Incremental Least-Square Temporal Difference Learning (iLSTD)

Alborz Geramifard, Michael Bowling, Richard S. Sutton
University of Alberta

Problem

- Markov Decision Process (MDP)

$$(\mathcal{S}, \mathcal{A}, \mathcal{P}^a_{ss'}, \mathcal{R}^a_{ss'}, \gamma)$$

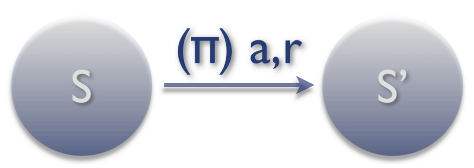
- We focus on online policy evaluation

$$V^\pi(s) = \sum_a \pi(s,a) \sum_{s'} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma V^\pi(s')\right]$$

Related Work

- Temporal Difference Learning: TD(0)

TD



$$\delta_t(V) = r_{t+1} + \gamma V(s_{t+1}) - V(s_t).$$

- Using Linear Function Approximation

$$V(s_t) = \boldsymbol{\phi}(s_t)^T \boldsymbol{\theta}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha_t \phi(s_t) \delta_t(V)$$

we assume k features are "on" on each time step

- Least-Square TD (LSTD)

LSTD

$$\boldsymbol{\mu}_t(\boldsymbol{\theta}) = \sum_{i=1}^{t} \phi_i \delta_i(V_{\boldsymbol{\theta}})$$

$$= \underbrace{\sum_{i=1}^{t} \phi_i r_{i+1}}_{\mathbf{b}_t} - \underbrace{\sum_{i=1}^{t} \phi_i(\phi_i - \gamma\phi_{i+1})^T}_{\mathbf{A}_t} \boldsymbol{\theta}$$

$$\boldsymbol{\theta}_{t+1} = \mathbf{A}_t^{-1}\mathbf{b}_t.$$

New Approach

TD
- Cheap - O(k) per time-step
- Relatively Data Inefficient

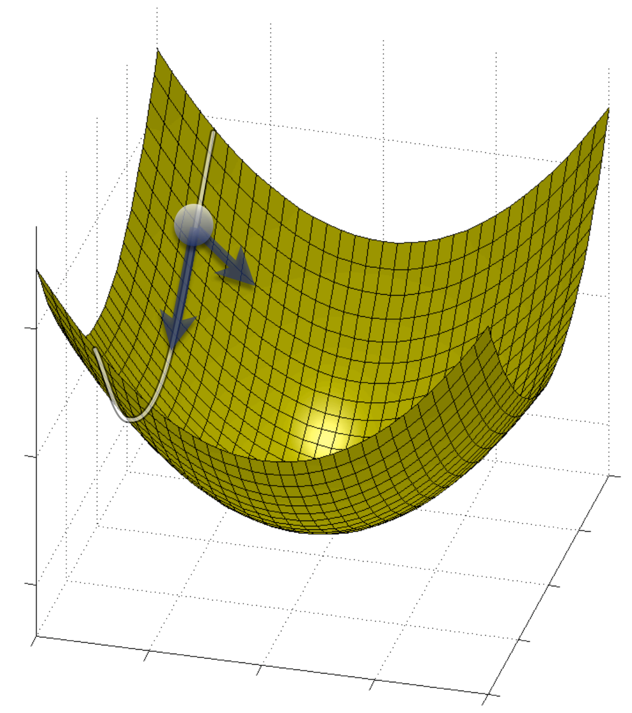
LSTD
- Expensive - O(n^2) per time-step
- Data Efficient

iLSTD
- Cheap - O(mn+k^2) per time-step
- Data Efficient

```
s ← s_0, A ← 0, μ ← 0, t ← 0
Initialize θ arbitrarily
repeat
   Take action according to π and observe r, s'
   t ← t + 1
   Δb ← φ(s)r
   ΔA ← φ(s)(φ(s) − γφ(s'))^T
   A ← A + ΔA
   μ ← μ + Δb − (ΔA)θ
   for i from 1 to m do
      j ← argmax(|μ_j|)
      θ_j ← θ_j + αμ_j
      μ ← μ − αμ_j A e_i
   end for
end repeat
```

iLSTD

- Would like to Update $\boldsymbol{\theta}$ by the sum of the TD updates ($\boldsymbol{\mu}$)
- Pick the dimension with the largest TD update
- Descend in that dimension



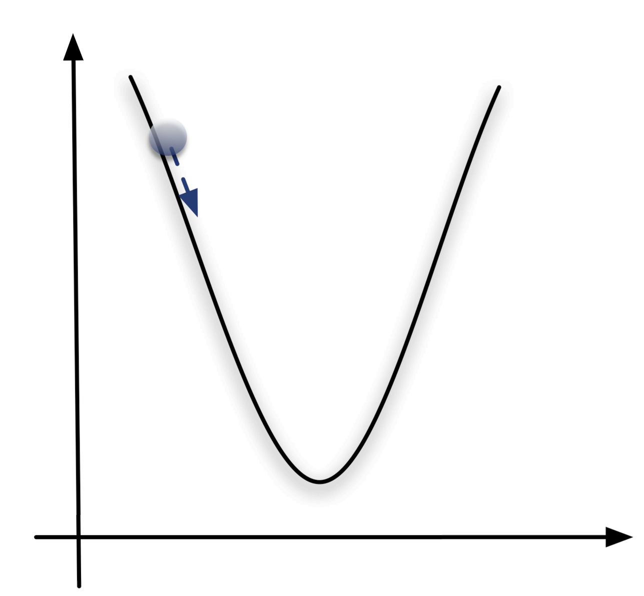
- Updates:

$$\boldsymbol{\mu}_t(\boldsymbol{\theta}_{t+1}) = \boldsymbol{\mu}_t(\boldsymbol{\theta}_t) - \mathbf{A}_t(\Delta\boldsymbol{\theta}_t)$$

$$\boldsymbol{\mu}_t(\boldsymbol{\theta}_t) = \boldsymbol{\mu}_{t-1}(\boldsymbol{\theta}_t) + \Delta\mathbf{b}_t - (\Delta\mathbf{A}_t)\boldsymbol{\theta}_t$$

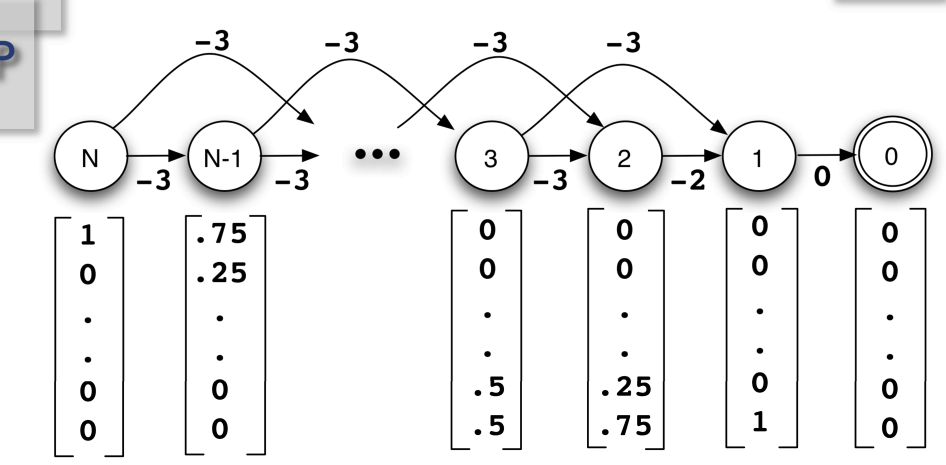
$$O(mn + k^2)$$

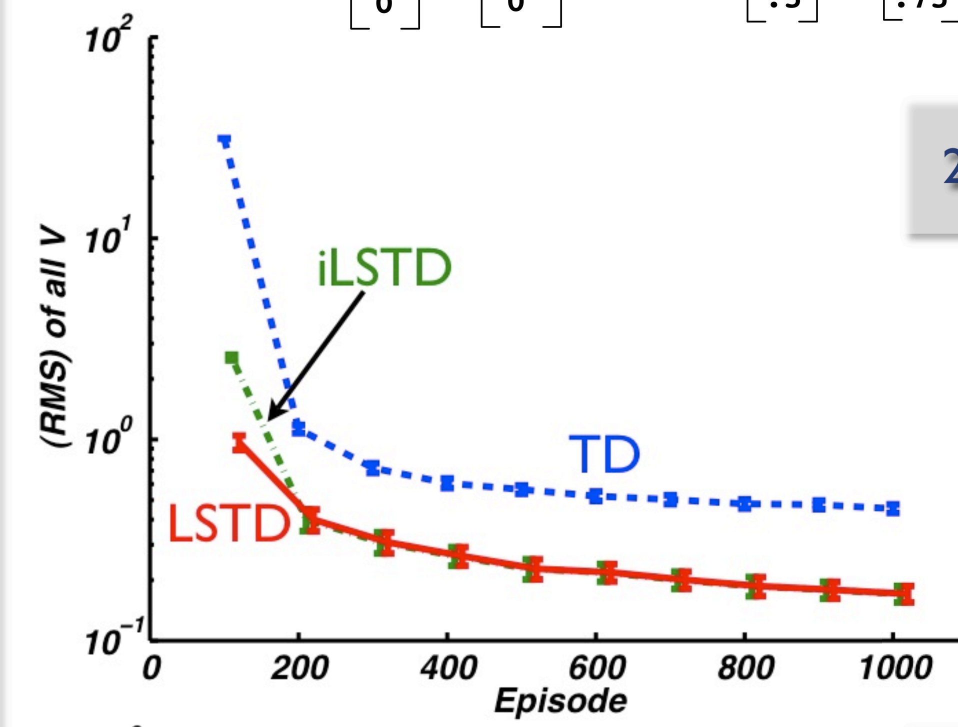
- k: Maximum number of "on" features
- n: Number of features
- m: Number of descents per time-step

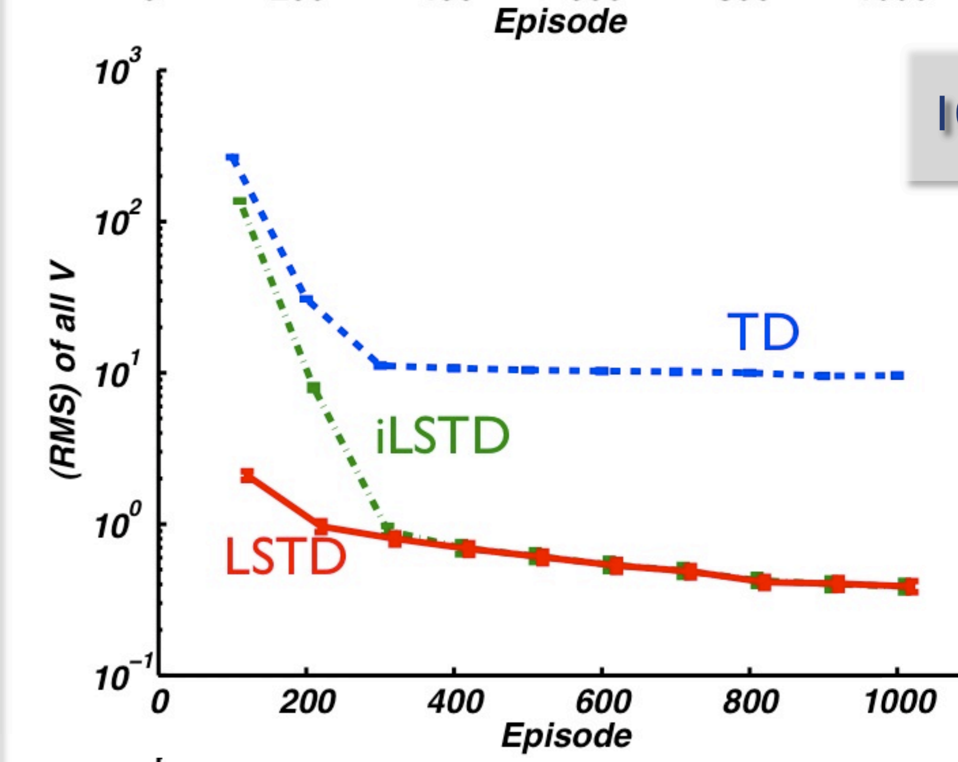


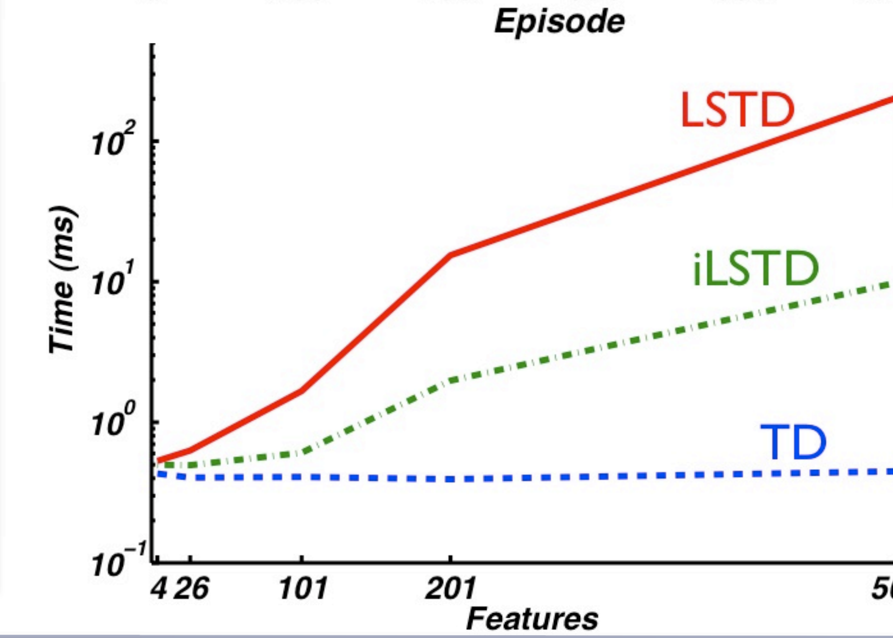
Results

Boyan's MDP









AAAI - 2006 Boston

NSERC CRSNG · CORE · ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING