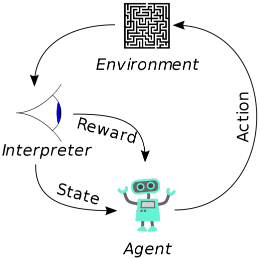
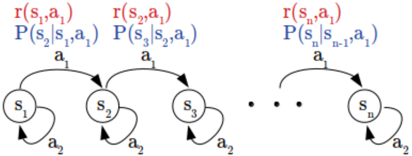
Linear Programming In Reinfocement Learning

Background

- What is Reinforcement Learning?
  - Machine Learning (think about regression) plus actions
  - Can be represented by state machines





  - As well as with a mathematical formula: $v(s) = r(s,a) + \gamma \sum_{j \in \mathcal{S}} P(s_j|s,a)v(j)$

Value Function

- It is a measure of how good is it to be at state s
- We like $v(s)$ to be as large as possible

$$v^*(s) = \max_a \{r(s,a) + \sum_{j \in \mathcal{S}} P(s_j|s,a)v(j)\}$$

- There are many ways to solve this equation, such as Value Iteration, Policy Iteration, Dynamic Programming, etc.
- The problem can also be formulated as a Linear Program

Value Function Optimization Using LP

- We are looking for the maximum of vector $v(s)$, which is of the size of the action set $|\mathcal{A}|$

$$\begin{aligned} &\min v(s) \\ s.t. \quad & v(s) \geq r(s,a) + \gamma \textstyle\sum_{j \in \mathcal{S}} P(s_j|s,a)v(j) \quad \forall a \in \mathcal{A} \end{aligned}$$

- We can do the same thing for all states

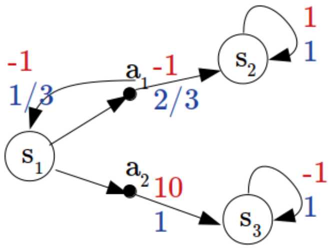
$$\begin{aligned} &\min \textstyle\sum_{j \in \mathcal{S}} \alpha(j)v(j) \\ s.t. \quad & v(s) \geq r(s,a) + \gamma \textstyle\sum_{j \in \mathcal{S}} P(s_j|s,a)v(j) \quad \forall a \in \mathcal{A} \quad \forall s \in \mathcal{S} \end{aligned}$$

- In vector form, with a bit of factorization and simplification

$$\begin{aligned} &\min \alpha \top v \\ s.t. \quad & \underbrace{(E - \gamma P)}_{A} v \geq r \end{aligned}$$

Value Function Optimization Using LP

Example:




```
Solved in 0 iterations and 0.23 seconds
Optimal objective  3.707317073e+02

Reward: 370.732
u[0] 1.46341
u[1] 18.5366
u[2] 0
u[3] 0
u[4] 0
u[5] 0
```

Robust MDPs

- We learned that v^* is

$$\begin{aligned} \min_v \; & \alpha^\top v \\ & Av \geq r \end{aligned}$$

- The dual of v^* is

$$\begin{aligned} \max_u \; & r^\top u \\ & A^\top u = \alpha \\ & u \geq 0 \end{aligned}$$

- In robust optimization, there are two agents. One is trying to maximize the objective function, and the other tries to minimize it
- In robust MDPs, nature plays the role of the second agent

$$\begin{aligned} \max_{r \in \mathcal{R}} \quad \max_u \; & r^\top u \\ & A^\top u = \alpha \\ & u \geq 0 \end{aligned}$$

Robust MDPs

- By strong duality we know

$$\min_{r\in\mathcal{R}} \max_{u} r^\top u = \max_{u} \min_{r\in\mathcal{R}} r^\top u$$
$$\begin{aligned} A^\top u &= \alpha & \qquad A^\top u &= \alpha \\ u &\geq 0 & \qquad u &\geq 0 \end{aligned}$$

- If the constraint over rewards are defined linearly $(Cr \leq d)$

$$\begin{array}{cc} \max\limits_{u} & \min\limits_{r} r^\top u \\ A^\top u = \alpha & Cr \leq d \\ u \geq 0 & \end{array}$$

- By writing the dual of the inner minimization and turn it into a maximization

$$\begin{aligned} &\max_{u,t} d^\top t \\ &A^\top u = \alpha \\ &u \geq 0 \\ &C^\top t = u \\ &t \geq 0 \end{aligned}$$

Thank You!