Lecture 4 (Video 9)

Date M T W T F S

Castelli

Bellman's equation

state value function :

A function that tells us the value of a state

value of state tells us how good and bad are state

value of state depends on the policy being followed

$$V_\pi(s) = \mathbb{E}(G_t | S_t = s)$$

Return is the discounted sum of reward.

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots$$

$$G_t = R_{t+1} + \gamma G_{t+1}$$

$$V_\pi(s) = \mathbb{E}(R_{t+1} + \gamma G_{t+1} | S_t = s)$$

$$G_{t+1} = \mathbb{E}(G_{t+1} | S_{t+1} = s')$$

$$V_\pi(s) = \mathbb{E}(R_{t+1} + \gamma (\mathbb{E}(G_{t+1} | S_{t+1} = s')) | S_t = s)$$

$$V_\pi(s)$$

$$V_\pi(s) = \mathbb{E}(R_{t+1} + \gamma V_\pi(s') | S_t = s)$$

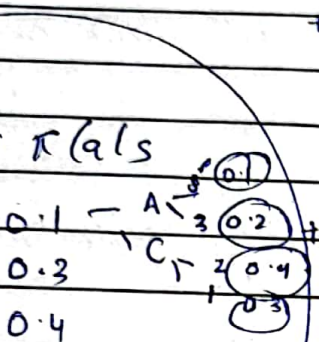
$$V_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma V_\pi(s')]$$

Date M T W T F S

4 states

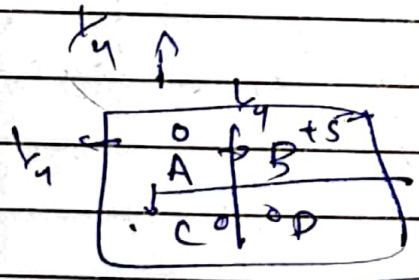
$V_\pi(A) = 0.1[0.1(5+0.9V_\pi A) + 0.2(3+0.9V_\pi A) + 0.4(2+0.9V_\pi C) + 0.3(\quad)] + 0.4[\quad] + 0.7[\quad]$

s	a	$\pi(a\|s)$
A	left	0.1
	right	0.3
	bottom	0.4
	up	0.2



For deterministic

$V_\pi(A) = \frac{1}{4}(0.9 V_\pi(C)) + \frac{1}{2}(0.9 V_\pi(A)) + \frac{1}{4}(5 + 0.9 V_\pi(B))$



book example

Date [M][T][W][T][F][S]

Action value

$$q_\pi(s,a) = \mathbb{E}(G_t \mid S_t = s, A_t = a)$$

$$= \mathbb{E}(R_{t+1} + \gamma G_{t+1} \mid S_t = s, A_t = a)$$

$$G_{t+1} = \mathbb{E}(G_{t+1} \mid S_{t+1} = s')$$

$$q_\pi(s,a) = \mathbb{E}(R_{t+1} + \gamma \underbrace{[\mathbb{E}(G_{t+1} \mid S_{t+1} = s)]}_{V_\pi(s')} \mid S_t = s, A_t = a)$$

$$q_\pi(s,a) = \mathbb{E}(R_{t+1} + \gamma V_\pi(s') \mid S_t = s, A_t = a)$$

~~$\mathbb{E}(R_{t+1} + \gamma V_\pi(s')$~~ $= \mathbb{E}$

$$V_\pi(s') = \mathbb{E}(G_{t+1} \mid S_{t+1} = s')$$

$$q_\pi(s,a) = \mathbb{E}(G_{t+1} \mid S_{t+1} = s', A_{t+1} = a')$$

$$V_\pi(s') = \sum_a \pi(a|s) \underbrace{\mathbb{E}(G_{t+1} \mid S_{t+1} = s', A_{t+1} = a')}_{q_\pi(s',a')}$$

$$q_\pi(s,a) = \mathbb{E}(R_{t+1} + \gamma \sum_a \pi(a|s) q_\pi(s',a') \mid S_t = s, A_t = a)$$

$$q_\pi(s,a) = \sum_{s'} \sum_r p(s',r|s,a) \left[r + \sum_a \pi(a|s) q_\pi(s',a') \right]$$

Date _____ M T W T F S

Castelli

Lecture 12 Video 10

Recap: Bellman Equations

$$V_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) [r + \gamma V_\pi(s')]$$

$$q_\pi(s, a) = \sum_{s'} \sum_r p(s', r|s, a) [r + \gamma V_\pi(s')]$$

$$q_\pi(s, a) = \sum_{s'} \sum_r p(s', r|s, a) [r + \gamma \sum_{a'} \pi(a'|s') q(s', a')]$$

A	B
C	D

→ Total 16 action equations 4 (actions left, right, up, bottom) for each state (A, B, C, D)

A → 4 states
B
C
D

Total 16 action state values $q_\pi(s', a')$

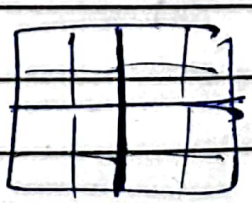
Here $p(s', r|s, a)$ is 1

$q(A, \text{left}) = 1[0 + 0.9 (\frac{1}{4} q(A, \text{left}) + \frac{1}{4} q(A, \text{right}) + \frac{1}{4} q(A, \text{up}) + \frac{1}{4} q(A, \text{down})$

$q(A, \text{right})$ 4

$q(A, \text{up})$ 4

$q(A, \text{down})$ 4

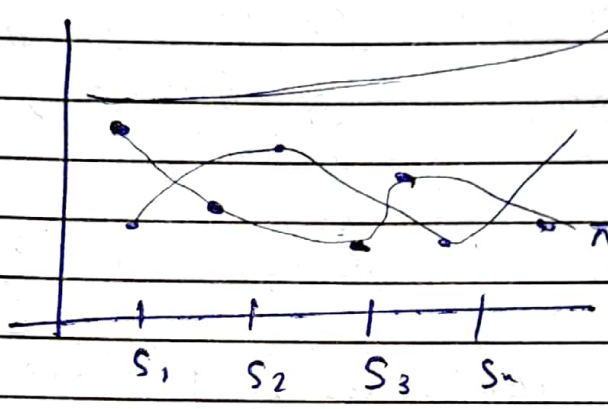


Date M T W T F S

Now these action value comes under a policy which is random so these maybe not be optimal values

When we say one policy is better than other value

π_1 π_2



In this case we can't determine which policy is better π_1 or π_2

$\pi_1 \geq \pi_2$

iff $V_{\pi_1}(s) \geq V_{\pi_2}(s)$ for each s belonging to S

$s \rightarrow S$

We can make a better policy by combining π_1 , π_2

MDP

Control → finding the best policy

Policy evaluation → Comparing different policy for finding the best policy

Date M T W T F S

→ Can more than one optimal policy exist

Yes

$$V_*(s) = \max_\pi V_\pi(s)$$

Value of the state under optimal policy

$$q_*(s,a) = \max_\pi q(s,a)$$

$$V_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s',r|s,a)[r + \gamma V_\pi(s')]$$

$$V_*(s) = \max_a \sum_{s'} \sum_r p(s',r|s,a)[r + \gamma V_\pi(s')]$$

What if two actions give you max value

↳ so we can choose from one of them

in this case even the deterministic policy become stochastic

optimal policy can be stochastic ?

Yes if there are more than one value/actions which gave you max value.

$$q_\pi(s,a) = \sum_{s'} \sum_r p(s',r|s,a)[r + \gamma \sum_{a'} \pi(a'|s') q_\pi(s',a')]$$

$$q_*(s,a) = \sum_{s'} \sum_r p(s',r|s,a)[r + \gamma \max_{a'} q_*(s',a')]$$

~~1, 2, 3, 4, 5, 6, 7, 8~~
~~9 10 11 12 13 14 15~~

Date (M T W T F S)

Castelli

$$V_\pi(s|a) \rightarrow q_\pi(s, a')$$

$$V_\pi(s') = \sum_a \pi(a'|s') q_\pi(s', a')$$

$$V_\pi(s) = \sum_a \pi(a|s) q_\pi(s, a)$$

$$V_*(s) = \max_a q_\pi(s, a)$$

max function is not a linear function
We can't solve it through system of linear equations
We solve it through DP.

V_* → next action from the π_* → π_*

Optimal policy

$$\pi_*(s) = \arg\max_a q_*(s, a)$$

Date M T W T F S

Castelli

Lecture 13 video 11

Revision

Optimal policy

$V_*(s) = \max_\pi V_\pi(s) \quad s \in S$

$q_*(s,a) = \max_\pi q_\pi(s,a)$

$V_*(s) = \max_a \sum_{s'} \sum_r p(s',r|s,a)\,[r + \gamma V_*(s')]$

$q_*(s,a) = \sum_{s'} \sum_r p(s',r|s,a)\,[r + \gamma \max_{a'} q_*(s',a')]$

$V_*(s)$ of each

20	18
13	15

$V_*(0,1)$

$q_*(s(0,1), up) = 1[-1 + 0.9(18)] = A$

$q_*(s(0,1), down = B$

$=$

which choose

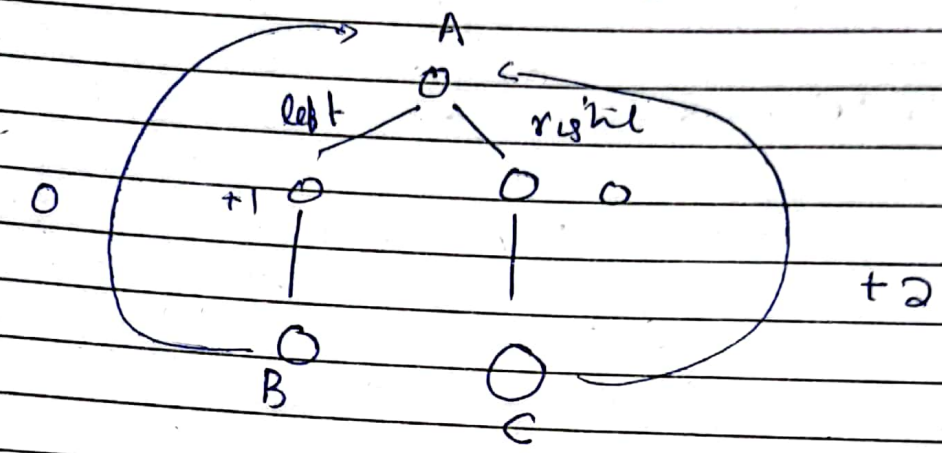
which is max

Date

M T W T F S

Castelli

$V_{\pi_1}(s) \geq V_{\pi_2}(s)$ for every $s \in S$



$\gamma = 0$

$$\sum_{s'} \pi(s,a) \sum_{s'} \sum_{r} p(s',r|s,a)[r + \gamma V_{\pi}(s'$$

$$V(A) = 1[1.\ (1 + 0(V(B))]$$

$$V_{left}(A) = 1$$

which value is greater left V_{left}

$$V_{right}(A) = 0$$

As γ is 0 we are not giving any value to the future rewards that's why on $\gamma = 0$ we got $V(left(A))$ greater the $V(right(A))$

Now if we want to give imp to future rewards

$\gamma = 0.9$

$V_{left}(A) = 1[1 + 0.9(V(B))]$

$V_{right}(A) = 1[0 + 0.9(V(C))]$

$V(B) = 1[0 + 0.9(V(A))]$

$V(C) = 1[2 + 0.9(V(A))]$

Date M T W T F S

$V_{left}(A) = [1 + 0.9\ (0.9\ V(A))]$

$V_L(A) = 1 + 0.81\ (V(A))$

~~$V_L(A) = 1$~~

$(1 - 0.81)\ V(A) = 1$

$V(A) = \frac{1}{1 - 0.81}$

$V(A) = G_t + R_{t+1}$

$V(A) = \sum_{k=t+1}^{\infty} \gamma^{k-t-1} R_{k+1}$

$V_{left}(A) = 1[1 + 0.9(0) + (0.9)^2(1) + (0.9)^3(0) + \dots]$

Geometric Sequence: $\sum_{n=0}^{\infty} r^n = \frac{1}{1-r}$

$= 1\left[\sum_{n=0}^{\infty} (1)(0.9)^{2n}\right]$

$= \sum_{n=0}^{\infty} (1)\ (0.9^2)^n$

$V(A) = \frac{1}{1 - (0.9^2)}$

$V(A)_{right} = [0 + (0.9)(2) + (0.9)^2(0) + (0.9)^3(2) + \dots]$

$= \sum_{n=0}^{\infty} (1)2(0.9)^{2n+1}$

$= 2 \sum_{n=0}^{\infty} (0.9)(0.9)^{2n}$

$= 1.8 \sum_{n=0}^{\infty} (0.9^2)^n$

$= 1.8 \left(\frac{1}{1 - 0.9^2}\right)$

Date M T W T F S

Castelli

Lecture # 14 Video 12

Dynamic Programing

MDP

$V_\pi(s)$
$q_\pi(s,a)$ } evaluating

Goal → find the optimal policy

Evaluation / prediction Control

$V_*(s) = \max_a \sum_{s'} \sum_r p(s', r | s, a)(r + V_*(s'))$

Dynamic Programming

sub problem divide
store the solution and reuse it

A	B
C	D

DP
→ Give intial value to each value of state
V(A) = V(B), V(C)
→ Initially all zero
→ repeat ~~iterate~~
→ iteratively improve
→ repeat until the value converge.
no longer change.

$V(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r | s, a)[r + V(s')]$

Date _____ M T W T F S

Castelli

A	B
C	D

$V(A) = \frac{1}{2}(0 + 0.9\,V(A)^{0}) +$

$\frac{1}{4}(0 + 0.9\,(V(C))^{0}) + \frac{1}{4}(0 + 0.9\,(V(B))^{0})$

Solve it

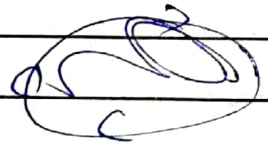
$V(A) = 5/4$

Find all

$V(B), V(C), V(D)$

repeat again putting these new value in equation.

Book Example.



0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

terminal state values are 0

$V(1) = \frac{1}{4}(-1 + 1\,(0)) +$ (left, right, 1)

$\frac{1}{4}(-1 + 0) +$ (V(1))

....

$= -\frac{1}{4} - \frac{1}{4} - \frac{1}{4} - \frac{1}{4}$

$V(1) = -1$

Date M T W T F S

Castelli

By Lecture # 15 (video 13)

Dynamic Programming :

$k = 0, \dots, N$

$$V_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r)[r + \gamma V_\pi(s')]$$

$$V_k(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma V_{k-1}(s')]$$

$$abs(V_k(s) - V_{k-1}(s)) < \text{threshold}$$

Iterative policy evaluation → Dynamic Programming for finding $V(s)$

$V_{\pi_1}(s) \geq V_{\pi_2}(s)$, for every $s \in S$

$\pi_1 \geq \pi_2$

Initialize $V(s)$, for all $s \in S^+$

$V(\text{terminal}) = 0$ terminal state = 0,

Loop:

$\Delta \leftarrow 0$ (kitna difference arraha)

Loop for each $s \in S$:

$v \leftarrow V(s)$

$V(s) \leftarrow \sum_a \pi(a|s) \sum_{s', r} p(s', r|s, a)[r + \gamma V(s')]$ ← Bellman's equation

$\Delta \leftarrow \max(\Delta, |v - V(s)|)$

until $\Delta < \theta$

suppose 25 states

(25 states me se ek ka previous iteration ki 25 states se difference, in 25 me se jo max value hogi wo Δ store hogi.)

(We choose max across all the states)

loop chlta rahega jb-k $\Delta < \theta$ in hojka

Date _____ M T W T F S

We use

$$V_k(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)\left[r + \gamma V_{k-1}(s')\right]$$

if we use previous iteration $V(s)$ value to calculate new values in next iteration then we need 2 data structures one for storing old values and one for new,

$V_{old}(s)$ $V_{new}(s)$

but we can also use the values if we find them early during our iteration.

The only one data structure needed

Control:

How to find a better policy

Control — Policy iteration.
Value iteration.

for deterministic policy

$$V_\pi(s) = \sum_{s'} \sum_r p(s', r|s, a)\left[r + \gamma V_\pi(s')\right]$$

$$q_\pi(s, a) = \sum_{s'} \sum_r p(s', r|s, a)\left[r + \gamma V_\pi(s')\right]$$

Are they same? No

$V(s)$ gives return when all actions are taken under the same policy

In $q(s, a)$ you first take action and then follow certain policy

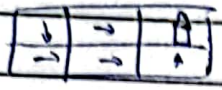
Date M T W T F S Castelli

but

$q_\pi(s, a)$ → first takes an action then follow a policy

↓

In q we fix an action



∴ if

a is not according to this $\pi(s)$

$q_\pi(s, \pi'(s)) \geq V_\pi(s)$

↓

this action better

~~If~~ w

Policy Iteration

1. Initialization: (random policy)

$V(s) \in \mathbb{R}$ and $\pi(s) \in A(s)$ arbitrarily for all $s \in S$

2. Policy Evaluation. (value of state func)

Loop:

$\Delta \leftarrow 0$

Loop for each $s \in S$:

$v \leftarrow V(s)$

$V(s) \leftarrow \sum_{s',r} p(s', r | s, \pi(s)) [r + \gamma V(s')]$

$\Delta \leftarrow \max(\Delta, |v - V(s)|)$

until $\Delta < \theta$

Once value converges we improve that

3. Policy Improvement.

policy-stable ← true

For each $s \in S$: choose action which maximize the value

old-action ← $\pi(s)$

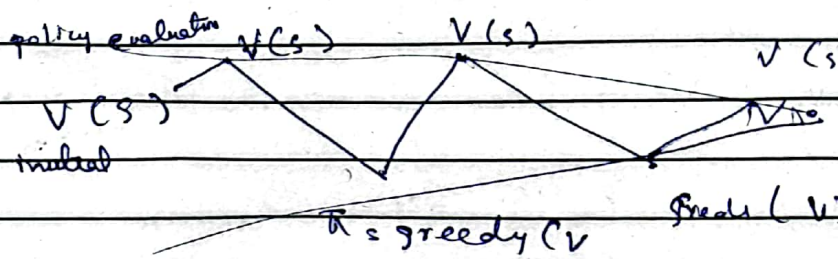
$\pi(s) \leftarrow \arg\max_a \sum_{s',r} p(s', r | s, a) [r + \gamma V(s')]$

If old-action $\neq \pi(s)$, then policy-stable ← false

If policy-stable, then stop and return $V \approx V_*$ and $\pi \approx \pi_*$

else go to 2.

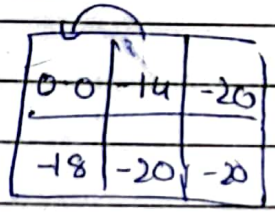
Date _____ M T W T F S

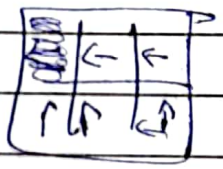


if you know reward and probability then your agent can find optimal pol. policy.
$P(s', r | s, a)$

These are Model-based methods ~~in rl~~

(max action)





Date

M T W T F S

Castelli

lecture 16 (video 14)

Iterative Policy Evaluation

(Prediction)

Determining the value of state using DP

Policy iteration

under a certain policy find V(s) on threashold satisfy

Now policy improvement

Now determine $\pi(s)$ which max $V(s)$ (making the policy)

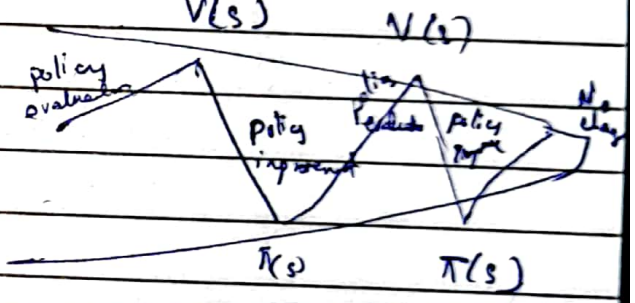
do until policy is stable

Control: finding the optimal policy

Policy Iteration

Here trivially finding the policy

Value Iteration



Bellmans equation

$$V_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s',r|s,a)[r+\gamma V(s')]$$

Bellmans optimality

$$V_\pi(s) = \max_a \sum_{s'} \sum_r p(s',r|s,a)[r+\gamma V(s')]$$

* policy converges faster than value convergence

time wasting in policy iteration

so value iteration is a solution.

Never bellmans optimality equation in value iteration

Value iteration

Initialize $V(s)$ for all $s \in S^+$

Loop:

$\Delta \leftarrow 0$

loop for each $s \in S$:

$v \leftarrow V(s)$

$V(s) \leftarrow \max_a \sum_{s',r} p(s',r|s,a)[r+\gamma V(s')]$ (already optimal value we set after the loop finished)

$\Delta \leftarrow \max(\Delta, |v - V(s)|)$

until $\Delta < \theta$

Output a deterministic policy $\pi \approx \pi_*$

$\pi(s) = \text{argmax}_a \sum_{s',r} p(s',r|s,a)[r+\gamma V(s')]$

Date _____ M T W T F S

In MDP → we are doing planning (information present)

Quick Recap:-

MAB — No true values — we use estimate

MDP — probabilities given — we find states

What if no Probabilities given

Sample-based Methods

We let the agent interact with environment

Monte Carlo Methods:

↳ randomness

is any experiment that have randomness and we tried to find some values by repeating the experiment a large number of times

Coin Example

A fair coin

P(Head) = 0.5 P(Tail) = 0.5

if we dont know then we find exp probabilities on the basis of experiment

we find value of state on the basis of episodes

run episodes multiple times

Episode start se terminal state

$V(s) = E(G_{t+1} \mid S_t = s)$

↓ probability x value

We take Average ↔ Expected value

Actual value experiment

∴ Average ≈ Expected value (if # repeated a large number of times)

Date M T W T F S

Castelli

Lecture 17 (video 15')

Recap:

control — policy iteration
- value iteration

Monte Carlo method (performing a large number of experiment for determining a value in a process that has randomness)

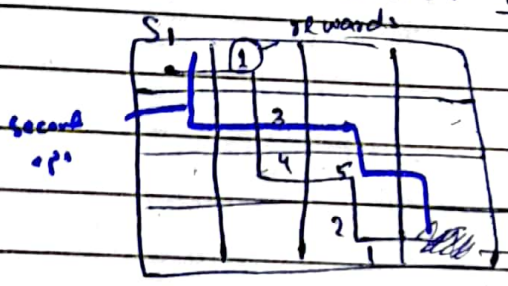
$V(s) = \mathbb{E}(G_t | S_t = s)$

$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \dots$

$G_t = R_{t+1} + \gamma G_{t+1}$

→ if probabilities not given we use Average

Experiment in RL: Episodes $t=0, \dots t=N$, random policy



$E1 \rightarrow S_0, A_0, R_1, S_1, A_1, R_2, S_2 \dots$ plus

return of a state: current state reward + all rewards in future state till termination

$S_0 = 1+3+4+5+2+1 = 16$

We start from terminal state to avoid repetition

Episode 1

$G(15) = 1$

$G(14) = 2 + G(15) = 3$

$G(10) = 5 + G(14) = 8$

$G(9) = 4 + G(10) = 12$

$G(6) = 3 + G(9) = 15$

$G(1) = 1 + G(6) = 16$

Episode 2: $G(15) = 3$, $G(1) =$ ...

Episode 3: $G(15) = 5$

$G(15) = \frac{1+5}{2}$ sum of state's value in all episodes / No of episodes

Two varieties of Monte Carlo

if within a episode a state is visited multiple times

- One approach to add the values everytime it visited
- one approach just use the first visited value

Date _____ M T W T F S

Castelli

First-visit MC

(only consider the state value when it is visited at first in an episode.)

Input a policy π to be evaluated

Initialize:

$V(s) \in \mathbb{R}$ for all $s \in S$

Returns(s) $\leftarrow$ empty list for all $s \in S$

Loop forever (means a large number of times) (for each episode):

Generate an episode following π: $S_0, A_0, R_1, S_1, A_1, \ldots S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

loop for each step of episode $t = T-1, T-2, \ldots 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless S_t appears in $S_0, S_1, \ldots, S_{t-1}$:

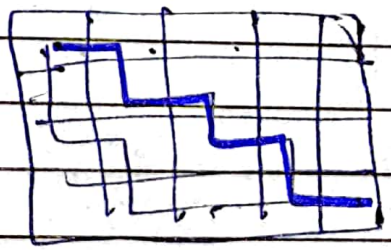
Append G to Returns(S_t)

$V(S_t) \leftarrow$ average (Returns(S_t)) → running average

if probability is not given we find values of state action values then find policy.

$v_\pi(s) = \mathbb{E}[G_t \mid S_t = s]$

$q_\pi(s,a) = \mathbb{E}[G_t \mid S_t = s, A_t = a]$



different episodes have ki reasons

→ policy stochastic

→ policy deterministic

$p(s', r)$ different, or rewards different

On average kitne return mile

Episode 1: $Q(1, \text{down}) = 25$

$Q(6, \text{right}) = 84$

Episode 2

$Q(1, \text{right}) = 4$

Date

M T W T F S

Castelli

if you didnot take an action X in any episode but what if that was a good action,

Monte Carlo ES (Exploring Starts) (Control)

Initialize:

$\pi(s) \in A(s)$ for all $s \in S$

$Q(s,a) \in R$ for all $s \in S, a \in A(s)$

Returns $(s,a) \leftarrow$ empty list, for all $s \in S, a \in A(s)$

Loop forever (for each episode):

Choose $S_0 \in S$ $A_0 \in A(S_0)$ randomly such that all pair have probability > 0

Generate an episode from S_0, A_0 following π

$G \leftarrow 0$

loop for each step of episod, $t = T-1, T-2, \dots 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless the pair S_t, A_t appears in $S_0, A_0, S_1, \dots S_{t-1}$:

Append G to Returns (S_t, A_t)

$Q(S_t, A_t) \leftarrow$ average (Returns (S_t, A_t))

$\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$

(we use Monte-Carlo for control)

(Exploring Start)

has episode ko ek naye state or action me start kiya

(ek naye)

has state or hr action ki ek return value aaye)

(Need to Search)

Date _____ M T W T F S

Castelli

Lecture 18 (video 6)

Monte Carlo Methods (Sample-base Method)

↳ policy evaluation
↳ control

Policy Evaluation — π (fix a policy)

$V_\pi(s)$

Finite num steps

Episode 1 : $S_0, A_0, R_1, S_1, A_1 \cdots S_T, A_T, R_T$

Episodes → (1) 2 3 4 5 ... N ...

States

S_1 Return(S_1) Return(S_1) — Ave(Return) = $V_\pi(s)$

S_2 Return(S_2)

S_3 Return(S_3) Return

S_4 Return(S_4)

Two Types of Monte Carlo → first-visit MC
→ Every-visit MC

Running average → simple avg $Q(a) = \frac{1}{n}\sum R_i$

↓

Incremental average

$$Q_{n+1} = Q_n + \frac{1}{n}(R_n - Q_n)$$

$$V_\pi(s) = V(s) + \frac{1}{n}(\text{Return} - V(s))$$

$$V(s) = V(s) + \frac{1}{n}(G_1 - V(s))$$

In previous model MDP we can find optimal value $\pi = \arg\max_a \left(\sum_{s'} p(s', r|s, a)[r + V(s')]\right)$

We can find control there because $p(s', r|s, a)$ but in Monte Carlo we can't as there is no $p(s', r|s, a)$ so for finding optimal value we take $Q(s, a)$

Date

M T W T F S

Castelli

Episodes 1, 2 3 4

S_0, a_0 Return ()

S_0, a_1

S_0, a_2

S_1, a_3 Return ()

S_1, a_1

S_2, a_2 Return ()

So there was the problem that some state, action pair will not be visited

Solution:

In epsiode start we randomly choose those pair whose s, a value probability is positive

at least choose every (s, a) value pair one time

but one time choose is still not enough

exploring start ↓

Monte Carlo (ES) so we want to do exploration here

↓

greedy approach → here it is not good

In MDP greedy action is not bad because there we have information about environment, probabilities were given

↓

Epsilon - greedy

ε - greedy

With ε, small value > 0 → choose a random action

with 1 - ε choose greedy action

Date M T W T F S

Castelli

A_1, A_2, A_3, A_4 $P(A_3)$ = max best

ε → random actn

ε = $P(A_1)$ or $P(A_2)$ or $P(A_3)$ or $P(A_4)$

$\varepsilon = P(A_1) + P(A_2) + P(A_3) + P(A_4)$ $P(A_1) = P(A_2) = P(A_3) = P(A_4)$

$\varepsilon = 4P(A_i)$

$P(A_i) = \varepsilon/4$ (choosing a random action probabilities

$|A|$ no of actn
↓ size of actn space

$P(A) = \frac{\varepsilon}{|A|}$, $\varepsilon/4$ if 4 actns

$P(A_1) = \frac{\varepsilon}{|A|} = P(A_2)$ probabily of choosing a random actn

Probabily of choosing a greedy actn

(As greedy actn can be choosn while exploratn

$P(A_3) = 1 - \varepsilon + \frac{\varepsilon}{|A|}$

$P(A_1) = P(A_2) = P(A_4) = \frac{\varepsilon}{|A|}$

In exploring start instead of greedy we ~~choose~~ use Epsilon greedy

Date

M T W T F S

Castelli

On-policy first MC control (for ε-soft policies)

small $\varepsilon > 0$

Initialize:

$\pi \leftarrow$ an ε-soft policy

$Q(s,a) \in \mathbb{R}$

Returns (s,a)

In start we give an ε-soft policy. In the end it becomes epsilon-greedy

Repeat forever (for each episode):

Generate an episode following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots S_{t-1}, A_{t-1}$:

Append G to Returns (S_t, A_t)

$Q(S_t, A_t) \leftarrow$ average (Returns (S_t, A_t))

$A^* \leftarrow \arg\max_a Q(S_t, a)$

For all $a \in A(S_t)$:

$$\pi(a|S_t) \leftarrow \begin{cases} 1-\varepsilon + \varepsilon/|A(S_t)| & \text{if } a = A^* \\ \varepsilon/|A(S_t)| & \text{if } a \neq A^* \end{cases}$$

Greedy

Non greedy

Epsilon-soft:

superset of epsilon-greedy

Every Epsilon greedy is Epsilon-soft but not every epsilon-soft is epsilon greedy.

Date____ M T W T F S

Lecture 19 (video 17)

Monte Carlo - Sample based

$Q(s,a)$

$\pi = \arg\max_a Q(s,a)$

① Exploring starts - episode start → state, action ek dfa explore kia ho

② Epsilon-greedy

$\varepsilon / |A(s)|$ ε -> non-greedy action

$1-\varepsilon+\varepsilon/|A(s)|$ $1-\varepsilon$ -> greedy action,

$Q_t(s,a) = Q_{t-1}(s,a) + \text{stepsize}(G_t - Q(s,a))$

$= Q_{t-1}(s,a) + \text{stepsize}((R_{t+1} + \gamma G_{t+1}) - Q(s,a))$

Epsilon-soft

$P(A) \geq \frac{\varepsilon}{|A|}$

Har action ki choose krne ki possibility at least $\frac{\varepsilon}{|A|}$ ho $\varepsilon = 0.1$, $|A| = 4$

$P(A_1) \geq 0.1/4$

$P(A_2) \geq 0.1/4$

$P(A_3) \geq 0.1/4$

$P(A_4) \geq 0.1/4$

ya ye $\frac{\varepsilon}{|A|}$ ya use zyada

Epsilon-greedy (soft is epsilon soft)

No-greedy

$P(A) = \frac{\varepsilon}{|A|}$

Greedy

$P(A) = 1-\varepsilon+\frac{\varepsilon}{|A|}$

Date

M T W T F S

Castelli

Equiprobable random policy is also epsilon soft

$P(A) = \frac{1}{|A|} \rightsquigarrow \varepsilon = 1$

has action k chosen have k equal chances hon

Epsilon - greedy proof

In previous also

greedy $\rightarrow \pi = \arg\max_a Q(s,a)$

Being greedy is being optimal as it may be wrong so for exploration we use epsilon-greedy

but greedy action maybe sometime a good action

Epsilon soft is stopping you from choosing greedy action even in the case when greedy action is a correct

So epsilon-soft is also not optimal

both greedy and epsilon-soft are not optimal but if we compare which is better

epsilon soft > greedy

So we want to improve epsilon soft

Date M T W T F S

if an agent is learning from its behavior then it may not be a good thing if the behavior is not optimal

Two policies — Behavior policy (non optimal) — may focus on the learning of behavior policy.

Target policy (non optimal)

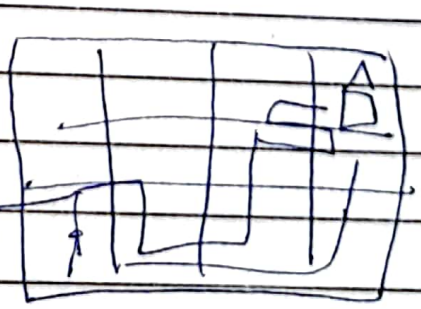
Until now the algorithms we see they both are equal

On policy ⟹ Behavior policy = Target policy

if behavior policy is not optimal then target is not optimal

how to separate both these

off-policy



behave — behavior policy

ek policy

k through k nae

or lean dusri se