**Fundamentals of Reinforcement Learning**

**Why use AI?** - For example, if you're playing bowling - you know that you need to hit the pins and a strike is a perfect shot. We know this because we are rewarded with points for hitting the pins down and we are "punished" with no points when your ball ends in the ditch. So, your brain projects those conditions of the environment onto your actions and that's how you know when you are doing good and when - not so much. And that's how we learn.

**Plan of Attack**

-Reinforcement Learning

-Bellman Equation

-The “Plan”

-Markov Decision Process (MDP)

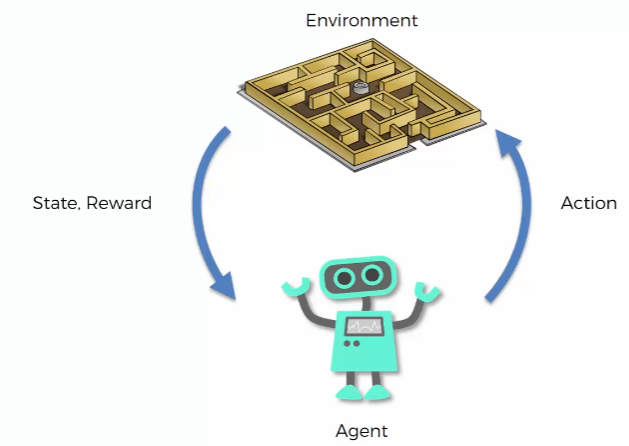
-Policy vs Plan

-Adding a “living Penalty”

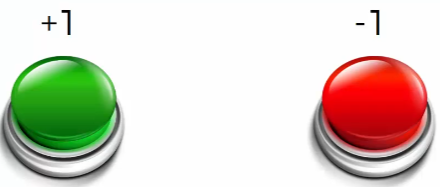
-Q-Learning Intuition

-Temporal Difference

-Q-Learning Visualization

**1) Reinforcement Learning**

Example- Training a dog, biscuit as a reward if he does something right, otherwise bad dog.



Goal is to not put a hardcore algorithm inside a “robodog”. Rewards +1 and bad movement is -1. Experiments and figuring out itself. But there is no hardcore algorithm fitted in it, something that it learns itself based on rewards.

Shape, table

Description automatically generated**2) Bellman Equation (Richard Ernest Bellman)**

- s – State

- a – Action

- R – Reward

- (gamma) – Discount

Text

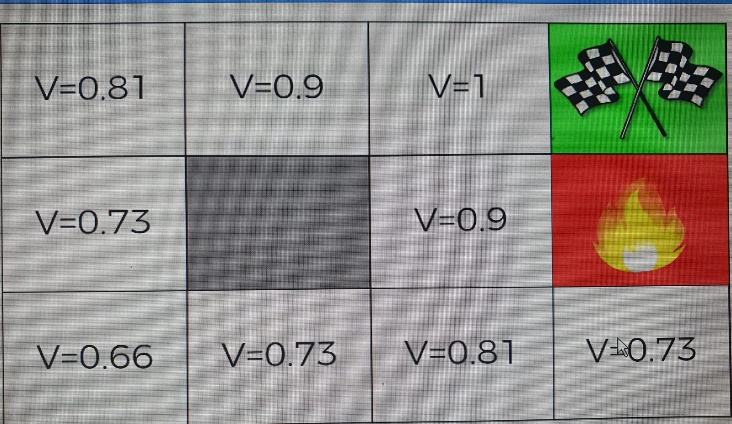
Description automatically generatedAssume a board with a fire (-1), and a green (+1). It will find possibilities to reach green. So as soon as it reaches green, the preceding states are V=1, and continues to precede ones to reach the previous preceding ones that is leading to green. Approach doesn’t work because V=1 can give multiple directions that could confuse the AI.

Table

Description automatically generatedComing to **Bellman equation,**

S’ – Following state that we will end up in

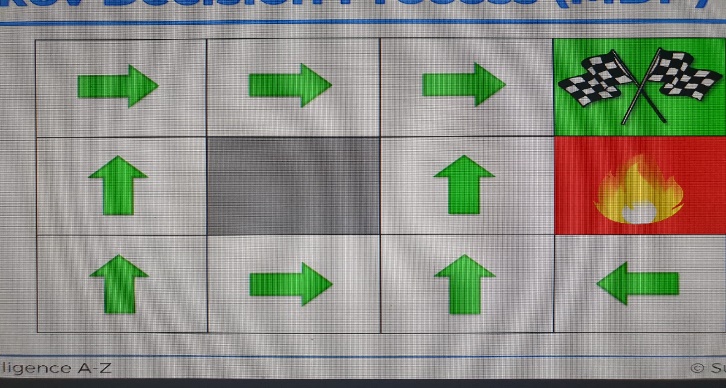
max- many actions can be taken

Gamma can take any value; each place preceding the final +1 Reward goes below the precedingly. So, preceding to +1 is better than preceding to itself and so on.

**3) The “Plan”**

Treasure map for the agent, Arrows will be a better choice to choose to show the direction where agent should go, from lower number to higher number.

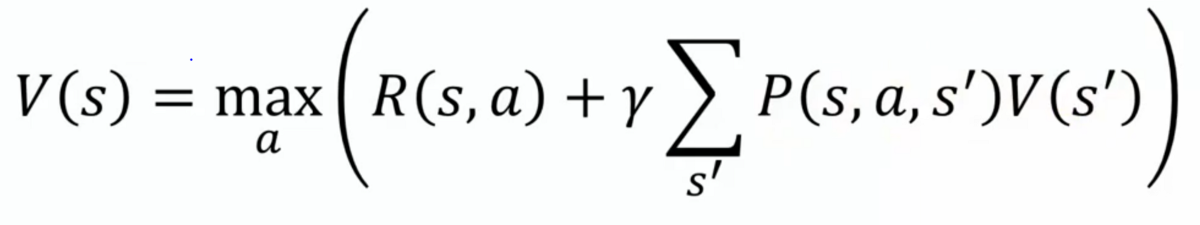
**4) Markov Decision Process (MDP)**

**Deterministic Search –** If an agent can go up, it will only go up.

**Non-Deterministic Search –** Agent can go up, or right or left or so on. Its non-determinant.

Randomness is there, not in control of agent.

**Markov Process-** A stochastic process has Markov property if the conditional probability distribution of future states of the process (conditional on both past and present states) depends only on the present state, not on the sequence of events that preceded it. A process with this property is called Markov Process.

**Markov Decision Process (MDPs)** provide mathematical framework for modeling decision making in situations where outcomes are partly random and partly under the control of a decision maker. (Sophisticated Bellman equation)

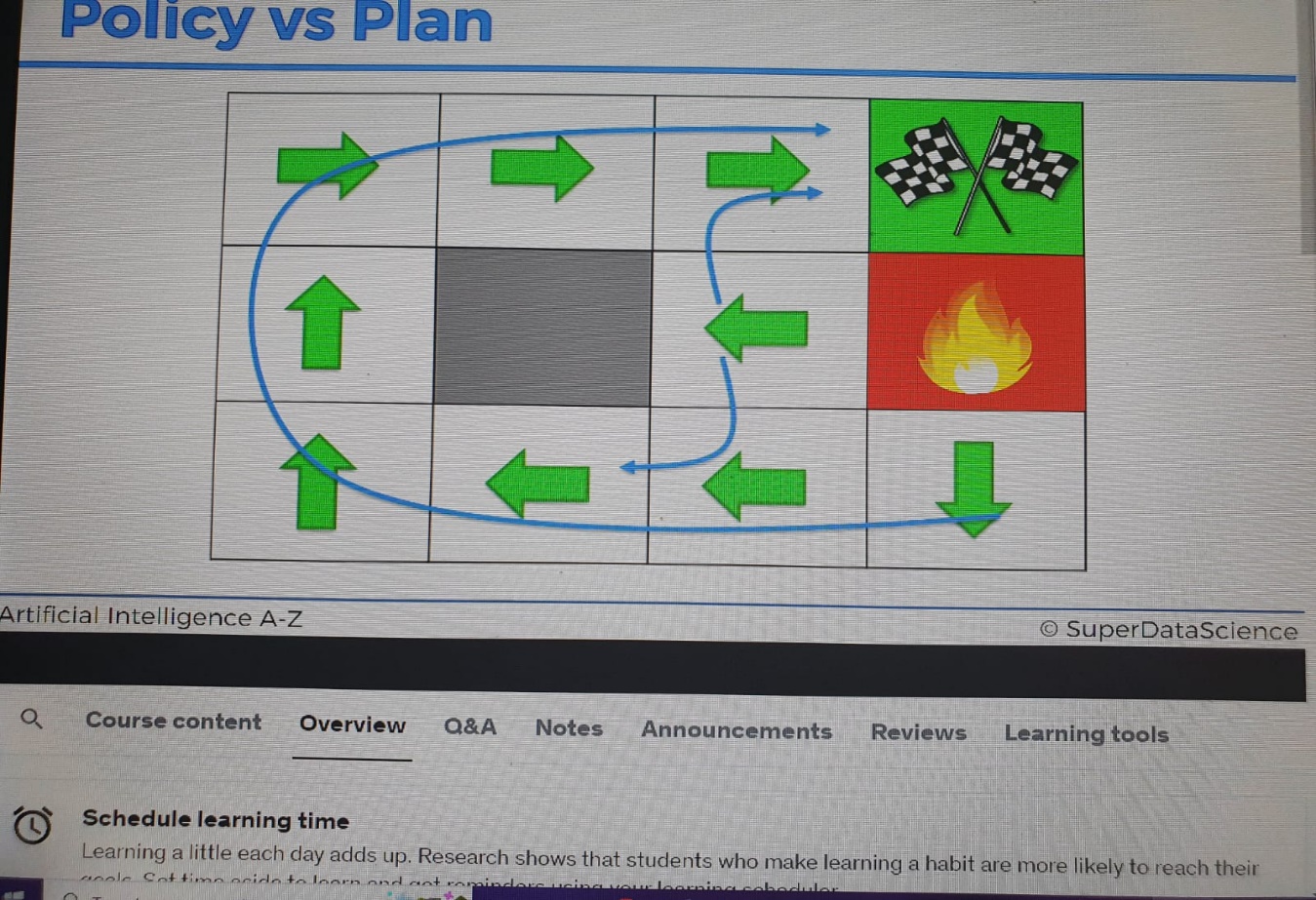
The sum of state is like an expected/ probability.

Text

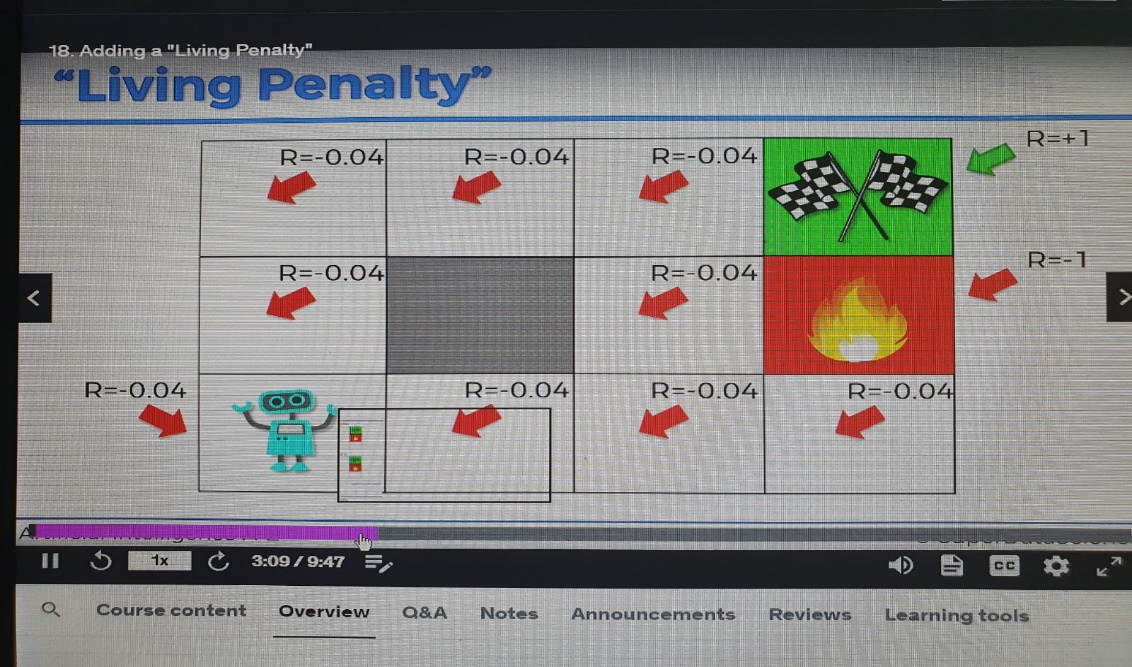
Description automatically generated

**5) Policy vs Plan**

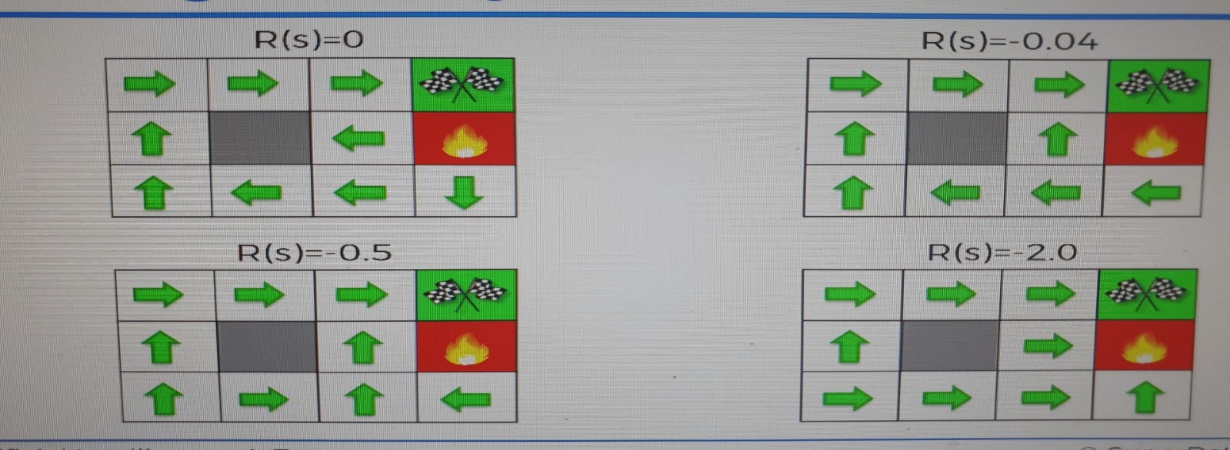
Plan only when you know what to do, but we are dealing with randomness more.

****

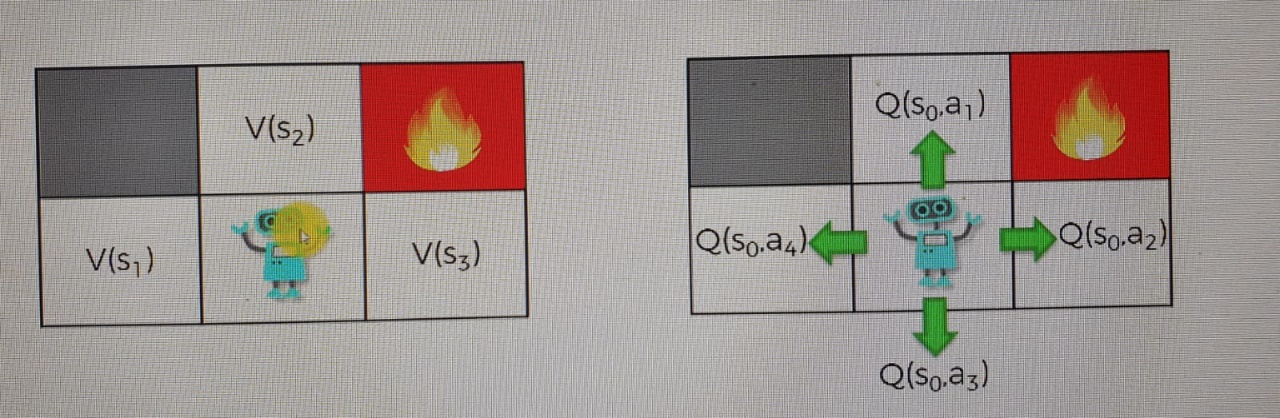
**6) Living Penalty**



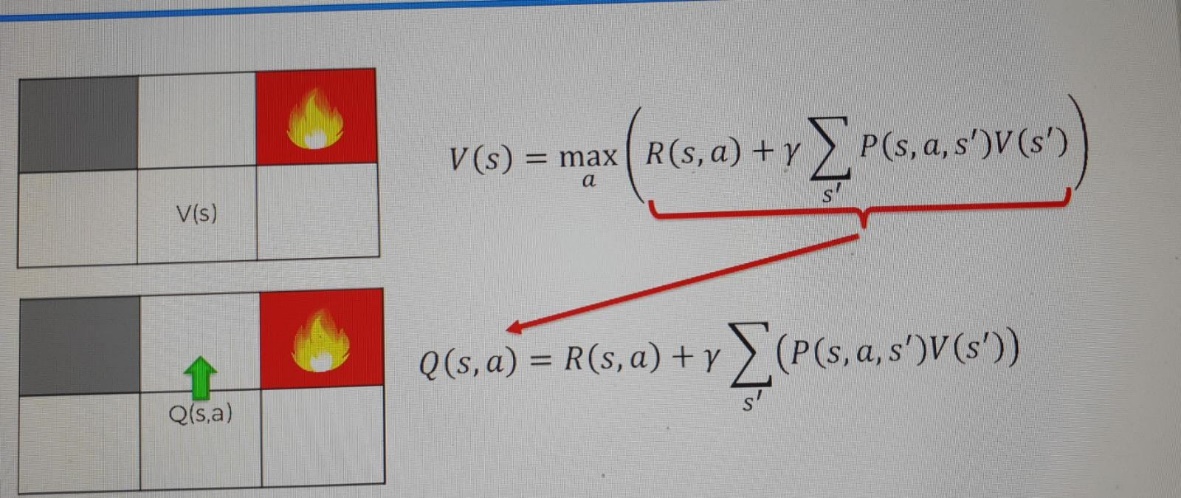
Four environments are to be noticed with different rewards

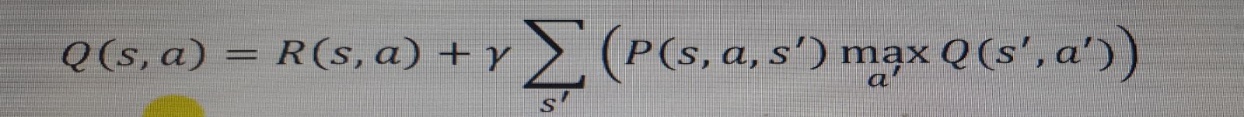


**7) Q-Learning Intuition**

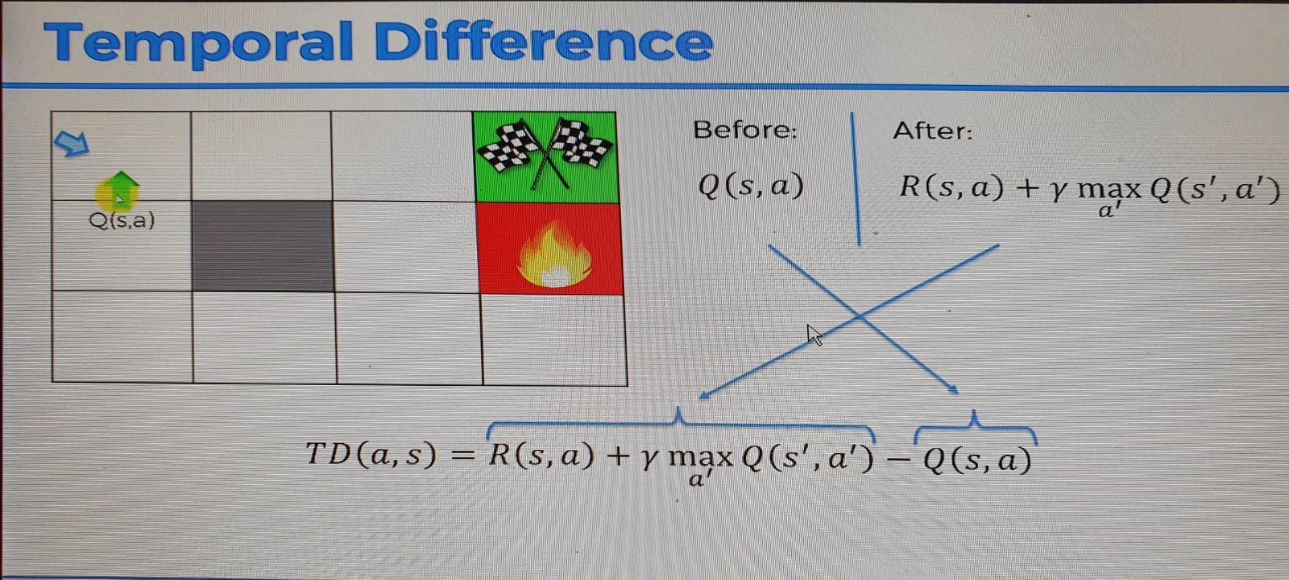


Q represents the quality of the action. Actions lead to states.





**8) Temporal difference**



We don’t want to remove Q(s,a) directly because it can be due to a randomness that new Q(s,a) which can be inconsistent.

