Function Approximation

VFA: Temporal Difference

Marius Lindauer

Leibniz Universität Hannover

Automated Machine Learning Hannover

Recall: Temporal Difference Learning w/ Lookup Table

- Uses bootstrapping and sampling to approximate V^π
- Updates $V^\pi(s)$ after each transition (s, a, r, s')

$$V^\pi(s) = V^\pi(s) + \alpha(r + \gamma V^\pi(s') - V^\pi(s))$$

- Target is $r + \gamma V^\pi(s')$, a biased estimate of the true value $V^\pi(s)$
- Represent value for each state with a separate table entry

Automated Machine Learning Hannover

Temporal Difference (TD(0)) Learning with Value Function Approximation

- Uses bootstrapping and sampling to approximate V^π
- Updates $V^\pi(s)$ after each transition (s, a, r, s')

$$V^\pi(s) = V^\pi(s) + \alpha(r + \gamma V^\pi(s') - V^\pi(s))$$

- Target is $r + \gamma V^\pi(s')$, a biased estimate of the true value $V^\pi(s)$
- In value function approximation, target is $r + \gamma \hat{V}^\pi(s'; \mathbf{w})$, a biased and approximated estimate of the true value $V^\pi(s)$
- 3 forms of approximation:
  - sampling
  - bootstrapping
  - VFA

Temporal Difference (TD(0)) Learning with Value Function Approximation

- In value function approximation, target is $r + \gamma \hat{V}^{\pi}(s'; \mathbf{w})$, a biased and approximated estimate of the true value $V^{\pi}(s)$
- Can reduce doing TD(0) learning with value function approximation to supervised learning on a set of data pairs

$$\langle s_1, r_1 + \gamma \hat{V}^{\pi}(s_2; \mathbf{w})\rangle, \langle s_2, r_2 + \gamma \hat{V}^{\pi}(s_3; \mathbf{w})\rangle, \ldots$$

- Find weights to minimize mean squared error

$$J(\mathbf{w}) = \mathbb{E}_{\pi}[(r_j + \gamma \hat{V}^{\pi}(s_{j+1}, \mathbf{w}) - \hat{V}(s_j; \mathbf{w}))^2]$$

Automated Machine Learning Hannover

Temporal Difference (TD(0)) Learning with Value Function Approximation

- In value function approximation, target is $r + \gamma \hat{V}^\pi(s'; \mathbf{w})$, a biased and approximated estimate of the true value $V^\pi(s)$
- Can reduce doing TD(0) learning with value function approximation to supervised learning on a set of data pairs

$$\langle s_1, r_1 + \gamma \hat{V}^\pi(s_2; \mathbf{w}) \rangle, \langle s_2, r_2 + \gamma \hat{V}^\pi(s_3; \mathbf{w}) \rangle, \ldots$$

- In linear TD(0):

$$\begin{aligned} \Delta \mathbf{w} &= \alpha(r + \gamma \hat{V}^\pi(s'; \mathbf{w}) - \hat{V}^\pi(s; \mathbf{w})) \nabla_{\mathbf{w}} \hat{V}^\pi(s; \mathbf{w}) \\ &= \alpha(r + \gamma \hat{V}^\pi(s'; \mathbf{w}) - \hat{V}^\pi(s; \mathbf{w})) \mathbf{x}(s) \\ &= \alpha(r + \gamma \mathbf{x}(s')^T \mathbf{w} - \mathbf{x}(s)^T \mathbf{w}) \mathbf{x}(s) \end{aligned}$$

Automated Machine Learning Hannover

Temporal Difference (TD(0)) Learning with Value Function Approximation

Initialize $\mathbf{w} = 0$, $k = 1$;
Loop

- Sample tuple (s_k, a_k, r_k, s_{k+1}) given π
- Update weights:

$$\mathbf{w} = \mathbf{w} + \alpha(r + \gamma \mathbf{x}(s')^T \mathbf{w} - \mathbf{x}(s)^T \mathbf{w})\mathbf{x}(s)$$

- $k = k + 1$

Convergence Guarantees for Linear Value Function Approximation for Policy Evaluation

- Define the mean squared error of a linear value function approximation for a particular policy π relative to the true value as

$$\text{MSVE}(\mathbf{w}) = \sum_{s \in S} d(s)(V^{\pi}(s) - \hat{V}^{\pi}(s; \mathbf{w}))^2$$

- where
  - $d(s)$: stationary distribution of π in the true decision process
  - $\hat{V}(s; \mathbf{w}) = \mathbf{x}(s)^T \mathbf{w}$, a linear value function approximation
- TD(0) policy evaluation with VFA converges to weights $\mathbf{w}_{TD}$ which is a constant factor of the minimum mean squared error possible:

$$\text{MSVE}(\mathbf{w}_{TD}) \leq \frac{1}{1-\gamma} \min_{\mathbf{w}} \sum_{s \in S} d(s)(V^{\pi}(s) - \hat{V}(s; \mathbf{w}))^2$$

Automated Machine Learning Hannover