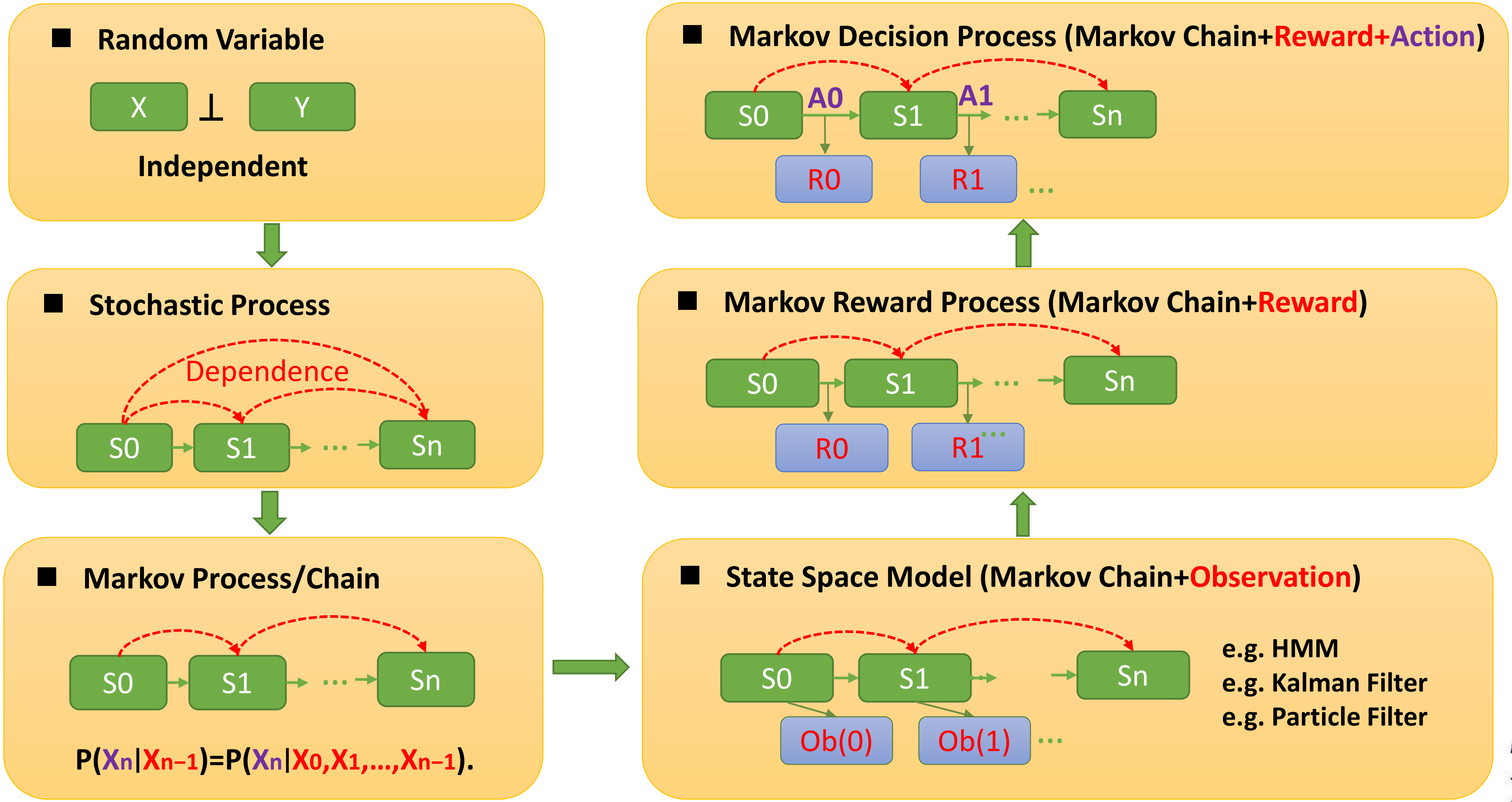
$$V^*(s) = \max_a \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma V^*(s')]$$

TP Reinforcement Learning

Adriana TAPUS and Juan Jose Garcia Cardinas and Adnan Saood

adriana.tapus@ensta-paris.fr & juan-jose.garcia@ensta.fr & adnan.saood@ensta-paris.fr

1. Basic knowledge



2. Terminologies

Return : cumulative future reward

$$U_t = R_t + R_{t+1} + R_{t+2} + \ldots\ldots$$

R_t *and* R_{t+1} *are important equally?*

Discounted Return : discounted cumulative future reward

$$U_t = R_t + \gamma \cdot R_{t+1} + \gamma^2 \cdot R_{t+2} + \ldots\ldots$$

Action-value function $Q_\pi(s_t, a_t)$

$$Q_\pi(s_t, a_t) = \mathbb{E}\,[U_t | S_t = s_t, A_t = a_t]$$

State-value function (value used in task 2)

$$V_\pi(s_t) = \mathbb{E}_A\,[Q_\pi(s_t, A)] = \sum_a \pi(a|s_t) \cdot Q_\pi(s_t, a)$$

Reinforcement Learning (RL)

- Learning through Interaction with Environment

- Agent is in State *s*
- Agent executes Action *a*
- Agent receives a *Reward r(s,a)* from the environment

- Goal: Maximize *long-term discounted Reward*

Value-Based RL

- Policy Iteration:
  - Start with random policy π_0
  - Estimate Value-Function of π_i
  - Improve $\pi_i \rightarrow \pi_{i+1}$ by making it greedy w.r.t. to the learned value function
  - Exploration: Try out random actions to explore the state-space
  - Repeat until Convergence

- Learning Algorithms:
  - Q-Learning (off-policy), SARSA (on-policy)
  - Actor-Critic Methods, etc.

3. Value iteration

Value iteration is a method of computing an optimal policy for an MDP (Markov Decision Process) and its value.

$$
\begin{aligned}
Q_{k+1}(s,a) &= R(s,a) + \gamma * \sum_{s'} P(s' \mid s,a) * V_k(s') \\
V_k(s) &= \max_a Q_k(s,a)
\end{aligned}
$$

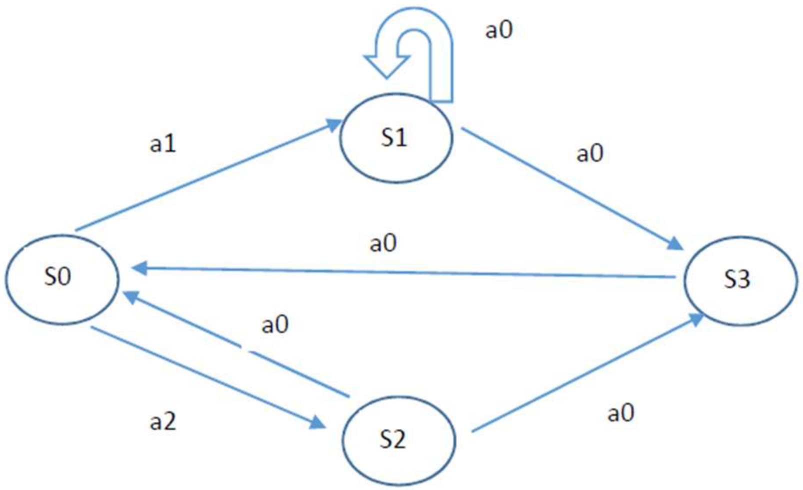
$$\pi[s] = \mathrm{argmax}_a R(s,a) + \gamma * \sum_{s'} P(s' \mid s,a) \cdot V_k[s']$$

[Interactive Example](https://perso.ensta-paris.fr/~saood/external/RL)
https://perso.ensta-paris.fr/~saood/external/RL

https://artint.info/2e/html/ArtInt2e.Ch9.S5.SS2.html

ENSTA IP PARIS

4. Task 1



In the figure above, the states are depicted by circles (S0, S1, S2, and S3) and the associated actions are indicated on the arrows: a0, a1, and a2. The transition functions for all the actions are shown below.

$$\begin{array}{ccc} & & \begin{array}{cccc} S0 & S1 & S2 & S3 \end{array} \ \to S' \text{ (future)} \\ T(S,a0,S') = & \begin{array}{c} S0 \\ S1 \\ S2 \\ S3 \\ \downarrow S(\text{Current}) \end{array} & \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1-x & 0 & x \\ 1-y & 0 & 0 & y \\ 1 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

$$T(S,a1,S') = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

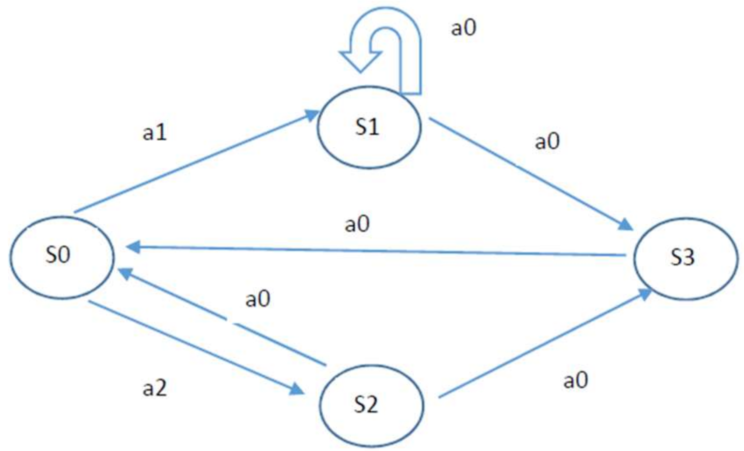
$$T(S,a2,S') = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Each of the parameters x and y are in the interval [0, 1], and the discounted factor $\gamma \in [0,1)$

The reward is:

$$R(s) = \begin{cases} 10, for\ state\ S3 \\ 1, for\ state\ S2 \\ 0, otherwise \end{cases}$$

4. Task 1



Question 1:

Enumerate all the possible policies

$\boldsymbol{\pi: s \rightarrow a}$

Question 2:

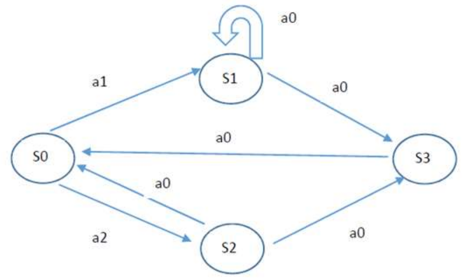
Write the equation for each optimal value function for each state

$(V^*(s0), V^*(s1), V^*(s2), V^*(s3))$

Reminder:

$$V^*(S) = R(s) + max_a \gamma \sum_{S\prime} T(S, a, S') V^*(S')$$

4. Task 1



Question 3:

Is there exist a value for x, that for all $\gamma \in [0,1), and\ y \in [0,1]$, $\pi^*(s0) = a2$. Justify your answer.

Reminder:

$$\pi^*(s) = \arg max_a \sum_{S'} T(S,a,S') V^*(S')$$

$R(s) \rightarrow$ not function of "a"

$$\boxed{V^*(S) = R(\cancel{a,}s) + max_a \gamma \sum_{S'} T(S,a,S') V^*(S')}$$

Question 4:

Is there exist a value for y, that for all $x > 0, and\ y \in [0,1]$, $\pi^*(s0) = a1$. Justify your answer.

Question 5: ***→ python code***

Using x=y=0.25 and $\gamma = 0.9$, calculate the π^* and V^* for all states.

Implement value iteration.

$$\begin{aligned} Q_{k+1}(s,a) &= R(s,a) + \gamma * \sum_{s'} P(s' \mid s,a) * V_k(s') \\ V_k(s) &= \max_a Q_k(s,a) \end{aligned}$$

Termination Rule: $|V_k(S) - V_{k-1}(S)|$<0.0001

Rule

--In groups

--Deadline: before Tuesday 21/10

Submit:

-- Code + Readme file with Q1 to Q4

-- Github or ENSTA gitlab

or

End!
Questions?