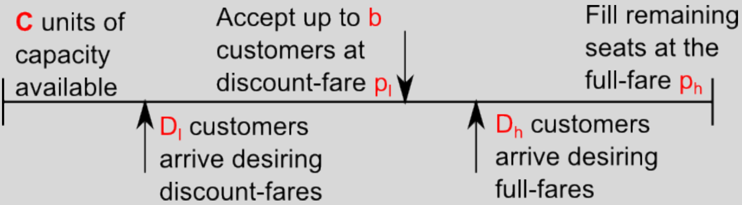
ORIE 4154 - Pricing and Market Design

Module 1: Capacity-based Revenue Management
(Intro to Stochastic Dynamic Programming)

Instructor: Sid Banerjee, ORIE

Cornell University

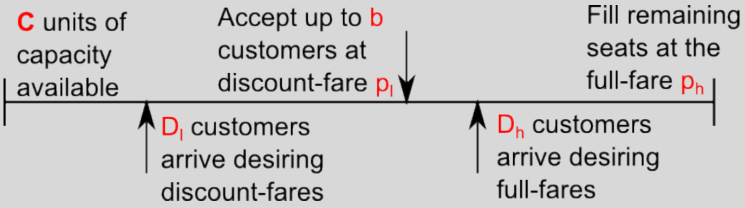
Single-resource two-stage capacity allocation



$$R(b, D_\ell, D_h) = p_\ell \min\{b, D_\ell\} + p_h \min\{D_h, \max\{C - b, C - D_\ell\}\}$$

Aim: Choose $b^* = \arg\max_{b \in [0,C]} \mathbb{E}[R(b, D_\ell, D_h)]$

Single-resource two-stage capacity allocation



$$R(b, D_\ell, D_h) = p_\ell \min\{b, D_\ell\} + p_h \min\{D_h, \max\{C - b, C - D_\ell\}\}$$

Aim: Choose $b^* = \arg\max_{b \in [0,C]} \mathbb{E}[R(b, D_\ell, D_h)]$

Littlewood's Rule

Assume D_ℓ, D_h are continuous, b can be fractional – then optimal booking limit b^* (or protection level $C - b^*$) satisfies:

$$C - b^* = y^* = F_h^{-1}\left(1 - \frac{p_\ell}{p_h}\right)$$

Single-resource two-stage capacity allocation

Alternate derivation of Littlewood's rule (Discrete RVs)

$$R(b, D_\ell, D_h) = p_\ell \min\{b, D_\ell\} + p_h \min\{D_h, \max\{C-b, C-D_\ell\}\}$$

Suppose D_h, D_ℓ are discrete RVs

Single-resource two-stage capacity allocation

Alternate derivation of Littlewood's rule (Discrete RVs)

$$R(b, D_\ell, D_h) = p_\ell \min\{b, D_\ell\} + p_h \min\{D_h, \max\{C-b, C-D_\ell\}\}$$

Suppose D_h, D_ℓ are discrete RVs

Main Idea: Analyze $\Delta r(b) = \mathbb{E}[R(b+1, D_\ell, D_h) - R(b, D_\ell, D_h)]$

Single-resource two-stage capacity allocation

Alternate derivation of Littlewood's rule (Discrete RVs)

$$R(b,D_\ell,D_h) = p_\ell \min\{b,D_\ell\} + p_h \min\{D_h, \max\{C-b, C-D_\ell\}\}$$

Suppose D_h, D_ℓ are discrete RVs
Main Idea: Analyze $\Delta r(b) = \mathbb{E}[R(b+1,D_\ell,D_h) - R(b,D_\ell,D_h)]$

$$\begin{aligned}\Delta r(b) = & p_\ell \mathbb{E}\Big[\min\{b+1,D_\ell\} - \min\{b,D_\ell\}\Big] + \\ & p_h \mathbb{E}\Big[\min\{D_h,\max\{C-(b+1),C-D_\ell\}\} - \\ & \min\{D_h,\max\{C-b,C-D_\ell\}\}\Big]\end{aligned}$$

Littlewood's Rule (Discrete RVs)

- $\min\{b+1, D_\ell\} - \min\{b, D_\ell\} = \begin{cases} 1 & \text{if } D_\ell \geq b+1 \\ 0 & \text{o.w.} \end{cases}$

Littlewood's Rule (Discrete RVs)

- $\min\{b+1, D_\ell\} - \min\{b, D_\ell\} = \begin{cases} 1 & \text{if } D_\ell \geq b+1 \\ 0 & \text{o.w.} \end{cases}$
- $\min\{D_h, \max\{C-(b+1), C-D_\ell\} - \min\{D_h, \max\{C-b, C-D_\ell\}$

$= \begin{cases} -1 & \text{if } \{D_\ell \geq b+1\} \text{ AND } \{D_h \geq C-b\} \\ 0 & \text{o.w.} \end{cases}$

Littlewood's Rule (Discrete RVs)

- $\min\{b+1, D_\ell\} - \min\{b, D_\ell\} = \begin{cases} 1 & \text{if } D_\ell \geq b+1 \\ 0 & \text{o.w.} \end{cases}$
- $\min\{D_h, \max\{C-(b+1), C-D_\ell\} - \min\{D_h, \max\{C-b, C-D_\ell\}$
$= \begin{cases} -1 & \text{if } \{D_\ell \geq b+1\} \text{ AND } \{D_h \geq C-b\} \\ 0 & \text{o.w.} \end{cases}$

- $\mathbb{E}\Big[\min\{b+1, D_\ell\} - \min\{b, D_\ell\}\Big] = \mathbb{P}[D_\ell \geq b+1]$
- $\mathbb{E}\Big[\min\{D_h, \max\{C-(b+1), C-D_\ell\}\} - \min\{D_h, \max\{C-b, C-D_\ell\}\}\Big]$
$= -\mathbb{P}[D_\ell \geq b+1]\mathbb{P}[D_h \geq C-b]$ (Note: via independence of D_h, D_ℓ)

Littlewood's Rule (Discrete RVs)

- $\min\{b+1, D_\ell\} - \min\{b, D_\ell\} = \begin{cases} 1 & \text{if } D_\ell \geq b+1 \\ 0 & \text{o.w.} \end{cases}$
- $\min\{D_h, \max\{C-(b+1), C-D_\ell\} - \min\{D_h, \max\{C-b, C-D_\ell\}$
$= \begin{cases} -1 & \text{if } \{D_\ell \geq b+1\} \text{ AND } \{D_h \geq C-b\} \\ 0 & \text{o.w.} \end{cases}$

- $\mathbb{E}\Big[\min\{b+1, D_\ell\} - \min\{b, D_\ell\}\Big] = \mathbb{P}[D_\ell \geq b+1]$
- $\mathbb{E}\Big[\min\{D_h, \max\{C-(b+1), C-D_\ell\}\} - \min\{D_h, \max\{C-b, C-D_\ell\}\}\Big]$
$= -\mathbb{P}[D_\ell \geq b+1]\mathbb{P}[D_h \geq C-b]$ (Note: via independence of D_h, D_ℓ)

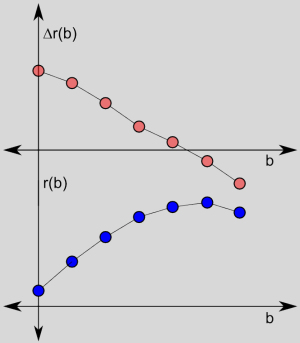
$$\Delta r(b) = \mathbb{P}[D_\ell \geq b+1]\left(p_\ell - p_h \mathbb{P}[D_h \geq C-b]\right)$$

Littlewood's Rule (Discrete RVs)

$$\Delta r(b) = r(b+1) - r(b) = \mathbb{P}[D_\ell \geq b+1]\,(p_\ell - p_h \mathbb{P}[D_h \geq C-b])$$

Littlewood's Rule (Discrete RVs)

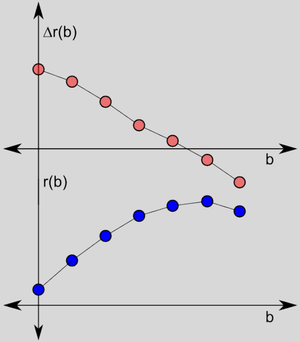
$$\Delta r(b) = r(b+1) - r(b) = \mathbb{P}[D_\ell \geq b+1]\,(p_\ell - p_h\mathbb{P}[D_h \geq C-b])$$



- Observe that $\Delta r(b)$ is non-increasing in b
- Thus $r(b)$ is maximized at $\min_b\{\Delta r(b) < 0\}$

Littlewood's Rule (Discrete RVs)

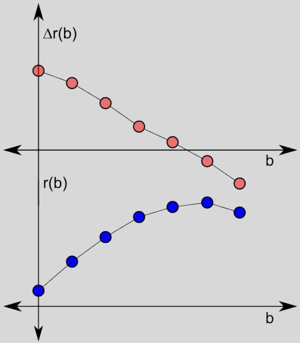
$$\Delta r(b) = r(b+1) - r(b) = \mathbb{P}[D_\ell \geq b+1] \left(p_\ell - p_h \mathbb{P}[D_h \geq C-b] \right)$$



- Observe that $\Delta r(b)$ is non-increasing in b
- Thus $r(b)$ is maximized at $\min_b \{\Delta r(b) < 0\}$
- In terms of protection level $y = C - b$, we have: $y^* = \max_{y \in \mathbb{N}} \left\{ \mathbb{P}[D_h \geq y] > \frac{p_\ell}{p_h} \right\}$

Littlewood's Rule (Discrete RVs)

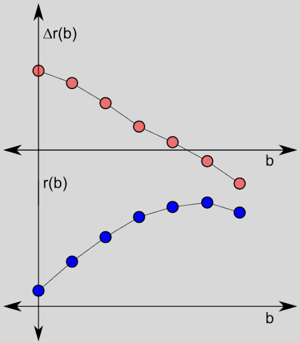
$$\Delta r(b) = r(b+1) - r(b) = \mathbb{P}[D_\ell \geq b+1]\left(p_\ell - p_h \mathbb{P}[D_h \geq C-b]\right)$$



- Observe that $\Delta r(b)$ is non-increasing in b
- Thus $r(b)$ is maximized at $\min_b\{\Delta r(b) < 0\}$
- In terms of protection level $y = C - b$, we have: $y^* = \max_{y \in \mathbb{N}} \left\{ \mathbb{P}[D_h \geq y] > \frac{p_\ell}{p_h} \right\}$
- Questions/Observations: Why is y^* independent of C? Why is y^* independent of D_ℓ?

Littlewood's Rule (Discrete RVs)

$$\Delta r(b) = r(b+1) - r(b) = \mathbb{P}[D_\ell \geq b+1]\,(p_\ell - p_h\mathbb{P}[D_h \geq C-b])$$



- Observe that $\Delta r(b)$ is non-increasing in b
- Thus $r(b)$ is maximized at $\min_b\{\Delta r(b) < 0\}$
- In terms of protection level $y = C - b$, we have: $y^* = \max_{y\in\mathbb{N}}\left\{\mathbb{P}[D_h \geq y] > \frac{p_\ell}{p_h}\right\}$
- Questions/Observations:
  Why is y^* independent of C?
  Why is y^* independent of D_ℓ?
  Perfect segmentation + dynamics

Problem: Single-resource three-stage capacity allocation

- Seller constraints:
  - (Setting) C seats, 3 fare-classes with prices $p_1 > p_2 > p_3$
  - (Control) Booking limits $\{b_2, b_3\}$

Problem: Single-resource three-stage capacity allocation

- Seller constraints:
  - (Setting) C seats, 3 fare-classes with prices $p_1 > p_2 > p_3$
  - (Control) Booking limits $\{b_2, b_3\}$
- Buyer behavior and information structure:
  - (Perfect Segmentation) Fare-classes arrive separately
  - (Dynamics) Sequential arrival in increasing order of fares
  - (Demand Distributions) Demand for fare-class i is $D_i \sim F_i$

Problem: Single-resource three-stage capacity allocation

- Seller constraints:
  - (Setting) C seats, 3 fare-classes with prices $p_1 > p_2 > p_3$
  - (Control) Booking limits $\{b_2, b_3\}$
- Buyer behavior and information structure:
  - (Perfect Segmentation) Fare-classes arrive separately
  - (Dynamics) Sequential arrival in increasing order of fares
  - (Demand Distributions) Demand for fare-class i is $D_i \sim F_i$

How can we solve for optimal booking limits?

Problem: Single-resource three-stage capacity allocation

- Seller constraints:
  - (Setting) C seats, 3 fare-classes with prices $p_1 > p_2 > p_3$
  - (Control) Booking limits $\{b_2, b_3\}$
- Buyer behavior and information structure:
  - (Perfect Segmentation) Fare-classes arrive separately
  - (Dynamics) Sequential arrival in increasing order of fares
  - (Demand Distributions) Demand for fare-class i is $D_i \sim F_i$

How can we solve for optimal booking limits?

Naive approach: Write expected revenue function $R(b_3, b_2)$, and maximize over $b_3 + b_2 \leq C$

- Seems somewhat daunting...

Problem: Single-resource three-stage capacity allocation

- Seller constraints:
  - (Setting) C seats, 3 fare-classes with prices $p_1 > p_2 > p_3$
  - (Control) Booking limits $\{b_2, b_3\}$
- Buyer behavior and information structure:
  - (Perfect Segmentation) Fare-classes arrive separately
  - (Dynamics) Sequential arrival in increasing order of fares
  - (Demand Distributions) Demand for fare-class i is $D_i \sim F_i$

How can we solve for optimal booking limits?

Naive approach: Write expected revenue function $R(b_3, b_2)$, and maximize over $b_3 + b_2 \leq C$

- Seems somewhat daunting...

Idea: If an oracle gave us X_3, the number of lowest-fare class seats sold – then we have a two fare-class problem with $C - X_3$ seats!

First, let us play a game

- Setup: A pile of 10 toothpicks
- You will be playing against an oblivious random adversary (called Computer).
- A Sequence of Events in Each Iteration:
  - You start first. You can take either 1 or 2 toothpicks from the pile.
  - After you make the decision, I will flip a random fair coin. If the coin lands HEAD, the Computer will remove 1 toothpick from the pile. Otherwise, the Computer will remove 2 toothpicks.
- The game proceeds until all toothpicks are removed from the pile.
- If you end up holding the last toothpick, you win $20. Otherwise, you get nothing.

Courtesy: Paat Rusmevichientong

(Aside: Variant of a game called Nim; see Youtube video for details)

Analyzing our game

Divide game into rounds: in each round, you go first followed by COMPUTER

In k^{th} round, computer picks X_k toothpicks ($X_k \sim$ UNIFORM$\{1,2\}$)

Observations

Analyzing our game

Divide game into rounds: in each round, you go first followed by COMPUTER

In k^{th} round, computer picks X_k toothpicks ($X_k \sim$ UNIFORM$\{1,2\}$)

Observations

- If the game starts with 1 or 2 toothpicks, then we win! (If game starts with 0 toothpicks, assume we lose.)

Analyzing our game

Divide game into rounds: in each round, you go first followed by COMPUTER

In k^{th} round, computer picks X_k toothpicks ($X_k \sim$ UNIFORM$\{1,2\}$)

Observations

- If the game starts with 1 or 2 toothpicks, then we win! (If game starts with 0 toothpicks, assume we lose.)
- Suppose after $k-1$ rounds, game has $S_k \geq 3$ toothpicks left, and let S_{k+1} be number of toothpicks left when we play next:
  - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
  - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

Analyzing our game

Divide game into rounds: in each round, you go first followed by COMPUTER

In k^{th} round, computer picks X_k toothpicks ($X_k \sim$ UNIFORM$\{1,2\}$)

Observations

- If the game starts with 1 or 2 toothpicks, then we win! (If game starts with 0 toothpicks, assume we lose.)
- Suppose after $k-1$ rounds, game has $S_k \geq 3$ toothpicks left, and let S_{k+1} be number of toothpicks left when we play next:
    - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
    - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

We will now see how to 'solve' this game (i.e., figure out an optimal set of moves) via Dynamic Programming.

Analyzing our game

- If after $k-1$ rounds, game has $S_k \geq 3$ toothpicks, and S_{k+1} is number of toothpicks when we play next:
  - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
  - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

  (where $X_k \sim$ UNIFORM$\{1,2\}$)

Let $V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

Analyzing our game

- If after $k-1$ rounds, game has $S_k \geq 3$ toothpicks, and S_{k+1} is number of toothpicks when we play next:
  - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
  - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

  (where $X_k \sim$ UNIFORM$\{1,2\}$)

Let $V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$

Analyzing our game

- If after $k-1$ rounds, game has $S_k \geq 3$ toothpicks, and S_{k+1} is number of toothpicks when we play next:
  - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
  - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

  (where $X_k \sim$ UNIFORM$\{1,2\}$)

Let $V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max \mathbb{E}[\text{Reward}]$ if round starts with 3 toothpicks

Analyzing our game

- If after $k-1$ rounds, game has $S_k \geq 3$ toothpicks, and S_{k+1} is number of toothpicks when we play next:
  - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
  - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

  (where $X_k \sim \text{UNIFORM}\{1,2\}$)

Let $V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max \mathbb{E}[\text{Reward}]$ if round starts with 3 toothpicks

$$= \max \left\{ \mathbb{E}[R \text{ if we pick 1 of 3}], \mathbb{E}[R \text{ if we pick 2 of 3}] \right\}$$

Analyzing our game

- If after $k-1$ rounds, game has $S_k \geq 3$ toothpicks, and S_{k+1} is number of toothpicks when we play next:
  - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
  - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

  (where $X_k \sim$ UNIFORM$\{1,2\}$)

Let $V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max \mathbb{E}[\text{Reward}]$ if round starts with 3 toothpicks

$$= \max \left\{ \mathbb{E}[R \text{ if we pick 1 of 3}], \mathbb{E}[R \text{ if we pick 2 of 3}] \right\}$$
$$= \max \left\{ \mathbb{E}[V(3-1-X)], \mathbb{E}[V(3-2-X)] \right\}$$

Analyzing our game

- If after $k-1$ rounds, game has $S_k \geq 3$ toothpicks, and S_{k+1} is number of toothpicks when we play next:
  - If we pick 1 match, then $S_{k+1} = S_k - 1 - X_k$
  - If we pick 2 match, then $S_{k+1} = S_k - 2 - X_k$

  (where $X_k \sim$ UNIFORM$\{1,2\}$)

Let $V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max \mathbb{E}[\text{Reward}]$ if round starts with 3 toothpicks

$$= \max \left\{ \mathbb{E}[R \text{ if we pick 1 of 3}], \mathbb{E}[R \text{ if we pick 2 of 3}] \right\}$$

$$= \max \left\{ \mathbb{E}[V(3-1-X)], \mathbb{E}[V(3-2-X)] \right\}$$

$$= \max \left\{ \left(\frac{V(1)+V(0)}{2} \right), \left(\frac{V(0)+V(-1)}{2} \right) \right\} = 10$$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\left\{0.5 \cdot \left(V(1) + V(0)\right), 0.5 \cdot \left(V(0) + V(-1)\right)\right\} = 10$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\left\{0.5 \cdot \big(V(1) + V(0)\big), 0.5 \cdot \big(V(0) + V(-1)\big)\right\} = 10$
- $V(4) = \max\left\{0.5 \cdot \big(V(2) + V(1)\big), 0.5 \cdot \big(V(1) + V(0)\big)\right\} = 20$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\left\{0.5 \cdot \big(V(1) + V(0)\big), 0.5 \cdot \big(V(0) + V(-1)\big)\right\} = 10$
- $V(4) = \max\left\{0.5 \cdot \big(V(2) + V(1)\big), 0.5 \cdot \big(V(1) + V(0)\big)\right\} = 20$
- $V(5) = \max\left\{0.5 \cdot \big(V(3) + V(2)\big), 0.5 \cdot \big(V(2) + V(1)\big)\right\} = 20$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\left\{0.5 \cdot \big(V(1) + V(0)\big), 0.5 \cdot \big(V(0) + V(-1)\big)\right\} = 10$
- $V(4) = \max\left\{0.5 \cdot \big(V(2) + V(1)\big), 0.5 \cdot \big(V(1) + V(0)\big)\right\} = 20$
- $V(5) = \max\left\{0.5 \cdot \big(V(3) + V(2)\big), 0.5 \cdot \big(V(2) + V(1)\big)\right\} = 20$
- $V(6) = \max\left\{0.5 \cdot \big(V(4) + V(3)\big), 0.5 \cdot \big(V(3) + V(2)\big)\right\} = 15$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\left\{0.5 \cdot \big(V(1) + V(0)\big), 0.5 \cdot \big(V(0) + V(-1)\big)\right\} = 10$
- $V(4) = \max\left\{0.5 \cdot \big(V(2) + V(1)\big), 0.5 \cdot \big(V(1) + V(0)\big)\right\} = 20$
- $V(5) = \max\left\{0.5 \cdot \big(V(3) + V(2)\big), 0.5 \cdot \big(V(2) + V(1)\big)\right\} = 20$
- $V(6) = \max\left\{0.5 \cdot \big(V(4) + V(3)\big), 0.5 \cdot \big(V(3) + V(2)\big)\right\} = 15$
- $V(7) = \max\left\{0.5 \cdot \big(V(5) + V(4)\big), 0.5 \cdot \big(V(4) + V(3)\big)\right\} = 20$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\left\{0.5 \cdot \left(V(1) + V(0)\right), 0.5 \cdot \left(V(0) + V(-1)\right)\right\} = 10$
- $V(4) = \max\left\{0.5 \cdot \left(V(2) + V(1)\right), 0.5 \cdot \left(V(1) + V(0)\right)\right\} = 20$
- $V(5) = \max\left\{0.5 \cdot \left(V(3) + V(2)\right), 0.5 \cdot \left(V(2) + V(1)\right)\right\} = 20$
- $V(6) = \max\left\{0.5 \cdot \left(V(4) + V(3)\right), 0.5 \cdot \left(V(3) + V(2)\right)\right\} = 15$
- $V(7) = \max\left\{0.5 \cdot \left(V(5) + V(4)\right), 0.5 \cdot \left(V(4) + V(3)\right)\right\} = 20$
- $V(8) = \max\left\{0.5 \cdot \left(V(6) + V(5)\right), 0.5 \cdot \left(V(5) + V(4)\right)\right\} = 20$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\left\{0.5 \cdot \big(V(1) + V(0)\big), 0.5 \cdot \big(V(0) + V(-1)\big)\right\} = 10$
- $V(4) = \max\left\{0.5 \cdot \big(V(2) + V(1)\big), 0.5 \cdot \big(V(1) + V(0)\big)\right\} = 20$
- $V(5) = \max\left\{0.5 \cdot \big(V(3) + V(2)\big), 0.5 \cdot \big(V(2) + V(1)\big)\right\} = 20$
- $V(6) = \max\left\{0.5 \cdot \big(V(4) + V(3)\big), 0.5 \cdot \big(V(3) + V(2)\big)\right\} = 15$
- $V(7) = \max\left\{0.5 \cdot \big(V(5) + V(4)\big), 0.5 \cdot \big(V(4) + V(3)\big)\right\} = 20$
- $V(8) = \max\left\{0.5 \cdot \big(V(6) + V(5)\big), 0.5 \cdot \big(V(5) + V(4)\big)\right\} = 20$
- $V(9) = \max\left\{0.5 \cdot \big(V(7) + V(6)\big), 0.5 \cdot \big(V(6) + V(5)\big)\right\} = 17.5$
- $V(10) = \max\left\{0.5 \cdot \big(V(8) + V(7)\big), 0.5 \cdot \big(V(7) + V(6)\big)\right\} = 20$

Analyzing our game

$V(x) = \max \mathbb{E}[\text{Reward}]$ if round starts with x toothpicks

- $V(-1) = V(0) = 0$, $V(1) = V(2) = 20$. Want to find $V(10)$
- $V(3) = \max\{0.5 \cdot (V(1) + V(0)), 0.5 \cdot (V(0) + V(-1))\} = 10$
- $V(4) = \max\{0.5 \cdot (V(2) + V(1)), 0.5 \cdot (V(1) + V(0))\} = 20$
- $V(5) = \max\{0.5 \cdot (V(3) + V(2)), 0.5 \cdot (V(2) + V(1))\} = 20$
- $V(6) = \max\{0.5 \cdot (V(4) + V(3)), 0.5 \cdot (V(3) + V(2))\} = 15$
- $V(7) = \max\{0.5 \cdot (V(5) + V(4)), 0.5 \cdot (V(4) + V(3))\} = 20$
- $V(8) = \max\{0.5 \cdot (V(6) + V(5)), 0.5 \cdot (V(5) + V(4))\} = 20$
- $V(9) = \max\{0.5 \cdot (V(7) + V(6)), 0.5 \cdot (V(6) + V(5))\} = 17.5$
- $V(10) = \max\{0.5 \cdot (V(8) + V(7)), 0.5 \cdot (V(7) + V(6))\} = 20$

Optimal policy: Move to nearest multiple of 3

We always win if $x \neq 0 \mod (3)$

(Stochastic) Dynamic Programming

General solution paradigm for sequential decision making
Problem: $\max_{a:\text{“Actions”}} \mathbb{E}_X[f(a,X)]$

(Stochastic) Dynamic Programming

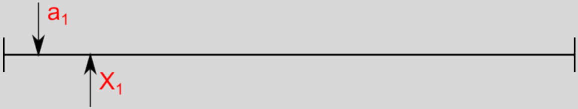
General solution paradigm for sequential decision making

Problem: $\max_{a:\text{“Actions”}} \mathbb{E}_X[f(a, X)]$

(Stochastic) Dynamic Programming

General solution paradigm for sequential decision making

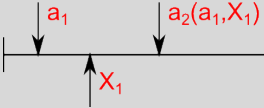
Problem: $\max_{a:\text{“Actions”}} \mathbb{E}_X[f(a,X)]$



(Stochastic) Dynamic Programming

General solution paradigm for sequential decision making

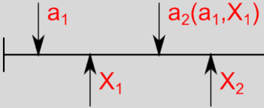
Problem: $\max_{a:\text{“Actions”}} \mathbb{E}_X[f(a,X)]$



(Stochastic) Dynamic Programming

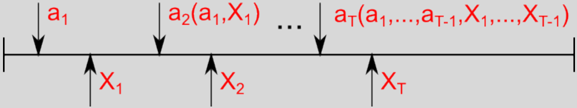
General solution paradigm for sequential decision making

Problem: $\max_{a:\text{“Actions”}} \mathbb{E}_X[f(a,X)]$



(Stochastic) Dynamic Programming

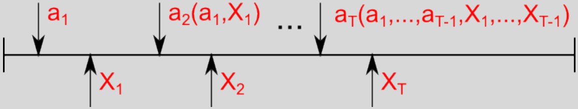
General solution paradigm for sequential decision making
Problem: $\max_{a:\text{"Actions"}} \mathbb{E}_X[f(a,X)]$



(Stochastic) Dynamic Programming

General solution paradigm for sequential decision making

Problem: $\max_{a:\text{"Actions"}} \mathbb{E}_X[f(a,X)]$

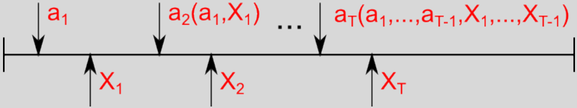


Main Ideas

(Stochastic) Dynamic Programming

General solution paradigm for sequential decision making

Problem: $\max_{a:\text{“Actions”}} \mathbb{E}_X[f(a,X)]$

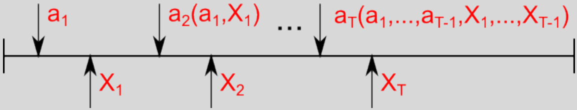


Main Ideas

- State: S - summary of history

(Stochastic) Dynamic Programming

General solution paradigm for sequential decision making
Problem: $\max_{a:\text{"Actions"}} \mathbb{E}_X[f(a,X)]$



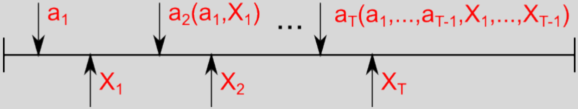
Main Ideas

- State: S - summary of history
- Value function: $V(\cdot)$ - 'value-to-go' for given state)

(Stochastic) Dynamic Programming

General solution paradigm for sequential decision making

Problem: $\max_{a:\text{“Actions”}} \mathbb{E}_X[f(a,X)]$



Main Ideas

- State: S - summary of history
- Value function: $V(\cdot)$ - 'value-to-go' for given state)
- Bellman Equation (or DP equation):

$$V(S_t) = \max_{a_t:\text{actions}} \left\{ R_t(S_t, a_t) + V\left(S_{t+1}(S_t, a_t)\right) \right\}$$