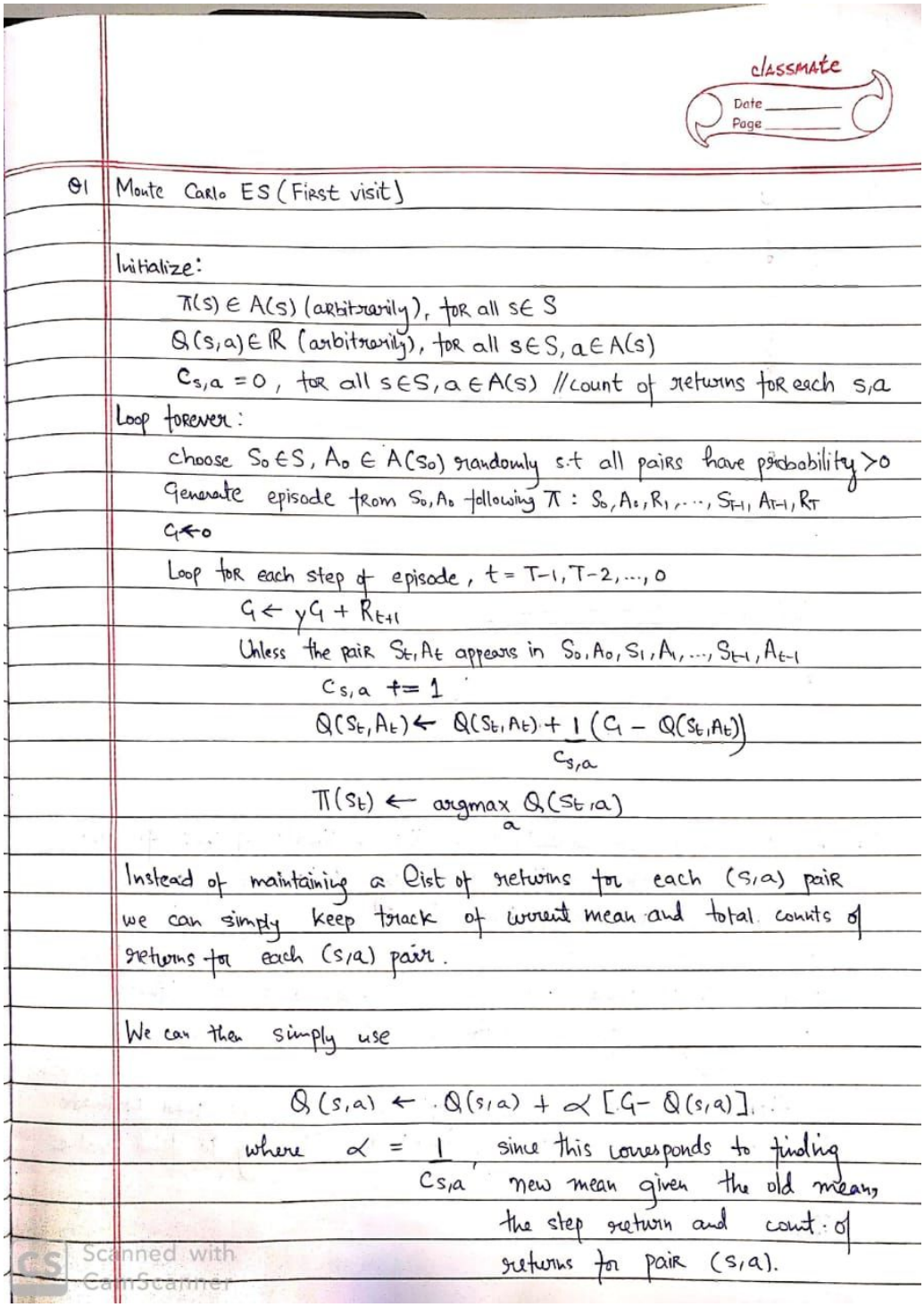
RL Assignment 3
Harsh Pathak
2016041

Q1.



Q1 Monte Carlo ES (First visit)

Initialize:

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

$Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in S, a \in A(s)$

$C_{s,a} = 0$, for all $s \in S, a \in A(s)$ // count of returns for each s,a

Loop forever:

choose $S_0 \in S$, $A_0 \in A(S_0)$ randomly s.t all pairs have probability > 0

Generate episode from S_0, A_0 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 $S_0, A_0, S_1, A_1, \ldots, S_{t-1}, A_{t-1}$

$C_{s,a} \mathrel{+}= 1$

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{C_{s,a}} (G - Q(S_t, A_t))$$

$$\pi(S_t) \leftarrow \underset{a}{\operatorname{argmax}}\, Q(S_t, a)$$

Instead of maintaining a list of returns for each (s,a) pair we can simply keep track of current mean and total counts of returns for each (s,a) pair.

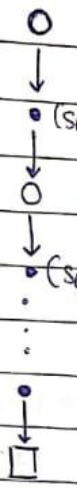
We can then simply use

$$Q(s,a) \leftarrow Q(s,a) + \alpha [G - Q(s,a)]$$

where $\alpha = \frac{1}{C_{s,a}}$, since this corresponds to finding new mean given the old mean, the step return and count of returns for pair (s,a).

Q2

Q2



Q3 ~~$Q(s,a) = \sum_{t \in T(s)} p_{t:T(t)=1}$~~

$$Q(s,a) = \frac{\sum_{t \in T(s)} p_{t+1:T(t)-1} G_{t+1}}{\sum_{t \in T(s)} p_{t+1:T(t)-1}}$$

Q5 Let us assume we already have the estimate of $V(s)$ for the time when we were in the old building. After moving to the new building and coming back home, since highway is a common entry point for both the routes, we can hence use the guess of $V(s)$ where s is the state when we are at highway to quickly estimate $V(s')$ for the new route change, where s' comes before s.

In other words, we are bootstrapping and using the guess estimate that we have for "remaining time" from highway to find value estimate of new states.

6.3 It is possible that the first episode was just A left
This is because A is the only (non terminal) state in this episode, hence only its value will decrease.
H changed by 0.05 and became 0.45 (since $\alpha = 0.1$)

6.4 Since TD and MC both are estimated on the basis of the episodes (hence the returns) observed at each episode, a large value of α will bias the V() estimates towards the latest rewards/returns. A large value of α means that the loss will never converge and keep changing in response to latest reward/returns.
A low value of alpha means that the loss converges but it will take a lot of time to converge. The best thing is to vary alpha on the basis of time steps (high to low α as time increases)

6.5 At large alphas, the value estimates are always biased towards the latest episodes. Hence the loss may go up as the no. of episodes are increased.
Hence the parameters need to carefully selected.
This can also be a function of how the value functions were initialized first.

6.12 SARSA $S \xrightarrow[\text{(greedy)}]{A,R} S' \xrightarrow[\text{(greedy)}]{A'} \{\text{update } Q\}$
begin with (S', A')

Q-learn $S \xrightarrow[\text{(greedy)}]{A,R} S' \{\text{update } Q\}$
begin with (S')

In SARSA, we have already picked the new S' and A' before updating Q {with $\epsilon = 0$, this is pure exploitation}
In Q-learn, we have update Q first and then selected S'. This way, a new A' selected for S' might be different from A' of SARSA. Hence there is a slight difference.

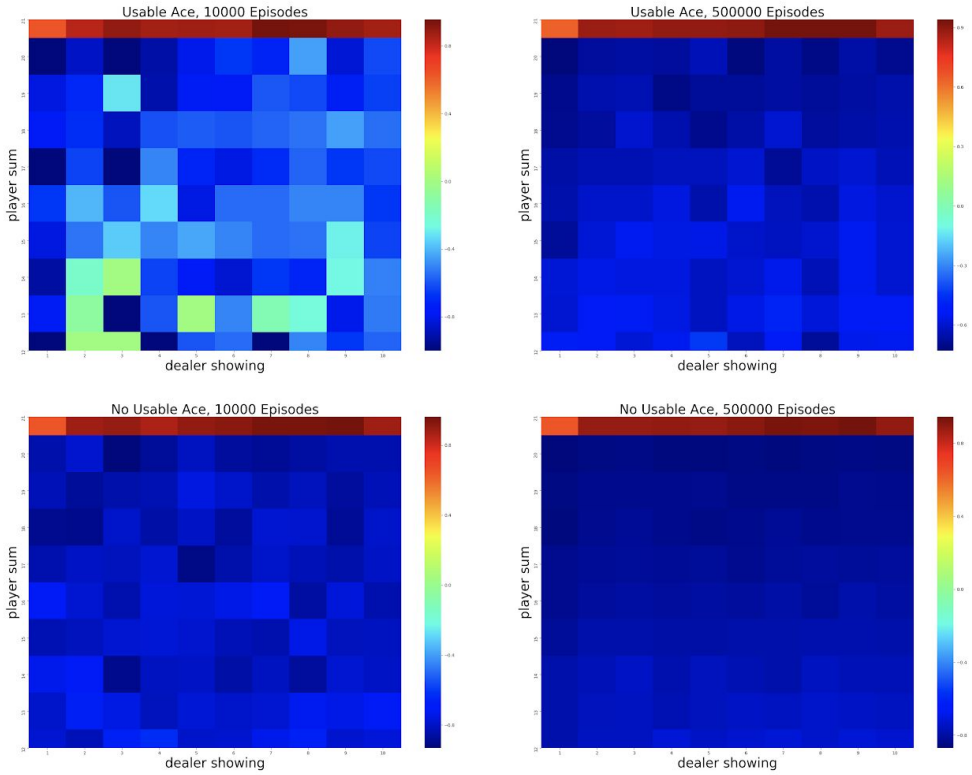

Usable Ace, 10000 Episodes
player sum
dealer showing
Usable Ace, 500000 Episodes
player sum
dealer showing
No Usable Ace, 10000 Episodes
player sum
dealer showing
No Usable Ace, 500000 Episodes
player sum
dealer showing

Fig 5.1

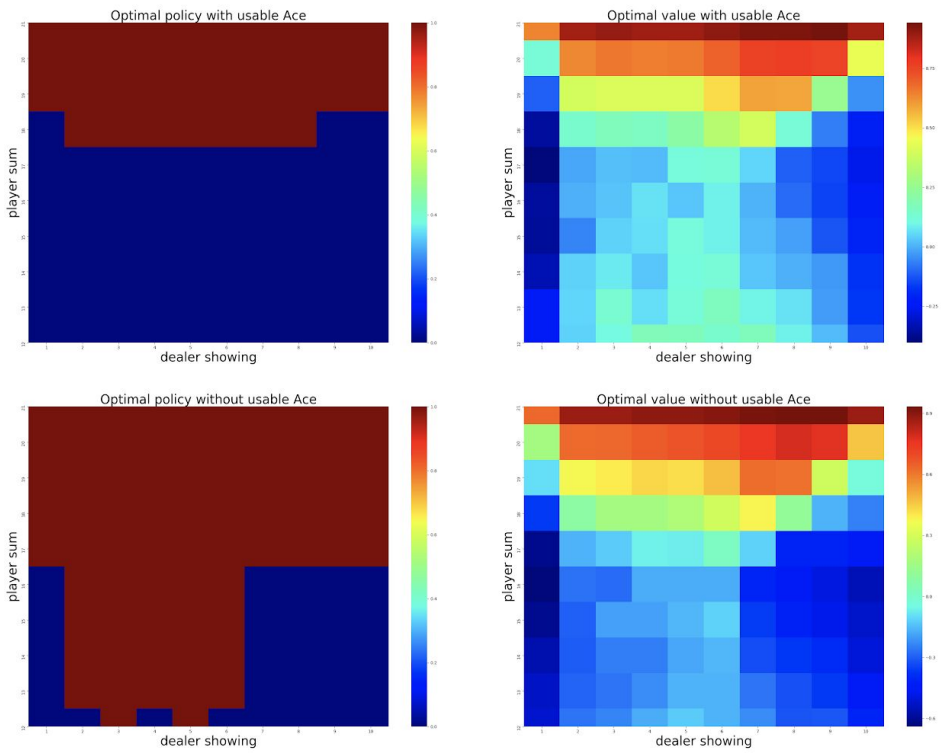

Optimal policy with usable Ace
player sum
dealer showing
Optimal value with usable Ace
player sum
dealer showing
Optimal policy without usable Ace
player sum
dealer showing
Optimal value without usable Ace
player sum
dealer showing

Fig 5.2

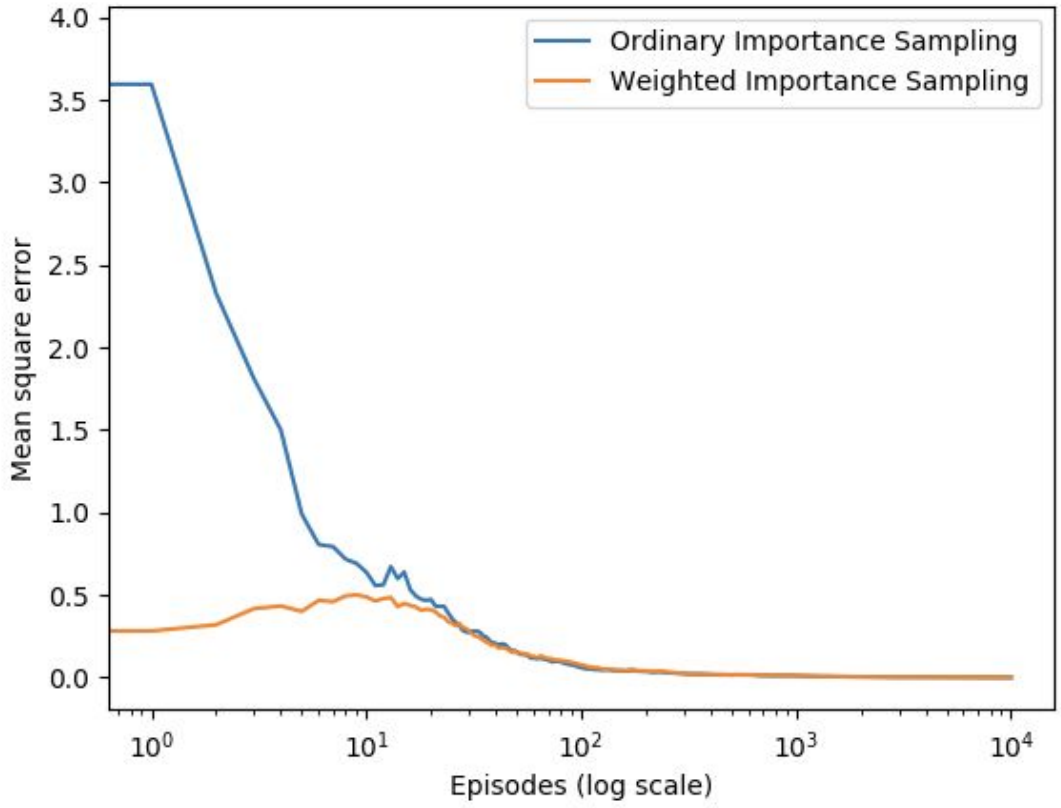
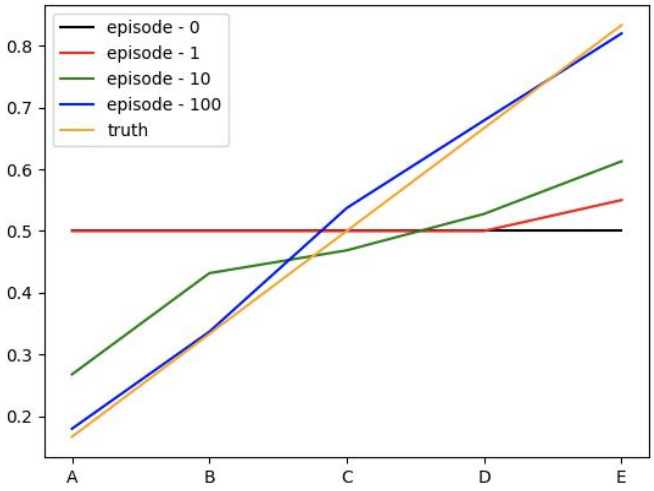

Ordinary Importance Sampling
Weighted Importance Sampling
Mean square error
Episodes (log scale)
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
10^0
10^1
10^2
10^3
10^4

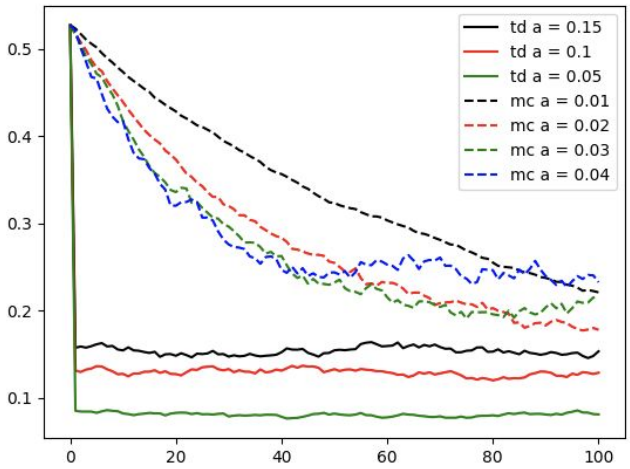
Fig 5_3

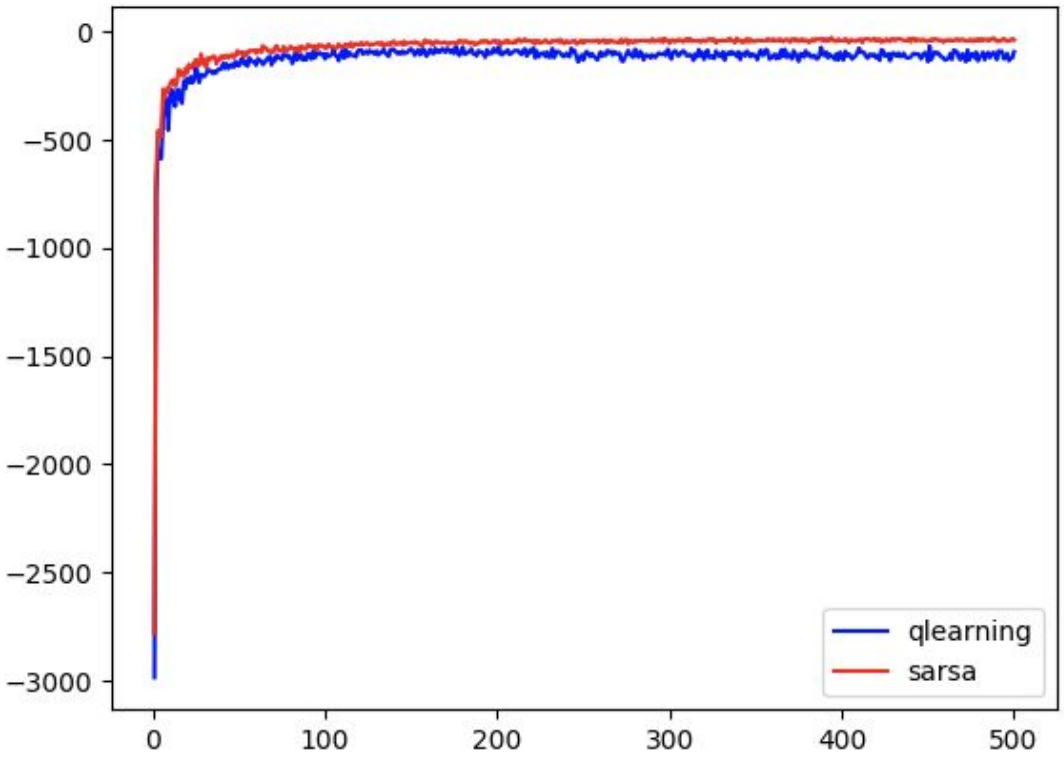
Q6

Q6


episode - 0
episode - 1
episode - 10
episode - 100
truth
0.8
0.7
0.6
0.5
0.4
0.3
0.2
A
B
C
D
E

Q6


td a = 0.15
td a = 0.1
td a = 0.05
mc a = 0.01
mc a = 0.02
mc a = 0.03
mc a = 0.04
0.5
0.4
0.3
0.2
0.1
0
20
40
60
80
100



Q7, sarsa vs Q-learning for the cliff problem