1 Overview

LP: essentially alternative to VI  PI (very hard to get guarantees with function approximation)

   LP: easy to get guarantees with function approximation

2 review of Value iteration

2.1 basic VI

\begin{align*}
\text{initialize } V &= 0 \\
\text{iterate : } &\forall s : \bar{V}(s) = \max_a \sum_{s'} P(s'|s,a)(R(s,a,s') + \gamma V(s')) \\
&V = \bar{V}
\end{align*}

the last two items are simply the bellman backup: \( V = TV \) This algorithm will converge to \( V^* \): \( V^* = TV^* \)

2.2 VI with function approximation

The algorithm with function approximation

\begin{align*}
\text{initialize } V &= 0 \\
\text{iterate : } &\forall s : \bar{V}(s) = \max_a \sum_{s'} P(s'|s,a)(R(s,a,s') + \gamma V(s')) \\
&V = \phi r \text{ for } r = \arg \min_r \| \bar{V} - \phi r \| \\
&\phi r \text{ for } r = \arg \min_r \sum_{s \in \bar{S}} (\bar{V}(s) - (\phi r)(s))
\end{align*}

Though this is just a theoretical algorithm, in practice one would do this:

\begin{align*}
r &= 0 \\
\text{iterate : } &\forall s : \bar{V}(s) = \max_a \sum_{s'} P(s'|s,a)(R(s,a,s') + \gamma V(s')) \\
&V = \phi r \text{ for } r = \arg \min_r \sum_{s \in \bar{S}} (\bar{V}(s) - (\phi r)(s))
\end{align*}

Where we’re only looking at some subset of the state space.
2.3 Tetris Example

cost: height of the current wall, discount = .9

22 features:

φ₀ to φ₉: height of columns 0 through 9
φ₁₀ to φ₁₈: height differential of consecutive columns
φ₁₉: holes
φ₂₀: max height
φ₂₁: 1

3 LP approach

3.1 relaxing Value Iteration

In Value iteration we solve for \( V \) in the following way
\[
V(s) = \max_a \sum_{s'} P(s'|s,a)(R(s,a,s') + \gamma V(s'))
\]

This is nonlinear, so let's investigate an alternative linear approach. We "relax" the problem and allow for a larger set of solutions

Instead we find \( V \) s.t.:
\[
\forall s : V(s) \geq \max_a \sum_{s'} P(s'|s,a)(R(s,a,s') + \gamma V(s'))
\]

This is equivalent to:
\[
\forall s, a : V(s) \geq \sum_{s'} P(s'|s,a)(R(s,a,s') + \gamma V(s'))
\]

We find a solution by solving
\[
\min_d d^T V \text{ s.t. } V \geq TV
\]
where \( d \) is an arbitrary vector

In shorthand this is: \( \min_d d^T V \text{ s.t. } V \geq TV \)

With this approach we have linear constraints and we can find a solution efficiently as it merely requires solving a linear optimization problem.

Our choice of \( d \) will affect which solution we find. Can we ensure that we find \( V^* \)?

For a feasible solution to LP we must have:
\[
V \geq TV
\] (14)

Recall
\[
V_1 \geq V_2 \Rightarrow TV_1 \geq TV_2
\] (15)

This implies
\[
TV \geq T^2 TV \geq T^3 V \geq ... \Rightarrow V \geq TV \geq T^2 V \geq ... \geq TV = V^*
\] (16)

Every feasible solution of \( V \) satisfies \( V \geq V^* \)
We can find \( V^* \) by solving the following linear program:
\[
\min_V \sum_s V(s) \text{ s.t. } V \geq TV
\] (17)

More generally this is:
\[
\min_V d^T VV \geq TV
\] (18)
This gives $V^*$ as a solution as long as $\forall s: d(s) > 0$

### 3.2 LP function approximation

\[
\begin{align*}
\min_{V, r} & \quad d^T V \\
\text{s.t.} & \quad V \geq TV \\
& \quad V = \phi r
\end{align*}
\]

\[
\downarrow
\]

\[
\begin{align*}
\min_{V, r} & \quad d^T \phi r \\
\text{s.t.} & \quad \phi r \geq T \phi r \\
& \quad V = \phi r
\end{align*}
\]

\[
\uparrow
\]

\[
\begin{align*}
\min_r & \quad \sum_{s \in \tilde{s}} d(s) \sum_i \phi_i(s) \cdot r_i \\
\text{s.t.} & \quad \forall s \in \tilde{s}, a: \\
& \quad \sum_i \phi_i(s) \cdot r_i \geq \sum_{s'} P(s'|s, a)[R(s, a, s') + \gamma \sum_j \phi_j(s') \cdot r_j]
\end{align*}
\]

lets call this solution $r^*$:

Is $r^* \in \arg \min_r \|V^* - \phi r^*\|$? No

$\|V^* - \phi r^*\|$ is comparable to $\min_r \|V^* - \phi r\|

### 3.3 relevant references

Ben Van Roy

de Farias Van Roy LP approach

Farias Van Roy Tetris case study