REINFORCEMENT LEARNING

Lecture 4 : Temporal Difference Methods

Ibrahim Sammour

December | 2023

Constant-α Monte Carlo Methods

- We wait until the end of the episode to update our value function

$$V(S_t) \leftarrow V(S_t) + \alpha(\bar{r}_t - V(S_t))$$

NewEstimate = oldEstimate + stepSize(target – oldEstimate)

- Note that the equation above is also applicable for the state-action value function.
- The target parameter is the main difference between monte carlo methods and temporal difference.

Temporal Difference

- Model free reinforcement learning
- Temporal difference = use of differences
  - Used in the learning process

Temporal Difference

Monte Carlo

- Wait until the end of the episode before updating the value function
- Only works for episodic tasks

Temporal Difference

- Learn even before reaching the terminal state
- Works for both episodic and continuing task

Temporal Difference

- We do not wait until the end of the episode to update our value function
- The “simplest form” of temporal difference “TD(0)” is given by:

$$V(S_t) \leftarrow V(S_t) + \alpha(r_t + \gamma V(S_{t+1}) - V(S_t))$$

NewEstimate = oldEstimate + stepSize(target – oldEstimate)

- We update the value function after each step instead of waiting till the end of the episode

- Note that the equation above is also applicable for the state-action value function.

Temporal Difference TD(0)

Algorithm

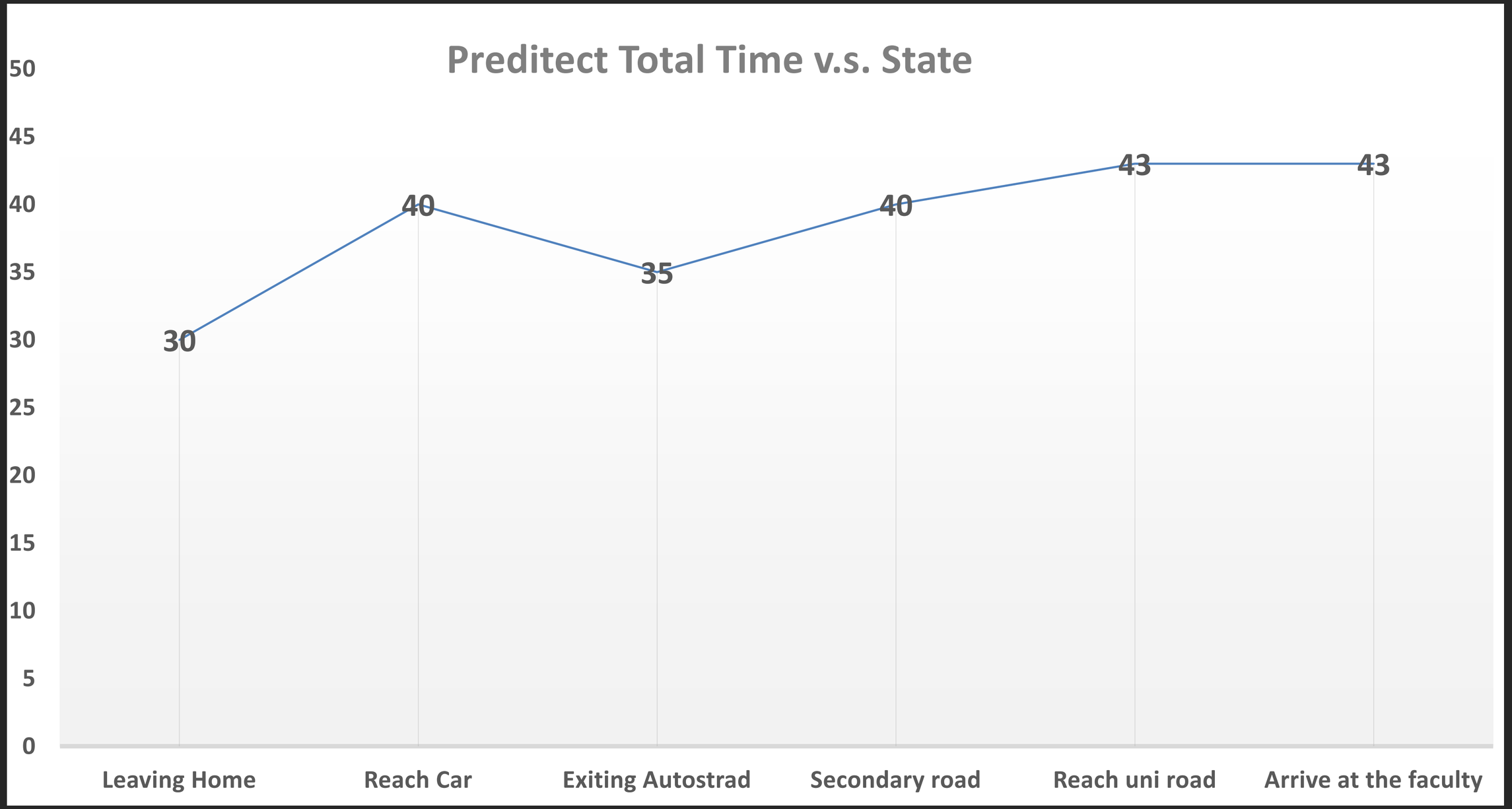
$Input: a\ policy\ \pi$
$Step\ size: \alpha \in (0, 1]$
$Initialize\ V(s)$
$Loop\ for\ each\ episode:$
$\quad s = s_0: first\ state$
$\quad Loop\ for\ each\ state\ in\ the\ episode$
$\quad\quad a \leftarrow action\ given\ by\ \pi\ for\ s$
$\quad\quad take\ action\ a\ then\ observe\ r\ and\ s'$
$\quad\quad V(s) \leftarrow V(s) + \alpha(r + \gamma V(s') - V(s)$
$\quad\quad s \leftarrow s'$
$\quad repeat\ until\ s\ is\ the\ terminal\ state$

Temporal Difference TD(0)

Example

State	Elapsed time Minutes	Predicted Time left	Predicted Total time
Leaving home at 10 am	0	30	30
Reach car, its raining	5	35	40
Exiting autostrad	20	15	35
Secondary road, traffic jam	30	10	40
Reach uni road	40	3	43
Arrive at the Faculty	43	0	43

Temporal Difference TD(0)



TD(λ) - Temporal Difference Learning with Eligibility Traces

- Introduces eligibility traces to extend the influence of states over multiple time steps.

$$V(s) = V(s) + \alpha \delta_t E_t(s)$$

$$\delta_t = r_t + \gamma V(S_{t+1}) - V(S_t)$$

$$E_t(s) = \lambda \gamma E_{t-1}(s) + 1(S_t = s)$$

- δ_t *is the TD error*
- $E_t(s)$ *is the eligbility trace of state* s
- λ *is the trace decay parameter*

Temporal Difference TD(λ)

Algorithm

$Input$: $a\ policy\ \pi$
$\alpha, \lambda \epsilon\ [0, 1]$
$Initialize\ V(s)$, E(s)
$Loop\ for\ each\ episode$:
$Loop\ for\ each\ state\ in\ the\ episode$
$a \leftarrow action\ given\ by\ \pi\ for\ s$
$take\ action\ a\ then\ observe\ r\ and\ s'$
$\delta_t = r_t + \gamma V(S_{t+1}) - V(S_t)$
$E_t(s) = \lambda\gamma E_{t-1}(s) + 1(S_t = s)$ for all states
$V(s) = V(s) + \alpha\delta_t E_t(s)$
$s \leftarrow s'$
$repeat\ until\ s\ is\ the\ terminal\ state$

SARSA

- Short for (state, action, reward, state, action)
- SARSA is an on-policy algorithm, meaning it learns from the policy it is currently following.
- Suitable for scenarios where exploration is crucial, such as in online learning or when the environment is unknown.
- Performs policy updates during the learning process.

$$Q(s,a) = Q(s,a) + \alpha(r + \gamma Q(s',a') - Q(s,a))$$

Q-Learning

- Q-learning is an off-policy algorithm, meaning it learns the value of the optimal policy regardless of the policy being followed.
- Suitable for scenarios where it's essential to learn an optimal policy for later exploitation.

$$Q(s,a) = Q(s,a) + \alpha\left(r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right)$$