

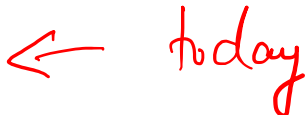
# Week 9: Temporal difference learning

COMP90054 – AI Planning for Autonomy

# Key concepts

- Q-learning and SARSA
- On-policy vs off-policy learning

# Model-based vs Model-free

- Model-based: Know the transition probability  $P_a(s'|s)$  and reward function  $r(s, a, s')$ 
  - E.g: Value Iteration
- Model-free: Don't know the transition probability and reward function   
SARSA, Q-learning

# Q-learning vs. SARSA

offline learning vs. online learning (next workshop).

Q-learning (Off-policy)	SARSA (On-policy)
$Q(s, a) = Q(s, a) + \alpha[r + \gamma \max_{a' \in A(s')} Q(s', a') - Q(s, a)]$	$Q(s, a) = Q(s, a) + \alpha[r + \gamma Q(s', \pi(s')) - Q(s, a)]$
<p>Update rule: Not update based on the policy. Update Q-function based on the assumption that the next action would be the action with the maximum Q.</p> <p>Optimistic: the greedy action will be chosen while in fact, the policy may choose an action other than the best</p>	<p>Update rule: Updated based on the policy. We know the action that it will execute next (whether it is best or not) when performing the update</p>
<ul style="list-style-type: none"> <li>Learning from prior experience</li> <li>The main advantage of off-policy approaches is that they can use samples from sources other than their own policy.</li> </ul>	<ul style="list-style-type: none"> <li>Learning on the job</li> <li>The main advantage of on-policy approaches is that they can learn optimal behaviour while operating in their environment.</li> </ul>

Combine Q-learning + SARSA.

- Should not start with a random policy
- + learn a sample policy based on Q-learning
- + Then, we optimise the policy in a real environment using SARSA.

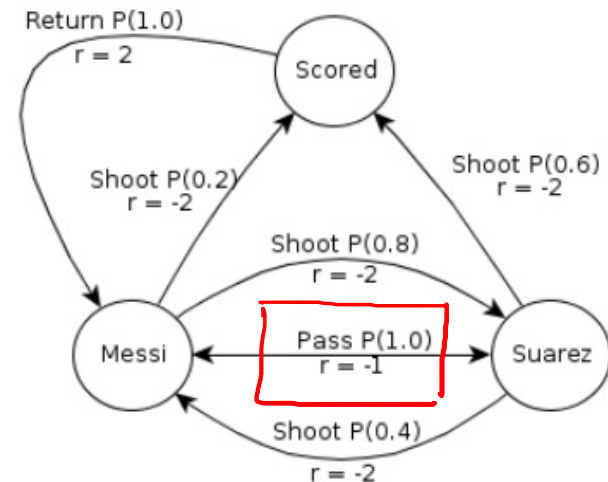
## Problem 2: Q-learning

Q-table

$$Q(\text{Suarez}, \text{shoot}) = -0.2$$

State \ action	Pass	Shoot	Return
Messi	-0.4	-0.8	-
Suarez	-0.7	-0.2	-
Scored	-	-	1.2

The following diagram shows the transition probabilities and rewards:



In the next step of the episode, from the state 'Suarez', Suarez passes the ball to Messi. Show the Q-learning update for this action using a discount factor  $\gamma = 0.9$  and learning rate  $\alpha = 0.4$

**Note:** Assume that this is a model-free problem, so the transition probabilities are not accessible to your algorithm.

$$Q(s, a) = Q(s, a) + \alpha [r + \gamma \max_{a' \in A(s')} Q(s', a') - Q(s, a)]$$

Suarez  $\xrightarrow{\text{pass}} \text{Messi}$   
 $s \quad a \quad s'$

$$\begin{aligned}
 Q(\text{Suarez}, \text{pass}) &= \underline{Q(\text{Suarez}, \text{pass})} + \alpha [r(\text{Suarez}, \text{pass}, \text{Messi}) + \gamma \underline{\max_{a' \in A(\text{Messi})} Q(\text{Messi}, a')}] - Q(\text{Suarez}, \text{pass}) \\
 &= -0.7 + 0.4 [-1 + 0.9 \times (-0.4) - (-0.7)] \\
 &= -0.964
 \end{aligned}$$

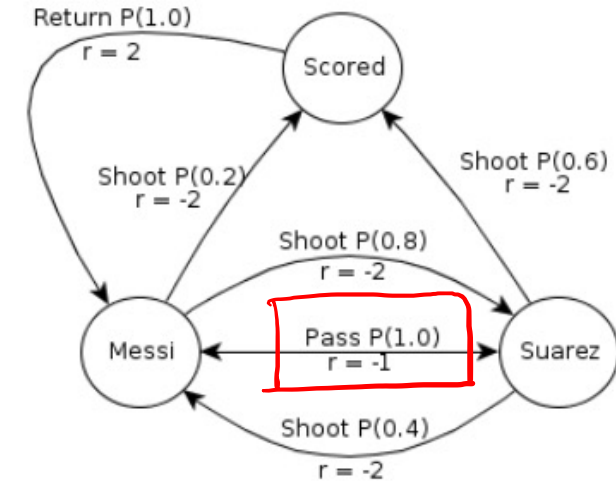
# Problem 3: SARSA

Q-learning

SARSA

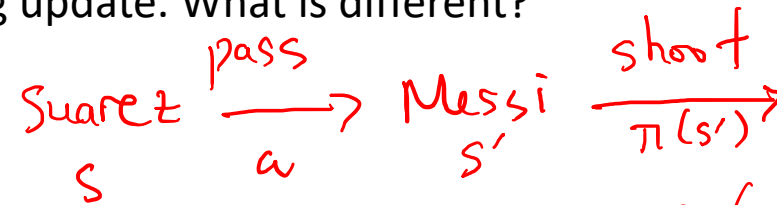
	Pass	Shoot	Return
Messi	-0.4	-0.8	-
Suarez	-0.7	-0.2	-
Scored	-	-	1.2

The following diagram shows the transition probabilities and rewards:



Consider again being in the state 'Suarez', Suarez passes the ball to Messi and then Messi decides to shoot. Show the SARSA update for the Pass action using a discount factor  $\gamma = 0.9$  and learning rate  $\alpha = 0.4$  and assuming  $a'$  (the next action to be execute) is **Shoot**. Compare to the Q-learning update. What is different?

$$Q(s, a) = Q(s, a) + \alpha [r + \gamma Q(s', \pi(s')) - Q(s, a)]$$



$$\begin{aligned}
 Q(\text{Suarez}, \text{pass}) &= Q(\text{Suarez}, \text{pass}) + \alpha [r(\text{Suarez}, \text{pass}, \text{Messi}) + \gamma Q(\text{Messi}, \text{shoot}) - Q(\text{Suarez}, \text{pass})] \\
 &= -0.7 + 0.4 [-1 + 0.9 \times (-0.8) - (-0.7)] \\
 &= -1.108
 \end{aligned}$$

# Problem 3: N-step TD (SARSA)

**N-step TD will not be examinable**