r \mid A_t = a, S_t = s) \tag{11}$$

which can be plugged in equation 10 to get

$$v_\pi^1(s_1) := \sum_{i=1}^{T} \gamma^{i-1} \sum_r r \sum_{s \in S} \sum_{a \in \mathcal{A}} \underbrace{\mathbb{P}(R_{i+1} = r \mid A_t = a, S_t = s)}_{\text{environment}} \underbrace{\mathbb{P}(A_t = a, S_t = s \mid S_1 = s_1)}_{\pi\text{-dependent}} \tag{12}$$

We have isolated the part of the equation which depends on the policy π - to enforce this concept let's mark it with a superscript:

$$\mathbb{P}^\pi(A_t = a, S_t = s \mid S_1 = s_1) \tag{13}$$

History is overrated

Now, suppose that the following holds:

Theorem 1 - For each history-dependent randomized policy π and for each $s_1 \in \mathcal{S}$ there exists a markovian randomized policy τ such that

$$\mathbb{P}^{\pi}(A_t = a, S_t = s \mid S_1 = s_1) = \mathbb{P}^{\tau}(A_t = a, S_t = s \mid S_1 = s_1) \tag{14}$$

for every $t \in \{1, 2, \dots\}$, for every $a \in \mathcal{A}$ and for every $s \in \mathcal{S}$.

Using equation 12 it would follow that

$$v^1_{\pi}(s_1) = v^1_{\tau}(s_1) \tag{15}$$

which can be used to show that an optimal policy in the class of markovian randomized policies is just as good as an optimal policy in the class of history-dependent randomized policies!

It is strikingly easy: let π^* be an optimal policy in Π^{HR} and let τ^* be an optimal policy in Π^{MR}.
We know that $\Pi^{MR} \subset \Pi^{HR}$ which implies that

$$v^1_{\pi^*}(s) \geq v^1_{\tau^*}(s) \qquad \text{for every } s \in \mathcal{S} \tag{16}$$

Nonetheless, for every $s \in \mathcal{S}$ there exists a policy α_s in Π^{MR} such that

$$v^1_{\pi^*}(s) = v^1_{\alpha_s}(s) \tag{17}$$

But τ^* is optimal in Π^{MR}:

$$v^1_{\tau^*}(s) \geq v^1_{\alpha_s}(s) = v^1_{\pi^*}(s) \geq v^1_{\tau^*}(s) \qquad \text{for every } s \in \mathcal{S} \tag{18}$$

$$\Rightarrow \quad v^1_{\pi^*}(s) = v^1_{\tau^*}(s) \qquad \text{for every } s \in \mathcal{S} \tag{19}$$

We have thus proved that if the environment is markovian and **Theorem 1** holds it is sufficient to search for an optimal policy in the class of markovian randomized policies Π^{MR}.

The proof of **Theorem 1** is not difficult and proceeds by induction - it can be found on Peterman's book as Theorem 5.5.1.

Interlude

Let's pause for a second to summarize what we have achieved and what we are still missing

We defined the concept of optimal policy as well as providing an extremely useful result to restrict the number of candidates for optimality.
Nonetheless we have not found yet an equation which gives us directly the optimal policy as a solution - we have no other way to find it, at the moment, than evaluating the state-value function of each policy in Π^{MR}.
Sadly, Π^{MR} is an infinite set, which means that we cannot find the optimal policy using brute

force.
What now?

Sometimes it is better to simplify and understand the matter at hand in a smaller context before facing the original more complex issue.

In our case, we are going to study a reinforcement learning problem with **fixed** and **finite** time-horizon T - not a random variable, but a natural number which does change between different simulation of our agent.
We shall derive a set of equations called **optimality equations** whose solution, unsurprisingly, is an **optimal policy** in this simplified setting.
The form of the equation will suggest us a more general formulation which can be used as an optimality equation for problems with arbitrary and random time-horizon T, the ones we are genuinely interested in.

Let's do it!

Optimality for fixed & finite time-horizons

Suppose, as we just suggested, that T is a fixed and finite natural number.
This means that S_T is the *terminal* state of our agent: no actions to be taken once we reach that point.
We assume that there exists a terminal reward function, $r_T : \mathcal{S} \to \mathbb{R}$: if $S_T = s$ is the terminal state of the simulation then the agent receives a reward equal to $r_T(s)$.

We proved that a policy π is **optimal** if $v^1_\pi(s) \geq v^1_\alpha(s)$ for all $\alpha \in \Pi^{MR}$ and for all $s \in \mathcal{S}$.
Define

$$v^t_*(s_t) := \sup_{\pi \in \Pi^{MR}} v^t_\pi(s_t) \quad \forall s_t \in \mathcal{S} \tag{20}$$

for every $t \in \{1, 2, \dots, T\}$.
It follows that v^1_* is a function (**not a policy**, be careful!) which satisfies

$$v^1_*(s) \geq v^1_\alpha(s) \qquad \forall\, \alpha \in \Pi^{MR}\,,\ \forall\, s \in \mathcal{S} \tag{21}$$

Even though equation 20 is pretty straight-forward it does not provide, by itself, a way to actually **compute** v^t_*.
Luckily enough, it can be proved *(Puterman - Th. 4.3.2)* that $\{v^t_*\}_{t=1}^T$ are a solution of the following **system** of equations:

$$
\begin{aligned}
&\forall t \in \{1, \dots, T-1\} : \\
&v_*^t(s_t) = \max_{a \in \mathcal{A}} \left\{ \mathbb{E}\left[R_{t+1} \mid A_t = a,\ S_t = s_t\right] + \sum_{s \in \mathcal{S}} v_*^{t+1}(s)\, \mathbb{P}(S_{t+1} = s \mid A_t = a,\ S_t = s_t) \right\} \\
&v_*^T(s_T) = r_T(s_T)
\end{aligned}
\tag{22}
$$

where we have implictly assumed that state transitions are markovian, i.e.

$$
\mathbb{P}(S_{t+1} = s \mid A_t = a,\ S_t = s_t) = \mathbb{P}(S_{t+1} = s \mid A_t = a,\ H_t = h_t) \tag{23}
$$

We have used a $\max$ over $\mathcal{A}$ because we are also assuming that the set of possible actions is finite.

Equations 22 are usually referred as **optimality equations**.
We shall see that they can be used to actually compute v_*^t, which for the moment stands as an upper-bound on the corresponding function for our hypothetical optimal policies. If we could prove that there exists a policy π^* such that

$$
v_*^t(s_t) = v_{\pi^*}^t(s_t) \qquad \forall s_t \in \mathcal{S} \tag{24}
$$

then we would be a good deal closer to the solution of the optimality problem for finite and fixed time-horizon systems.
Puterman comes in our aid once again, with **Theorem 4.3.3** (which slightly generalize).

Theorem 4.3.3 - Let $\{v_*^t\}_{t=1}^T$ be a solution of the optimality equations and suppose that $\pi^* = (d_1^*, \dots, d_{T-1}^*) \in \Pi^{MR}$ satisfies

$$
\begin{aligned}
&\mathbb{E}_{\pi^*}\left[R_{t+1} \mid S_t = s_t\right] + \sum_{s \in \mathcal{S},\, a \in \mathcal{A}} v_{\pi^*}^{t+1}(s)\, b_{d_t^*(s_t)}(a)\, \mathbb{P}(S_{t+1} = s \mid A_t = a,\ S_t = s_t) = \\
&\qquad = \max_{a \in \mathcal{A}} \left\{ \mathbb{E}\left[R_{t+1} \mid A_t = a,\ S_t = s_t\right] + \sum_{s \in \mathcal{S}} v_*^{t+1}(s)\, \mathbb{P}(S_{t+1} = s \mid A_t = a,\ S_t = s_t) \right\}
\end{aligned}
\tag{25}
$$

for $t \in \{1, \dots, T\}$. Then

$$
v_*^t(s_t) = v_{\pi^*}^t(s_t) \qquad \forall s_t \in \mathcal{S},\ \forall t \in \{1, \dots, T\} \tag{26}
$$

which implies that π^* is an **optimal policy**.

We can actually say something more.
Let's rewrite 25 in a slightly different but equivalent way:

$$
\begin{aligned}
&\sum_{a \in \mathcal{A}} b_{d_t^*(s_t)}(a) \left\{ \mathbb{E}_{\pi^*}\left[R_{t+1} \mid A_t = a, S_t = s_t\right] + \sum_{s \in \mathcal{S}} v_{\pi^*}^{t+1}(s)\, \mathbb{P}(S_{t+1} = s \mid A_t = a,\ S_t = s_t) \right\} = \\
&\qquad = \max_{a \in \mathcal{A}} \left\{ \mathbb{E}\left[R_{t+1} \mid A_t = a,\ S_t = s_t\right] + \sum_{s \in \mathcal{S}} v_*^{t+1}(s)\, \mathbb{P}(S_{t+1} = s \mid A_t = a,\ S_t = s_t) \right\}
\end{aligned}
\tag{27}
$$

But a very simple calculus lemma stays the following:

Lemma - Let $\{w_i\}_{i \in I}$ be a finite collection of real numbers and let $\{q_i\}_{i \in I}$ be a probability distribution over I (i.e. $q_i \in [0, 1]$ and $\sum_i q_i = 1$); then

$$
\max_{i \in I} w_i \geq \sum_{i \in I} q_i w_i \tag{28}
$$

Equality holds if and only if

$$
q_i = 0 \tag{29}
$$

for every i such that $w_i < \max_{i \in I} w_i$.

In our case $\{w_i\}_{i \in I} = \mathcal{A}$ and $b_{d_t^*}(\cdot)$ is our probability distribution, which implies that a policy $\pi^* = (d_1^*, \dots, d_{T-1}^*)$ is optimal if and only if it chooses with positive probability only those actions which maximize the right hand side of equation 22.
In other words, solving the system of optimality equations provides us with the optimal actions to be taken at each step of our reinforcement learning problem, which in turn completely determine the set of optimal policies (which can even be chosen to be deterministic, instead of randomized!).

Solving equations 22 is not so difficult, considering that we are assuming that $\mathcal{A}$ is finite: a possible recipe is given by the **Backward Induction Algorithm**.
It determines

$$A_t^*(s_t) := \underset{a \in \mathcal{A}}{\operatorname{argmax}} \left\{ \mathbb{E}\left[R_{t+1} \mid S_t = s_t,\ A_t = a\right] + \sum_{s \in \mathcal{S}} v_*^{t+1}(s_t, s, a)\, \mathbb{P}(S_{t+1} = s \mid A_t = a,\ S_t = s_t) \right\}$$

for all $t \in \{1, \dots, T-1\}$ and $s_t \in \mathcal{S}$ - i.e. the set of actions which maximize the expression in curly brackets.

Here is the pseudo-code:

procedure BACKWARD INDUCTION ALGORITHM
    $t \leftarrow T$
    for $s_T \in \mathcal{S}$ **do**
        $v_*^T(s_T) \leftarrow r_T(s_T)$
    end for
    while $t \geq 1$ **do**
        for $s_t \in \mathcal{S}$ **do**
            Compute and memorize $A_t^*(s_t)$
            $a^* \leftarrow$ first element in $A_t^*(s_t)$
            $v_*^t(s_t) \leftarrow \mathbb{E}\left[R_{t+1} \mid A_t = a^*,\ S_t = s_t\right] + \sum_{s \in \mathcal{S}} v_*^{t+1}(s)\, \mathbb{P}(S_{t+1} = s \mid A_t = a^*,\ S_t = s_t)$
        end for
        $t \leftarrow t - 1$
    end while
end procedure

Optimality equations for an arbitrary time-horizon

We have seen that the solution of

$$\begin{aligned} &\forall t \in \{1, \dots, T-1\},\ \forall s \in \mathcal{S}: \\ &v_*^t(s) = \max_{a \in \mathcal{A}} \left\{ \mathbb{E}\left[R_{t+1} \mid A_t = a,\ S_t = s\right] + \sum_{s' \in \mathcal{S}} v_*^{t+1}(s')\, \mathbb{P}(S_{t+1} = s' \mid A_t = a,\ S_t = s) \right\} \\ &v_*^T(s) = r_T(s) \end{aligned} \tag{31}$$

provides us with the state-value function of an optimal policy (as well as the policy itself!) if the time-horizon is fixed and finite.

Is there a way to reformulate this system to accomodate the general case - where T is a random variable, potentially finite?

So far we have supposed that rewards and state transition probabilities were markovian. We want to make two more assumptions on the environment:

- rewards are stationary:

$$\mathbb{E}\left[R_{t+1} \mid A_t = a,\ S_t = s\right] = \mathbb{E}\left[R_{j+1} \mid A_j = a,\ S_j = s\right] \tag{32}$$

for every $t, j \in \{1, \dots, T-1\}$;

- state transition probabilities are stationary:

$$\mathbb{P}\left(S_{t+1} = s' \mid A_t = a,\ S_t = s\right) = \mathbb{P}\left(S_{j+1} = s' \mid A_j = a,\ S_j = s\right) \tag{33}$$

for every $t, j \in \{1, \dots, T-1\}$.

It make sense, then, to define:

$$r(a, s) := \mathbb{E}\left[R_2 \mid A_1 = a,\ S_1 = s\right] \tag{34}$$
$$p(s' \mid a, s) := \mathbb{P}\left(S_{t+1} = s' \mid A_t = a,\ S_t = s\right) \tag{35}$$

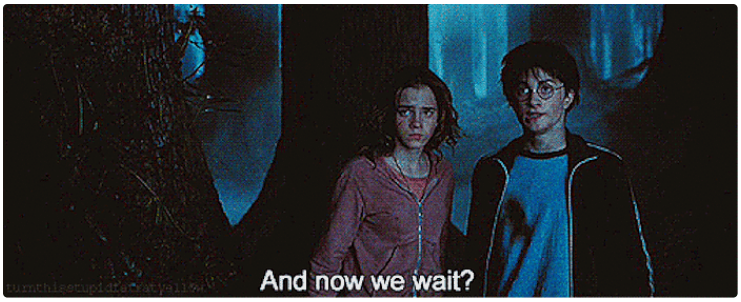
which can plugged into the optimality equations leading to:

$$\begin{aligned} &\forall t \in \{1, \dots, T-1\},\ \forall s \in \mathcal{S} : \\ &v_*^t(s) = \max_{a \in \mathcal{A}} \left\{ r(a, s) + \sum_{s' \in \mathcal{S}} v_*^{t+1}(s')\, p(s' \mid a, s) \right\} \\ &v_*^T(s) = r_T(s) \end{aligned} \tag{36}$$

What would happen if we were to pass to the limit now, for $t \to +\infty$?
The only terms depending on t in the first equation are v_*^t and v^{t+1}, thanks to our stationarity assumption.
If the sequence $\{v^t\}_t$ (which is infinite, if $T = \infty$) converged to a proper limit v_* for $t \to +\infty$ we would find that

$$v_*(s) = \max_{a \in \mathcal{A}} \left\{ r(a, s) + \sum_{s' \in \mathcal{S}} v_*(s')\, p(s' \mid a, s) \right\} \tag{37}$$

Equation 37 is called **Bellman optimality equation** and it does actually provide us with an optimal state-value function in the general case.
But this will be topic of the next episode.



Reinforcement Learning series index

- Part 0
- Part 1
- Part 2 *(this post)*
- *Coming soon…*