PWEA (WMA & RWMA)

Lecturer: Kris Kitani   Scribes: Roshini Rajesh Kannan , Moneish Kumar (Group 1)

1 Review

In the previous lecture, we saw the PWEA greedy algorithm, halving algorithm and randomized greedy algorithm along with an analysis of their mistake bounds. All these algorithms had an assumption of realizability. But, when these assumptions are violated, these algorithms could break. The concept of regret, a new special case mistake bound was introduced that would compare the performance of the algorithm with the best hypothesis. We would like an algorithm whose average regret is bounded as T goes to infinity and for this the regret must grow sub-linearly. In this lecture, we relax the realizability assumption for the first time. We will see two algorithms: Weighted Majority Algorithm and Random Weighted Majority Algorithm in this lecture along with their mistake bound calculation.

1.1 Prediction with Expert Advice

The problem of Prediction with Expert Advice(PWEA) [2] is a type online learning problem where the algorithm makes prediction based on advice from experts. The characteristics of PWEA are:

1. One-Shot: The data arrives as a stream and the outcome of one stage doesn't affect the next

2. Instructive: Since a fully observable loss is received, it is instructive

3. Exhaustive: The action and state space are both finite and can be fully accessed.

1.2 Important Definitions

Definition 1. Realizability Assumption that a perfect hypothesis exists.

Previously under the assumption of realizability we calculated mistake bound which tells upper bound on the number of mistakes. We had seen previously the Majority algorithm with halving , which maintains a set of hypothesis (the version space) consistent with past evidence, i.e. it predicts according to the majority vote of hypothesis from the version space. The mistake bound of the halving algorithm ($log_2 N$) was logarithmic in the number of hypotheses, since it removes at least half the hypotheses after one mistake.

However, we will now relax this realizability assumption and allow even the best expert to make mistakes. With the new condition of non-realizability, we will evaluate the performance of the algorithm in terms of the *regret bound* instead of the *mistake bound*. In such a case, we would like to design algorithms demonstrating small regrets. Following is the definition of regret below.

Definition 2. Regret of the learner, $R^{(T)}(H)$, is the cumulative loss for not following the best hypothesis in the hypothesis class H. Mathematically formulation of regret is:

$$R^{(T)}(H) = \sum_{t=1}^{T} l(\hat{y}^{(t)}, y^{(t)}) - \min_{h \in H} \sum_{t=1}^{T} l(h(x^{(t)}), y^{(t)}) \tag{1}$$

where, $\sum_{t=1}^{T} l(\hat{y}^{(t)}, y^{(t)})$ represents the cumulative loss of the learner, and $\min \sum_{t=1}^{T} l(h(x^{(t)}), y^{(t)})$ represents the loss of best single hypothesis.

In this lecture, we will use the regret bound to characterize online algorithms and conversely, we will be interested in finding good algorithms as characterized by this regret bound. We recall that in previous lectures, we had assumed at least one perfect solution, which implies realizability. However, in this lecture, the first algorithm we will consider is under the relaxed assumption of non-realizability, namely, the Weighted Majority Algorithm (WMA).

2 Summary

2.1 Weighted Majority Algorithm (WMA)

Our assumption of realizability is dismissed for the first time in the Weighted Majority Algorithm [1]. The Algorithm 1 below shows the steps in Weighted Majority Algorithm.

Algorithm 1 Weighted Majority Algorithm (WMA)

1: $\mathbf{w}^{(1)} \leftarrow \{w_n^{(1)} = 1\}_{n=1}^{N}$ ▷ Weights vector initialization
2: $\eta \leq \frac{1}{2}$ ▷ Penalty parameter initialization
3: **for** $t = 1, \cdots, T$ **do**
4:     RECEIVE $(\mathbf{x}^{(t)} \in \{-1, 1\}^N)$ ▷ Receive expert predictions
5:     $\hat{y}^{(t)} = \text{sign}\left(\sum_{n=1}^{N} x_n^{(t)} \cdot w_n^{(t)}\right) \in \{-1, 1\}$ ▷ Prediction by algorithm
6:     RECEIVE $(y^{(t)} \in \{-1, 1\})$ ▷ Receive true value
7:     $w_n^{(t+1)} \leftarrow w_n^{(t)}\left(1 - \eta \cdot \mathbf{1}[y^{(t)} \neq x_n^{(t)}]\right)$ ▷ Updating weights
8: **end for**

Let us derive the mistake bound and regret bound for WMA. For this we will be using the following two identities:

Lemma 3. *For $0 < x < 1$, following two inequalities hold (refer to appendix A detailed proof):*

$$-x - x^2 \leq \log(1 - x) \tag{2}$$
$$\log(1 - x) \leq -x \tag{3}$$

Theorem 4. *If $M^{(T)}$ is the number of mistakes made by the learner, $m_n^{(t)}$ is the mistake made by n^{th} expert till time t, N is the number of experts and η is the penalty parameter, then the mistake bound for WMA is:*

$$M^{(T)} \leq 2(1 + \eta)m_n^{(t)} + \frac{2 \log N}{\eta}$$

Proof. Defining the potential function $\Phi^{(t)}$ for WMA:

The potential function is an equivalent of the size of the version space, but is a non-integer.

$$\Phi^{(t)} = \sum_{n=1}^{N} w_n^{(t)} \tag{4}$$

The potential function here is the sum of weights of all experts at time t.

Upper bound potential:

According to WMA, atleast half of the experts would be mistaken and hence atleast half of the weights will be penalized by $(1 - \eta)$.

$$\Phi^{(t+1)} \le \Phi^{(t)} \left(\frac{1}{2} + \frac{1}{2}(1 - \eta) \right) \tag{5}$$

$$\Phi^{(t+1)} \le \Phi^{(t)} \left(1 - \frac{\eta}{2} \right) \tag{6}$$

By substituting t=1 and using induction we would get the following equation.At t=1, all experts will have weights initialised to 1 and $\Phi^{(1)}$ would equal to the number of experts N

$$\Phi^{(2)} \le \Phi^{(1)} \left(1 - \frac{\eta}{2} \right)^{M^{(1)}} = N \left(1 - \frac{\eta}{2} \right)^{M^{(1)}} \tag{7}$$

The lower bound function is derived using the fact that a single element is always smaller than or equal to the sum.

$$w_i(t) \le \phi^{(t)} = \sum_{n} w_n(t) \tag{8}$$

Using the update rule for the line 7 in algorithm, we get:

$$w_i^{(t+1)} \le \phi^{(t+1)} \tag{9}$$

$$(1 - \eta)^{(m_i^{(t)})} \le \phi^{(t+1)} \tag{10}$$

By combining Eq.(7) and Eq.(10), we can now derive the combined bound for WMA:

$$(1 - \eta)^{m_i^{(t)}} \le N \left(1 - \frac{\eta}{2} \right)^{M^{(t)}} \tag{11}$$

By performing the following algebraic operations, we obtain the mistake bound of WMA:

$$m_i^{(t)} \ln(1 - \eta) \le \ln N + M^{(t)} \ln \left(1 - \frac{\eta}{2} \right) \quad \text{(by applying natural log to Eq. (11))}$$

$$m_i^{(t)} (-\eta - \eta^2) \le \ln N + M^{(t)} \ln \left(1 - \frac{\eta}{2} \right) \quad (\because \text{ Lemma 3})$$

$$-m_i^{(t)} (\eta + \eta^2) \le \ln N + M^{(t)} \left(-\frac{\eta}{2} \right) \quad (\because \text{ Lemma 3})$$

$$M^{(t)} \left(\frac{\eta}{2} \right) \le \ln N + m_i^{(t)} (\eta + \eta^2)$$

$$M^{(t)} \le \frac{2 \ln N}{\eta} + 2 m_i^{(t)} (1 + \eta) \quad \forall n$$

Therefore, we have obtained the mistake bound for WMA. This is dependent on the performance of an expert unlike the previous algorithms. This happened as a result of no realizability. $\square$

Theorem 5. *The regret bound of WMA is:* $R_(h_n) \leq m_n^{(T)} + 2\eta m_n^{(T)} + \frac{2\ln N}{\eta}$

Proof. From the mistake bound we have:

$$M^{(t)} \leq 2m_i^{(t)}(1+\eta) + \frac{2\ln N}{\eta} \tag{12}$$

$$M^{(t)} \leq 2m_i^{(t)} + 2m_i^{(t)}\eta + \frac{2\ln N}{\eta} \tag{13}$$

By getting the $m_i^{(t)}$ from RHS to LHS, we get:

$$M^{(t)} - m_i^{(t)} \leq m_i^{(t)} + 2m_i^{(t)}\eta + \frac{2\ln N}{\eta} \tag{14}$$

$$R(h_n) \leq m_i^{(t)} + 2m_i^{(t)}\eta + \frac{2\ln N}{\eta} \tag{15}$$

After getting the regret bound now we need to analyse how this regret bound changes with time and check if WMA is a no-regret algorithm. A no-regret algorithm means that the average regret bound would tend to zero as time T tends to ∞. From Eq. (15) let us split the RHS as 3 terms, namely $m_i(T)$ as term 1, $2m_i(T)\eta$ as term 2 and $\frac{2\ln N}{\eta}$ as term 3. Term 1 and term 2 grow linearly over time T and have bound O(T). Term 3 has a constant time bound of O(1). Even with a suitable penalty parameter such as $\frac{1}{t}$ or $\frac{1}{\sqrt{t}}$ would only make the term 2 constant and sub-linear respectively. The term 1 remains linear irrespective of the penalty parameter. Since there is linear growth, WMA has a bounded or controlled regret but not no-regre. This leaves a scope of improvement that leads to Randomized Weighted Majority Algorithm (RWMA) discussed next.

2.2 Randomized Weighted Majority Algorithm (RWMA)

We have looked at Weighted Majority Algorithm (WMA) which had a bounded regret. This means that the error grows linearly with time. Natural question to ask is if we can do better than this ? Yes, in fact the only change that needs to be done is that we need to randomly sample the hypothesis space (in our case the Experts) biased on the weights of each one. Randomized Weighted majority Algorithm (RWMA) [1], listed as Algorithm 2 below. The only difference this algorithm has from the Weighted Majority Algorithm is in line 5 where a multinomial distribution is created using the weights and the learner prediction is done via sampling from this distribution. This algorithm is also known as Randomized Weighted Majority, Generalize Weighted Majority, Exponentiated Weighted Majority, Hedge Algorithm.

Algorithm 2 Randomized Weighted Majority Algorithm (RWMA)

1: $\mathbf{w}^{(1)} \leftarrow \{w_n^{(1)} = 1\}_{n=1}^N$ ▷ Weight initialization
2: $0 \leq \eta \leq 1$ ▷ Penalty rate initialization
3: **for** $t = 1, \cdots, T$ **do**
4:     Receive $(\mathbf{x}^{(t)} \in \{-1, 1\}^N)$ ▷ Receive experts predictions
5:     $I \sim \text{Multinomial}(\mathbf{w}^{(t)}/\Phi^{(t)})$, where $\Phi^{(t)} = \sum_{n=1}^N w_n^{(t)}$
6:     $\hat{y}^{(t)} = h_i(\mathbf{x}^{(t)})$ ▷ Make learner prediction via sampling
7:     Receive $(y^{(t)} \in \{-1, 1\})$ ▷ Receive actual answer
8:     $w_n^{(t+1)} \leftarrow w_n^{(t)}\big(1 - \eta \cdot \mathbf{1}[y^{(t)} \neq h_n(\mathbf{x})^{(t)}]\big)$ ▷ Weight update
9: **end for**

We will calculate the mistake bound for RWMA. We will also show that this mistake bound is better than WMA by a factor of 2. Which means that asymptotically RWMA is bound to make two times less mistakes as compared to WMA. Before calculating the mistake bound lets look at lemma that helps in the derivation of mistake bound.

Lemma 6. $e^x \geq 1 + x$ for all $x \in \mathbb{R}$.

Proof. The lemma can be proved using the taylor expansion of e^x. Refer to the Appendix B for the proof.

Theorem 7. *Let* $M^{(t)}, m_i^{(t)}$ *respectively be the number of mistakes that have been made by the RWMA learner and the i-th hypothesis until the time step t, and* N, η *be the number of experts and the penalty rate. Then, expected mistakes of learner* $\mathbb{E}[M^{(t)}]$ *is upper-bounded as:*

$$\mathbb{E}[M^{(t)}] \leq (1 + \eta)m_i^{(t)} + \frac{\log N}{\eta}$$

Proof. Step 1: define the potential function $\Phi^{(t)}$ for RWMA as the sum of weights:

$$\Phi^{(t)} = \sum_{n=1}^N w_n^{(t)} \tag{16}$$

Step 2: calculate upper bound - By the weight update rule of RWMA, $w_n^{(t+1)}$ can be analytically calculated from $w_n^{(t)}$ as follows:

$$w_n^{(t+1)} = \big(1 - \eta \cdot \mathbf{1}[y^{(t)} \neq y_n^{(t)}]\big)w_n^{(t)} = \big(1 - \eta\alpha_n^{(t)}\big)w_n^{(t)} \quad (\text{where } \alpha_n^{(t)} = \mathbf{1}[y^{(t)}) \neq y_n^{(t)}] \tag{17}$$

By combining Eq. (17) & (16), one can derive the mathematical relation between $\Phi^{(t+1)}$ and $\Phi^{(t)}$

as follows:

$$\Phi^{(t+1)} = \sum_n w_n^{(t+1)} = \sum_n (1 - \eta \alpha_n^{(t)}) w_n^{(t)} \quad (\because \text{Eq. (17)})$$

$$= \sum_n w_n^{(t)} - \sum_n \eta \alpha_n^{(t)} w_n^{(t)}$$

$$= \Phi^{(t)} - \frac{\Phi^{(t)}}{\Phi^{(t)}} \sum_n \eta \alpha_n^{(t)} w_n^{(t)}$$

$$= \Phi^{(t)} - \Phi^{(t)} \eta \sum_n \alpha_n^{(t)} \frac{w_n^{(t)}}{\Phi^{(t)}}$$

$$= \Phi^{(t)} - \Phi^{(t)} \eta \sum_n \alpha_n^{(t)} p_n^{(t)} \quad (\text{where } p_n^{(t)} = \frac{w_n^{(t)}}{\Phi^{(t)}})$$

$$= \Phi^{(t)} \left(1 - \eta \sum_n \alpha_n^{(t)} p_n^{(t)} \right) \tag{18}$$

Assuming all weights are initialized to 1, $\Phi^{(t+1)}$ can be calculated by recursively applying Eq. (18) as follows:

$$\Phi^{(T+1)} = \Phi^{(1)} \prod_{t=1}^{T} \left(1 - \eta \sum_n \alpha_n^{(t)} p_n^{(t)} \right) = N \prod_{t=1}^{T} \left(1 - \eta \sum_n \alpha_n^{(t)} p_n^{(t)} \right) \tag{19}$$

We can now derive the upper bound of $\Phi^{(T+1)}$ by combining Lemma 6 and Eq. (19) as follows:

$$\Phi^{(T+1)} = N \prod_{t=1}^{T} \left(1 - \eta \sum_n \alpha_n^{(t)} p_n^{(t)} \right)$$

$$\leq N \prod_{t=1}^{T} \exp\left(-\eta \sum_n \alpha_n^{(t)} p_n^{(t)} \right) \quad (\because \text{Lemma 6})$$

$$\leq N \exp\left(-\eta \sum_{t=1}^{T} \mathbb{E}\big[\mathbf{1}[y^{(t)} \neq \hat{y}^{(t)}]\big] \right) \quad (\because \sum_n \alpha_n^{(t)} p_n^{(t)} = \mathbb{E}_p\big[\mathbf{1}[y^{(t)} \neq \hat{y}^{(t)}]\big])$$

$$\leq N \exp\left(-\mathbb{E}[M^{(T)}] \right) \quad (\because \text{definition of } M^{(t)}) \tag{20}$$

Step 3: Lower bound calculation - As shown before in equation 8 , single element of the weight is always lesser than or equal to the sum.

$$w_i(t) \leq \phi^{(t)} = \sum_n w_n(t) \tag{21}$$

Using the update rule for the line 7 in algorithm, we get:

$$w_i^{(t+1)} \leq \phi^{(t+1)} \tag{22}$$

$$(1 - \eta)^{(m_i^{(T)})} \leq \phi^{(T+1)} \tag{23}$$

By combining Eq. (20) & (23), we can now derive the mathematical relation between $\mathbb{E}[M^{(t)}]$ and $m_n^{(t)}$ as follows:

$$(1 - \eta)^{m_n^{(T)}} \leq \Phi^{(T+1)} \leq N \exp\left(-\mathbb{E}[M^{(T)}]\right) \tag{24}$$

Step 4 and 5: Computing Upper bound of expected mistakes $\mathbb{E}[M^{(t)}]$ by combining the bounds of $\Phi^{(t+1)}$ (Eq. (24)) and Lemma 3 as follows:

$$m_n^{(T)} \log(1 - \eta) \leq \log N - \eta \mathbb{E}[M^{(T)}]$$
$$m_n^{(T)}(-\eta - \eta^2) \leq \log N - \eta \mathbb{E}[M^{(T)}] \quad (\because \text{ Lemma 3})$$
$$-m_n^{(T)}(1 + \eta) \leq \frac{\log N}{\eta} - \mathbb{E}[M^{(T)}]$$
$$\mathbb{E}[M^{(T)}] \leq (1 + \eta)m_n^{(T)} + \frac{\log N}{\eta}$$

Therefore, we have obtained the upper bound of expected mistakes for RWMA.

We are now interested in the performance that is the regret bound of RWMA algorithm.

Theorem 8. *RWMA is a no-regret algorithm.*

Proof. By the definition of regret and Theorem 7, we could obtain the upper bound of *expected regret* of RWMA, $\mathbb{E}[R]$, as follows:

$$\mathbb{E}[R] = \mathbb{E}[M^{(T)}] - m_n^{(T)} \leq \eta m_n^{(T)} + \frac{\log N}{\eta} \tag{25}$$

If we set η to $\frac{1}{\sqrt{T}}$, both $\eta m_n^{(T)}$ and $\frac{\log N}{\eta}$ follow $O(\sqrt{T})$.

Thus, $\frac{\mathbb{E}[R]}{T} = \left(\eta m_n^{(T)} + \frac{\log N}{\eta}\right) \propto \frac{1}{\sqrt{T}} \to 0$ as $T \to \infty$.

3 Discussions in class:

3.1 Why use regret and not loss ?

The idea of regret comes into picture when the loss computation is not trivial or changes over time, then the next best thing we can calculate is who is our algorithm compared to the best possible solution.

3.2 Why are we comparing the worst case regret of WMA with the expected regret in case of RWMA ?

We are looking are the performance of algorithms asymptotically i.e their behaviour as time goes to infinity and in the case of randomised algorithms this boils down to expectation.

References

[1] N. Littlestone and M. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

[2] M. L. Littman. Reinforcement learning improves behaviour from evaluative feedback. *Nature*, 521(7553):445–451, 2015.

A Proof of $-x - x^2 \leq \log(1-x)$ and $\log(1-x) \leq -x \; \forall x \in (0,1)$

The Taylor series expansion of log(1-x) is:

$$\log(1-x) = -x - \frac{x^2}{2} - \frac{x^3}{3} + \ldots$$

$$\log(1-x) \geq -x - x^2 \because \begin{cases} \frac{x^2}{2} \geq 0 (\text{ adding and subtracting } \frac{x^2}{2} \forall x \in (0,1)) \\ \sum_{k=1}^{\infty} -\frac{x^{2k+1}}{2k+1} - \frac{x^{2k+2}}{(2k+2)} \geq 0 \; \forall x \in (0,1) \end{cases} \tag{26}$$

$$\log(1-x) \leq -x \because \left\{ \sum_{k=1}^{\infty} -\frac{x^{2k}}{2k} - \frac{x^{2k+1}}{(2k+1)} \leq 0 \; \forall x \in (0,1) \right. \tag{27}$$

When x ranges from zero to one, -x will be subtracted by values further in the series and hence $\log 1 - x$ is smaller than $-x$. But, since these values are very small they wouldn't become greater than $-x - x^2$.

B Proof of $e^x \geq 1 + x \; \forall x \in [-\infty, \infty]$

The taylor expansion of e^x is

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

$$e^x = \frac{x^0}{0!} + \frac{x^1}{1!} + \frac{x^2}{2!} + \ldots$$

$$e^x \geq \frac{x^0}{0!} + \frac{x^1}{1!} \quad \because \begin{cases} \frac{x^k}{k!} \geq 0 \; \forall x \geq 0 \; \& \; k \geq 2 \\\\ \sum_{k=1}^{\infty} \frac{x^{2k}}{2k!} + \frac{x^{2k+1}}{(2k+1)!} \geq 0 \; \forall x \in (-1,0) \\\\ \sum_{k=2}^{\infty} \frac{x^{2k}}{(2k)!} + \frac{x^{2k-1}}{(2k-1)!} \geq 0 \; \& \; \frac{x^2}{(2)!} \geq 0 \; \forall x \in (-\infty, -1] \end{cases}$$