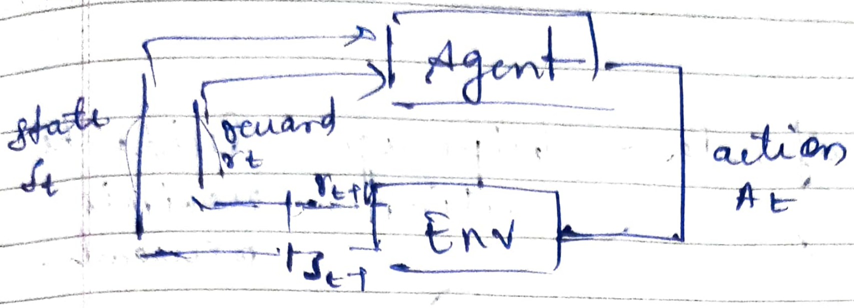
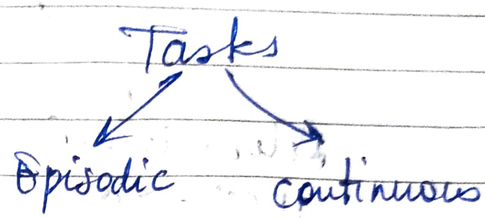
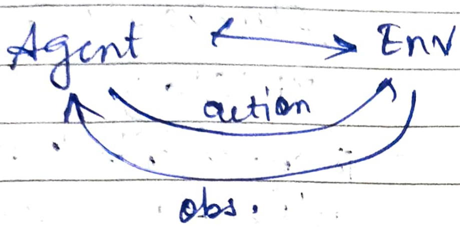
RL - Module 1 & 2

1) RL :
- feedback based ML technique
- agent learns to take actions — env — max. rewards
- agent — accomplishing — tasks — env — +ve/-ve rewards







2) - State vector — list of features — helps agent take action.
↳ collection of relevant observations.

3) Objectives of RL agents —

- episodic tasks — find seq. of actions — make — majority of episodes successful
- continuous task — break into multiple episodes — find actions — maximise avg. rewards — from those episodes.

4) Action $\xrightarrow{\text{consequence}}$ change of state
$\qquad\qquad\qquad\qquad\searrow$ reward

Action + consequence $\rightarrow$ knowledge Base

5) Policy

policy – set of rules – helps agent decide the action – maximise rewards.

Deterministic	Stochastic
$\pi(S) \rightarrow a$	$\pi(S/a)$
– given state – what action to maximize rewards ?	– prob. distributn over actions for each state
– always produce same action	– o/p a set prob. of each possible action.
– outcome is fully determined	– outcome uncertain.
– optimal action known with certainity	– multiple optimal outcomes.

6) Exploitation - greedy approach - agent takes the same actions it has taken before which have proven to get some rewards

- best action in the face of current knowledge
- doesn't improve the knowledge of the agent

Exploration - agent focuses on improving the knowledge of the env.
- takes steps it has never taken before to get to know the env.
- reaps long term benefits

7) Markov State:
Markovian Assumption - current state contains all the necessary info about the past states & past actions
- current state captures info abt past states

$$P[S_{t+1} | S_t] = P[S_{t+1} | S_1, \ldots S_t]$$

8) Markov Decision Process
- stochastic decision making process
- uses mathematical framework - model decision making
- MDP evaluate which action the decision maker should take.

9) Value Function:
- Value of a state - total reward an agent can expect to accumulate - starting from that state.

State value function
- $V_\pi(s)$
- Exp. returns starting from state 's' following π till we reach terminal state
- $V_\pi(s) = E[R_\pi(s)]$
- How good it is to be in a particular state?
- evaluation done to maximize total rewards

Action value function
- $q_\pi(s,a)$
- The value of taking an action
- Exp. returns starting from state 's' following π & take action 'a'
- $q_\pi(s,a) = E[R_\pi(s,a)]$
- calculates the value of performing an action.

10) Optimal Policy: $\forall \pi : \pi \geq \pi'$

$\pi > \pi'$ if $\forall s$: $V_\pi(s) \geq V_{\pi'}(s)$

11) Model of the env:
$P(s', r \mid s, a)$
- describes the behaviour of env in response to the actions

Model Based
- implicit
- model maps consequences to actions

Model Free
- explicit
- it cannot.

12) RL Equations

1) State Value Function:

$$V_\pi(s) = E[R_\pi(s)]$$
$$q_\pi(s,a) = E[R_\pi(s,a)]$$

Value of state 's' $= R_\pi(s)$

Value ~~for each~~ each action 'a' in state 's' following $\pi(a/s)$ $= R_\pi(s,a)$

$$\therefore R_\pi(s) = \sum_{a \sim \pi(a/s)} \pi(a/s)\ R_\pi(s,a)$$

$$\therefore V_\pi(s) = \sum_{a \sim \pi(a/s)} \pi(a/s)\ q_\pi(s,a)$$

2) Action Value Function:

$$V_\pi(s) = \sum_a \pi(s/a)\ q_\pi(s,a)$$
$$q_\pi(s,a) = r + V_\pi(s')$$

\# Discount factor $(0 < \gamma < 1)$

$$q_\pi(s,a) = r + \gamma V_\pi(s')$$
$$= r + \gamma \sum_a \pi(a/s)\ q_\pi(s',a)$$

$$\therefore q_\pi(s,a) = \sum_{s'} \sum_{r} p(s'r \mid sa)\ (r + \gamma V_\pi(s'))$$

13) Bellman's Eq^n of optimality

$$\pi^*(a^*/s) = \begin{cases} 1, & a^* = \text{argmax}\ (q^*(a,s)) \\ 0, & \text{otherwise} \end{cases}$$

$$V_\pi^*(s) = \sum_a \pi^*(a/s)\ q^*(a,s)$$

$$q^*(s,a) = \sum_{s'} \sum_r p(s'r \mid sa)(r + \gamma V_\pi^*(s)).$$