

## Markov Processes: Prediction<sup>1</sup>

Our aim in this series of lectures is to extend our suite of heuristic search and optimization algorithms to the case in which transitions to successor states are non-deterministic. We introduce Markov processes (MPs) and Markov decision processes (MDPs) as modeling tools in the study of non-deterministic state-space search problems. These models provide frameworks for computing optimal behavior in uncertain worlds. For example, we might be interested in planning an optimal route to work, or optimizing our blackjack strategy, or maximizing the returns on an investment in the stock market. Solutions may involve linear programming or dynamic programming methods, in the case where the non-deterministic nature of the process is known with certainty and the state and action space is sufficiently small; otherwise, they may be solved using Monte Carlo simulations or reinforcement learning (*e.g.*, TD-learning and Q-learning).

Our discussion of Markov (decision) processes is divided into two parts: the first is concerned with computing state-values  $V$ , and the second action-values  $Q$ . This division coincides with two related problems, namely:

1. the (*passive*) *prediction problem*, or *policy evaluation*: compute the state-value function  $V^\pi$ , given policy  $\pi$
2. the (*active*) *control problem*: find an optimal policy  $\pi^*$ , by computing the optimal action-value function  $Q^*$

### 1 Definitions and Examples

An agent operating in a non-deterministic environment transitions from state to state, possibly obtaining rewards along the way, as follows: at time  $t$ ,

1. state is  $s_t$
2. receive reward  $r_{t+1}$
3. transition to state  $s_{t+1}$  with probability  $P[s_{t+1}|s_t, \dots, s_0]$

In this lecture, we restrict our attention to *stationary* processes: *i.e.*, processes in which transition probabilities are independent of time.

---

<sup>1</sup>Copyright © Amy Greenwald, 2001-03

A discrete-time *stochastic process* is a tuple  $\langle S, r, P \rangle$ , where time is discrete: *i.e.*,  $t \in T = \{0, 1, \dots\}$ , and

- $S$  is a (finite) set of states
- $r : S \rightarrow \mathbb{R}$  is a reward function
- $P$  is a probability transition function that describes transitions between states, conditioned on past states: *e.g.*,  $P[s_{t+1}|s_t, \dots, s_0]$

A *Markov process* is a stochastic process whose probability transition function satisfies the Markov property:  $\forall t, \forall s_0, \dots, s_t \in S$ ,

$$P[s_{t+1}|s_t, \dots, s_0] = P[s_{t+1}|s_t] \quad (1)$$

The *Markov property* is sometimes called the memoryless property, since it implies that probability transitions to future states, such as  $s_{t+1}$ , depend only on the present state  $s_t$ , but are independent of the remote past, namely  $s_{t-1}, \dots, s_0$ .

**Example 1** *Gambler's Ruin* is an example of a Markov process. A gambler gambles until he either wins a set amount of money (in this example \$4), or loses all his money. At time  $t$ , his wealth increases by \$1 with probability  $1/3$ , and decreases by \$1 with probability  $2/3$ .

The set of states is defined by the worth of the gambler:  $S = \{0, 1, 2, 3, 4, \text{END}\}$ . The rewards are defined as  $r(0) = r(1) = r(2) = r(3) = r(\text{END}) = 0$  but  $r(4) = 1$ . Finally, the transition probabilities are such that  $P[i + 1|i] = 1/3$  and  $P[i - 1|i] = 2/3$ , for  $i = 1, 2, 3$ ,  $P[i|\text{END}] = 0$ , for  $i = 0, 4$ , and  $P[\text{END}|\text{END}] = 1$ .

Given initial state  $i \in 1, 2, 3$ , what is the probability that the gambler wins: *i.e.*, reaches state 4?  $\square$

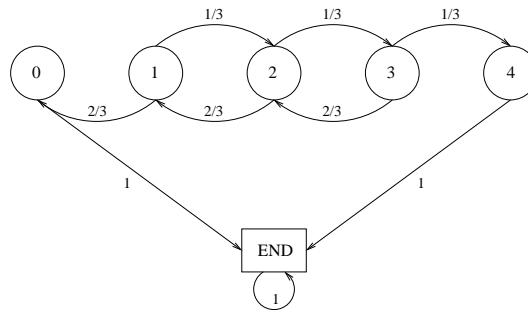


Figure 1: Gambler's Ruin.

An *absorbing state*  $s \in S$  is *s.t.*  $P[s|s] = 1$ . The END state is an absorbing state in Gambler's Ruin.

**Example 2** Consider the sentence: “Today is the first day of the rest of your life.” This sentence induces the Markov process:

- $T = \{0, 1, \dots\}$
- $S = \{\text{Today, is, the, first, day, of, rest, your, life.}\}$
- $r(s) = 0$ , for all  $s \in S$
- $P[\text{Today}|\epsilon] = 1$   
 $P[\text{is}|\text{Today}] = 1$   
 $P[\text{the}|\text{is}] = 1$   
 $P[\text{first}|\text{the}] = 1/2$   
 $P[\text{day}|\text{first}] = 1$   
 $P[\text{of}|\text{day}] = 1$   
 $P[\text{the}|\text{of}] = 1/2$   
 $P[\text{rest}|\text{the}] = 1/2$   
 $P[\text{of}|\text{rest}] = 1$   
 $P[\text{your}|\text{of}] = 1/2$   
 $P[\text{life}|\text{your}] = 1$   
 $P[\text{Today}|\text{life.}] = 1$

On this input, Doctor Nerve’s Markov chain program ([www.doctornerve.org/nerve/pages/interact/mrkvform.shtml](http://www.doctornerve.org/nerve/pages/interact/mrkvform.shtml)) generates the output: *Today is the first day of your life. Today is the first day of the first day of your life. Today is the rest of your life.* □

## 2 Expected Return

The expected return  $R_t$  at time  $t$  is the expected stream of future rewards  $r_t, r_{t+1}, r_{t+2}, \dots$ . In the case of a finite horizon, say of length  $T < \infty$ , expected return can be computed simply as the sum of *expected* future rewards: *i.e.*,

$$R_t = r_t + r_{t+1} + r_{t+2} + \dots + r_T = \sum_{i=0}^{T-t} r_{t+i} \quad (2)$$

In the case of infinite horizons, however, the sum of expected future rewards is potentially infinite. If all trajectories are *proper* (a trajectory is called *proper* iff it transitions to a zero-reward, absorbing state with positive probability), once again, expected return can be computed as the sum of expected future rewards, as in Equation 2. Otherwise, expected return is computed as the *discounted* sum of expected future rewards, assuming discount factor  $0 \leq \gamma < 1$ : *i.e.*,

$$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots = \sum_{i=0}^{\infty} \gamma^i r_{t+i} \quad (3)$$

Assuming bounded rewards, expected return as defined by Equation 3 is finite. **(Exercise)** But even in the case of finite horizons or proper trajectories,  $R_t$  is often computed with discounting, because of the following economic intuition.

The motivation for discounting future rewards can be simply stated: *a dollar today is worth more than a dollar tomorrow*. For example, given an interest rate of  $x\%$  per annum,  $d$  dollars today are worth  $(1+x)d$  dollars 365 days from now. Thus,  $d$  dollars 365 days from now are worth only  $d/(1+x)$  dollars today. The discount factor  $\gamma$  is inversely related to the interest rate:  $\gamma = 1/(1+x)$ .

Intuitively,  $\gamma$  determines the relative worth of immediate vs. future rewards. If  $\gamma = 0$ , only immediate rewards are deemed relevant; agents that attempt to maximize expected return in these circumstances are called *myopic*. As  $\gamma \rightarrow 1$ , future rewards are weighted more and more heavily; agents that aim to maximize discounted rewards based on high values of  $\gamma$  exhibit *foresight*.

### 3 Bellman's Theorem

We now derive Bellman's theorem for Markov processes: the state-value  $V(s_t)$  at state  $s_t$  is the sum of the immediate reward obtained in state  $s_t$  and the discounted sum of the rewards obtained thereafter:

$$V(s_t) = r_t + \gamma \mathbb{E}_{s_{t+1}}[V(\cdot)] \quad (4)$$

By definition, the state-value  $V(s_t)$  is the expected return  $R_t$ . This return is the expected stream of future rewards, where expectation is computed with respect to probabilities over (proper or finite) trajectories  $\tau = (s_t, s_{t+1}, s_{t+2}, \dots)$ : *i.e.*,

$$R_t = \sum_{\tau} P[\tau|s_t] R_t^{\tau} \quad (5)$$

The value  $R_t^{\tau}$  is the discounted sum of expected rewards along trajectory  $\tau$ : *i.e.*,  $R_t^{\tau} = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots$ . Bellman's theorem can be derived as follows:

$$\begin{aligned} V(s_t) &= \sum_{\tau} P[\tau|s_t] R_t^{\tau} \\ &= r_t + \gamma \sum_{\tau'} P[\tau'|s_t] R_{t+1}^{\tau'} \\ &= r_t + \gamma \sum_{\tau'} \sum_{s_{t+1} \in S} P[s_{t+1}|s_t] P[\tau'|s_t, s_{t+1}] R_{t+1}^{\tau'} \\ &= r_t + \gamma \sum_{s_{t+1} \in S} P[s_{t+1}|s_t] \sum_{\tau'} P[\tau'|s_{t+1}] R_{t+1}^{\tau'} \\ &= r_t + \gamma \sum_{s_{t+1} \in S} P[s_{t+1}|s_t] V(s_{t+1}) \\ &= r_t + \gamma \mathbb{E}_{s_{t+1}}[V(\cdot)] \end{aligned}$$

The first line follows from the definition of  $V$ ; the second line follows the fact that  $r_t$  is known with certainty; the third line follows from the definition of marginal probability; the fourth line follows from the Markov property; the fifth line follows from the definition of  $V$ ; the sixth line is simply an abbreviation.

### 3.1 Bellman's Equations

The value  $V(s)$  of state  $s$  is defined as the expected return associated with state  $s$ . Bellman's theorem states that *the expected return of state  $s$  equals the immediate rewards obtained at state  $s$  plus the discounted sum of expected future rewards earned thereafter*. By stationarity, this theorem gives rise to the following system of  $|S|$  equations, known as Bellman's equations: for all  $s \in S$ ,

$$V(s) = r(s) + \gamma \sum_{s'} P[s'|s]V(s') \quad (6)$$

The proof that Bellman's system of equations has a solution relies on Banach's fixed point theorem, also called the contraction mapping theorem. Given a metric space  $(X, d)$ , a mapping  $f : X \rightarrow X$  is a *contraction* iff there exists some  $0 \leq k < 1$  s.t.  $d(f(x), f(y)) \leq kd(x, y)$ , for all  $x, y \in X$ .

**Theorem 3.1** *Given complete, metric space  $(X, d)$  and contraction mapping  $f : X \rightarrow X$ , (i) there exists unique  $x^* \in X$  s.t.  $f(x^*) = x^*$ ; and, (ii) for arbitrary  $x^0 \in X$ , the sequence  $\{x^n\}$  defined by  $x^{n+1} = f(x^n) = f^{n+1}(x^0) \rightarrow x^*$ .*

Define the mapping  $f : X \rightarrow X$  as follows:

$$f(x)(s) = r(s) + \gamma \sum_{s'} P[s'|s]x(s') \quad (7)$$

**Theorem 3.2** *The mapping  $f$  in Equation 7 is a contraction in the  $L_\infty$ -norm.*

**Corollary 3.3** *Bellman's system of optimality equations indeed has a fixed point solution, and the iterative application of  $f$  converges to this solution.*

**Proof 3.4** Let the metric  $d$  be the max norm: i.e.,  $\|x - y\| = \max_i |x_i - y_i|$ . For arbitrary state  $s \in S$ ,

$$\begin{aligned} |f(x)(s) - f(y)(s)| &= \left| r(s) + \gamma \sum_{s'} P[s'|s]x(s') - \left( r(s) + \gamma \sum_{s'} P[s'|s]y(s') \right) \right| \\ &= \gamma \sum_{s'} P[s'|s] |x(s') - y(s')| \\ &\leq \gamma \sum_{s'} P[s'|s] \|x - y\| \\ &= \gamma \|x - y\| \end{aligned}$$

Therefore,  $\|f(x) - f(y)\| \leq \gamma \|x - y\|$ .  $\square$

## 4 State Values

Policy evaluation is a dynamic programming method that computes state-values via iterative updates based on Bellman's equations:

$$V(s) \leftarrow r(s) + \gamma \sum_{s'} P[s'|s]V(s') \quad (8)$$

Gauss-Seidel's version of this algorithm incorporates in-place updating: *i.e.*, updating with  $V$ , as shown in Table 2, rather than  $V'$ , as shown in Table 1.

POLICY_EVALUATION(MP, $\gamma$ , $\epsilon$ )	
Inputs	discount factor $\gamma$ convergence test $\epsilon$
Output	state-value function $V$
Initialize	$V = 0$ and $V' = \infty$
<b>while</b> $\max_s  V(s) - V'(s)  > \epsilon$ <b>do</b>	
1. $V' = V$	
2. for all $s \in S$	
(a) $V(s) = r(s) + \gamma \sum_{s'} P[s' s]V'(s')$	
<b>return</b> $V$	

Table 1: Policy Evaluation.

GAUSS_SEIDEL(MP, $\gamma$ , $\epsilon$ )	
Inputs	discount factor $\gamma$ convergence test $\epsilon$
Output	state-value function $V$
Initialize	$V = 0$ and $V' = \infty$
<b>while</b> $\max_s  V(s) - V'(s)  > \epsilon$ <b>do</b>	
1. $V' = V$	
2. for all $s \in S$	
(a) $V(s) = r(s) + \gamma \sum_{s'} P[s' s]V(s')$	
<b>return</b> $V$	

Table 2: Gauss-Seidel.

## 4.1 Example: Gambler's Ruin

Assuming  $\gamma = 1$ , the extent of the gambler's ruin is indicated by the state-values in the table below. These values are computed via policy evaluation as follows:

V	0	1	2	3	4	END
0	0	0	0	0	0	0
1	0	0	0	0	1	0
2	0	0	0	$\frac{1}{3}$	1	0
3	0	0	$\frac{1}{9}$	$\frac{1}{3}$	1	0
4	0	$\frac{1}{27}$	$\frac{1}{9}$	$\frac{11}{27}$	1	0
5	0	$\frac{1}{27}$	$\frac{13}{81}$	$\frac{11}{27}$	1	0

With in-place computation *à la* Gauss-Seidel, working backwards from END to state 4 down to state 0, the computation proceeds as follows:

V	0	1	2	3	4	END
0	0	0	0	0	0	0
1	0	$\frac{1}{27}$	$\frac{1}{9}$	$\frac{1}{3}$	1	0
2	0	$\frac{13}{243}$	$\frac{13}{81}$	$\frac{11}{27}$	1	0
3	0	$\frac{133}{2187}$	$\frac{133}{729}$	$\frac{107}{243}$	1	0
100	0	0.0667	0.2	0.4667	1	0

Since rewards are zero everywhere unless the gambler is *not* ruined, the final state-values can be interpreted as the probability that the gambler is *not* ruined.

## Problems

- #1 (a) Show that for all MPs, the value function  $V$  is well-defined if  $0 \leq \gamma < 1$ .
- (b) Show that there exists an MP *s.t.* the value function  $V$  is not necessarily well-defined if  $\gamma = 1$ .
- (c) Prove that any MP  $M_1$ , which for discount factor  $0 \leq \gamma < 1$  yields value function  $V_1$ , can be transformed into another MP  $M_2$ , *s.t.* the *undiscounted* value function  $V_2 = V_1$ . [Hint: Introduce a zero-reward, absorbing state END in  $M_2$  *s.t.* all state-action pairs in  $M_1$  transition to END with probability  $1 - \gamma$ .]