Bandit Multiclass Classification

1 A Regret Lower Bound for A Certain Type of Algorithms

In this section, we try to construct an error lower bound for a certain type of algorithms. This type of algorithms does not make update when it makes a wrong prediction. For simplicity, we only consider binary classification. More formally, the algorithms we consider satisfy the following assumption.

Assumption 1 (Algorithm)**.** *Let $p_t(x)$ be the algorithm's probability of predicting class 1 (recall we consider binary classification) at round t if it receives the feature vector $x \in \mathcal{X} \subset \mathbb{R}^d$. We assume $p_t(\cdot)$ is totally determined by all previous **correct** examples. In other words, $p_t(\cdot)$ is determined by the tuple $((x_{\tau_1}, y_{\tau_1}), \ldots, (x_{\tau_N}, y_{\tau_N}))$ where $1 \le \tau_1 < \tau_2 \ldots < \tau_N < t$ are the rounds that the learner makes correct prediction.*

Assumption 2 (linearly separable with a margin)**.** *We assume the samples are linearly separable with margin γ (i.e., any two points with different labels have distance no less than γ).*

Definition 1 (Free space)**.** *The free space at time t is the set of points whose label is still under-determined given $(x_1, y_1), \ldots, (x_{t-1}, y_{t-1})$. For example, the γ-ball centered around any already presented point is excluded from the free space. Denote the free space at time t by FS_t.*

The free space's definition simply means that at time t, the adversary can pick any point x_t in FS_t and assign the label y_t to either 1 or 2 without violating the linearly separable and the γ-margin assumption.

Below we present the Adversary's strategy of constructing (x_t, y_t).

Algorithm 1: Adversary's strategy

```
Pick x_1 randomly from 𝒳, and let y_1 = 1.
for t = 2, ..., T do
    if ỹ_{t-1} ≠ y_{t-1} then
        Let (x_t, y_t) = (x_{t-1}, y_{t-1})
    else if FS_t is not empty then
        Pick x_t ∈ FS_t. Because of this x_t, the free space's volume is reduced. We denote the
          reduction amount by V_t = v(FS_{t+1}) − v(FS_t) ≤ V. (i.e., V is a global upper bound of V_t)
        If p_t(x_t) ≥ 1 − max{√V, 1/√T}, then label y_t = 2; otherwise, label y_t = 1.
    else
        Randomly assign (x_t, y_t) with some value that does not violate the assumption.
```

Definition 2 (history)**.** *Let $\mathcal{H}_t$ be the history before time t: $\mathcal{H}_t = \{(x_s, y_s, \tilde{y}_s)\}_{s=1}^{t-1}$. We use $\mathbb{E}_t[\cdot]$ to denote $\mathbb{E}[\cdot|\mathcal{H}_t]$.*

Lemma 3. *If $\exists t$ such $p_t(x_t) \ge 1 - \max\left\{\sqrt{V}, \frac{1}{\sqrt{T}}\right\}$, then $\mathbb{E}_t\left[\sum_{s=t}^{T} \mathbf{1}[\tilde{y}_s \neq y_s]\right] \ge \Omega\left(\min\left\{\frac{1}{\sqrt{V}}, \sqrt{T}\right\}\right)$.*

Proof. By of the condition and the adversary strategy, we have $y_t = 2$. Therefore, the learner will predict the true label with probability $\le \max\left\{\sqrt{V}, \frac{1}{\sqrt{T}}\right\}$. And note that if the learner predicts

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

incorrectly at time t, then at time $t+1$ the feature vector remains the same $(x_{t+1} = x_t)$, and the learner's probability of prediction also remains the same $(p_{t+1}(\cdot) = p_t(\cdot))$. Therefore, the expected number of mistakes before the first correct guess is (roughly) larger than $\frac{1}{\max\left\{\sqrt{V}, \frac{1}{\sqrt{T}}\right\}} = \min\left\{\frac{1}{\sqrt{V}}, \sqrt{T}\right\}$. □

Lemma 4. *If* $\forall s,\ p_s(x_s) \leq 1 - \max\left\{\sqrt{V}, \frac{1}{\sqrt{T}}\right\}$, *then* $\sum_{s=1}^{T} \mathbb{E}_s[\mathbf{1}[\tilde{y}_s \neq y_s]] = \Omega\left(\min\left\{\sqrt{T}, \frac{1}{\sqrt{V}}\right\}\right)$.

Proof. By the condition and the adversary strategy, we know that the probability of error is larger than $\max\left\{\sqrt{V}, \frac{1}{\sqrt{T}}\right\}$ at all time t *before the free space is used up*. Since each time the free space only reduces by V, in the first $\frac{1}{V}$ rounds (assume the total volume is 1), the free space is still all available. Therefore,

$$\sum_{s=1}^{T} \mathbb{E}_s[\mathbf{1}[\tilde{y}_s \neq y_s]] \geq \min\left\{T, \frac{1}{V}\right\} \times \max\left\{\sqrt{V}, \frac{1}{\sqrt{T}}\right\} = \min\left\{\sqrt{T}, \frac{1}{\sqrt{V}}\right\}.$$

□

Discussion. We believe that V can be in the order of $\mathcal{O}\left(\gamma^d\right)$ (some weaker thing like $\mathcal{O}\left(\gamma^{d/2}\right)$ is also acceptable). Basically, we need to figure out the following question: can we construct a sequence of points $\{x_1, x_2, \ldots\}$, such that for every t, x_t's distance with conichull$\{x_1, \ldots, x_{t-1}\}$ is larger than γ, but the volume difference $v(\text{conichull}\{x_1, \ldots, x_t\}) - v(\text{conichull}\{x_1, \ldots, x_{t-1}\})$ is upper bounded by V?

This lower bound does not rule out those algorithms that change its probability vector based on the **count** of consecutive errors. This type of algorithm may still easy to be implemented (like QBC). Also it does not rule out the banditron algorithm.

2 Biased Halving: Trading Error with Complexity

Algorithm 2: Banditron

1 **Define**: $\Omega = \{W \in \mathbb{R}^{Kd} : \|\mathbf{e}_i^\top W\|_2 \leq D\}$.
2 For a set S of W's, $S(i|x)$ is the subset of S that outputs class i given feature vector x, i.e., $S(i|x) = \{W \in S : (Wx)_i \geq (Wx)_j\ \forall j\}$
3 $|S|$ denotes the volumn of S.
4 **parameter**: $\alpha \in (0, 1)$
5 $\Omega_1 = \Omega$.
6 **for** $t = 1, \ldots, T$ **do**
7   **if** $\text{argmax}_i \frac{|\Omega_t(i|x_t)|}{|\Omega_t|} \geq 1 - \alpha$ **then**
8   Let $\tilde{y}_t = i$.
9   **if** $\tilde{y}_t \neq y_t$ **then**
10             $\Omega_{t+1} = \Omega_t \backslash \Omega_t(\tilde{y}_t|x_t)$.
11   **else**
12   Let $\tilde{y}_t \sim \text{unif}([K])$.
13   **if** $\tilde{y}_t = y_t$ **then**
14             $\Omega_{t+1} = \Omega_t(\tilde{y}_t|x_t)$.

Conjecture(should be true): If the volume of Ω_t becomes smaller than $\frac{|\Omega|}{N}$, then the algorithm won't make any error anymore. N should be in the order of $\Theta(\frac{1}{\gamma^{Kd}})$.

Rough analysis:

Each time the algorithm makes an error in Line 7, the volume becomes α times the original volume. So the algorithm will not make more than $\frac{\ln N}{\ln \frac{1}{\alpha}}$ mistakes in this case.

In the case of Line 10, $K \ln \frac{1}{\delta}$ errors will accompany with a $(1-\alpha)$-factor shrinkage in the volume. Therefore, the number of errors occurred in this case is upper bounded by $\frac{K \ln \frac{1}{\delta} \ln N}{\ln \frac{1}{1-\alpha}} \leq \frac{K \ln \frac{1}{\delta} \ln N}{\alpha}$.

Now we discuss about the complexity. The main issue is how to maintain Ω_t. Each time the algorithm enters Line 10, Ω_t becomes more and more fragmented. But if Ω_t can be maintained with M convex cones, then Ω_{t+1} can be maintained with $(K-1)M \leq KM$ convex cones. And we assume each cone's volume can be computed in poly(T) time. Each time the algorithm enters Line 14, the number of convex cones does not increase.

By the above discussion, there will be no more than $K^{\frac{\ln N}{\ln \frac{1}{\alpha}}}$ convex cones to maintain. And the error bound is in the order of $\frac{K \ln \frac{1}{\delta} \ln N}{\alpha}$ for some $\alpha < \frac{1}{2}$. Let's try to balance the number of errors and computational complexity. Let

$$
\begin{aligned}
& K^{\frac{\ln N}{\ln \frac{1}{\alpha}}} \approx \frac{K \ln \frac{1}{\delta} \ln N}{\alpha} \\
\Rightarrow & \frac{\ln N}{\ln \frac{1}{\alpha}} \ln K \approx \ln \left(K \ln \frac{1}{\delta} \ln N \right) + \ln \frac{1}{\alpha} \\
\Rightarrow & \text{pick } \ln \frac{1}{\alpha} = \sqrt{\ln N}.
\end{aligned}
$$

Thus the computational complexity is in the order of $K^{\sqrt{\ln N}} \times \text{poly}(T) = K^{\sqrt{Kd \ln \frac{1}{\delta}}}$. The error bound is $\mathcal{O}\left(e^{\sqrt{Kd \ln \frac{1}{\delta}}} K^2 d \ln \frac{1}{\delta} \ln \frac{1}{\gamma}\right)$.

Another viewpoint: let $\frac{1}{\alpha} = K^{\beta}$, then the complexity is $\left(\frac{1}{\gamma}\right)^{\frac{Kd}{\beta}} \times \text{poly}(T)$ and the error bound is $K^{\beta+1} \ln \frac{1}{\delta} \ln N$.

3 How hard it is to use the feedback only from wrong guesses?

The halving algorithm can actually run if we can do "uniform sampling" over the version space. But it is even unknown whether we can efficiently pick a model from the version space. The problem is that we get a lot of feedback in the form of "feature x_t does not belong to class $\tilde{y}_t$", which we don't know how to use.

The following is just an attempt (not successful but might be interesting...) to say that it might be not easy to figure out a model if the learner is only presented with this kind of "error message".

The problem is formulated as follows. Given N points in a row, each one with a class $c_i \in [K]$, $\forall i \in [N]$. We call these N points *separable* if the following statement holds:

$$\text{If } c_i = c_j \text{ for some } i \leq j, \text{ then } c_i = c_{i+1} = \cdots = c_j.$$

For example, if $N = 5, K = 3$, then $(c_1, c_2, c_3, c_4, c_5) = (3, 3, 1, 1, 1)$ is separable, but $(c_1, c_2, c_3, c_4, c_5) = (2, 1, 2, 2, 2)$ is not.

Now you have N conditions, in which the i-th condition only says something like "$c_i \neq k$" for some k.
(1) Can you efficiently decide whether there exists an assignment of $(c_1, \ldots, c_N)$ such that these N points are separable and satisfy all the conditions?
(2) If it is guaranteed that there are separable solutions, can you efficiently find one of them?
(Efficient: the complexity is polynomial in N and K)

Example 1.
$N = 5, K = 3$:
$c_1 \neq 1$
$c_2 \neq 2$
$c_3 \neq 3$
$c_4 \neq 1$
$c_5 \neq 2$

$\Rightarrow (c_1, c_2, c_3, c_4, c_5) = (2, 1, 1, 3, 3)$ or $(3, 3, 2, 2, 1)$ are separable solutions.

Example 2.
$N = 7, K = 3$:
$c_1 \neq 1$
$c_2 \neq 2$
$c_3 \neq 3$
$c_4 \neq 1$
$c_5 \neq 2$
$c_6 \neq 3$
$c_7 \neq 1$
$\Rightarrow$ There is no separable solution.

It turns out this specific 1-dimensional problem is equivalent to identify a **missing permutation** of $[K]$ as a subsequence in the given sequence. In Example 1, the existing permutations are $(1, 2, 3), (1, 3, 2), (2, 3, 1), (3, 1, 2)$, the missing ones are $(2, 1, 3)$ and $(3, 2, 1)$. So the solutions can be $(2, 1, 3)$ or $(3, 2, 1)$ (with some repetition). In Example 2, all permutations are as subsequences, so there is no solution.

We can prove this equivalence considering two directions:
(1) If there is a solution with the class labels following a permutation, then that permutation cannot be a subsequence of the given sequence.
(2) If there is a missing permutation in the given sequence, then there is a class assignment that follows this permutation.
They should be straightforward by trying some examples.

4 Cone Algorithm

Algorithm 3: Banditron

definition: $K \triangleq$ number of classes, $\gamma \triangleq$ margin
Initialize: $\mathcal{S}_1 = \cdots = \mathcal{S}_K = \phi$ (empty set)
for $t = 1, \ldots, T$ **do**
  Receive $x_t \in \mathbb{R}^d$.
  Define the cone $\mathcal{C}_i = \left\{ x \in \mathbb{R}^d : x = \sum_{j=1}^{|\mathcal{S}_i|} \alpha_j y_j, \text{ where } y_j \in \mathcal{S}_i, \alpha_j \geq 0 \right\}$
  (that is, $\mathcal{C}_i$ is the conic hull of $\mathcal{S}_i$)
  Check whether x_t belongs to, or has distance smaller than γ, to one of $\mathcal{C}_1, \ldots, \mathcal{C}_K$.
  If so, classify x_t to the corresponding class (say class i), and let $\mathcal{S}_i \leftarrow \mathcal{S}_i \cup \{x_t\}$. This prediction will be correct for sure by our margin assumption.
  If not, let $\tilde{y}_t \sim$ unif($[K]$) and predict $\tilde{y}_t$. If $\tilde{y}_t = y_t$, then $\mathcal{S}_{y_t} \leftarrow \mathcal{S}_{y_t} \cup \{x_t\}$.

5 One-against-all Perceptron

Same as Chicheng's writeup's Section 5: Fixed-Threshold Perceptron.

6 Gradient Descent [TODO]

Assumption 3. $\|x_t\|_2^2 \leq 1$. *There is a* $W^* \in \mathcal{W}$ *such that* $\ell_t(W^*) \leq 0$ *for all* t *(*ℓ_t *and* $\mathcal{W}$ *are defined below).*

Algorithm 4: Banditron

Input: $D \geq 2$, ϵ (picked in a later lemma).

Definition:

$$\begin{aligned}\ell_t(W) &\triangleq [1 - (Wx_t)_{y_t} + \max_{r \neq y_t} (Wx_t)_r]_+^2 \quad \text{(squared hinge loss)} \\ &= \Phi_t(Wx_t),\end{aligned}$$

where $\Phi_t(z) \triangleq [1 - \mathbf{e}_{y_t}^\top z + \max_{r \neq y_t} \mathbf{e}_r^\top z]_+^2$.

Also, define $\mathcal{W} = \{W \in \mathbb{R}^{K \times d} : \|\mathbf{e}_i^\top W\|_2 \leq D \text{ for all } i \in [K]\}$.

Initialization: $W_1 = 0, M_1 = I$.

for $t = 1, \ldots, T$ **do**

- Observe x_t.
- **if** $\|x_t\|_{M_t^{-1}} \geq \epsilon$ *and* $\|W_t - W^*\|_F \geq 1$ **then**
  - Draw $\tilde{y}_t \sim \text{unif}([K])$.
- **else**
  - Draw $\tilde{y}_t = \hat{y}_t \triangleq \text{argmax}_{r \in [K]} (W_t x_t)_r$.
- **if** $\tilde{y}_t = y_t$ **then**
  - $Z_t \leftarrow 1$,
  - $M_{t+1} \leftarrow M_t + Z_t \ell_t(W_t) x_t x_t^\top$,
  - $W_{t+1} \leftarrow \Pi_{\mathcal{W}}(W_t - \eta_{t+1} \nabla \ell_t(W_t))$, where $\eta_{t+1} = \frac{1}{8}$.
  - ($\Pi_{\mathcal{W}}$ is the projection operator onto $\mathcal{W}$ w.r.t. Frobenius norm)
- **else**
  - $Z_t \leftarrow 0$,
  - $M_{t+1} \leftarrow M_t$,
  - $W_{t+1} \leftarrow W_t$.

Lemma 5. $\|\nabla \ell_t(W)\|_F^2 \leq 8\ell_t(W)$.

Proof. $\|\nabla \ell_t(W)\|_F^2 = \|\nabla \Phi_t(Wx_t) x_t^\top\|_F^2 \leq \left(2\sqrt{\Phi_t(Wx_t)}\right)^2 \times 2\|x_t\|_2^2 \leq 8\ell_t(W)$. □

Lemma 6. *Let* $L_{t+1} \triangleq \sum_{s=1}^t Z_s \ell_s(W_s)$*. Then* $\|W_{t+1} - W^*\|_F^2 \leq \exp\left(-\frac{L_{t+1}}{32KD^2}\right)$.

Proof. Let $Z_t = 1$.

$$\begin{aligned}\|W_{t+1} - W^*\|_F^2 &\leq \|W_t - \eta_{t+1}\nabla \ell_t(W_t) - W^*\|_F^2 \\ &= \|W_t - W^*\|_F^2 - 2\eta_{t+1} \langle \nabla \ell_t(W_t), W_t - W^* \rangle_F + \eta_{t+1}^2 \|\nabla \ell_t(W_t)\|_F^2.\end{aligned}$$

By the separable assumption we have $\ell_t(W^*) \leq 0$. Since ℓ_t is convex, $\langle \nabla \ell_t(W_t), W_t - W^* \rangle \geq \ell_t(W_t) - \ell_t(W^*) \geq \ell_t(W_t)$. Continuing the above calculation and using Lemma 5, we get

$$\begin{aligned}\|W_{t+1} - W^*\|_F^2 &\leq \|W_t - W^*\|_2^2 - 2\eta_{t+1}\ell_t(W_t) + 8\eta_{t+1}^2 \ell_t(W_t) \\ &\leq \|W_t - W^*\|_F^2 - \frac{1}{8}\ell_t(W_t) \\ &\leq \|W_t - W^*\|_F^2 \left(1 - \frac{\ell_t(W_t)}{32KD^2}\right) \quad \text{because } \|W_t - W^*\|_F^2 \leq 4KD^2 \\ &\leq \|W_t - W^*\|_F^2 \exp\left(-\frac{\ell_t(W_t)}{32KD^2}\right)\end{aligned}$$

By induction, we can get

$$\|W_{t+1} - W^*\|_F^2 \leq KD^2 \exp\left(-\frac{L_{t+1}}{32KD^2}\right)$$

□

Definition 7. $\|W\|_M^2 \triangleq \sum_{i=1}^K \left\|\mathbf{e}_i^\top W\right\|_M^2$.

With this definition we have $\|Wx_t\|_2^2 = \sum_{i=1}^K (\mathbf{e}_i^\top W x_t)^2 \leq \sum_{i=1}^K \|\mathbf{e}_i^\top W\|_M^2 \|x_t\|_{M^{-1}}^2 \leq \|W\|_M^2 \|x_t\|_{M^{-1}}^2$

Lemma 8.

$$\|W_t - W^*\|_{M_t}^2 \leq (1 + L_t) K^2 D^2 \exp\left(-\frac{L_t}{32KD^2}\right) \leq 32K^3 D^4.$$

Proof. Because we assume $\|x_t\|_2^2 \leq 1$, it holds that $M_t \preceq (1 + L_t)I$. Therefore $\|W_t - W^*\|_{M_t}^2 \leq (1+L_t)\|W_t - W^*\|_I^2 = (1+L_t)\sum_{i=1}^K \|\mathbf{e}_i^\top (W_t - W^*)\|_2^2 \leq (1+L_t)K\|W_t - W^*\|_F^2$. By Lemma 6 this is bounded by $(1+L_t)K^2D^2 \exp\left(-\frac{L_t}{32KD^2}\right)$, which can further be bounded by a constant related to K and D. For example, using the property $\exp(-x) \leq \frac{1}{(1+x)^2}$ for all $x > 0$, it can be upper bounded by $(1 + L_t)K^2D^2 \times \frac{(32KD^2)^2}{(L_t+32KD^2)^2} \leq \frac{32^2K^4D^6}{32KD^2+L_t} \leq 32K^3D^4$. ☐

Lemma 9. *If* $\|x_t\|_{M_t^{-1}} \leq \epsilon = \frac{1}{4D\sqrt{32K^3D^4}}$, *then* $\hat{y}_t = y_t$.

Proof. By the convexity of ℓ_t,

$$\begin{aligned}
\ell_t(W_t) \leq \ell_t(W_t) - \ell_t(W^*) &\leq \langle \nabla \ell_t(W_t), W_t - W^* \rangle \\
&= \left\langle \nabla \Phi_t(W_t x_t) x_t^\top, W_t - W^* \right\rangle \\
&= \langle \nabla \Phi_t(W_t x_t), W_t x_t - W^* x_t \rangle \\
&\leq 4D \left\|W_t x_t - W^* x_t\right\|_2 \\
&\leq 4D \left\|W_t - W^*\right\|_{M_t} \left\|x_t\right\|_{M_t^{-1}} \leq 1.
\end{aligned}$$

This implies $\hat{y}_t = y_t$. ☐

Therefore, when we do not explore, we know W_t will predict correctly! Thus we only need to bound the number of errors occurred in exploration rounds, which is calculated by the following lemma.

Lemma 10. $\sum_{t=1}^T \mathbf{1}[\tilde{y}_t \neq y_t] \leq ???$ *with probability at least* $1 - \delta$.

Proof. By the above discussion, $\sum_{t=1}^T \mathbf{1}[\tilde{y}_t \neq y_t] \leq N \triangleq \sum_{t=1}^T Z_t$, the number of exploration rounds.

$$\begin{aligned}
N &= \sum_{t=1}^T \mathbf{1}\left[\|x_t\|_{M_t^{-1}} > \epsilon\right] \\
&\leq \left(K \ln \frac{1}{\delta}\right) \sum_{t=1}^T \mathbf{1}\left[\|x_t\|_{M_t^{-1}} > \epsilon\right] Z_t \qquad \text{(when } \|x_t\|_{M_t^{-1}} \geq \epsilon,\ \tilde{y}_t = y_t \text{ with probability } \tfrac{1}{K}\text{)} \\
&\leq \frac{K \ln \frac{1}{\delta}}{\epsilon^2} \sum_{t=1}^T \|x_t\|_{M_t^{-1}}^2 Z_t \leq \max_{t \in [T]} \left(\frac{1}{\ell_t(W_t)}\right) \times \frac{K \ln T \ln \frac{1}{\delta}}{\epsilon^2}.
\end{aligned}$$

☐

Discussion. In the calculation of Lemma 9, we can actually get $\ell_t(W_t)^2 \leq \|\nabla\Phi_t(W_t x_t)\|_2^2 \|W_t - W^*\|_{M_t}^2 \|x_t\|_{M_t^{-1}}^2$. Similar to the calculation in Lemma 6, $\|\nabla\Phi_t(W_t x_t)\|_2^2$ is bounded by constant times $\ell_t(W_t)$. So the exploration criterion could potentially become $\ell_t(W_t)\|x_t\|_{M_t^{-1}}^2 \geq \frac{1}{\epsilon^2}$, which makes Lemma 10 go through. The problem is just we do not know $\ell_t(W_t)$ in general.

7 Continuous EXP4 with Uniform Exploration

Algorithm 5: Banditron

Parameters: feasible set $\Omega \subset \mathbb{R}^{K\times d}$

Definitions: $\ell_t(W) \triangleq [1-(Wx_t)_{y_t} + \max_{r\in[K]}(Wx_t)_r]_+$ (hinge loss)

for $t = 1, \ldots, T$ **do**

Receive $x_t \in \mathbb{R}^d$.

Define

$$q_t(W) = \frac{\exp(-\alpha \sum_{s=1}^{t-1} \hat{\ell}_s(W))}{\int_{U\in\Omega} \exp(-\alpha \sum_{s=1}^{t-1} \hat{\ell}_s(U)) dU}, \quad \forall W \in \Omega,$$

where $\hat{\ell}_s(W) = \mathbf{1}[\tilde{y}_s = y_s]\left(\frac{\mathbf{1}[\hat{y}_s=y_s]\ell_s(W)}{1-\gamma+\frac{\gamma}{K}} + \frac{\mathbf{1}[\hat{y}_s\neq y_s]\ell_s(W)}{\frac{\gamma}{K}}\right)$.

Sample $W_t \sim q_t$, and let $\hat{y}_t = \operatorname{argmax}_{r\in[K]}(W_t x_t)_r$.

Let $\tilde{y}_t = \hat{y}_t$ with probability $1-\gamma$, and $\tilde{y}_t \sim \text{unif}([K])$ with probability γ.

Lemma 11. $\mathbb{E}_{\tilde{y}_t}[\hat{\ell}_t(W)] = \ell_t(W)$ *for all* W.

Proof.

$$\begin{aligned}
\mathbb{E}_{\tilde{y}_t}[\hat{\ell}_t(W)] &= \mathbb{E}_{\tilde{y}_t}\left[\mathbf{1}[\hat{y}_s = y_s]\frac{\mathbf{1}[\tilde{y}_s = y_s]\ell_t(W)}{1-\gamma+\frac{\gamma}{K}} + \mathbf{1}[\hat{y}_s \neq y_s]\frac{\mathbf{1}[\tilde{y}_s = y_s]\ell_t(W)}{\frac{\gamma}{K}}\right] \\
&= \mathbf{1}[\hat{y}_s = y_s]\ell_t(W) + \mathbf{1}[\hat{y}_s \neq y_s]\ell_t(W) = \ell_t(W).
\end{aligned}$$

□

Plugging these lemmas in the previous hedge bound, we can get

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^T \ell_t(W_t)\right] \\
&= \mathbb{E}\left[\sum_{t=1}^T \int_{W\in\Omega} q_t(W)\ell_t(W)dW\right] \\
&= \mathbb{E}\left[\sum_{t=1}^T \mathbb{E}_{\tilde{y}_t}\left[\int_{W\in\Omega} q_t(W)\hat{\ell}_t(W)dW\right]\right] \\
&\le \mathbb{E}\left[\sum_{t=1}^T \mathbb{E}_{\tilde{y}_t}\left[\hat{\ell}_t(W^*)\right] + \frac{\textbf{Ent}(q_1||\delta(W^*))}{\alpha} + \alpha\sum_{t=1}^T \mathbb{E}_{\tilde{y}_t}\left[\int_{W\in\Omega} q_t(W)\hat{\ell}_t(W)^2 dW\right]\right] \\
&\le \mathbb{E}\left[\sum_{t=1}^T \mathbb{E}_{\tilde{y}_t}\left[\hat{\ell}_t(W^*)\right] + \frac{\textbf{Ent}(q_1||\delta(W^*))}{\alpha} + \frac{2K\alpha}{\gamma}\sum_{t=1}^T \mathbb{E}_{\tilde{y}_t}\left[\int_{W\in\Omega} q_t(W)\hat{\ell}_t(W) dW\right]\right]
\end{aligned}$$

... to bound the regret, it would be something like bounding $\frac{1}{1-\frac{K\alpha}{\gamma}}\left(\frac{1}{\alpha} + \frac{K\alpha}{\gamma}L^* + \gamma T\right)$, which gives $(L^*T)^{1/3} + \sqrt{T}$ regret bound.

Discussion. We can change $\ell_t(\cdot)$ to any reasonable convex loss (e.g., logsitic loss or second-order loss).