Problem: arbitrarily large state spaces

↓

Solution: find a good approximate solution

Using limited computational resources

to learn a better policy

"generalization" function approximation

Approximate V_π from experience generated using a known policy π

parameterized function:

$\hat{v}(s, w) \approx v_\pi(s)$

- linear model
- DNN
- Decision Tree

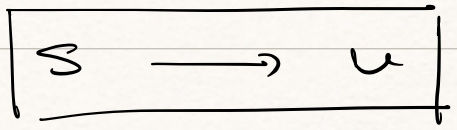
$w \in \mathbb{R}^d$ weight vector.

$d \ll |\mathcal{S}|$

- d: the number of weights
- $|\mathcal{S}|$: the number of states

RL update to an estimated value function that shift its value at particular states toward a

"back-up value" or "update target" for that state



MC : $S_t \longrightarrow G_t$

TD(0) : $S_t \longrightarrow R_{t+1} + \gamma \hat{v}(S_{t+1}, w)$

n-step TD : $S_t \longrightarrow G_{t:t+n}$

DP : $s \longrightarrow \mathbb{E}_\pi[R_{t+1} + \gamma \hat{v}(S_{t+1}, w_t) \mid S_t = s]$

Prediction Objective $\overline{VE}$

Mean Squared Value Error ($\overline{VE}$)

$$\overline{VE}(w) \doteq \sum_{s \in \mathcal{S}} \mu(s) \left[v_\pi(s) - \hat{v}(s, w) \right]^2$$

$\mu(s)$: fraction of time spent in s

(on-policy distribution)

$\mu(s) \geq 0$, $\sum_s \mu(s) = 1$

$$\eta(s) = h(s) + \sum_{\bar{s}} \eta(\bar{s}) \sum_{a} \pi(a|\bar{s})\, p(s|\bar{s}, a)$$

$$\mu(s) = \frac{\eta(s)}{\sum_{s'} \eta(s')}$$

VE : find a global optimum

Stochastic gradient and semi-gradient Method.

Stochastic gradient descent (SGD):

$$w \doteq (w_1, w_2, \dots, w_d)^T$$

$\hat{v}(s, w)$: differentiable function of w for all s

$$S_t \longrightarrow V_\pi(S_t)$$

$$w_{t+1} \doteq w_t - \frac{1}{2}\alpha \nabla \left[V_\pi(S_t) - \hat{v}(S_t, w_t) \right]^2$$

$$= w_t + \alpha \left[V_\pi(S_t) - \hat{v}(S_t, w_t) \right] \nabla \hat{v}(S_t, w_t)$$

$$\nabla f(w) \doteq \left(\frac{\partial f(w)}{\partial w_1}, \frac{\partial f(w)}{\partial w_2}, \cdots, \frac{\partial f(w)}{\partial w_d} \right)^\top$$

target output: $U_t \in \mathbb{R}$

$$w_{t+1} \doteq w_t + \alpha \left[U_t - \hat{v}(S_t, w_t) \right] \nabla \hat{v}(S_t, w_t)$$

U_t : unbiased estimate

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

$$w \leftarrow w + \alpha \left[G_t - \hat{v}(S_t, w) \right] \nabla \hat{v}(S_t, w)$$

Gradient MC:

$$w \leftarrow w + \alpha \left[G_t - \hat{v}(S_t, w) \right] \nabla \hat{v}(S_t, w)$$

Semi-gradient TD(0):

$$w \leftarrow w + \alpha \left[R + \gamma \hat{v}(S', w) - \hat{v}(S, w) \right] \nabla \hat{v}(S, w)$$

Linear Models

$$x(s) \doteq (x_1(s), x_2(s), \ldots, x_d(s))^\top$$

$$\hat{v}(s, w) \doteq w^\top x(s) \doteq \sum_{i=1}^{d} w_i x_i(s)$$

(linear in the weights)

$x(s)$: feature vector

$$\nabla \hat{v}(s, w) = x(s)$$

$$w_{t+1} \doteq w_t + \alpha \left[U_t - \hat{v}(S_t, w_t) \right] x(S_t)$$

$$\doteq w_t + \alpha \left(R_{t+1} + \gamma w_t^\top x_{t+1} - w_t^\top x_t \right) x_t$$

$$= w_t + \alpha \left(R_{t+1} x_t - x_t (x_t - \gamma x_{t+1})^\top w_t \right)$$

$$\mathbb{E}[w_{t+1} \mid w_t] = w_t + \alpha (b - A w_t)$$

$$b \doteq \mathbb{E}[R_{t+1} x_t] \in \mathbb{R}^d$$

$$A \doteq \mathbb{E}\left[x_t (x_t - \gamma x_{t+1})^\top \right] \in \mathbb{R}^d \times \mathbb{R}^d$$

$$b - A w_{TD} = 0$$

$b = A w_{TD}$

$w_{TD} \doteq A^{-1} b$ (TD fixed point)

ANN: Artificial Neural Networks

Semi-linear units

Weighted sum ⟶ activation function (nonlinear function)

S-shaped (sigmoid)

logistic function $f(x) = \frac{1}{1+e^{-x}}$

rectifier nonlinearity $f(x) = \max(0, x)$

"Modify the objective function to discourage complexity of the approximation"

"The dropout method efficiently approximates this combination by multiplying each outgoing weight of a unit by the probability that that unit

was retained during training.

Method:
- gradient descent
- Semi-gradient descent

Linear Methods

Nonlinear Function Approximation: ANN

Least-Squares TD

Memory-based Function Approximation

Kernel-based Function Approximation