water filling

Joshua Talbot

November 2025

1 The CASE $\ell = 1$

For the water-filling method, in the case where $\ell = 1$,

$$value(p^i) = \frac{i - t}{\displaystyle\sum_{j=1}^{i} \frac{1}{v_j}}$$

$$MaximumValue = \max_{t+1 \leq i \leq n} \Big(value(p^i) \Big)$$

Where i refers to a specific box

p^i is the probability that box i is selected.

For example, in the puzzle posited by the paper with values $8, 7, 5, 4$, and $t = 1$:

Testing $i = 2$: $\frac{2-1}{\frac{1}{8}+\frac{1}{7}} \approx 3.73$

Testing $i = 3$: $\frac{3-1}{\frac{1}{8}+\frac{1}{7}+\frac{1}{5}} \approx 4.27$.

Testing $i = 4$: ≈ 4.18.

From this, we determine that we should randomize over the first 3 boxes, and ignoring the forth in order to maximize potential value.

2 The Case $\ell > 1$

For each box i, let $p'_i = \mathbb{P}_{S \sim p}(i \in S)$ be the probability that each box i is included in the final set S.

The set of marginal probabilities $p' = (p'_1, ...p'_n)$ forms a pseudo-distribution $p' \in \Delta_\ell([n])$.

Each p'_i is between 0 and 1.

The sum of all marginals is exactly ℓ.

3 Pseudocode for water_filling

function *water_filling*(values, t, ℓ):
  $n \leftarrow length(values)$
  if $t \geq \ell \wedge (l = n)$, return values
  if $l = 1$:
    $max\_val, optimal\_p\_prime \leftarrow solve\_l1(values, t, n)$
  else:
    $max\_val, optimal\_p\_prime \leftarrow solve\_l(values, t, n)$
  return $max\_val, optimal\_p\_prime$

4 Pseudocode for solve_l1

function *solve_l1*(values, t, ℓ):
  $max\_value \leftarrow 0$
  $optimal\_i \leftarrow t$
  $inverse\_v\_sum = 0$
  for $(i = t+1,\ i = n+1,\ i++)$:
    $inverse\_v\_sum \leftarrow inverse\_v\_sum + \frac{1}{v[i-1]}$
    $current\_value \leftarrow \frac{i-t}{inverse\_v\_sum}$
    if current_value > max_value:
      max_value = current_value
      optimal_i = i
  if $max\_value > 0$:
    $p\_prime \leftarrow [0, ..., 0]$ (of length n)
    $C \leftarrow \sum_{j=0}^{optimal\_i} (\frac{1}{v_j})$
    $final\_p \leftarrow \sum_{j=0}^{n} (\frac{1}{v_j} \cdot \frac{1}{C})[j < optimal\_i]$ (using anderson notation)
    return $max\_value, final\_p$
  else return $0, [0, ...0]$