22/07/25

U - 1 : INTRODUCTION

INTRODUCTION :

* Machine Learning
  - Supervised Learning
  - Unsupervised Learning
  - Reinforcement Learning

* Supervised - dataset input data ~~mapped~~ mapped to output
labelled data, regression, classification

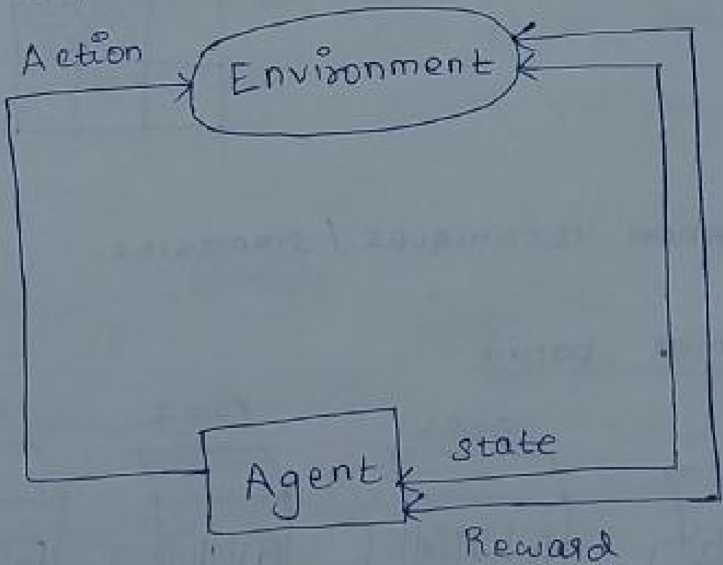
regression:
- linear
- log
- ridge
- Lasso
- Poly
- Multilinear

classification:
- decision
- KNN
- SVM
- Random forest
- Ensemble

* Unsupervised - no proper label / map, input & output
  - k-means cluster
  - DBSCAN cluster
  - PCA

* Why RL? → no data pts needed (model learn by itself)
  → too much data not knowing reliability
  → not practical for real envi
  → Learn by trial & error

1. Exploration 2. Exploitation



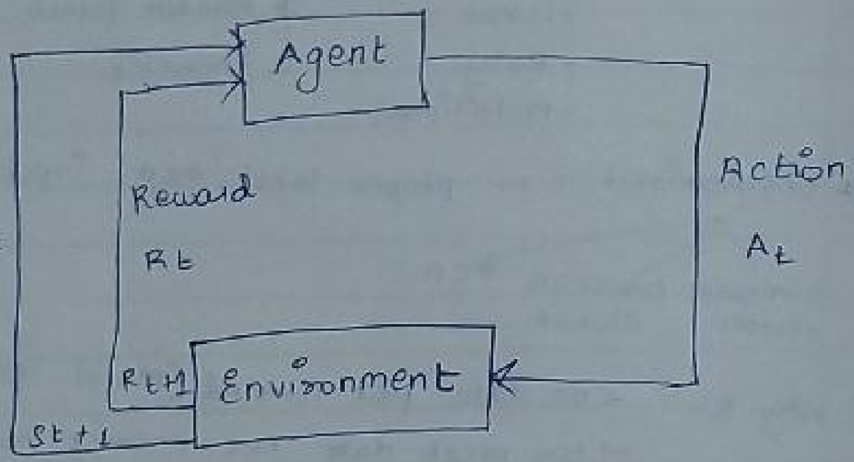
RL is sequence decision-making

Exploration :

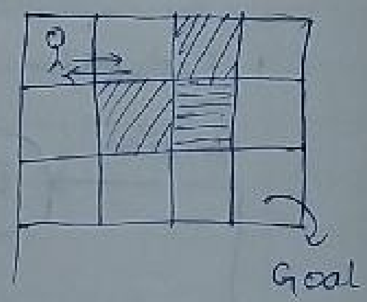
Exploitation :

* State: Agent resides / current state
* Action: agent moves from start to next state (actor/, affects environment)
* Reward: agent gets in return, based on goals / actions, maybe +ve, -ve
* Agent: Decision-making entity that acts - max rewards
* Environment: External system where agent interacts and learns

24/07/25

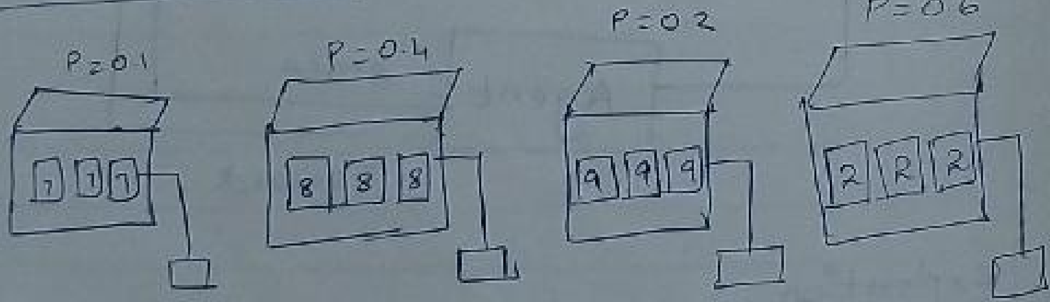


* Policy: strategy followed to maximize the rewards.



EXPLORATION TECHNIQUES / STRATEGIES:

1. One armed bandit



* Without any previous knowledge, trying everything, trial - exploration

2. Multi-armed bandit

Professor
300 days

R1 → 10 units happiness

R2 → 8 units happiness

R3 → 5 units of happiness

① Only exploration:

Professor explores the 3 restaurants for 300 days

$100 \times 10 + 100 \times 8 + 100 \times 5$

$= 1000 + 800 + 500$

$= 2300$ units of happiness

max happiness → [3000 - 2300]

regret → $= 700$ units

② Only Exploitation:

$1 + 2 + 3$ → 297

* 297 days, he uses the same restaurants which he has explored

* first day R1, second day R2, third day R3. ~~that day~~ then he decides and uses those ~~the~~ restaurant

③ ε - greedy strategy

ε = 10 %

$\boxed{300}$ ⇒ 30 days →
10 days - R1
10 days - R2
10 days - R3

rest 270 days selected restaurant

29/07/25

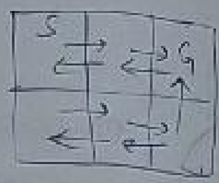
→ finding many possible solution - exploration

→ finding one solution and using it repeatedly without finding other solutions - exploitation.

ε → exploration

1 - ε → exploitation

Q - LEARNING ALGORITHM:



for each s, a initialize table entry
$\hat{Q}(s, a)$ to zero

observe current state s

Do forever

- Select an action 'a' and execute it
- Receive immediate reward 'r'
- observe the new state s'
- update the table new entry for $\hat{Q}(s, a)$

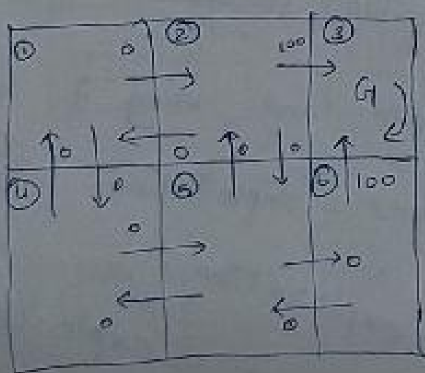
$$\hat{Q}(s, a) \longleftarrow r + \gamma \max Q(\hat{s}, \hat{a})$$

γ → discounting factor

- $s \leftarrow s'$

can be 0 to [0.9]

Eg:
* higher the Q value - ~~the~~ close to the goal
* Grid, goal is door, action is up, down, right, left.
* Each action gives a Q value, higher the q value, more close to reaching the goal.



near the goal
higher the reward
so value near goal is 100
and rest is 0

r(s, a) reward values

Reward Matrix 6x6

$$
\begin{array}{c|cccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 \\
\hline
1 & -1 & 0 & -1 & 0 & -1 & -1 \\
2 & 0 & -1 & 100 & -1 & 0 & -1 \\
3 & -1 & -1 & 0 & -1 & -1 & -1 \\
4 & 0 & -1 & -1 & -1 & 0 & -1 \\
5 & -1 & 0 & -1 & 0 & -1 & 0 \\
6 & -1 & -1 & 100 & -1 & 0 & -1
\end{array}
$$

* no connection b/w (3,1)/(1,3) so [-1]

* reward matrix is 100

* connected room (1,2)/(3,2) ... but no reward so zero.

* Initialize Q ~~value~~ matrix to zero.

$$
\begin{array}{c|cccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 0 \\
3 & 0 & 0 & 0 & 0 & 0 & 0 \\
4 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 \\
6 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}
\rightarrow \text{Q table.}
$$

$$Q(\overset{s,\ a}{2,3}) = R(\overset{s,\ a}{2,3}) + \gamma \max\left[\overset{Q(s',a')}{Q(3,3)}\right]$$

$$= 100 + (0.9)(0) = 100.$$

value from reward matrix then value from Q table

then update Q table

updated Q table

$$
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 100 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$Q(6,3) = R(6,3) + \gamma \max[Q(3,3)]$

$= 100 + (0.9)(0)$

$= 100$

update (6,3) value in Q table.

updated

	1	2	3	4	5	6
1	0	90	0	0	0	0
2	81	0	100	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	100	0	0	0

$Q(1,2) = R(1,2) + \gamma \max[Q(2,1) Q(2,3) Q(2,5)]$

$= 0 + 0.9 \max[0, 100, 0]$

$= 0 + 0.9 \times 100$

$= 90$

update $Q(1,2)$ as 90 in Q table.

see updated table.

$Q(2,1) = R(2,1) + \gamma \max[Q(1,2)\ Q(1,4)]$

$= 0 + 0.9 \max[90, 0]$ ~~= 0 + 0.9 × 0~~

$=$ ~~0.9~~ $0 + 0.9 \times 90$

$= 81$

$Q(1,4) = R(1,4) + \gamma \max[Q(4,1)\ Q(4,5)]$

$= 0 + 0.9 \max[0, 0]$

$= 0.$

$$Q(5,2) = R(5,2) + \gamma \max[Q(2,1)\ Q(2,3)\ Q(2,5)]$$

$$= 0 + 0.9 \max[81, 100, 0]$$

$$= 0 + 0.9 \times 100$$

$$= 90$$

$$Q(5,4) = R(5,4) + \gamma \max[Q(4,1)\ Q(4,5)]$$

$$= 0 + 0.9 \max[0]$$

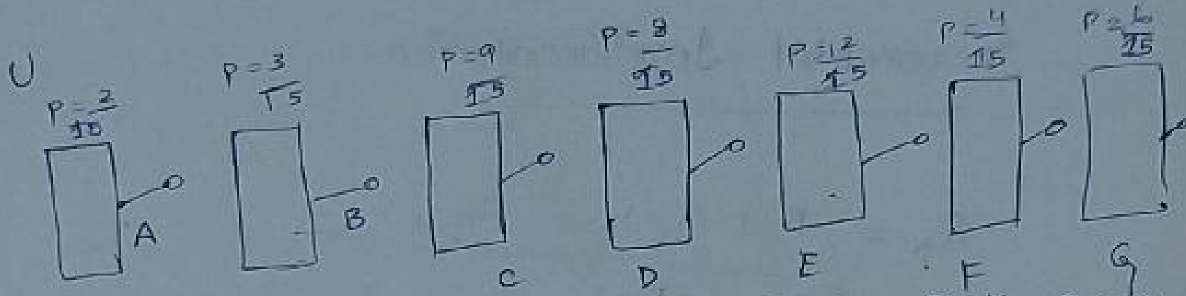
$$= 0$$

Final updated Q value after

	1	2	3	4	5	6
1	0	90	0	0 72	0	0
2	81	0	100	0	81	0
3	0	0	0	0	0	0
4	81	0	0	0	81	0
5	0	90	0	72	0	90
6	0	0	100	0	81	0

Model based RL	Model free RL

21/7/25



$P = \frac{2}{15}$ $P = \frac{3}{15}$ $P = \frac{9}{15}$ $P = \frac{8}{15}$ $P = \frac{12}{15}$ $P = \frac{4}{15}$ $P = \frac{6}{15}$

CC   NC   NN   NN   NN   CN   CC

coin, No coin.

[Multi-armed bandit problem]

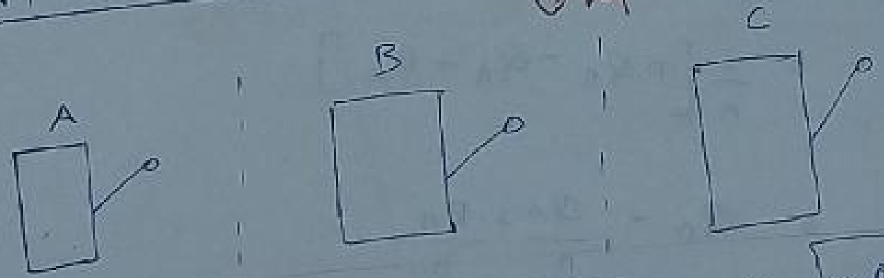
$\varepsilon \rightarrow$ exploration

10%. $1 - \varepsilon$ [0.9]

based on the resut $\rightarrow$ exploitation

UPPER CONFIDENCE BOUND $\rightarrow$ [no PE]

hope



$$Q(A) + \sqrt{\frac{2 \ln N}{n_A}} \quad Q(B) + \sqrt{\frac{2 \ln N}{n_B}} \quad Q(C) + \sqrt{\frac{2 \ln N}{n_C}}$$

100   ($\ln \rightarrow$ log arithmetic)

bound value

$$N = n_A + n_B + n_C$$

[Total no of machines] (no. of times playing for machine A, B, C)

* Bound + Estimated Q value should be maximum, [Bound value $\rightarrow \sqrt{\frac{2 \ln N}{n_A}}$]

* Deterministic approach [fixed action at a state]

Incremental Implementation:

$$Q_n = \frac{R_1 + R_2 + \dots R_{n-1}}{n-1}$$

$$Q_{n+1} = \frac{R_1 + R_2 + \dots R_n}{n}$$

2 3rd

$$Q_{n+1} = \frac{1}{n} \sum_{i=1}^{n} R_i = \frac{1}{n} \sum_{i=1}^{n-1} R_i + R_n$$

$$= \frac{1}{n} \times \frac{n-1}{n-1} \sum_{i=1}^{n-1} R_i + R_n \Rightarrow \frac{n-1}{n-1} \sum_{i=1}^{n-1} R_i$$

Q_n

$$= \frac{1}{n} \left[(n-1) \times Q_n + R_n \right]$$

$$= \frac{1}{n} \left[n Q_n - Q_n + R_n \right]$$

$$= Q_n - \frac{Q_n}{n} + \frac{R_n}{n}$$

$$Q_{n+1} = Q_n + \frac{1}{n} \left[R_n - Q_n \right]$$

300

10%

ϵ Epsilon greedy

ϵ

$1-\epsilon$

ϵ 0.1

$1 - 0.9$