High Rollers: A Dyanmic Programming Exercise

Investment, Finance and Asset Prices ECON5068

Thomas Walsh

Adam Smith Business School

Overview

- Turn a **complicated asset-valuation problem** into an optimal **action rule** (and value function)

- Useful across economics, finance, and beyond!

- Seen as assessment exercise in financial recruitment

The Game

You work as an analyst for **Kelvingrove Asset Management Co.**

We have deep pockets and only care about **expected value**,

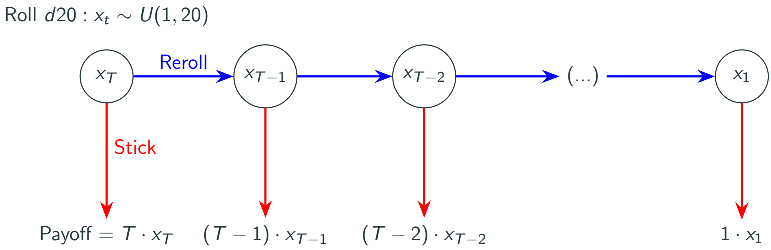
The Boss has asked you to value an asset with the following structure:

Problem Setup

- You have a fair 20-sided die ($d20$).
- **Horizon**: T rounds to score.
- Each round:
  - If you **Stick (S)**: collect the current (face value $\times$ number of rounds remaining).
  - If you **Reroll (R)**: sacrifice the current payoff, advance one round, roll again.
- **Goal**: maximize the expected **total payoff** over T rounds.
- **Goal**: What's the value of this game?
- We will set up the **game recursively** and use **dynamic programming** to solve for optimal actions

Sequence of the game

Let x_t be the score on the die with t periods remaining.



Definitions

- **Value** of game with T **rounds remaining** and d20-face x showing.

$$V^T(x)$$

- Two **actions, S or R**.
- Stick, gives the current score times number of remaing turns

$$S^T(x) = T \cdot x \quad \texttt{(stick)}$$

- Reroll has the unconditional mean value, with one turn fewer

$$R^T = \mathbb{E}_y[V^{T-1}(Y)] \quad (\texttt{reroll}, \text{with } Y \sim \text{Uniform}\{1, \ldots, 20\})$$

- Value under optimal play:

$$V^T(x) = \max\{S^T(x), R^T\} = \max\{S^T(x), \mathbb{E}V^{T-1}(x')\}$$

Building Intuition: Start at the end

- **Last turn**, d20-die shows x, Stick payoff

$$S^1(x) = x \quad (V^0 = 0)$$

- **Game has no future value** so

$$R^1 = \frac{1}{20}\sum_{y=1}^{20} V^0(y) = 0.$$

- **Value** of game at $t = 1$ remaining period:

$$V^1(x) = \max\{x, 0\} = x \tag{1}$$

$$\Rightarrow R^2 = \mathbb{E}(V^1(y)) = \mathbb{E}(d20) = 10.5 \tag{2}$$

- Now we move to **2 periods remaining**:

$$V^2(x) = \max\{S^2(x), R^2\} = \max\{2x, 10.5\}$$

3 turns left

- Summarise what we have so far:

$$V^0 = 0 \tag{3}$$

$$V^1(x) = x \tag{4}$$

$$V^2(x) = \max\{2x, 10.5\} \tag{5}$$

$$R^3 = \mathbb{E}(V^2(y)) = \frac{1}{20}\sum_{y=1}^{20} V^2(y) \tag{6}$$

3 turns left

We can calculate this reroll value R^3, for $x = 1, 2, ..5$, we accept the reroll with expected value 10.5, otherwise for $x = 6, .., 20$ stick and earn $2y$

$$R^3 = \frac{1}{20} \sum_{y=1}^{20} \max\{2y, 10.5\} \tag{7}$$

$$= \sum_{y=1}^{5} 10.5 + \sum_{6=1}^{20} 2y \tag{8}$$

$$= (5 \times 10.5) + (12 + 14 + ... + 40) \tag{9}$$

Hard to keep track of everything relative to the dynamic programming solution (easy)

Recursive Formulation

- Terminal Condition:

$$V^1(x) = x \quad (V^0 = 0)$$

- For $t \geq 2$:

$$V^t(x) = \max\{t \cdot x,\ R^t\},$$

  where

$$R^t = \mathbb{E}_y[V^{t-1}(y)] = \frac{1}{20}\sum_{y=1}^{20} V^{t-1}(y).$$

- Threshold rule:

$$\text{Stick if } x \geq \frac{R^t}{t}.$$

Algorithm / Pseudocode

1. Initialize $V_1(x) = x$.
2. For $t = 2, \dots, T$:
   - Take Value for one-period-less V_{t-1}
   - Compute Reroll Value $R_t = \frac{1}{20} \sum_y V_{t-1}(y)$.
   - Set threshold (ceiling function) $x_t^\star = \lceil R_t/t \rceil$. *e.g.* $\lceil 2.4 \rceil = 3$.
   - For each face x:

$$V^t(x) = \max\{t \cdot x, R^t\}.$$

3. **Policy**: stick if $x \geq x_t^\star$, otherwise reroll.

Check out `high_rollers.m` on moodle for the code!

Comparing Sticking with Rerolling with T rolls left

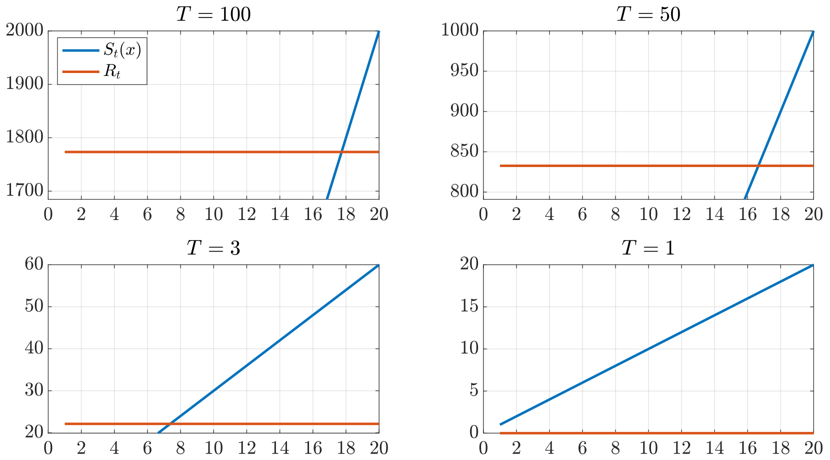


Figure 1: Value of Stick and Reroll

Policy and Value Functions

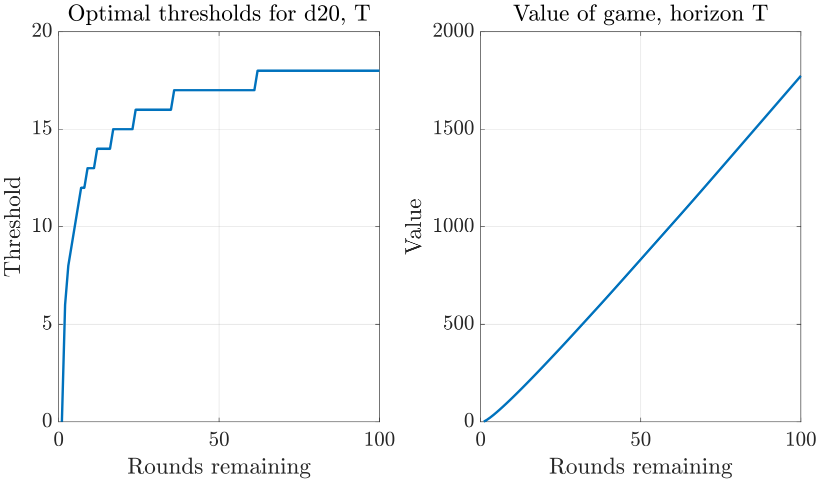


Figure 2: Optimal Policy and Value of game

Monte Carlo Simulation (N=1M players, T=33)

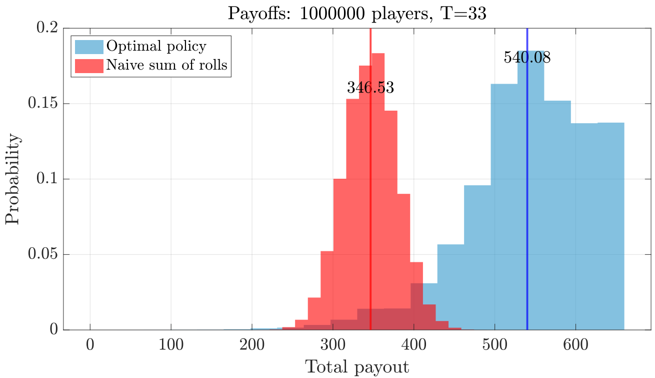


Figure 3: Monte Carlo Simulation of Many Games: Naive Sum versus Optimal Play

Naive Sum is $\mathbb{E}_y(V^T(y))$ expected value of immediately cashing out.

Exercise

Question

What is the expected payoff under the optimal play for T-rounds?
(with this we know the maximum to pay for such a prospect)

Comments

- **Compare to naive expected value $E(d20) \times T$.**
  - Typically much higher than Naive sum
- **Why is the optimal policy sometimes lower on the left tail but much higher on the right tail?**
  - some people get stuck rolling many low scores, and keep rerolling, which sacrifices this turns payoff, so they end up burning a lot of potential value from low scores chasing the higher payoffs later
- **Discuss tradeoff: rerolling $\Rightarrow$ sacrifice today's payoff + lose one period.**
  - when there are many periods left, it makes sense to be highly selective and only accept very high scores, turns are abundant, we can waste a few to improve our score.
  - Later on, burning each score to reroll can be detrimental
  - Optimal policy reflects this falling choosiness as time runs out

Takeaways

- **Dynamic programming** simplifies **sequential** stopping problems into simple rules.
- **Threshold rule**: stick iff current value > continuation.
- Variance of outcomes increases under optimal play: high risk, high reward.
- Many problems have this Optimal Stopping aspect:
    - **IPO timing**: go public now or hold out for better, markets may lose interest
    - **Finding a job** with limited savings
    - **Apartment** hunting when the move-in date approaches
    - Buying **airline, train, concert tickets** on a given date