IV. Online ADP Algorithm

① TD Error
② Value Fcn Approx. $\Big\}$ → Policy Itr. ↪ Online ADP

At each time step k, collect data $(x_k, x_{k+1}, c(x_k, \pi(x_k)))$. Consider value fcn approx:

$$V_\pi(x) = W^T\phi(x).$$

Then the TD err is

$$e_k = c(x_k, \pi(x_k)) + \gamma \cdot W^T\phi(x_{k+1}) - W^T\phi(x_k)$$

corresponds to linear regression model:

$$\Rightarrow c(x_k, \pi(x_k)) = [\phi(x_k) - \gamma \cdot \phi(x_{k+1})]^T \underline{w}$$

re-arranged Bellman eqn: $V_k = c + \gamma \cdot V_{k+1}$

We can now write our first <u>online</u> RL algo that performs policy eval via supervised learning.

<u>Online Policy Iter.</u>

0) Initialization: Select an admissible control policy π^0. Set $m = 0$.

1) **Policy Eval**: Run control policy π^m on environ/system for one episode. Collect L measured data tuples $(x_k, x_{k+1}, c(x_k, \pi^m(x_k)))$. Find least squares solution w.r.t. W_m for regression model (a.k.a Bellman Eqn)

$$L > n_w \qquad \underbrace{\begin{bmatrix} \vdots \\ c(x_k, \pi^m(x_k)) \\ \vdots \end{bmatrix}^{=G}}_{L\times 1} = \underbrace{\begin{bmatrix} \vdots \\ \underline{\left[\phi(x_k) - \gamma \cdot \phi(x_{k+1})\right]^T} \\ \vdots \end{bmatrix}^{=\Phi}}_{L\times n_w} \underbrace{W}_{n_w \times 1}$$

written compactly as $G = \Phi \underline{W}$

For example, you can perform ordinary lsq.

$$W_m \leftarrow W^* = \left[\Phi^T \Phi\right]^{-1} \Phi G$$

2) Policy Improve: Find an improved policy via

$$\pi^{m+1} = \arg\min_{\pi(\cdot)} \left\{ c(x_k, \pi(x_k)) + \gamma \cdot W_m^T \phi(x_{k+1}) \right\}$$

where $x_{k+1} = f(x_k, \pi(x_k)) \quad \forall x_k \in X$

Set $m \leftarrow m+1$. Go to Step 1.

Rem: Besides OLS, you can also use recursive lsq, gradient method, ridge regression, LASSO regression.

Rem: In online ADP, the regressor

$$\left[\phi(x_t) - \gamma \cdot \phi(x_{t+1})\right]$$ must be

"persistently excited" for a soln to exist for lsq. This is a sufficient condition for $\Phi^T \Phi$ to be invertible.

Rem: Observe that Step 1 Policy Eval is model-free. We only require data $(x_k, x_{k+1}, c(\cdot,\cdot))$

However, Step 2 Policy improvement is NOT model-free. We are required to solve:

$$\frac{\partial c}{\partial u}(x_k, \pi(x_k)) + \gamma \cdot W^T \frac{\partial \phi}{\partial x}(x_{k+1}) \cdot \frac{\partial f}{\partial u}(x_k, \pi(x_k)) = 0$$

which requires knowledge of $c(\cdot,\cdot)$, $f(\cdot,\cdot)$

Rem Step 2 still requires minimization for all $x_k \in X$. So we have only partially avoided the curse of dimensionality

This motivates fcn approx. for the control policy fcn $\pi(\cdot)$.

Called "actor neural net" by Werbos & Bertsekas.