BUILDING A GOOD CACHE BY COMBINING THE INITIAL SEGMENTS OF TWO LISTS

<table>
<thead>
<tr>
<th>C</th>
<th>T1</th>
<th>T2</th>
</tr>
</thead>
</table>

\[ 1T_1 + 1T_2 = C \]

ARING: HEURISTIC FOR SETTING A GOAL COMBINATION
- NO REFETCHING
  ADJUST CACHE TOWARDS GOAL
- W. REFETCHING
  SET CURRENT CACHE TO GOAL

RELATIVE LOSS BOUNDS

A SEQUENCES OF REQUEST S

\[ M_{\text{alg}}(S) \leq \alpha \cdot M^{*}(S) + \text{SMALL ADDITIONAL TERMS} \]

IDEALLY 1
ONE EXPERT PER COMBINED CACHE

LOSES ∈ [0, 13]  HIT OR MISS

PROBLEM:
- How do we produce a cache from the weight vector over experts?
- Can't use weighted average

\[ \bar{w}_t, \bar{x}_t \]

↑ Binary, only available after request
n+1 weights \( w_0, \ldots, w_n \) on gaps

**Idea 1:** \( \hat{y}_t = \sum_{i=0}^{n} w_{t,i} \cdot i \) \text{ MEAN}

- Why \( i \)

- When miss, need to half of total weight by \( \beta = e^{-\mu} \)

\[
\begin{align*}
w_0 & \quad w_1 & \quad w_2 & \quad w_3 & \quad w_4 & \cdots & \quad w_n \\
0 & \quad 2 & \quad \Delta & \quad 3 & \quad 4 & \cdots & \quad n
\end{align*}
\]

**Gaps**

**Mean** \( \alpha = \sum_{i=0}^{n} w_{t,i} \cdot i \)

---

Left \text{ TOTAL} \quad Right \text{ TOTAL}

\[
\text{Not necessarily} \quad 50 \: 50 \quad \text{Split} \]
Idea 2: Predict w, weighted median

\[ \frac{w_0 \cdot w \cdot w_2 \cdot \ldots \cdot w_{i-1} \cdot w_i \cdot w_{i+1} \cdot w_{n-1} \cdot w_n}{2} \]

\[ \frac{1}{2} \quad \text{MEDIAN} \]

\[ \text{MEDIAN} \quad \text{MEDIAN} \]

EITHER MEDIAN UPWARDS OR DOWNWARDS MULTIPLIED BY \( \frac{1}{2} \)

Bounds work fine for det. case

\[ M_A \leq 2 M_k + O\left( \sqrt{k} \ln n + \ln n \right) \]

Ideally median does not move too fast?
RANDOMIZED HEDGE ALG:
- \( i \sim w_i \)
- CONSTANT OF 1 IN FRONT OF \( M^* \)
- TOO MUCH REFETCHING

OPEN: IS CONSTANT OF 1 POSSIBLE W. NO REFETCHING, I.E.

\[
E(M_A) \leq 1 \cdot M^* + O(\sqrt{M^* \ln n + 2n \ln n})
\]

\[\sum_i \text{INITIAL SEGMI} = n\]

\[O(n^{u-1})\] EXPERTS

HEDGE ALG. W. ONE WEIGHT PER COMPOSITE CACHE
- DYNAMIC PROGR. \( O(kn^2) \) - FANCY \( O(k\ln n) \)
- TOO MUCH REFETCHING
- EXPENSIVE DISRUPTION

FOR ANY DETERM. ALG, \# MISSES CAN BE \( \geq k \cdot M^* \)