Tabular methods and dynamic programming

Optimal control problem

$$\max_{\rho} V^{\rho}(x) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r(X_k, \rho(X_k)) \,\middle|\, X_0 = x \right]$$

condition

$$\text{s.t.} \quad X_{k+1} \sim f(x_{k+1} | x_k, u_k)$$

$$r_k = r(x_k, u_k)$$

The optimal policy:

$$\rho^*(x) = \arg\max_{\rho} V^{\rho}(x)$$

Let's define $\hat{\rho}(u|\eta)$ — concatenation of an action u with an arbitrary tail policy η

Let's denote $X_+^u := f(x_+ | x, u)$

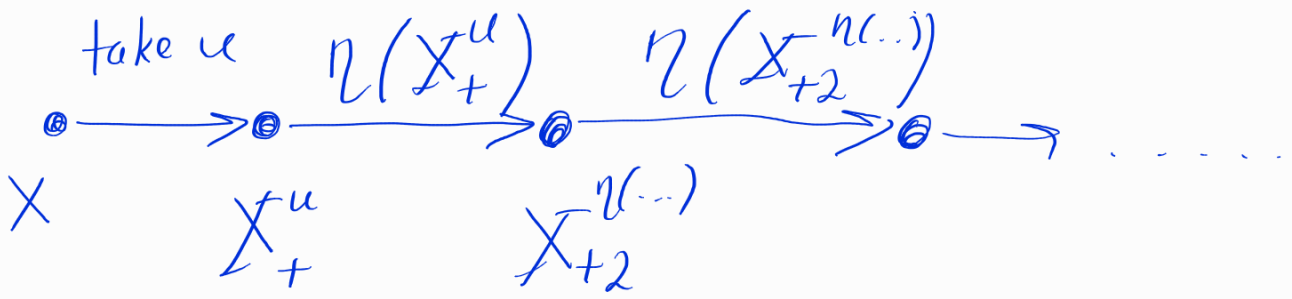
(next state upon applying the action u)

Let's take a look at the optimal value:

$$V^*(x) = \max_\rho V^\rho(x)$$

$$= \max_\rho \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r(X_k, \rho(X_k)) \,\middle|\, X_0 = x\right]$$

$$= \max_{\hat{\rho}} \left\{ r(x,u) + \gamma \mathbb{E}\left[V^\rho(X_+^u)\right] \right\}$$



$$\hat{\rho} = \langle \text{first apply } u, \text{ then follow } \eta \rangle$$

$$V^*(x) = \max_{\hat{\rho}} \left\{ r(x,u) + \gamma \mathbb{E}\left[V^\eta(X_+^u)\right] \right\}$$

$$\leq \max_{u} \left\{ r(x,u) + \gamma \mathbb{E}\left[V^*(X_+^u)\right] \right\}$$

On the other hand,

$$\max_{\hat{\rho}} \left\{ r(x,u) + \gamma \mathbb{E}\left[V^\eta(X_+^u) \right] \right\} \geq$$

$$r(x,v) + \gamma \mathbb{E}\left[V^\xi(X_+^v) \right]$$

$\forall$ action v and tail policy ξ

If we take $\sigma = \rho^*(x)$,

ξ = tail of ρ^*, then

$$\max_{\hat{\rho}} \left\{ r(x,u) + \gamma \mathbb{E}\left[V^{\hat{\rho}}(X_+^u) \right] \right\} \geq$$

$$r(x, \rho^*(x)) + \gamma \mathbb{E}\left[V^{\rho^*}(X_+^{\rho^*(x)}) \right] =$$

$$V^*(x) \, !$$

Combining the two inequalities that we obtained, we conclude:

$$V^*(x) = \max_u \left\{ r(x,u) + \gamma \mathbb{E}\left[V^*(X_+^u) \right] \right\}$$

↑

Dynamic programming principle

"DP"

Let's introduce a DP operator:

for an arbitrary function W,

$$T[W](x) := \max_u \{ r(x,u) + \gamma \mathbb{E}[W(X_+^u)] \}$$

↑

maps functions to functions

It so happens that T satisfies the following two conditions:

(1)

$\forall x, u, W_1, W_2$

$$r(x,u) + \gamma \mathbb{E}[W_1(X_+^u)] \leq r(x,u) + \gamma \mathbb{E}[W_2(X_+^u)]$$ whenever

$W_1(x) \leq W_2(x), \forall x$

And so

$$T[W_1] \leq T[W_2]$$

$\Rightarrow$ monotonicity condition

(the 1st Blackwell's contractive mapping condition)

② (T is discounting)

$$T[W+a](x) =$$

$$\max_u \{ r(x,u) + \gamma \mathbb{E}[W(x_+^u)] + \gamma a \}$$

$$= T[W] + \gamma a$$

Thus, T is contractive with modulus γ, and by the fixed-point theorem, a successive application of T to itself converges to a unique fixed point

So if we do iterations like

$$V_i := \mathcal{T}[V_{i-1}], \quad i \in \underbrace{\mathbb{N}_{+0}}_{\text{natural numbers with zero}}$$

starting at any initial guess V_0, we converge to a limit with the rate

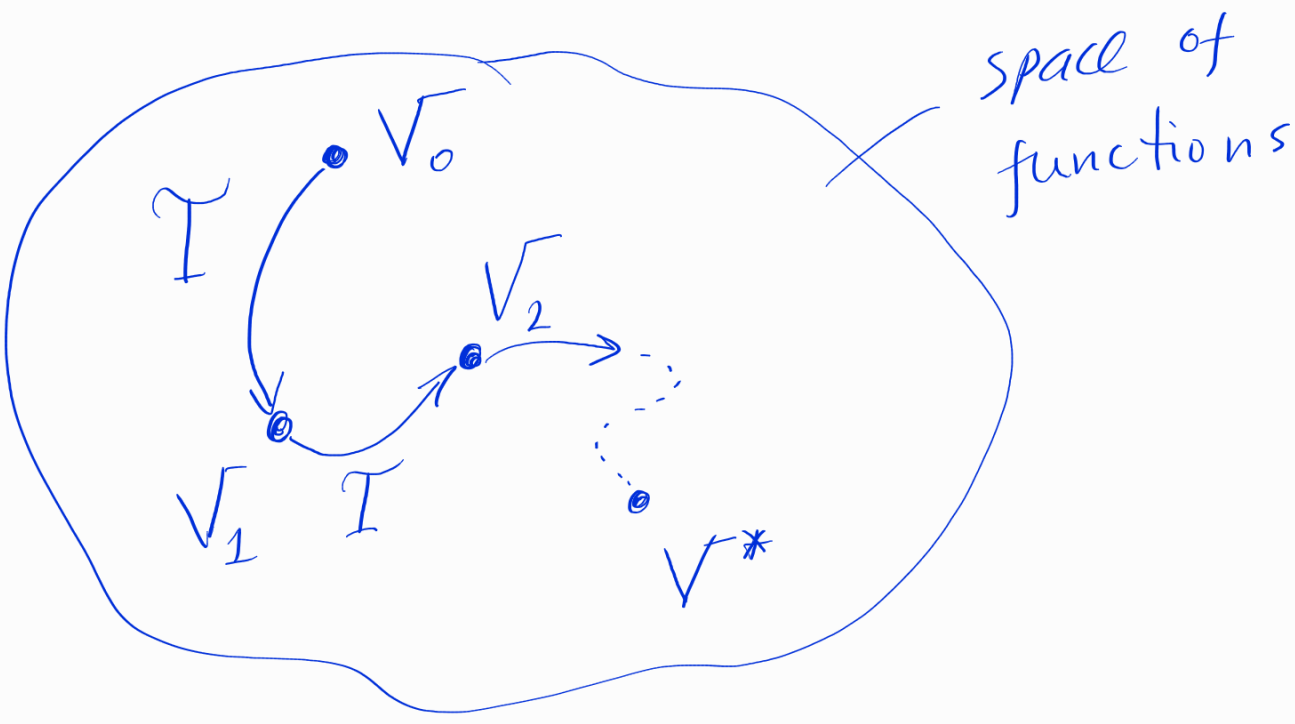
($\mathcal{T}^i$: i-fold application of $\mathcal{T}$)

$$\| \mathcal{T}^i[V_0] - V^* \| \leq \frac{\gamma^i}{1-\gamma} \| \mathcal{T}[V_0] - V_0 \|$$

($\|\cdot\|$: sup-norm)

sup-norm:

$$\| f - g \| \triangleq \sup_{x \in \mathbb{X}} |f(x) - g(x)|$$



Alg-m 1 ("Value iteration")

Start with a V_0

apply T succ-ly

after suff-t accuracy, say,

$\| V_i - V_{i-1} \| \leq \varepsilon$ is achieved,

compute a maximizer

ρ_i of V_i and that's

appr-ly the optimal policy

Algorithm 2 (Value iteration)

- Init value estimate V_0
- do until $\| V_i - V_{i-1} \| \leq \varepsilon$

  for all states x:

$$\rho_i(x) := \arg\max_u \{ r(x,u) + \gamma \mathbb{E}[V_i(X_+^u)] \}$$

$$V_{i+1}(x) := r(x, \rho_i(x)) + \gamma \mathbb{E}[V_i(X_+^{\rho_i(x)})]$$

Algorithm 3 (Policy iteration)

- Init policy ρ_0
- do until $\| V_i - V_{i-1} \| \leq \varepsilon$
  - *Solve (value update)

$$V_i(x) = r(x, \rho_i(x)) + \gamma \mathbb{E}[V_i(X_+^{\rho(x)})]$$

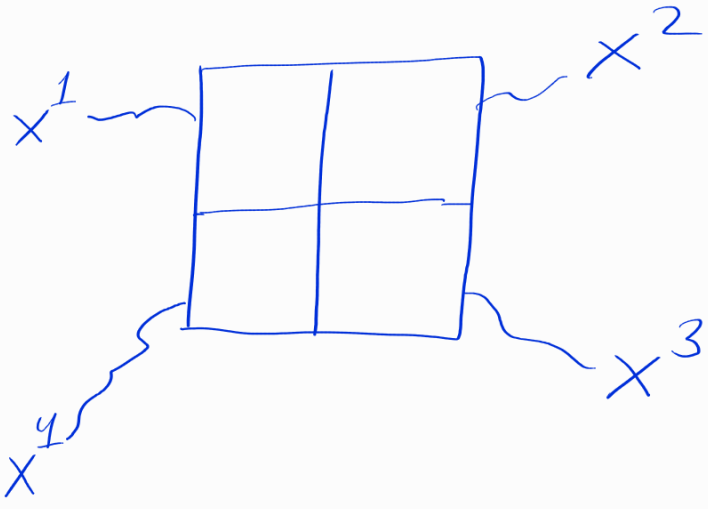
  - $\rho_{i+1}(x) := \arg\max_u \{ r(x,u) + \gamma \mathbb{E}[V_i(X_+^u)] \}$

* start with a guess V_i^0, and do (for j):

$$V_i^j(x) := r(x, \pi_i(x)) + \gamma \mathbb{E}\left[V_i^{j-1}\left(X_+^{\pi_i(x)}\right)\right]$$

Example: a small grid world

$\mathbb{X}$ - state space

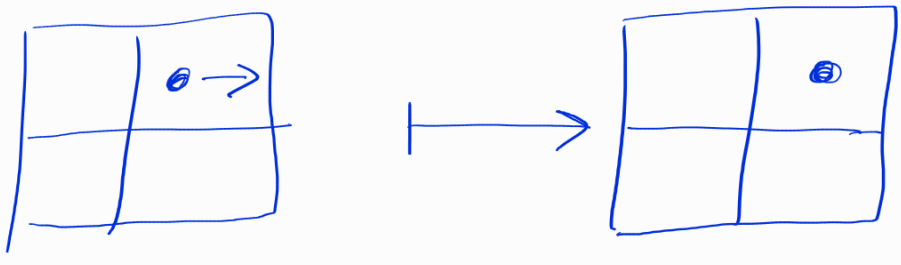


$\mathbb{U}$ - action space

$\mathbb{U} = \{\leftarrow, \rightarrow, \uparrow, \downarrow\}$

Environment dynamics:

in a cell, go to the next one if allowed, or stay if not



Reward

1	-100
1	100

Say x^3 is terminal, i.e., any action will force the agent to stay at x^3

Let's do value iteration.

Init V_0

0	0
0	0

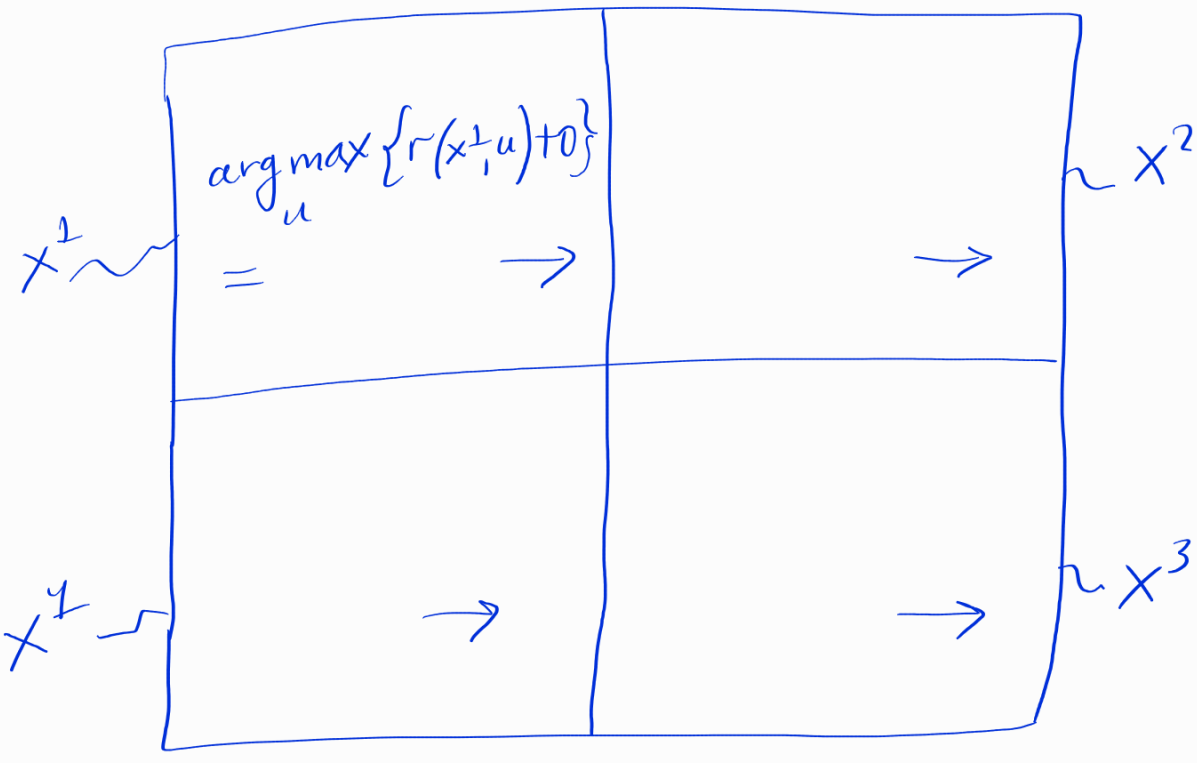
Let's calculate ρ_0:

algorithm recap

$$\rho_i(x) := \arg\max_u \{r(x,u) + \gamma \mathbb{E}[V_i(X_+^u)]\}$$

Take $\gamma = 1$. $\mathbb{E}$ can be dropped.

ρ_0



Reward

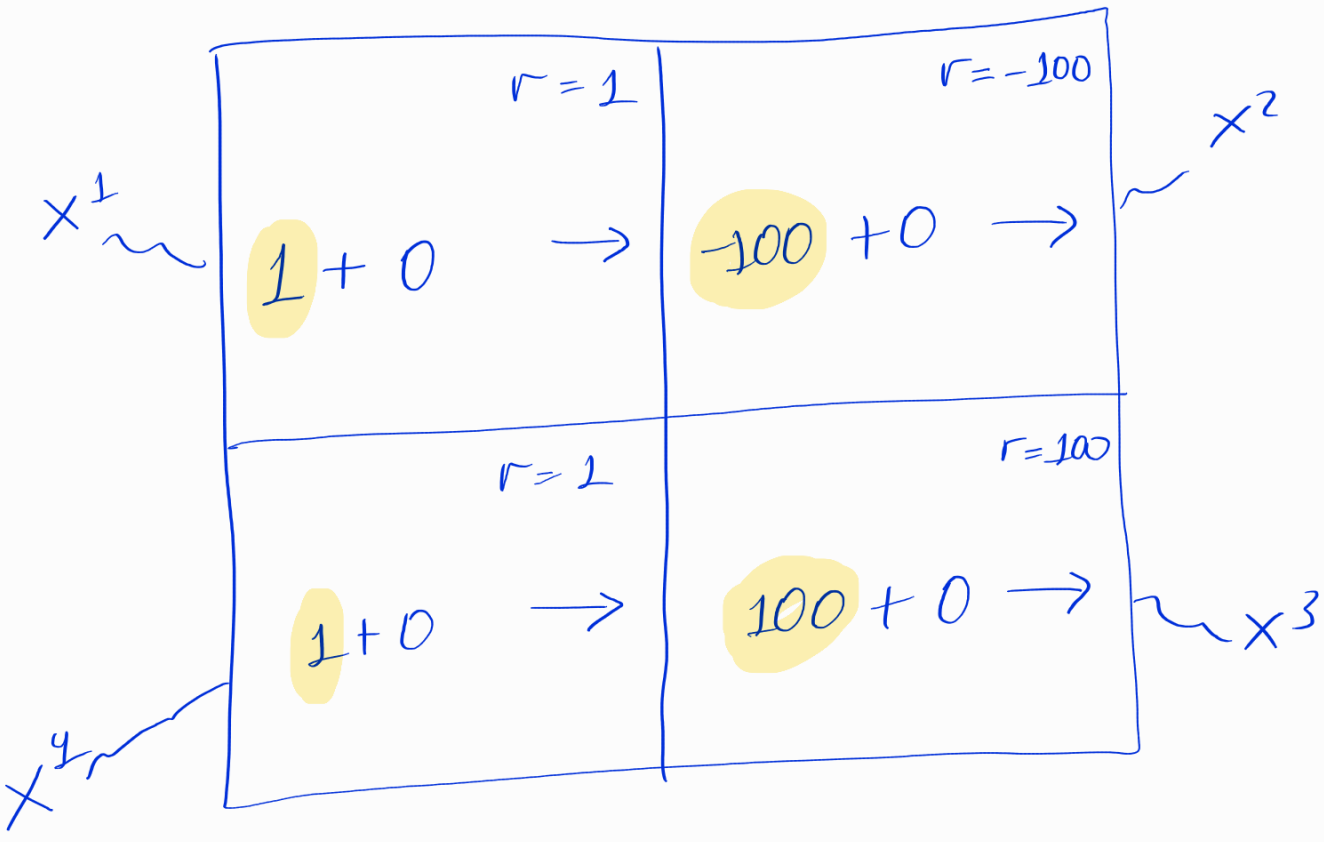
1	-100
1	100

Step 1

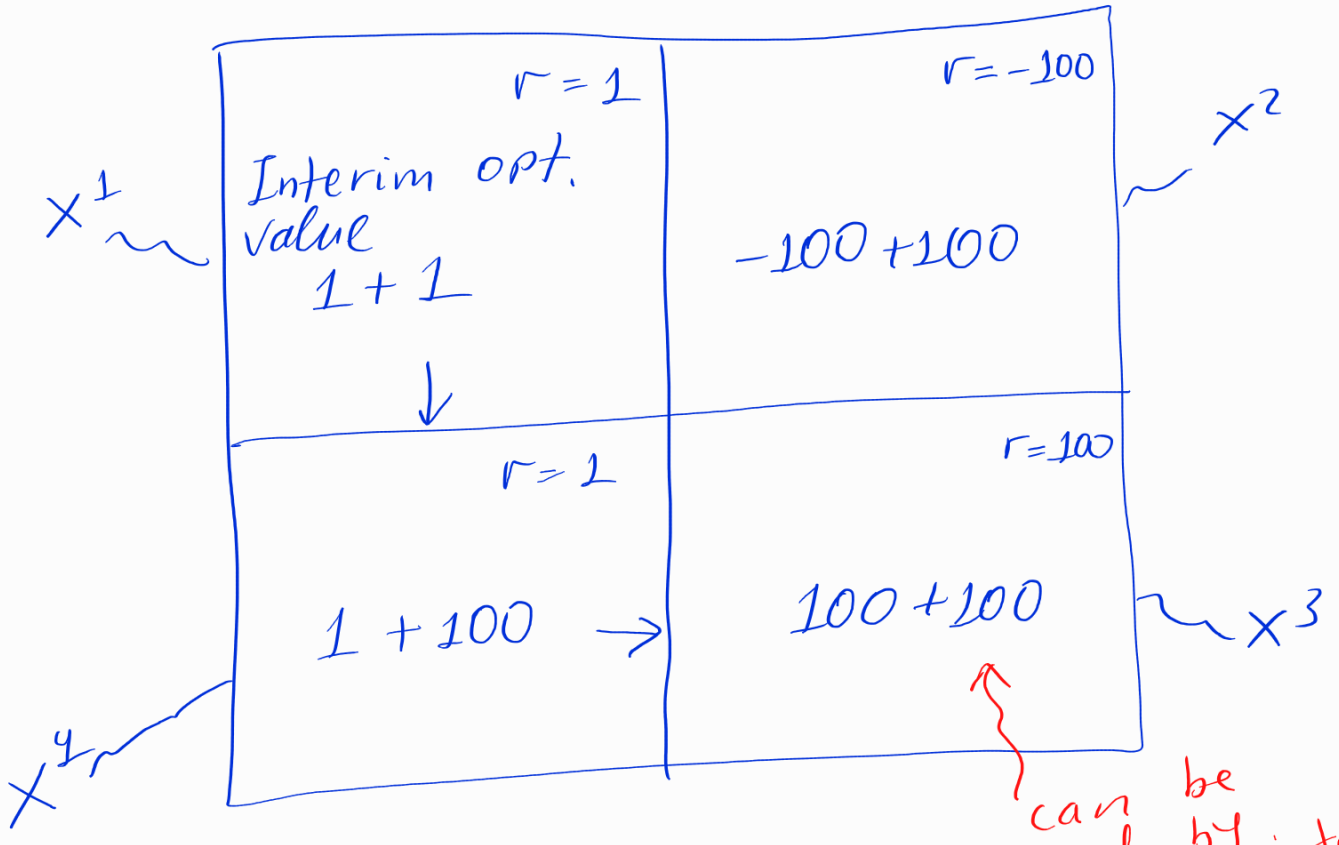
V_1

Algorithm recap

$$V_{i+1}(x) := r(x, \rho_i(x)) + \gamma \mathbb{E}\left[V_i\left(X_t^{\rho_i(x)}\right)\right]$$



ρ_1



can be fixed by teleporting into x^5 with $r^5 = 0$

algorithm recap

$$\rho_i(x) := \arg\max_u \{r(x,u) + \gamma \mathbb{E}[V_i(X_+^u)]\}$$

V^*

r = 1	r = -100
102	2
r = 1	r = 100
101	100

If game stops after 3 steps

V^*

r = -1	r = -100
98	0
r = -1	r = 100
99	100

Let's pretend the reward table was

-1	-100
-1	100