

Introduction of Topic

Topic: Trying to decide which expert to listen to try to minimize the regret. Algorithms where we want to minimize our regret - which is how well we performed compared to the best expert.

N experts all offering advice over some time period $\{1, \dots, t\}$. Every time period, we choose some expert to believe.

Goal: Minimize how often we take the “wrong” advice.

1 Algorithm 1

Strategy: Every round, “forget” forever every expert who was ever wrong.

Example We have experts $[A, B, C]$

$t = 1$, we choose A. A is wrong, so we remove them. Now our pool consists of $[B, C]$

$t=2$, we choose C. C is right, so we don't change the pool. Our pool is $[B, C]$

$t=3$, we choose B. B is wrong, so we remove them. Now our pool is $[C]$.

$t=4$, we choose C again. C is wrong, so we remove them.

Because now we have no experts left, we reset the pool by putting everyone back in and repeat the process over again.

Question If there is an expert who is never wrong, how many times could I be wrong?

Answer: Every time I'm wrong, some “expert” was wrong, and they are deleted from the pool. This can happen at most $N - 1$ times.

This algorithm seems a little limited, we want to try to expand it.

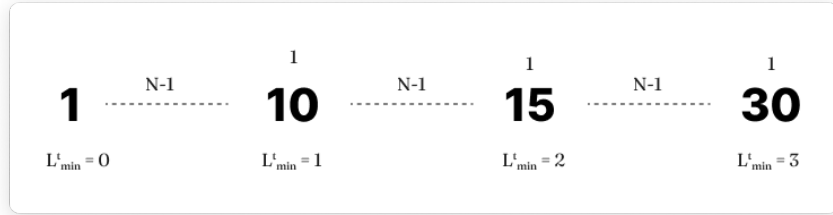
2 Algorithm 2

Strategy: Favor the people who are wrong less, but only pick among those who have been wrong the least.

1. Let L_i^t be the number of times expert i was wrong by time t .
2. Let $L_{min}^t = \min_i L_i^t$ be the number of times the best expert has been wrong.
3. Let S^t be the pool of experts who have been wrong L_{min}^t number of times.
4. We are only picking from S^t at a time t .

How many times will I be wrong over t time steps compared to L_{min}^t ?

Ideal: Break-up the time in blocks based on when L_{min}^t increases.



Our cost:

$$L_{us}^t = N \cdot L_{min}^t + N \cdot 1 (\text{i.e. the number of times we are wrong})$$

Ideally I would want the N in $N \cdot L_{min}^t$ be much smaller, but can't be done deterministically i.e. non-randomly.

Example. 3 choices: A,B,C every round, 1 or more is revealed to be correct.

$t = 1$ all appears the same so $S^1 = \{A, B, C\}$. At time 1, A is wrong but B and C were right. Thus:

$$L_A^2 = 1, L_B^2 = 0 = L_C^2$$

$t = 2 : S^2 = \{B, C\}$ At time 2, A and C were right so:

$$L_A^3 = 1, L_B^3 = 1, L_C^3 = 0$$

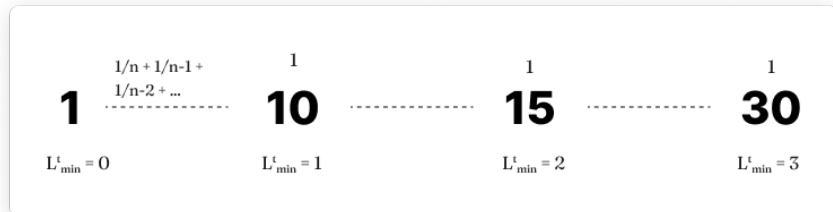
$t = 3 : S^3 = \{C\}$

Alg 2 felt bad because in our bounding, one expert in S^t was wrong and we had accidentally chose them every round.

Idea: Do Alg 2, but use *randomness*.

3 Algorithm 3

Strategy: Similar to Algorithm 2 (2) but pick from S_t uniformly at random, ie. pick from these with equal probability.



If k people were wrong then the probability I chose the wrong answer is

$$\frac{k}{|S^t|} = k \cdot \left(\frac{1}{|S^t|} \right) = \frac{1}{|S^t|} + \frac{1}{|S^t|} + \frac{1}{|S^t|} + \dots \leq \frac{1}{|S^t|} + \frac{1}{|S^t| - 1} + \dots + \frac{1}{|S^t| - k}$$

$$\Leftrightarrow \frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \cdots + \frac{1}{1} \leq \ln n + 1$$

$$S_0 \cdot L_{us}^t \leq (\log n + 1)L_{min}^t + \log n$$

Example for Algorithm 3:

Again, we have 3 choices: A, B, and C at every round.

$t = 1$ all appear the same so $S^1 = \{A, B, C\}$. At time 1, let's use the same outcome - A is wrong, but B and C were right. We choose uniformly at random from A, B, and C. Let's assume we chose B. Thus, we were correct this time.

For time 2:

$$L_A^2 = 1, L_B^2 = 0 = L_C^2$$

$t = 2$: $S^2 = \{B, C\}$ At time 2, A and C were right. We choose uniformly at random from B and C. Let's assume we chose C. Thus, we were correct again.

For time 3:

$$L_A^3 = 1, L_B^3 = 1, L_C^3 = 0$$

$t = 3$: $S^3 = \{C\}$. Since there's only one expert left, we choose C. The outcome depends on whether C is correct or not, but given the way this example has progressed, C has been consistently right and hence we could be correct again.

By incorporating randomness, Algorithm 3 provides a chance to avoid a situation where one might consistently pick the wrong expert even if that expert is part of the best performing set S^t . Instead, by choosing uniformly at random from the set of best-performing experts, we give ourselves the chance to spread out our risks and possibly perform better on average over time.

Observation: It feels too limiting to pick only from S^t

4 Algorithm 4

Strategy: In Algorithm 3, S_t was too limiting. Assign each expert a weight w_i^t , where at the start they are all the $w_i^1 = 1$ and:

$$w_i^{t+1} = \begin{cases} w_i^t & \text{if they are right} \\ (1 - \eta)w_i^t & \text{if they are wrong, where } 0 < \eta \leq 1/2 \end{cases}$$

we will select expert i with probability

$$p_i^t = \frac{w_i^t}{W^t}$$

where W^t is the combined weight of everyone.

$$\begin{aligned}
W^{t+1} &= \sum_{\text{correct experts } i} w_i^{t+1} + \sum_{\text{wrong experts } i} w_i^{t+1} \\
&= \sum_{\text{correct experts } i} w_i^t + (1 - \eta) \sum_{\text{wrong experts } i} w_i^t \\
&= \sum_{\text{correct experts } i} w_i^t + \sum_{\text{wrong experts } i} w_i^t - \eta \sum_{\text{wrong experts } i} w_i^t \\
&= W^t - \eta \sum_{\text{wrong experts } i} w_i^t
\end{aligned}$$

“Expected cost to us” = Probability of us being wrong at time t

$$\begin{aligned}
\text{“Expected cost to us”} &= \sum_{\text{wrong experts } i} p_i^t \\
&= \frac{\sum_{\text{wrong experts } i} w_i^t}{W^t} \\
&:= F^t
\end{aligned}$$

$$\begin{aligned}
W_{t+1} &= W^t - \eta F^t W^t \\
&= W^t (1 - \eta F^t) \\
&= W^{t-1} \cdot (1 - \eta F^t) (1 - \eta F^{t-1}) \\
&= W^{t-2} \cdot (1 - \eta F^t) (1 - \eta F^{t-1}) (1 - \eta F^{t-2}) \\
&= W^1 \prod_{i=1}^t (1 - \eta F^i) \\
&= N \prod_{i=1}^t (1 - \eta F^i)
\end{aligned}$$

The combined weight of everyone is at greater or equal than the weight of any expert, including the best expert. Keep in mind that getting to a logarithmic cost from a linear cost is a **huge** improvement.

$$W^{t+1} \geq \max_i w_i^{t+1} = (1 - \eta)^{L_{min}^t}$$

$$(1 - \eta)^{L_{min}^t} \leq N \prod_{i=1}^t (1 - \eta F^i)$$

Applying $\ln$ on both sides we get:

$$L_{min}^t \ln(1 - \eta) \leq \ln N + \sum_{i=1}^t \ln(1 - \eta F^i)$$

Keep in mind:

$$\ln A \cdot B = \ln A + \ln B$$

$$\ln A^B = B \ln A$$

and

$$\ln 1 - x \leq -x$$

$$-\ln(1 - x) \leq x + x^2, \text{ for } 0 < x \leq \frac{1}{2}$$

Thus, we can see that:

$$L_{min}^t \ln(1 - \eta) \leq \ln N + \sum_{i=1}^t \ln(1 - \eta F^i)$$

$$\begin{aligned} L_{min}^t &\geq \frac{\ln N}{\ln(1 - \eta)} + \frac{1}{\ln(1 - \eta)} \sum_{i=1}^t \ln(1 - \eta F^i) \\ &\geq \frac{\ln N}{\ln(1 - \eta)} + \frac{1}{\ln(1 - \eta)} \eta \sum_{i=1}^t F^i \\ &\geq \frac{\ln N}{\eta} - L_{us}^t \end{aligned}$$

$$\begin{aligned} L_{us}^t &\leq -L_{min}^t \frac{\ln(1 - \eta)}{\eta} + \frac{\ln N}{\eta} \\ &\leq L_{min}^t \frac{\eta + \eta^2}{\eta} + \frac{\ln N}{\eta} \\ &= L_{min}^t(1 + \eta) + \frac{\ln N}{\eta} \end{aligned}$$

If we take:

$$\eta = \min\left(\frac{1}{2}, \sqrt{\frac{\ln N}{T}}\right)$$

Then, by substituting η in the equation mentioned above, we get that:

$$L_{us}^t \leq L_{min}^t + L_{min}^t \sqrt{\frac{\ln N}{T}} + \frac{\ln N}{\sqrt{\frac{\ln N}{T}}}$$

and since $L_{min}^t \leq t$, this gives us:

$$L_{us}^t \leq L_{min}^t + 2\sqrt{t \ln(N)}$$