CSC 696H: Topics in Bandits and Reinforcement Learning Theory **Fall 2024**

Lecture 4: Online Learning: Exponential Weight Algorithm

Lecturer: Chicheng Zhang *Scribe: Rethvick Sriram Yugendra Babu*

1 Introduction

The Exponential Weight Algorithm (EWA) is a fundamental approach in online learning, where the goal is to make sequential decisions that minimize regret compared to the best-fixed predictor in hindsight. EWA maintains a weighted combination of predictors and updates these weights exponentially based on incurred loss. This algorithm is widely used in adversarial settings and serves as a building block for more complex algorithms in online learning and reinforcement learning.

2 Recap: Online to Batch Conversion Theorem

The theorem provides a method to convert an online learning algorithm with low regret into a batch learning algorithm with low expected loss.

Let O be an online learning algorithm with regret bound $\text{Reg}_o(F, T)$ and $L_D(\hat{f})$ denote the expected loss of a predictor $\hat{f}$:

$$L_D(\hat{f}) \leq L_D(\hat{f}) + B\sqrt{\frac{\ln \frac{1}{\delta}}{T}} + \frac{\text{Reg}_o(F,T)}{T},$$

Proof Outline:

- A union bound is applied to control the probability of events F_1 and F_2.

  The union bound is stated as:

  $$P(A \cup B) \leq P(A) + P(B)$$

  Union bounds allow one to argue that the chance that two highly likely events happen simultaneously is high:

  $$P(F_1) \geq 0.99,\, P(F_2) \geq 0.98 \implies P(F_1 \cap F_2) \geq 1 - 0.01 - 0.02.$$

- The result shows that as $T \to \infty$, the expected loss of the converted batch predictor approaches the loss of the best predictor in hindsight.

3 A Better Idea: Exponential Weight Algorithm (EWA)

The Exponential Weight Algorithm (EWA) provides a soft, weighted combination of models based on their performance. This method continuously updates the weights of predictors to favor models with lower incurred loss, offering a robust mechanism for online decision-making.

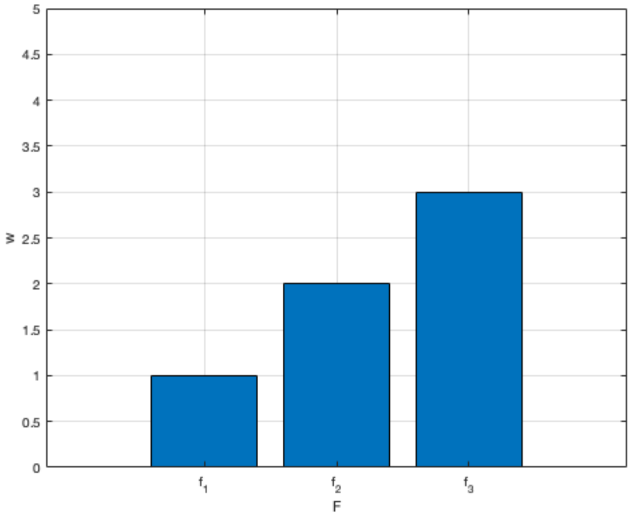

Figure 1: Bar graph illustrating the weight distribution over functions f_1, f_2, f_3

3.1 Algorithm Outline

- **Input:** Learning rate $\eta > 0$, prediction class $\mathcal{F}$.
- **Initialization:** $w_1(f) = 1$ for all $f \in \mathcal{F}$.

For $t = 1, 2, \ldots, T$:

- Compute the distribution:

$$q_t(f) = \frac{w_t(f)}{W_t}, \quad \text{where} \quad W_t = \sum_{f \in \mathcal{F}} w_t(f),$$

  where $q_t \in \Delta(\mathcal{F})$ is a probability distribution over $\mathcal{F}$.

- Compute predictor:

$$\hat{f}_t(x) = \sum_{f \in \mathcal{F}} q_t(f) f(x).$$

- Observe (x_t, y_t) and incur loss $\ell(\hat{f}_t(x_t), y_t)$.
- Update weights:

$$w_{t+1}(f) = w_t(f) \cdot e^{-\eta \ell(f(x_t), y_t)}.$$

Notes:

- $q_t(f)$ encodes how "credible" f is based on historical data.

- $w_t(f)$ can be explicitly expressed as:

$$w_t(f) = e^{-\eta \sum_{i=1}^{t-1} \ell(f(x_i), y_i)}.$$

- Consequently,

$$q_t(f) \propto w_t(f).$$

4 Behavior of EWA with Different Learning Rates

The Exponential Weight Algorithm (EWA) behaves differently depending on the choice of the learning rate η. Below are insights into how the algorithm responds to different values of η:

- As $\eta \to 0$: The algorithm does no learning and assigns equal weights to all models, resulting in a uniform distribution over $\mathcal{F}$.
- As $\eta \to +\infty$: The algorithm places all weight on the single best-performing model, effectively concentrating all probability mass on the leader at each time step.
- q_t: Represents the probability distribution, adjusting based on the observed performance of models.

EWA with $\eta \to +\infty$ behaves like Follow the Leader (FTL)

4.1 Graphical Representation

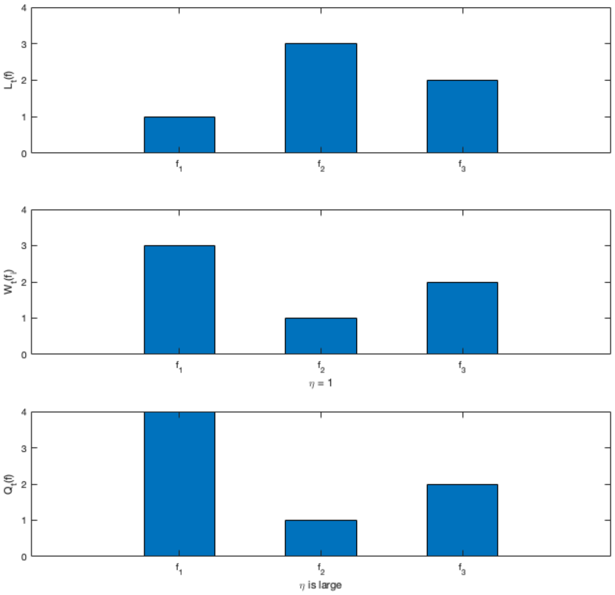

Figure 2: Graphs illustrating the weight distribution over functions f_1, f_2, f_3 with varying learning rates.

- **Top Graph:** Shows cumulative loss up to step $t-1$ for each model, $L_{t-1}(f) = \sum_{s=1}^{t-1} \ell(f(x_s), y_s)$.
- **Middle Graph:** Illustrates the weight distribution adjusting after observing losses, where better-performing models gain more weight.

- **Bottom Graph:** Depicts a highly concentrated weight distribution, indicating a large η, focusing on the best-performing model.

4.2 Comparative Weight Analysis

- As time progresses, models that perform better in past rounds see their weights increase exponentially:

$$L_{t-1}(f_1) < L_{t-1}(f_2) \implies w_t(f_1) > w_t(f_2).$$

4.3 Regret Analysis

For any prediction class $\mathcal{F}$, EWA with an appropriate choice of η exhibits the following behavior:

1. For absolute loss $\ell(\hat{y}, y) = |\hat{y} - y|$ and $\hat{y} \in \{-1, 1\}$, the regret is bounded by:

$$\text{Regret}(F, T) \leq O\left(\sqrt{T \ln |\mathcal{F}|}\right), \quad \text{`Slow rate''}.$$

2. Assuming $\hat{y} \to e^{-\alpha d(\hat{y})}$ is concave (this property, α-exp-concavity, is true for squared loss $\ell(\hat{y}, y) = (\hat{y} - y)^2$, and log loss):

$$\text{Regret}(F, T) \leq O\left(\frac{1}{\eta} \ln |\mathcal{F}|\right), \quad \text{`Fast rate''}.$$

Implications of Learning Rate - The first scenario (slow rate) implies that even with conservative learning, the regret grows sublinearly, showcasing EWA's adaptivity over time.

- The second scenario (fast rate) shows that when the losses have additional favorable structure, appropriately tuning η allows the algorithm to focus on minimizing regret more aggressively.

The choice of η is critical for maintaining a balance between adapting to the best models and ensuring stable performance.

4.4 Refresher on Convex and Concave Functions

Understanding convexity and concavity is essential in analyzing optimization algorithms like the Exponential Weight Algorithm (EWA). Below is a refresher on these concepts.

4.4.1 Definition of Convex Function

A function $f : \mathcal{X} \to \mathbb{R}$ is called a convex function if, for any $x_1, x_2 \in \mathcal{X}$ and $\lambda \in [0, 1]$, the following inequality holds:

$$f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2).$$

Conversely, f is concave if:

$$f(\lambda x_1 + (1 - \lambda)x_2) \geq \lambda f(x_1) + (1 - \lambda)f(x_2).$$

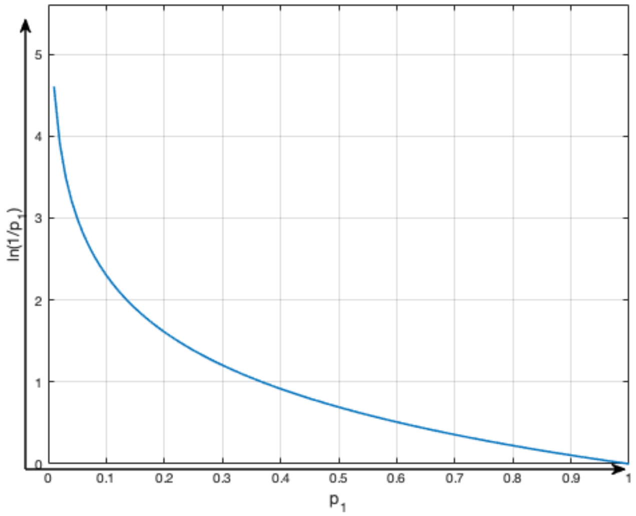

Figure 3: Graph of $\ln \frac{1}{p_1}$, showing the relationship between p_1 and its logarithmic transformation.

4.5 Further Explanation on α-exp-Concavity

We can now interpret α-exp-concavity with the definition of concave functions. When we say that a loss function ℓ is α-exp-concave, we mean the following:

For any predictions $\hat{y}_1, \hat{y}_2 \in \hat{\mathcal{Y}}$ and $\lambda \in [0,1]$:

$$-e^{-\alpha\ell(\lambda\hat{y}_1+(1-\lambda)\hat{y}_2,y)} \geq \lambda e^{-\alpha\ell(\hat{y}_1,y)} + (1-\lambda)e^{-\alpha\ell(\hat{y}_2,y)}.$$

Rearranging the above inequality, we get:

$$\ell(\lambda\hat{y}_1 + (1-\lambda)\hat{y}_2, y) \leq \frac{1}{-\alpha}\ln\left(\lambda e^{\alpha\ell(\hat{y}_1,y)} + (1-\lambda)e^{\alpha\ell(\hat{y}_2,y)}\right).$$

Implication: Mixing predictions doesn't hurt prediction performance significantly, meaning that combining predictions can still yield good results under α-exp-concave conditions.

4.6 Examples of α-exp-Concave Losses

Understanding α-exp-concave losses helps in analyzing various optimization problems where specific loss functions are used. Below are examples of commonly encountered α-exp-concave losses:

- **Squared Loss:** When $y, \hat{y} \in [-1,1]$, the squared loss function $\ell(\hat{y}, y) = (\hat{y} - y)^2$ is α-exp-concave with $\alpha = \frac{1}{8}$.
- **Logarithmic Loss:** When $\hat{y} = p$, the logarithmic loss is defined as $\ell(p, y) = \ln \frac{1}{p_y}$, where $y \in \{0,1\}$ and $p = (p_0, p_1)$. For this loss function, $\alpha = 1$.

4.7 Relationship of Convex Losses

The Venn diagram illustrates the relationship among various types of losses:

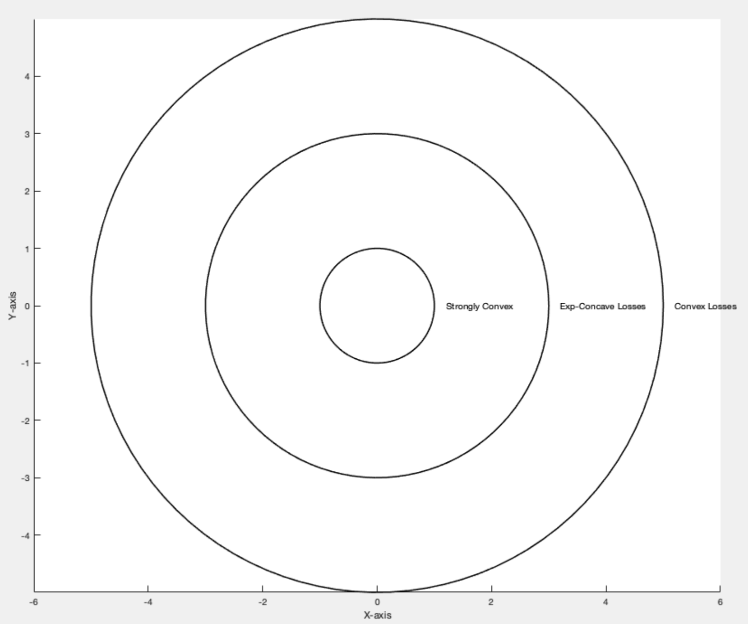

Figure 4: Visualization of the relationship among convex, α-exp-concave, and strongly convex loss functions.

- **Convex Losses:** The outermost circle represents general convex losses.
- **α-exp-Concave Losses:** A subset of convex losses, with special properties that make them useful in online learning.
- **Strongly Convex Losses:** The innermost circle, representing losses that have the strongest form of convexity, often used for faster convergence guarantees in optimization.

4.8 Jensen's Inequality

Jensen's inequality provides a fundamental relationship between convex functions and expectations. It states that for any (convex or concave) function F and random variable Z:

$$F\left(\mathbb{E}[Z]\right) \leq \mathbb{E}[F(Z)] \quad \text{if } F \text{ is convex.}$$

Example Application: Let Z's PMF be given as $Z = (x_1, x_2, \ldots, x_n)$ with probabilities $(p_1, p_2, \ldots, p_n)$. By applying Jensen's inequality, we have:

$$F\left(\mathbb{E}[Z]\right) = F\left(\sum_{i=1}^{n} p_i x_i\right) \leq \sum_{i=1}^{n} p_i F(x_i).$$

Using $F(Z) = \ln \frac{1}{p_1}$ as an example function, Jensen's inequality illustrates how expectations of logarithmic functions behave in probability and statistical settings.

5 Regret Analysis for EWA

In online learning, regret measures how much worse the learning algorithm performs compared to the best possible fixed predictor in hindsight.

$$\text{Regret}(F, T) = \sum_{t=1}^{T} \ell(f_t(x_t), y_t) - \ell(f^*(x_t), y_t), \quad f^* = \arg\min_{f \in \mathcal{F}} \sum_{t=1}^{T} \ell(f(x_t), y_t).$$

5.1 Potential-based Amortized Analysis

This analysis is used to assess the performance of an online learning algorithm by relating it to potential functions and Kullback-Leibler (KL) divergence, also known as relative entropy.

5.2 Definition: Kullback-Leibler Divergence (KL)

Given two distributions $u, v \in \Delta(\mathcal{F})$, the KL divergence is defined as:

$$K(u,v) = \sum_{f \in \mathcal{F}} u(f) \ln \frac{u(f)}{v(f)}.$$

Properties:

- $K(u,v) \geq 0$.
- $K(u,v) = 0 \iff u = v$.
- KL divergence is not symmetric: $K(u,v) \neq K(v,u)$ in general.

5.3 Example Calculations

Let $\mathcal{F} = \{f_1, f_2, f_3\}$, with specific distributions $u = (1,0,0)$ and $v = \left(\frac{1}{2}, \frac{1}{2}, 0\right)$.

$$K(u,v) = 1 \ln \frac{1}{\frac{1}{2}} + 0 \ln \frac{0}{\frac{1}{2}} + 0 \ln \frac{0}{0} = \ln 2.$$

Another example with $u = (1,0)$ and $v = \left(\frac{1}{2}, \frac{1}{2}\right)$:

$$K(u,v) = 1 \ln \frac{1}{\frac{1}{2}} + 0 \ln \frac{0}{\frac{1}{2}} = \ln 2 + 0 = \ln 2.$$

$$K(v,u) = \frac{1}{2} \ln \frac{\frac{1}{2}}{1} + \frac{1}{2} \ln \frac{\frac{1}{2}}{0} = \frac{1}{2} \ln \frac{1}{2} + \infty = +\infty.$$

In general, if v has nonzero probability on points not supported by u, $K(v,u) = +\infty$.

Note: We define $0 \ln 0 = 0$ to handle cases where the probability is zero.

5.4 Definition of Potential Function

Define the potential function $\Phi_t = K(q_t, e_{f^*})$, where q_t denotes the distribution over $\mathcal{F}$ at time t, and e_{f^*} represents the probability distribution that assigns probability 1 to f^* and 0 to all other predictors.

6 Regret Analysis and Proof for Absolute Loss Setting

In this section, we analyze the regret bounds in online learning using the potential method and present a key result showing the dependency of regret on the learning rate η.

6.1 Regret Bound Analysis

We define regret as:

$$\text{Regret}(F,T) = \sum_{t=1}^{T} \ell(f_t(x_t), y_t) - \sum_{t=1}^{T} \ell(f^*(x_t), y_t),$$

where $f^* = \arg\min_{f\in\mathcal{F}} \sum_{t=1}^{T} \ell(f(x_t), y_t)$. This is the regret against the best fixed predictor in hindsight from the hypothesis class $\mathcal{F}$.

Denote by the instantaneous regret of the algorithm at round t by

$$\text{reg}_t := \ell(f_t(x_t), y_t) - \ell(f^*(x_t), y_t).$$

The basic observation for potential analysis is:

$$\Phi_{T+1} \geq 0 \quad \text{(KL divergence is non-negative)}.$$

In addition,

$$\Phi_1 = K(e_{f^*}, q_1) = \ln|\mathcal{F}|.$$

Key Claim:

$$\eta\text{reg}_t \leq \Phi_t - \Phi_{t+1} + \frac{\eta^2}{2}.$$

This claim relates the potential functions to the learning rate and observed losses.

6.2 Behavior of the Learning Rate η

- When η is large: - The algorithm reacts quickly to recent predictions.
- When η is small: - The algorithm balances past and recent data, adapting more slowly.

6.3 Proof Outline

Summing over all t, we establish the relationship:

$$\eta \cdot \text{Regret}(F,T) \leq \Phi_1 - \Phi_{T+1} + \frac{\eta^2}{2}T.$$

Simplifying further:

$$\text{Regret}(F,T) \leq \frac{\ln|\mathcal{F}|}{\eta} + \frac{\eta T}{2}.$$

Choosing the optimal η to minimize this bound:

$$\text{Regret}(F,T) \leq \sqrt{2T\ln|\mathcal{F}|}.$$

6.4 Graphical Representation of Potential Functions

First, q_t does not always converge to e_{f^*} – note that in online learning, we make no assumption on the data generation process. For example, if the environment shows examples with the same feature but alternating labels in odd and even rounds respectively, $\{q_t\}$ may oscillate.

The win-win interpretation of the equation

$$\eta\text{reg}_t \leq \Phi_t - \Phi_{t+1} + \frac{\eta^2}{2},$$

as presented in the lecture, was:

- If reg_t is small, this is already good since the algorithm does well in step t. (win)
- Otherwise, if reg_t is large, this inequality implies that $\Phi_t - \Phi_{t+1}$ is large – in other words, $K(e_{f^*}, q_{t+1})$ is significantly smaller than $K(e_{f^*}, q_t)$, i.e. q_{t+1} is closer to e_{f^*} than q_t, making us making future prediction more accurately. (win)

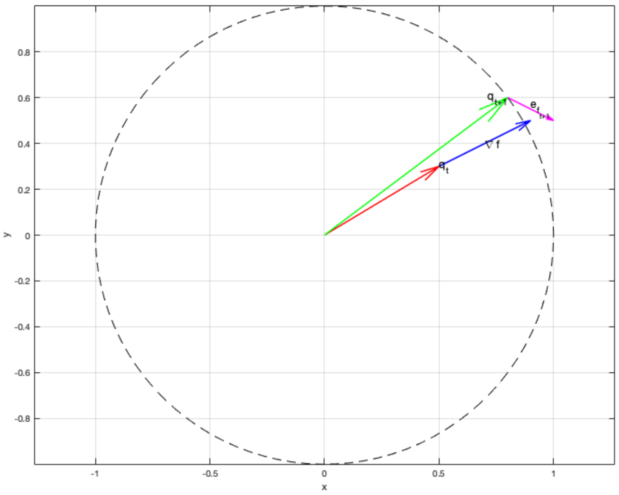

Figure 5: Graphical representation illustrating the progression of potential functions, showing how regularization and potential differences impact future prediction accuracy.

7 Proof of Key Claim

We now prove the claim by analyzing the difference between potential functions at consecutive time steps and bounding this difference using the KL divergence and observed losses.

7.1 Proof of the Claim

We analyze the key difference:

$$\Phi_t - \Phi_{t+1} = \ln \frac{q_{t+1}(f^*)}{q_t(f^*)} = \underbrace{\ln \frac{w_{t+1}(f^*)}{w_t(f^*)}}_{(a)} - \underbrace{\ln \frac{W_{t+1}}{W_t}}_{(b)} \tag{1}$$

Let us now lower bound (a) and (b) respectively. Using:

$$w_{t+1}(f) = e^{-\eta \ell(f(x_t), y_t)} w_t(f),$$

Term (a) is

$$\ln \frac{w_{t+1}(f^*)}{w_t(f^*)} = -\eta \ell(f^*(x_t), y_t) \tag{2}$$

For term (b), observe that

$$\ln \frac{W_{t+1}}{W_t} = \ln(\frac{\sum_{f \in \mathcal{F}} w_t(f) e^{-\eta \ell(f(x_t), y_t)}}{W_t}) = \ln(\sum_{f \in \mathcal{F}} q_t(f) e^{-\eta \ell(f(x_t), y_t)})$$

Note that, if we can upper bound $\ln \frac{W_{t+1}}{W_t}$ by, say $-\eta\ell(f_t(x_t), y_t)$ (perhaps with some slack), we are close to proving the claim.

7.2 Key Observations

Observation 1:

Let $f^1, \ldots, f^N$ be an enumeration of the elements in $\mathcal{F}$, where $N = |\mathcal{F}|$. Consider random variable Z that has the following probability mass function:

Z	$\ell(f^1(x_t), y_t)$	$\ldots$	$\ell(f^N(x_t), y_t)$
$P(Z = z)$	$q_t(f^1)$	$\ldots$	$q_t(f^N)$

By the construction of Z, we can now write

$$\ln \frac{W_{t+1}}{W_t} = \ln\mathbb{E}\left[e^{-\eta Z}\right].$$

Observation 2:

$$\ell(f_t(x_t), y_t) = \sum_{f\in\mathcal{F}} q_t(f)\ell(f(x_t), y_t).$$

This uses the property of the absolute loss, and it may not hold in general cases (this was left as an exercise).

Now again, by the construction of Z, we can write

$$\ell(f_t(x_t), y_t) = \mathbb{E}[Z]$$

So, if we can upper bound $\ln \mathbb{E}\left[e^{-\eta Z}\right]$ in terms of $\mathbb{E}[Z]$, that would fulfill our goal. Did this ring a bell? In our very first lecture, we introduced the subgaussian condition that provides bounds of moment generating functions of a random variable. Specifically, if Z is b^2-subgaussian, then

$$\mathbb{E}e^{-\eta(Z-\mathbb{E}Z)} \le e^{\frac{\eta^2 b^2}{2}}$$

Multiply by constant $e^{-\eta\mathbb{E}[Z]}$ on both sides,

$$\mathbb{E}e^{-\eta Z} \le e^{\frac{\eta^2 b^2}{2}} \cdot e^{-\eta\mathbb{E}[Z]}$$

Taking log on both sides,

$$\ln \mathbb{E}e^{-\eta Z} \le \frac{\eta^2 b^2}{2} - \eta\mathbb{E}[Z]$$

Now back to our random variable Z, is it subgaussian? Yes – since it is bounded – Z takes values in $[0, 2]$, and thus it is 1-subgaussian. In other words, for our Z, we have

$$\ln \mathbb{E}e^{-\eta Z} \le \frac{\eta^2}{2} - \eta\mathbb{E}[Z],$$

and putting this into our online learning notation, this gives

$$\ln \frac{W_{t+1}}{W_t} \le \frac{\eta^2}{2} - \eta\ell(f_t(x_t), y_t). \tag{3}$$

The claim now follows by combining Eqs. (1), (2), and (3).

8 Proof of Regret Bound for α-exp-Concave Losses

We now analyze the regret bound for online learning algorithms with α-exp-concave losses.

8.1 Regret Definition

The regret over T rounds is given by:

$$\text{Regret}(F,T) = \sum_{t=1}^{T} \ell(f_t(x_t), y_t) - \sum_{t=1}^{T} \ell(f^*(x_t), y_t),$$

where $f^* = \arg\min_{f\in\mathcal{F}} \sum_{t=1}^{T} \ell(f(x_t), y_t)$.

8.2 Proof of Regret Bound for α-exp-Concave Losses

For α-exp-concave losses, we set $\eta = \alpha$ and prove the following claim:

Claim: Setting $\eta = \alpha$ ensures that for all t:

$$\alpha \cdot \text{reg}_t \leq \Phi_t - \Phi_{t+1}.$$

Summing over all t, we conclude:

$$\alpha \cdot \text{Regret}(F,T) \leq \Phi_1 - \Phi_{T+1} \leq \ln|\mathcal{F}|.$$

8.3 Proof of the Claim

First, note that we can reuse the proof for the absolute loss and have the same expression for $\Phi_t - \Phi_{t+1}$ (with $\eta = \alpha$):

$$\Phi_t - \Phi_{t+1} = -\alpha\ell(f^*(x_t), y_t) - \ln\left(\sum_{f\in\mathcal{F}} q_t(f) e^{-\alpha\ell(f(x_t),y_t)}\right)$$

Using Jensen's inequality and the fact that $F(\hat{y}) := e^{-\alpha\ell(\hat{y},y)}$ is a concave function in $\hat{y}$, we derive:

$$\sum_{f\in\mathcal{F}} q_t(f) e^{-\alpha\ell(f(x_t),y_t)} \leq e^{-\alpha\ell(f_t(x_t),y_t)}.$$

Plugging this into the equality above, this proves the desired bound.

Conclusion: The result demonstrates that for α-exp-concave losses, setting $\eta = \alpha$ achieves an optimal balance between learning speed and regret minimization.