

# Value-based methods: overview

The simplest model-based approach are *Value-based* methods.

They revolve around the idea of

- assigning a \*State Value function  $\text{statevalfun}_\pi(\text{state})$  to each state  $\text{state} \in \text{States}$   
 $\text{statevalfun}_\pi : \text{States} \rightarrow \text{Reals}$
- $\text{statevalfun}_\pi(\text{state})$  is an approximation of  $\text{E}_\pi(G \mid \text{stateseq} = \text{state})$   
the expected return achievable from state  $\text{state}$

Given  $\text{statevalfun}_\pi$ , the optimal deterministic policy is

$$\begin{aligned} & \pi^*(\text{state}) \\ = & \underset{\text{act}}{\operatorname{argmax}} \quad \text{statevalfun}_\pi(\text{state}') \\ & \quad \text{transp}(\text{state}', \text{rew} | \text{state}, \text{act}) \neq 0 \end{aligned}$$

- From state  $\text{state}$
- Choose the action  $\text{act}$
- that results in next state  $\text{state}'$
- with maximal  $\text{statevalfun}_\pi(\text{state}')$

Note: the  $\operatorname{argmax}$  results in a *deterministic* policy.

Using a value-based method is practical only if we can discover the state value function  $V_\pi$ , which is initially unknown.

Most methods are iterative in nature and create a sequence of increasingly accurate approximations of  $V_\pi$

$$V_{\pi,0} \cdots V_{\pi,k} \cdots$$

In the limit

$$\lim_{k \rightarrow \infty} V_{\pi,k} = V_\pi$$

As we obtain an improved approximation  $\backslash\text{statevalfun}_{\pi,k+1}$

- we may reflect this improved knowledge by updating the policy

Thus, we periodically improve the policy based on improved approximations of  $\backslash\text{statevalfun}_{\pi}$

This results in a sequence of increasingly accurate approximations of the policy  $\pi$

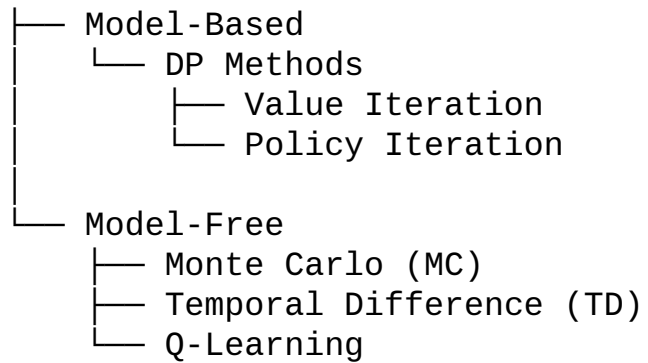
$$\pi_0, \dots, \pi_p, \dots$$

which hopefully converges to  $\pi^*$ .

We will show two broad classes of Value-based methods

- Model-based
- Model-free

#### Value-Based Methods



A common characteristic of both Model-based and Model-free Value methods

- use of *Dynamic Programming* type solutions.

In Dynamic Programming

- there is a *Bellman update* equation
- which relates the value of a state  $V(s)$
- to the values of each of the potential successor states  $V(s')$ 
  - which the environment may return in response to the Learner taking an action in state  $s$

Information flowing from successor to predecessor state is called a *backup*

The Bellman update equation asserts that

$$V_{\pi}(\text{state})$$

can be derived from

- the immediate reward  $R(\text{state}, \text{act}, \text{state}')$ 
  - received by taking action  $\text{act}$  in state  $\text{state}$  and transitioning to state  $\text{state}'$
- and the discounted (by  $\gamma$ ) value of the successor state  $V_{\pi}(\text{state}')$

## The main difference between Model-based and Model-free methods

- a Model-based method *does not need to actively gather* experience
  - if the model is completely known
  - all necessary information is available without actively interacting with the Environment
    - $\text{transp}(\text{state}' \mid \text{state}, \text{act})$  is known  
 $\forall \text{state}, \text{state}' \in \text{States}, \text{act} \in \text{Actions}$
- a Model-free method *must* actively interact with the Environment
  - gathers a *sample* from  $\text{transp}(\text{state}' \mid \text{state}, \text{act})$  for each  $\text{state}, \text{act}$  pair in the episode
  - through multiple episodes
    - the *effect* of the true  $\text{transp}(\text{state}' \mid \text{state}, \text{act})$  is derived

Method	Category	Model-based?	Update Equation	Key Characteristics	On-/Off-policy	Convergence Target
Value Iteration	Value-based	Yes	$(V(s) \leftarrow \max_a \sum_{s'} P(s') [R(s,a,s') + \gamma V(s')])$	$s,a[R(s,a,s') + \gamma V(s')]$	Uses full model; synchronous updates	N/A $(V^*), (\pi^*)$
Policy Iteration	Policy-based	Yes	Eval: $(V^\pi(s) \leftarrow \sum_{s'} P(s') [R(s,\pi(s),s') + \gamma V^\pi(s')])$ Improve: $(\pi(s) \leftarrow \arg\max_a Q^\pi(s,a))$	$s,\pi(s)[R + \gamma V^\pi(s')]$	Alternates evaluation/policy update	N/A $(V^*), (\pi^*)$



Method	Category	Model-based?	Update Equation	Key Characteristics	On-/Off-policy	Convergence Target
TD(0)	Value-based	No	$V(s) \leftarrow V(s) + \alpha [r + \gamma V(s') - V(s)]$	Incremental via samples, bootstraps	On-policy	$(V^{\pi})$
Q-learning	Value-based	No	$Q(s,a) \leftarrow Q(s,a) + \alpha [r + \gamma \max_{a'} Q(s',a') - Q(s,a)]$	Off-policy, learns optimal Q	Off-policy	$(Q^*), (\pi^*)$
Monte Carlo	Value-based	No	$V(s) \leftarrow V(s) + \alpha [G - V(s)], \text{ where } (G) = \text{episode return}$	No bootstrapping; full episodes	On-policy	$(V^{\pi})$

In [2]: `print("Done")`

Done

