Homework 2

Yu-Chieh Kuo B07611039[†]

[†]Department of Information Management, National Taiwan University

Problem 1: PAC-Bayes

1.(a)

One of the possible learning problems satisfying the assumptions in the problem statement is the *learning axis-aligned rectangles*. [1] Consider a simple learning game that aims to learn an unknown axis-aligned rectangle in the $\mathbb{R}^2$ plane. The goal of Learner is to use as few examples as possible to pick a hypothesis rectangle that approximates the true rectangle. In this learning problem,

- $z_1, \cdots, z_n \in \mathcal{Z}$ are the data points in $\mathbb{R}^2$ plane.
- $\mathcal{H}$ is the hypothesis class that is the collection of all axis-aligned rectangles.
- $\lambda : \mathcal{H} \times \mathcal{Z} \to \{0, 1\}$ is the loss function, where $\lambda(h, z) = 1$ if the data point $(z_i)_{i=1,\cdots,n}$ is in the rectangle and 0 otherwise.
- The risk $R : \mathcal{H} \to [0, 1]$ is the expected loss for any possible rectangle $h \in \mathcal{H}$.

A visualization can be accessible by Wu (2021), and the original example of the axis-aligned rectangle was from Kearns and Vazirani (1994).

1.(b)

Given $\eta\varphi(h)$, applying the change of measure inequality yields

$$\begin{aligned}
& \mathbb{E}_{h\sim\hat{\pi}}[\eta\varphi(h)] \leq D(\hat{\pi} \parallel \pi) + \log \mathbb{E}_{h\sim\pi}\left[e^{\eta\varphi(h)}\right] \\
\iff & \mathbb{E}_{h\sim\hat{\pi}}[\eta\varphi(h)] \leq D(\hat{\pi} \parallel \pi) + \log \frac{1}{\delta} \mathbb{E}_{z_1,\cdots,z_n}\left[\mathbb{E}_{h\sim\pi}\left[e^{\eta\varphi(h)}\right]\right] \\
& \qquad \text{(By Markov's inequality)} \\
\iff & \mathbb{E}_{h\sim\hat{\pi}}[\varphi(h)] \leq \frac{1}{\eta} D(\hat{\pi} \parallel \pi) + \frac{1}{\eta} \log \frac{C_n(\eta)}{\delta}
\end{aligned}$$

with probability at least $1 - \delta$. Note that for a random variable $\xi \geq 0$, the Markov's inequality states

$$\mathbb{P}(\xi \geq y) \leq \frac{\mathbb{E}[\xi]}{y} \iff \mathbb{P}\left(\xi \leq \frac{\mathbb{E}[\xi]}{\delta}\right) \geq 1 - \delta;$$

[1]The idea refers to Professor Hung-Yi Lee's ML slide, and this lecture was taught by Professor Pei-Yuan Wu.

that is,

$$\mathbb{P}\left(\log \mathbb{E}_{h\sim\pi}\left[e^{\eta\varphi(h)}\right] \le \log \frac{1}{\delta}\, \mathbb{E}_{z_1,\cdots,z_n}\left[\mathbb{E}_{h\sim\pi}\left[e^{\eta\varphi(h)}\right]\right]\right) \ge 1-\delta.$$

1.(c)

The hint suggests checking the convexity of $\delta(p \parallel q)$ to satisfy Jensen's inequality. I separate $\delta(p \parallel q)$ into the former and latter parts, which are denoted by

$$f(p,q) = p\log\frac{p}{q} \quad \text{and} \quad g(p,q) = (1-p)\log\frac{1-p}{1-q}.$$

A function is convex if and only if its Hessian matrix is positive semi-definite. The Hessian matrices of $f(p,q)$ and $g(p,q)$ are

$$\nabla^2 f(p,q) = \begin{pmatrix} \frac{1}{p} & -\frac{1}{q} \\ -\frac{1}{q} & \frac{p}{q^2} \end{pmatrix} \quad \text{and} \quad \nabla^2 g(p,q) = \begin{pmatrix} \frac{1}{1-p} & -\frac{1}{1-q} \\ -\frac{1}{1-q} & \frac{1-p}{(1-q)^2} \end{pmatrix},$$

and their corresponding eigenvalues are

$$\lambda_f = \begin{pmatrix} 0 & \frac{p^2+q^2}{pq^2} \end{pmatrix} \quad \text{and} \quad \lambda_g = \begin{pmatrix} 0 & \frac{-(p-1)^2-(q-1)^2}{(p-1)(q-1)^2} \end{pmatrix},$$

where λ_f and λ_g are eigenvalue pairs. As the elements of λ_f and λ_g are all non-negative for any $p, q \in (0,1)$, it yields the positive semi-definition of the Hessian matrix and thus concludes the convexity of $f(p,q)$ and $g(p,q)$. Hence, $\delta(p \parallel q)$ is also convex due to the additivity of convexity. Lastly, it shows the statement

$$\mathbb{E}[\delta(u \parallel v)] \ge \delta(\mathbb{E}[u] \parallel \mathbb{E}[v])$$

by Jensen's inequality.

1.(d)

First, I observe that the statement includes two parts: one with the square root and another without. It's always challenging to deal with the problem of combining terms with and without the square root. Hence, it's natural to think about separate terms.

Next, we define $\Delta_R(h) \equiv R(h) - \hat{R}_n(h)$ as the difference between the expected loss and the empirical risk. Applying the change of measure over $2n\Delta_R^2(h)$ gives

$$\begin{aligned}
& \mathbb{E}_{h\sim\hat{\pi}}\left[2n\Delta_R^2(h)\right] && \le D(\hat{\pi} \parallel \pi) + \log \mathbb{E}_{h\sim\pi}\left[e^{2n\Delta_R^2(h)}\right] \\
\iff\ & \exp\left\{2n\, \mathbb{E}_{h\sim\hat{\pi}}\left[\Delta_R^2(h)\right] - D(\hat{\pi} \parallel \pi)\right\} && \le \mathbb{E}_{h\sim\pi}\left[\exp\left\{2n\Delta_R^2(h)\right\}\right] \\
\iff\ & \mathbb{E}_S\left[\exp\left\{2n\, \mathbb{E}_{h\sim\hat{\pi}}\left[\Delta_R^2(h)\right] - D(\hat{\pi} \parallel \pi)\right\}\right] && \le \mathbb{E}_S\left[\mathbb{E}_{h\sim\pi}\left[\exp\left\{2n\Delta_R^2(h)\right\}\right]\right] \\
\iff\ & \mathbb{E}_S\left[\exp\left\{2n\, \mathbb{E}_{h\sim\hat{\pi}}\left[\Delta_R^2(h)\right] - D(\hat{\pi} \parallel \pi)\right\}\right] && \le \mathbb{E}_{h\sim\pi}\left[\mathbb{E}_S\left[\exp\left\{2n\Delta_R^2(h)\right\}\right]\right],
\end{aligned}$$

where S is some set related to data, and the expectation over S can be exchanged with the expectation over $h \sim \pi$ since π is independent with data.

In addition, we define $V(P, Q) \equiv \sup\{P(A) - Q(A)\}$ to represent the total variation distance, where P, Q are two probability distributions on a measurable space (X, Σ), and $A \in \Sigma$ is a measurable event. [2] Moreover, let $p(h) = (R(h), 1 - R(h))$ and $\hat{p}(h) = \left(\hat{R}_n(h), 1 - \hat{R}_n(h)\right)$ be two probability vectors. [3] Since

$$\Delta_R^2(h) = R^2(h) + \hat{R}_n^2(h) - 2R(h)\hat{R}_n(h) \quad \text{and} \quad \left\|p(h) - \hat{p}(h)\right\|_1^2 = 4\left(R^2(h) + \hat{R}_n^2(h) - 2R(h)\hat{R}_n(h)\right),$$

it yields [4]

$$\begin{aligned}
\mathbb{E}_S\left[\exp\left\{2n\Delta_R^2(h)\right\}\right] &= \mathbb{E}_S\left[\exp\left\{2n \cdot \frac{1}{4}\left\|p(h) - \hat{p}(h)\right\|_1^2\right\}\right] \\
&\leq \mathbb{E}_S\left[\exp\left\{n\left\|p(h) - \hat{p}(h)\right\|_1^2\right\}\right] \\
&\leq \mathbb{E}_S[\exp\{nV(\hat{p}(h) - p(h))\}] \\
&\leq \mathbb{E}_S\left[\exp\left\{n\sqrt{\frac{1}{2}D(\hat{p}(h) \parallel p(h))}\right\}\right] \quad \text{(By Pinsker's inequality)} \\
&\leq \mathbb{E}_S\left[\exp\left\{n\sqrt{\frac{1}{2}\delta(\hat{p}(h) \parallel p(h))}\right\}\right] \\
&\leq \mathbb{E}_S[\exp\{n\delta(\hat{p}(h) \parallel p(h))\}] \\
&\leq 2\sqrt{n} \quad \forall\, n \geq 8 \quad \text{(By Theorem 1).}
\end{aligned}$$

Hence, by Markov's inequality, we have

$$\begin{aligned}
&\mathbb{E}_S\left[\exp\left\{2n\,\mathbb{E}_{h\sim\hat{\pi}}\left[\Delta_R^2(h)\right] - D(\hat{\pi} \parallel \pi)\right\}\right] \leq 2\sqrt{n} \\
\iff\ &\mathbb{P}\left(\exp\left\{2n\,\mathbb{E}_{h\sim\hat{\pi}}\left[\Delta_R^2(h)\right] - D(\hat{\pi} \parallel \pi)\right\} > \frac{2\sqrt{n}}{\delta}\right) \leq \delta;
\end{aligned}$$

that is, with probability at least $1 - \delta$,

$$2n\,\mathbb{E}_{h\sim\hat{\pi}}\left[\Delta_R^2(h)\right] \leq D(\hat{\pi} \parallel \pi) + \log\frac{2\sqrt{n}}{\delta}.$$

By Jensen's inequality, it alters to

$$(\mathbb{E}_{h\sim\hat{\pi}}[\Delta_R(h)])^2 \leq \mathbb{E}_{h\sim\hat{\pi}}\left[\Delta_R^2(h)\right] \leq \frac{D(\hat{\pi} \parallel \pi) + \log\frac{2\sqrt{n}}{\delta}}{2n},$$

and finally we prove the statement

$$\begin{aligned}
&\mathbb{E}_{h\sim\hat{\pi}}[R(h)] - \mathbb{E}_{h\sim\hat{\pi}}\left[\hat{R}_n(h)\right] \leq \sqrt{\frac{D(\hat{\pi} \parallel \pi) + \log\frac{2\sqrt{n}}{\delta}}{2n}} \\
\iff\ &\mathbb{E}_{h\sim\hat{\pi}}[R(h)] \leq \mathbb{E}_{h\sim\hat{\pi}}\left[\hat{R}_n(h)\right] + \sqrt{\frac{D(\hat{\pi} \parallel \pi) + \log\frac{2\sqrt{n}}{\delta}}{2n}}.
\end{aligned}$$

[2] This definition refers to the wikipedia page of Pinsker's inequality.

[3] Yu-Chieh thanks to TA Chung-En's tremendous help!

[4] Yu-Chieh thanks for the helpful and insightful discussion with my friendly classmates Chao-Hsun Yang (r10922148) and Cheng-Kang Chou (b09705011)!

After completing the proof of the statement, I tried to find the original paper for this inequality. I first searched the keyword *parameters free PAC-Bayes bound* since this bound includes no learning rates, but I found no identical result. Alquier (2023) gives a user-friendly introduction to various PAC-Bayes bounds. The most similar bound I found related to the problem is McAllesters bound, which was proposed and discussed by McAllester (1998), Maurer (2004), and McAllester (1999). Conducting the literature review provides me with more knowledge in this field and an understanding of PAC-Bayes learning. Additionally, I genuinely admire scholars developing algorithm bounds, which is always so difficult for me.

Problem 2: Online Convex Optimization

2.(a) Shifting Regret's definition, algorithms, and applications

I start from Cesa-Bianchi et al. (2012)[5] to see the shifting regret for the online convex optimization problem. I quote the explanation of shifting regret in the abstract of Cesa-Bianchi et al. (2012) here:

> ... A much harder criterion to minimize is shifting regret, which is defined as the difference between the learners cumulative loss and the cumulative loss of an arbitrary sequence of elements in $\mathcal{S}$. Shifting regret bounds are typically expressed in terms of the shift, a notion of regularity measuring the length of the trajectory in $\mathcal{S}$ described by the comparison sequence (i.e., the sequence of elements against which the regret is evaluated).

$\mathcal{S}$ denotes a fixed convex set, and then a convex loss function is defined on the same set $\mathcal{S}$. Cesa-Bianchi et al. (2012) multiply cited Herbster and Warmuth (2001); therefore, I also read Herbster and Warmuth (2001) to better understand the definition of shifting regret. Herbster and Warmuth (2001) explain the shifting bounds and the method to obtain such bounds more comprehensively. The definition of shifting regret is described as Definition 1.

Definition 1. A shifting regret, compared with a classical problem of prediction whose goal is to minimize the difference between Learner's cumulative loss and the cumulative loss of the best constant action in hindsight, is to compare the difference between Learner's cumulative loss and the cumulative loss of an arbitrary sequence of actions.

The goal of using the shifting regret is usually to track the best expert in the setting of prediction with experts. This form of the problem might be first proposed by Herbster and Warmuth (1998). Although the authors didn't use the term *shifting regret*, their objective is to bound the difference between Learner's cumulative loss and the loss of the best expert, which is exact the idea of the shifting regret.

In addition, Zhang et al. (2017) illustrate the shifting more easily:

> ... In the setting of prediction with expert advice, the dynamic regret is also referred to as tracking regret or shifting regret [Herbster and Warmuth, 1998, Cesa-bianchi et al., 2012]. The path-length of the comparator sequence is named as shift, which is just the number of times the expert changes.

[5] The authors seemed to change the article's title. I found another version called **A New Look at Shifting Regret**, which is the reason I started my exploration of shifting regret from this article.

A natural question is how to connect the online algorithms between the common regret and the shifting regret settings. Herbster and Warmuth (1998) and Herbster and Warmuth (2001) projected the online algorithm onto a constraint set: see **Definition 8** and **Definition 9** in Herbster and Warmuth (2001) for the process of the projection, and **Theorem 10** in Herbster and Warmuth (2001) for the shifting regret bound. Following the definitions and theorems in Herbster and Warmuth (2001), therefore, an online learning algorithm has a shifting regret guarantee if

1. An online algorithm $\mathscr{A}$ is based on a convex function F has an amortized analysis.
2. There exists a convex constraint set Γ over F with related parameters.
3. The shifting regret for $\mathscr{A}$ can be bounded in some bounds.

Why Herbster and Warmuth (2001) use a complex dynamic projection skill to show the shifting bound is to keep mirror descent from choosing points too close to the simplex boundary for tackling the behavior of the regularizer at the boundary of the simplex.

2.(b)

I give an algorithm proposed by Cesa-Bianchi et al. (2012). The assumptions on $x_t \in \mathcal{X}$ is $x_t = (x_{1,t}, \cdots, x_{D,t}) \in \Delta_D$ where D is the total number of experts and Δ_D denotes the simplex

$$\Delta_D = \left\{ q \in [0,1]^n \;:\; \left\| q \right\|_1 = 1 \right\}.$$

The loss function $f_t(\cdot) : \Delta_D^\top \times [0,1]^D \to \mathbb{R}$ is the inner product of $x_t^\top$ and ℓ_t, where $\ell_t = (\ell_{1,t}, \cdots, \ell_{D,t}) \in [0,1]^D$ can be regarded as a loss over all experts.

Cesa-Bianchi et al. (2012) use the generalized share algorithm to solve this problem. The algorithm is demonstrated as Algorithm 1.

Algorithm 1 The Generalized Share Algorithm

```
Input: learning rate η > 0 and a coefficient 0 ≤ α ≤ 1.
Initialize: x_1 ← w_1 = (1/D, ··· , 1/D).
for each round t = 1, ··· , T do
    LEARNER announces a prediction x_t.
    LEARNER suffers a loss f_t(x_t) = x_t^⊤ ℓ_t.
    for each j = 1, ··· , D do
        w_{j,t+1} ← p_{j,t} e^{−η ℓ_{j,t}} / Σ_{j=1}^{n} p_{i,t} e^{−η ℓ_{i,t}}     ▷ The current pre-weights.
        x_{j,t+1} ← α/D + (1 − α) w_{j,t+1}                                          ▷ Update the share.
    end for
end for
```

Here authors measure the regularity of the sequence $x^\otimes = (x_1, \cdots, x_T)$ in terms of the quantity

$$m\left(x^\otimes\right) = \sum_{t=1}^{T} d\left(x_{t+1}, x_t\right),$$

where $d(a,b) = \frac{1}{2}\|a-b\|_1$ for $a,b \in \Delta_D$. For all compared sequences $y_1, \cdots, y_T \in \Delta_D$ with $m(y^{\otimes}) \le m_0$, the shifting regret bound of Algorithm 1 is

$$\sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(y_t) \le \sqrt{\frac{T}{2}\left((m_0+1)\ln D + (T-1)h\left(\frac{m_0}{T-1}\right)\right)} \in \mathcal{O}\left(\sqrt{T\left(\ln D + (T-1)\ln(T-1)\right)}\right),$$

whenever η and α are optimally chosen in terms of m_0 and T, and

$$h(x) = -x\ln x - (1-x)\ln(1-x)$$

denotes the binary entropy for $x \in [0,1]$. Note that the statement of Problem 2 is equivalent to set $y_1 = \cdots = y_T$ under the setting of Cesa-Bianchi et al. (2012).

2.(c)

One of the applications that satisfies the assumptions above is to track the best expert, in which there is a small number of *base* experts and the goal of the Learner is to predict as well as the best *compound* expert. Consider D finance experts who predict the likelihood that the stock market will rise or fall. Each expert offers its predictions, and the master expert (Learner) decides the investment strategy among the prediction over all experts based on their performance and tries to predict as well as the best compound expert. In this setting, the stock market (Reality) acts based on the previous investment strategy and has a different action function $f_t(x)$ in each trial t. In addition, the master expert (Learner) aggregates all experts' suggestions and announces its strategy. Li and Hoi (2013) give a comprehensive survey, and Gaivoronski and Stella (2000) is an example.

2.(d) Adaptive Regret's definition, algorithms, and applications

Following the **Definition 1.1** in Hazan and Seshadhri (2009), which firstly introduced the concept of an adaptive regret, I describe an adaptive regret as Definition 2.

Definition 2. The adaptive regret of an online convex optimization algorithm $\mathscr{A}$ is defined as the maximum regret it achieves over any contiguous time interval. That is,

$$\mathit{Adaptive-Regret}_T(\mathscr{A}) := \sup_{I=[r,s]\subseteq[T]} \left\{ \sum_{t=r}^{s} f_t(x_t) - \min_{x_I^\star \in \mathcal{X}} \sum_{t=r}^{s} f_t(x_I^\star) \right\},$$

where $\mathcal{X} \subseteq \mathbb{R}^D$ is a convex domain, $x_t \in \mathcal{X}$, and $x_I^\star$ can vary arbitrarily with any interval I.

The adaptive regret is designed to measure how algorithm $\mathscr{A}$ performs compared to the optimum in hindsight for the same interval over every interval of time. A crucial point of the adaptive regret is that the comparison in cost is for a different optimum for any interval within $[T]$, which intuitively captures how well algorithm $\mathscr{A}$ tracks the progress of the dynamic nature since nature may undergo many different changes rather than stay statically permanently.

2.(e)

Although I cite Hazan and Seshadhri (2009) to illustrate adaptive regret, I use the same Algorithm 1 under the setting of Cesa-Bianchi et al. (2012) as an example here to ease note. The assumptions and notations on $\mathcal{X}$ and the loss function $f_t(\cdot)$ follow the same setting in **Problem 2.(b)**. Here I slightly modify the form of adaptive regret $Adaptive - Regret_T$ as

$$Adaptive - Regret_T^{\tau_0}(\mathscr{A}) := \max_{[r,s]\subset[1,T],\ s+1-r\leq\tau_0} \left\{ \sum_{t=r}^{s} f_t(x_t) - \min_{y^\star\in\Delta_D} \sum_{t=r}^{s} f_t(y^\star) \right\},$$

where $\tau_0 \in \{1, \cdots, T\}$, to satisfy the setting in Cesa-Bianchi et al. (2012). By Algorithm 1, the adaptive regret is bounded by

$$Adaptive - Regret_T^{\tau_0}(\mathscr{A}) \leq \sqrt{\frac{\tau_0}{2}\left(\tau_0 h\left(\frac{1}{\tau_0}\right) + \ln D\right)} \leq \sqrt{\frac{\tau_0}{2}\ln(eD\tau_0)} \in \mathcal{O}\left(\sqrt{\ln D}\right)$$

whenever η and α for Algorithm 1 are chosen optimally depending on τ_0 and T.

2.(f)

Since the algorithm in **Problem 2.(e)** is in the same setting and the algorithm in **Problem 2.(b)**, which follows Cesa-Bianchi et al. (2012), the same applications in **Problem 2.(c)** should fit the algorithm in **Problem 2.(e)**. Hence, I offer the same application as **Problem 2.(c)**.

One of the applications that satisfies the assumptions above is to track the best expert, in which there is a small number of *base* experts and the goal of the LEARNER is to predict as well as the best *compound* expert. Consider D finance experts who predict the likelihood that the stock market will rise or fall. Each expert offers its predictions, and the master expert (LEARNER) decides the investment strategy among the prediction over all experts based on their performance and tries to predict as well as the best compound expert. In this setting, the stock market (REALITY) acts based on the previous investment strategy and has a different action function $f_t(x)$ in each trial t. In addition, the master expert (LEARNER) aggregates all experts' suggestions and announces its strategy. Li and Hoi (2013) give a comprehensive survey, and Gaivoronski and Stella (2000) is an example.

2.(g) Dynamic Regret's definition, algorithms, and applications

Following the definition in Zhang et al. (2020) and Zinkevich (2003), the dynamic regret is demonstrated as Definition 3.

Definition 3. The dynamic regret is defined as the difference between the cumulative loss of LEARNER and that of a sequence of comparators $y_1, \cdots, y_T \in \mathcal{X}$:

$$Dynamic - Regret_T(\mathscr{A}) := \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(y_t).$$

2.(h)

I provide the algorithm proposed by Zhang et al. (2017). They propose the squared path length and compare it with the original path length demonstrated in Zinkevich (2003), but they also slightly modify the dynamic regret into a more restricted version in Definition 4, defined with respect to a sequence of minimizers of the loss functions due to its greater mathematical tractability.

Definition 4. The *restricted* dynamic regret is defined as the difference between the cumulative loss of LEARNER and that of a sequence of *local minimizer*:

$$Dynamic - Regret_T^\star(\mathscr{A}) := \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} \min_{y\in\mathcal{X}} f_t(y)$$

Before demonstrating the algorithm, let us define the notations. The path length of the comparator sequence is

$$\mathcal{P}_T^\star := \sum_{t=2}^{T} \left\|x_t^\star - x_{t-1}^\star\right\|_1,$$

where $x_t^\star \in \arg\min_{x\in\mathcal{X}} f_t(x)$. The *squared* path length of the comparator sequence is

$$\mathcal{S}_T^\star := \sum_{t=2}^{T} \left\|x_t^\star - x_{t-1}^\star\right\|_1^2,$$

which could be much smaller than $\mathcal{P}_T^\star$ when the local variations are small. Zhang et al. (2017) assume $\mathcal{X}$ is a convex set, and each f_t is μ-strongly convex and ℓ-smooth over $\mathcal{X}$ and $\left\|\nabla f_t(x)\right\|_2 \le G$ for all $x \in \mathcal{X}$ with G denoting the maximum norm of the gradient of $f_t(x)$ (also see Tomaso et al. (2019)); $\mathrm{Proj}(\cdot)$ denotes the projection onto the nearest point in $\mathcal{X}$. The algorithm is described as Algorithm 2.

Algorithm 2 Online Multiple Gradient Descent (OMGD)

```
Input: the step size η > 0 and the number of inner iterations K.
Initialize: x_1 ← any point in X.
for each round t = 1, ··· , T do
    LEARNER announces a prediction x_t.
    LEARNER suffers a loss f_t(x_t).
    Let u_t^1 ← x_t.
    for each j = 1, ··· , K do
        u_t^{j+1} ← Proj_X( u_t^j − η∇f_t(u_t^j) )
    end for
    x_{t+1} ← u_t^{K+1}
end for
```

By setting $\eta \le \frac{1}{\ell}$ and $K = \left\lceil \frac{\frac{1}{\eta}+\mu}{2\mu} \ln 4 \right\rceil$, Algorithm 2 achieves $\mathcal{O}\left(\min\left\{\mathcal{P}_T^\star, \mathcal{S}_T^\star\right\}\right)$.

2.(i)

I sincerely think the *track the best expert* scenario can be still regarded as an application in this algorithm. Therefore, I still use the same description as **Problem 2.(c)**.

One of the applications that satisfies the assumptions above is to track the best expert, in which there is a small number of *base* experts and the goal of the LEARNER is to predict as well as the best *compound* expert. Consider D finance experts who predict the likelihood that the stock market will rise or fall. Each expert offers its predictions, and the master expert (LEARNER) decides the investment strategy among the prediction over all experts based on their performance and tries to predict as well as the best compound expert. In this setting, the stock market (REALITY) acts based on the previous investment strategy and has a different action function $f_t(x)$ in each trial t. In addition, the master expert (LEARNER) aggregates all experts' suggestions and announces its strategy. Li and Hoi (2013) give a comprehensive survey, and Gaivoronski and Stella (2000) is an example.

References

Alquier, Pierre (2023) "User-friendly introduction to PAC-Bayes bounds."

Cesa-Bianchi, Nicolò, Pierre Gaillard, Gábor Lugosi, and Gilles Stoltz (2012) "Mirror Descent Meets Fixed Share (and Feels No Regret)," in *Proceedings of the 25th International Conference on Neural Information Processing Systems*, 980–988.

Gaivoronski, Alexei and Fabio Stella (2000) "Stochastic Nonstationary Optimization for Finding Universal Portfolios," *Annals of Operations Research*, 100 (1), 165–188.

Hazan, Elad and Comandur Seshadhri (2009) "Efficient Learning Algorithms for Changing Environments," in *Proceedings of the 26th Annual International Conference on Machine Learning*, 393–400.

Herbster, Mark and Manfred K. Warmuth (1998) "Tracking the Best Expert," *Machine Learning*, 32, 151–178.

——— (2001) "Tracking the Best Linear Predictor," *Journal of Machine Learning Research*, 1, 281–309.

Kearns, M. J. and U. V. Vazirani (1994) *An Introduction to Computational Learning Theory*: MIT Press.

Li, Bin and Steven C. H. Hoi (2013) "Online Portfolio Selection: A Survey."

Maurer, Andreas (2004) "A Note on the PAC Bayesian Theorem."

McAllester, David A. (1998) "Some PAC-Bayesian Theorems," in *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, 230–234.

——— (1999) "PAC-Bayesian model averaging," in *Annual Conference Computational Learning Theory*.

Tomaso, Poggio, Rosasco Lorenzo, Rakhlin Alexander, and Banburski Andrzej (2019) "Lecture note in Statistical Learning Theory and Applications."

Wu, Pei-Yuan (2021) "Lecture note in Probably Approximately Correct Learning."

Zhang, Lijun, Shiyin Lu, and Tianbao Yang (2020) "Minimizing Dynamic Regret and Adaptive Regret Simultaneously."

Zhang, Lijun, Tianbao Yangt, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou (2017) "Improved Dynamic Regret for Non-Degenerate Functions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 732–741.

Zinkevich, Martin (2003) "Online Convex Programming and Generalized Infinitesimal Gradient Ascent," in *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, 928–935.