Normal-Hedge
The Hedge Algorithm

[Freund & Schapire 1997]

based on [Littlestone and Warmuth 1989, the Weighted Majority Algorithm]

Initial weights: \( w^1 = \left\langle \frac{1}{N}, \ldots, \frac{1}{N} \right\rangle \)

Weights update rule: \( w_{i}^{t+1} = w_{i}^{t} e^{-\eta \ell_{i}^{t}} \)

Alternatively: \( w_{i}^{t+1} = \frac{1}{N} e^{-\eta L_{i}^{t}} \neq \frac{1}{N} \prod_{s=1}^{t} p_{i}^{s}(x^{s}) \)

Learning rate

Not Bayes!
Potential-based bound

Potential: \( W^t = \sum_{i=1}^{N} w_i^t \)

Theorem: \( L_A^T \leq \frac{-\log W^{T+1}}{1 - e^{-\eta}} \)
Tuning the learning rate

\[ \forall i, L^T_A \leq \frac{\eta L^T_i + \ln N}{1 - e^{-\eta}} \]

If we set \( \eta = \sqrt{\frac{2 \ln N}{T}} \)

Then we guarantee \( L^T_A \leq \min_i L^T_i + \sqrt{2T \ln N} + \ln N \)

Equivalently \( \forall i, R^T_i \leq \sqrt{2T \ln N} + \ln N; \lim_{T \to \infty} \frac{\sqrt{2T \ln N} + \ln N}{T} = 0 \)

Achieved our goal!
Tuning the learning rate

∀i, \( L_A^T \leq \frac{\eta L_i^T + \ln N}{1 - e^{-\eta}} \)

If we set \( \eta = \sqrt{\frac{2 \ln N}{T}} \)

Then we guarantee \( L_A^T \leq \min_i L_i^T + \sqrt{2T \ln N} + \ln N \)

Equivalently \( \forall i, R_i^T \leq \sqrt{2T \ln N} + \ln N \); \( \lim_{T \to \infty} \frac{\sqrt{2T \ln N} + \ln N}{T} = 0 \)

Can we do better?
Lower bound

Each instantaneous loss $l_i^t$ is chosen IID -1/+1 with prob $\frac{1}{2}, \frac{1}{2}$

Cumulative loss defines a random walk.

Optimal weighting is always uniform.
Optimal cumulative loss is always zero.
Lower bound

Each instantaneous loss $l_i^t$ is chosen IID -1/+1 with prob $\frac{1}{2}, \frac{1}{2}$

Cumulative loss defines a random walk.

Optimal weighting is always uniform.
Optimal cumulative loss is always zero.

With high probability one of the $N$ actions has cumulative loss smaller than $-\sqrt{2T \ln N}$
The problem with Hedge

If we set \( \eta=\sqrt{\frac{2\ln N}{T}} \)

Then we guarantee \( L_A^T \leq \min_i L_i^T + \sqrt{2T \ln N} + \ln N \)

Different setting of \( \eta \) for different \( T \)
N is not a real parameter

• If N actions consist of M groups, where in each group the behavior is identical, we want the bound to depend on M, not on N.

• If there uncountably many actions, we want a bound that depends on the fraction of actions that perform well.

• We want an algorithm with the optimal performance guarantee uniformly for N and for T.
ε-quantile instead of N

Instead of regret relative to best action, compare performance to the best ε-quantile i.e. \( L_\varepsilon \) s.t. for \( \varepsilon \) fraction of the actions \( L_\theta < L_\varepsilon \)

For Hedge we get:

\[
W^t = \int_{[0,1]} e^{-L^t_\theta} \, dW(\theta) \geq \int_{\theta: L^t_\theta \leq L_\varepsilon} e^{-L^t_\theta} \, dW(\theta) \geq w(\theta : L^t_\theta \leq L_\varepsilon) e^{-L_\varepsilon}
\]

If we set \( \eta = \sqrt{\frac{-2 \ln \varepsilon}{T}} \)

Then we guarantee \( L^T_A \leq L_\varepsilon + \sqrt{-2T \ln \varepsilon} - \ln \varepsilon \)

But we don’t know either \( \varepsilon \) or \( T \) a-priori, so we don’t know how to set \( \eta \)
The NormalHedge potential

Potential: \( \psi(r,c) = \begin{cases} 
\exp\left(\frac{r^2}{2c}\right) & \text{if } r \geq 0 \\
1 & \text{if } r \leq 0
\end{cases} \)

Weight: \( w(r,c) = \frac{\partial}{\partial r} \psi(r,c) = \begin{cases} 
\frac{r}{c} \exp\left(\frac{r^2}{2c}\right) & \text{if } r \geq 0 \\
0 & \text{if } r \leq 0
\end{cases} \)
NormalHedge algorithm

for t=0,1,2,...

if $\forall i, R_i^t \leq 0$ then $w_i^t = 1 / N$

else

set $c(t)$ so that $\frac{1}{N} \sum_{i=1}^{N} \psi(R_i^t, c(t)) = e$

$w_i^t = w(R_i^t, c(t))$

Incur instantaneous losses: $\langle l_1^t, l_2^t, ..., l_N^t \rangle$

Algorithm loss: $l_A^t = \frac{\sum_{i=1}^{N} w_i^t l_i^t}{\sum_{i=1}^{N} w_i^t}$

Update regrets: $R_i^{t+1} = R_i^t + l_A^t - l_i^t$
Illustrative Example

\[ \begin{align*}
\text{Expert} & \quad \exp(\eta G) \\
\text{Algorithm} & \quad \begin{cases}
\exp\left(\frac{R^2}{2c}\right) & \text{if } R \geq 0 \\
1 & \text{if } R \leq 0
\end{cases}
\end{align*} \]
Illustrative Example

\[
\begin{align*}
\text{Expert} & \quad \exp(\eta G) \\
\text{Algorithm} & \quad \begin{cases}
\exp\left(\frac{R^2}{2c}\right) & \text{if } R \geq 0 \\
1 & \text{if } R \leq 0
\end{cases}
\end{align*}
\]
Illustrative Example

\[ \text{exp}(\eta G) \]

\[ \begin{cases} \exp\left(\frac{R^2}{2c}\right) & \text{if } R \geq 0 \\ 1 & \text{if } R \leq 0 \end{cases} \]

iteration \( t+1 \)

Cumulative Gain

potential
Illustrative Example

\[ \text{exp}(\eta G) \]

\[ \begin{cases} 
\exp \left( \frac{R^2}{2c} \right) & \text{if } R \geq 0 \\
1 & \text{if } R \leq 0 
\end{cases} \]

\( \text{calc } c(t+1) \)

Cumulative Gain

potential
Normal-Hedge Performance bound

[Chaudhuri, Freund & Hsu 2009]

The regret of NormalHedge is upper bounded by

\[ O\left(\sqrt{T \ln N + \ln^3 N}\right) \]
Performance on flip-flop
Combining experts, the binary prediction case

• Goal is to predict a binary sequence, making as few mistakes as possible.

• There are $N$ experts.

• All predictions are binary and deterministic.

• A-priori knowledge: there is an expert that never makes more than $k$ mistakes.

• $k=0$ corresponds to the halving algorithm.
Combining experts as a drifting game

[ Cesa-Bianchi, Freund, Helmbold, Warmuth 96 ]

Binary instantaneous loss $l_i^t, l_A^t \in \{0, 1\}$

Bin $s$ contains all experts for which $L_i^t = s$
The game lattice

$L^t_i$

Tuesday, January 19, 2010
Initial configuration $t=1$
Experts predictions $t=1$

Algorithm predicts with majority = 0

Suppose algorithm is wrong

Predict 0

Predict 1

Tuesday, January 19, 2010
configuration at $t=2$
Experts predictions $t=2$

How should the algorithm predict? 0 or 1?
configuration $t=3$

Prediction is 0 and outcome is 1

$L_i^3$
Experts predictions $t=2$

How should the algorithm predict? 0 or 1?
configuration $t=3$

Prediction is 1 and outcome is 0
If an error will lead to this configuration then an error is not possible
\[ \Rightarrow \] this is a safe prediction

Algorithm’s goal is to get to an illegal configuration with the smallest number of mistakes.
Helping the adversary.

- Assume that the set of experts is continuous, arbitrarily divisible.
- a-priori knowledge: $\frac{1}{N}$ fraction of the expert “mass” have cumulative loss at most $k$
- Find algorithm with the tightest uniform upper bound on the cumulative loss.
If an error will lead to this configuration then an error is not possible
\[\Rightarrow\] this is a safe prediction

Algorithm's goal is to get to an illegal configuration with the smallest number of mistakes.

illegal configuration

Total volume < $\frac{1}{N}$

Tuesday, January 19, 2010
An optimal adversarial strategy

- Split each bin to two equal parts. Algorithm’s prediction is always incorrect.
- Equivalently: predictions of each expert are IID 0,1 with probabilities 1/2, 1/2
An optimal adversarial strategy

- Split each bin to two equal parts. Algorithm’s prediction is always incorrect.

- Equivalently: predictions of each expert are IID 0,1 with probabilities 1/2, 1/2

- Same adversarial strategy was used to prove general lower bound on BLG

Link to lower bound
Optimal prediction strategy

- Assume that adversary will play optimally from the next iteration until the end of the game.
- Choose as the next configuration the one that would end the game faster.
- Relevant only when adversary plays sub-optimally, when adversary plays optimally the two next configurations are identical.
Potentials

- **potential for bin $i$** = the fraction of the experts in the bin that will have $<k$ mistakes in $r$ iterations.
  \[
  \psi(i,r) = \text{Binom}(k - i, r); \text{Binom}(l, m) = \frac{1}{2^m} \sum_{j=0}^{l} \binom{m}{j}
  \]

- $f(i)$ = the fraction of the experts currently in bin $i$

- **potential for a configuration** = weighted sum of bin potentials.
  \[
  \Psi(\text{configuration}) = \sum_{j=1}^{k} f(i) \psi(i,r)
  \]
properties of the potential

\[ \psi(i,r) = \frac{\psi(i,r-1) + \psi(i+1,r-1)}{2} \]

**End of game**

\[ \psi(i,0) = \begin{cases} 
1 & i \leq k \\
0 & i > k 
\end{cases} \]

Illegal configuration:

\[ \Psi(\text{configuration}) = \sum_{j=1}^{k} f(i) \psi(i,0) < \frac{1}{N} \]

**Beginning of game**

Number of errors if adversary always plays optimally

\[ r - 1, \text{ where } r \text{ is the smallest integer for which} \]

\[ \psi(0,r) < \frac{1}{N} \]
BW Prediction algorithm

- **Initialization**: set $r$ to be the number of errors against optimal adversary.
- Given expert predictions: choose prediction that will result (assuming error) in a lower-potential configuration.
- Decrease $r$ if possible.
Main properties

• If algorithm is followed, the potential of the configuration never increases - is always $\leq 1/N$

• Algorithm is min/max optimal.
  
  • Removing assumption that expert set is divisible min/max optimality holds if $N > 2^k$
  
  • Based on relation to Ulam’s game with k lies [Spencer 92]
Alternative Representation

The difference between the two configurations can be represented as a weighted sum

\[ \Psi(\text{configuration 1}) - \Psi(\text{configuration 0}) = \sum_{j=0}^{k} f(i)w(i,r) \]

\[ w(i,r) = \psi(i + 1, r - 1) - \psi(i, r - 1) = \frac{1}{2^{r-1}} \binom{r-1}{k-i} \]

The optimal prediction is according to the sign of this weighted sum.
The BW algorithm

• Better error bound than exponential weights.
• A-priori assumption that one of the experts has loss at most $k$, we want a bound on the regret without any a priori assumptions.
• Instantaneous loss is restricted to $\{0, 1\}$, we want it to be any number in $[-1, +1]$. 
Design of NormalHedge

- BW: potential function depends on loss and number of remaining mistakes
- Normal-Hedge: Potential function based on regret and variance of the positive regrets
What next?

• I came up with the NormalHedge algorithm by considering the continuous time limit.

• The discrete-time proof is very technical and gives little insight.

• In the continuous time limit, the analysis is simple and insightful and the bound is much tighter.