Online learning – CMPUT 654

Gábor Bartók Dávid Pál Csaba Szepesvári István Szita

October 20, 2011

Contents

Chapter 1

Shooting Game

Imagine the following repeated game—shooting game. In each round $t = 1, 2, \ldots, n$

- We choose a point $\widehat{p}_t \in \mathbb{R}^d$
- Then, the environment chooses a point $y_t \in \mathbb{R}^d$ with Euclidean norm $\|y_t\| \leq 1$
- We suffer a loss $\ell(p_t, y_t) = \|\widehat{p}_t - y_t\|^2$

If we knew the sequence $y_1, y_2, \ldots, y_n$ ahead of time and wanted to use the best *fixed* point to predict, we would choose the point that minimizes the total loss:

$$p_n^* = \operatorname*{argmin}_{p \in \mathbb{R}^d} \sum_{t=1}^{n} \ell(p, y_t) = \operatorname*{argmin}_{p \in \mathbb{R}^d} \sum_{t=1}^{n} \|p - y_t\|^2$$

For the loss function $\ell(p, y) = \|p - y\|^2$ it is not hard to calculate p_n^* explicitly:

$$p_n^* = \frac{1}{n} \sum_{t=1}^{n} y_t$$

(Do the calculation!)

An online algorithm does not have the luxury of knowing the sequence ahead of time, and thus it needs to choose $\widehat{p}_t$ based on $y_1, y_2, \ldots, y_{t-1}$ only. The *regret* is the extra loss that it suffers compared to p_n^*:

$$R_n = \sum_{t=1}^{n} \ell(\widehat{p}_t, y_t) - \sum_{t=1}^{n} \ell(p_n^*, y_n) \ .$$

A particularly good algorithm for the shooting game is the Follow-The-Leader (FTL) algorithm, which chooses the point that minimizes the loss on the points $y_1, y_2, \ldots, y_{t-1}$ seen so far:

$$\widehat{p}_t = \operatorname*{argmin}_{p \in \mathbb{R}^d} \sum_{t=1}^{n} \ell(p, y_t) = \operatorname*{argmin}_{p \in \mathbb{R}^d} \sum_{s=1}^{t-1} \|p - y_s\|^2$$

(The point $\widehat{p}_t$ is the current "leader".) Again, there exists an explicit formula for $\widehat{p}_t$:

$$\widehat{p}_t = \frac{1}{t-1} \sum_{s=1}^{t-1} y_s$$

Note that, using our notation, $\widehat{p}_t = p^*_{t-1}$. For concreteness, we define $\widehat{p}_1 = p^*_0 = 0$.

We analyze the regret of FOLLOW-THE-LEADER algorithm by using the following lemma:

Lemma 1.1 (Hannan's "Follow The Leader–Be The Leader" lemma)**.** *For any sequence $y_1, y_2, \ldots, y_n \in \mathbb{R}^d$ we have*

$$\sum_{t=1}^{n} \ell(p^*_t, y_t) \leq \sum_{t=1}^{n} \ell(p^*_n, y_t)\,.$$

Proof. We prove the lemma by induction on n. For $n = 1$ the inequality is equivalent to:

$$\ell(p^*_1, y_1) \leq \ell(p^*_1, y_1)\,,$$

which is trivially satisfied with equality. Now, take some $n \geq 2$ and assume that the desired inequality holds up to $n-1$. Our goal is to prove that it also holds for n, i.e., to show that

$$\sum_{t=1}^{n} \ell(p^*_t, y_t) \leq \sum_{t=1}^{n} \ell(p^*_n, y_t)\,.$$

If we cancel $\ell(p^*_n, y_n)$ on both sides, we get an equivalent statement

$$\sum_{t=1}^{n-1} \ell(p^*_t, y_t) \leq \sum_{t=1}^{n-1} \ell(p^*_n, y_t)\,,$$

which we prove by writing it as a chain of two inequalities, each of which is easy to justify:

$$\sum_{t=1}^{n-1} \ell(p^*_t, y_t) \leq \sum_{t=1}^{n-1} \ell(p^*_{n-1}, y_t) \leq \sum_{t=1}^{n-1} \ell(p^*_n, y_t)$$

The first inequality holds by the induction hypothesis. The second follows from that $p^*_{n-1} = \operatorname{argmin}_{p \in \mathbb{R}^d} \sum_{t=1}^{n-1} \ell(p^*_{n-1}, y_t)$ is a minimizer of the sum losses, so the sum of losses evaluated at p^*_n must be at least as large as the sum evaluated at p^*_{n-1}. This finishes the proof. □

We use the lemma to upper bound the regret of the FOLLOW-THE-LEADER algorithm:

$$R_n = \sum_{t=1}^{n} \ell(\widehat{p}_t, y_t) - \sum_{t=1}^{n} \ell(p^*_n, y_t) = \sum_{t=1}^{n} \ell(p^*_{t-1}, y_t) - \sum_{t=1}^{n} \ell(p^*_n, y_t) \leq \sum_{t=1}^{n} \ell(p^*_{t-1}, y_t) - \sum_{t=1}^{n} \ell(p^*_t, y_t)\,.$$

Now, the rightmost expression can be written as:[1]

$$\begin{aligned}\sum_{t=1}^{n}\ell(p_{t-1}^*,y_t)-\sum_{t=1}^{n}\ell(p_{t-1}^*,y_t)&=\sum_{t=1}^{n}\|p_{t-1}^*-y_t\|^2-\sum_{t=1}^{n}\|p_{t-1}^*-y_t\|^2\\&=\sum_{t=1}^{n}\langle p_{t-1}^*-p_t^*,p_{t-1}^*+p_t^*-2y_t\rangle\,,\end{aligned}$$

where we have used that $\|u\|^2-\|v\|^2=\langle u-v,u+v\rangle$. The last expression can be further upper bounded by the Cauchy-Schwarz inequality, that states that $|\langle u,v\rangle|\le\|u\|\cdot\|v\|$, and the triangle inequality that states that $\|u+v\|\le\|u\|+\|v\|$:

$$\begin{aligned}\sum_{t=1}^{n}\langle p_{t-1}^*-p_t^*,p_{t-1}^*+p_t^*-2y_t\rangle&\le\sum_{t=1}^{n}\|p_{t-1}^*-p_t^*\|\cdot\|p_{t-1}^*+p_t^*-2y_t\|\\&\le\sum_{t=1}^{n}\|p_{t-1}^*-p_t^*\|\cdot\big(\|p_{t-1}^*\|+\|p_t^*\|+2\|y_t\|\big)\,.\end{aligned}$$

We use that $y_1,y_2,\ldots,y_n$ have norm at most 1 and thus so do the averages $p_0^*,p_1^*,\ldots,p_n^*$:

$$R_n\le\sum_{t=1}^{n}\|p_{t-1}^*-p_t^*\|\cdot\big(\|p_{t-1}^*\|+\|p_t^*\|+2\|y_t\|\big)\le4\sum_{t=1}^{n}\|p_{t-1}^*-p_t^*\|\,.$$

It remains to upper bound $\|p_{t-1}^*-p_t^*\|$, which we do by substituting $p_t^*=\frac{(t-1)p_{t-1}^*+y_t}{t}$, using the triangle inequality and $\|y_t\|\le1$, $\|p_{t-1}^*\|\le1$:

$$\|p_{t-1}^*-p_t^*\|=\left\|p_{t-1}^*-\frac{(t-1)p_{t-1}^*+y_t}{t}\right\|=\frac{1}{t}\left\|p_{t-1}^*-y_t\right\|\le\frac{1}{t}\left(\|p_{t-1}^*\|+\|y_t\|\right)\le\frac{2}{t}\,.$$

In summary,

$$R_n\le4\sum_{t=1}^{n}\|p_{t-1}^*-p_t^*\|\le8\sum_{t=1}^{n}\frac{1}{t}\le8(1+\ln n)\,.$$

The last inequality, $\sum_{t=1}^{n}\frac{1}{t}\le1+\ln n$, is a well known inequality for Harmonic numbers.

Theorem 1.2 (FTL Regret Bound)**.** *If $y_1,y_2,\ldots,y_n$ is any sequence of points of the unit ball of $\mathbb{R}^d$ then the* Folllow-The-Leader *algorithm that in round t predicts $\widehat{p}_t=p_{t-1}^*=\frac{1}{t-1}\sum_{s=1}^{t-1}y_s$ has regret at most $8(1+\ln n)$.*

1.1 Exercises

Exercise 1.1.

[1]For vectors $u,v\in\mathbb{R}^d$, the inner product of u and v is $\langle u,v\rangle=u^\top v$. Further, $\|v\|=\sqrt{\langle v,v\rangle}$.

(a) Consider an *arbitrary* deterministic online algorithm A for the shooting game. Show that for any $n \geq 1$ there exists a sequence $y_1, y_2, \ldots, y_n$ such that A has non-positive regret. Justify your answer.

(b) Consider an *arbitrary* deterministic online algorithm A for the shooting game. Show that for any $n \geq 1$ there exists a sequence $y_1, y_2, \ldots, y_n$ in the unit ball of $\mathbb{R}^d$ such that A has non-negative regret. Justify your answer.

Hint: Imagine that the algorithm is given to you as a subroutine, which you can call at any time. More precisely, you will have two functions: `init` and `predict`. The first initializes the algorithm's internal variables (whatever they are) and returns the algorithm's first prediction and the second, taking the previous outcome gives you the new prediction. That the algorithm is deterministic means that if you call it with the same sequence of outcomes, it will give the same sequence of predictions. Can you write a code that will produce an outcome sequence that makes the regret behave as desired? Remember, your code is free to use the above two functions! Do not forget to justify why your solution works.

Exercise 1.2. (Follow-The-Leader) Consider the FOLLOW-THE-LEADER (FTL) algorithm for the shooting game which chooses $\widehat{p}_1 = \mathbf{0}$ in round $t = 1$.

(a) Prove that for any sequence $y_1, y_2, \ldots, y_n$, $n \geq 1$, the regret of the FTL algorithm is nonnegative.

(b) According to Theorem 1.2, the regret of the FTL algorithm on any sequence $y_1, y_2, \ldots, y_n$ in the unit ball of $\mathbb{R}^d$ is at most $O(\log n)$. This problem asks you to show that the upper bound $O(\log n)$ for FTL is tight. More precisely, for any $n \geq 1$ construct a sequence $y_1, y_2, \ldots, y_n$ in the unit ball such that FTL has regret at least $\Omega(\log n)$.

Hint for Part (b): Consider the sequence $y_t = (-1)^t v$ where v is an arbitrary unit vector. First solve the case when n is even. The inequality $1 + 1/2 + \cdots + 1/n \geq \ln n$ might be useful.

Exercise 1.3. (Stochastic model) Consider the situation when $y_1, y_2, \ldots, y_n$ are generated i.i.d. (independently and identically distributed) according to some probability distribution. For this case, we have seen in class an alternative definition of regret:

$$\widetilde{R}_n = \sum_{t=1}^{n} \ell(\widehat{p}_t, y_t) - \min_{p \in \mathbb{R}^d} \sum_{t=1}^{n} \mathbf{E}[\ell(p, y_t)] \; .$$

Prove that for any online learning algorithm $\mathbf{E}[\widetilde{R}_n] \leq \mathbf{E}[R_n]$.

(Morale: The result that you're asked to prove shows the strength of the non-stochastic framework in which no distributional assumptions are placed on $y_1, y_2, \ldots, y_n$. Namely, it shows that any upper bound $R_n \leq f(n)$ in the non-stochastic framework implies an upper bound $\mathbf{E}[\widetilde{R}_n] \leq f(n)$ in the stochastic model.)
Hint: Write what you have to prove. Cancel identical terms.

Chapter 2

Weighted Majority Algorithm

Consider the prediction problem, in which we want to predict whether tomorrow it's going be rainy or sunny. For our disposal, we have N experts that predict rain/sunshine. We do this repeatedly over a span of n days. In round (day) $t = 1, 2, \ldots, n$:

- Each expert $i = 1, 2, \ldots, N$ predicts $f_{i,t} \in \{0, 1\}$
- We predict $\widehat{p}_t \in \{0, 1\}$ based on the experts' predictions
- Then, nature reveals $y_t \in \{0, 1\}$
- We suffer the loss $\ell(\widehat{p}_t, y_t) = \mathbb{I}\{\widehat{p}_t \neq y_t\}$ and
  each expert $i = 1, 2, \ldots, N$ suffers loss $\ell(f_{i,t}, y_t) = \mathbb{I}\{\widehat{f}_{i,t} \neq y_t\}$

(The symbol $\mathbb{I}$ denotes the indicator function. If $\ell(\widehat{p}_t, y_t) = 1$ we make a mistake and if $\ell(f_{i,t}, y_t) = 1$, expert i makes a mistake.) We are interested in algorithms which make as few mistakes as possible.

2.1 The Halving Algorithm

If we know that there is an expert that never makes a mistakes (but we don't know which one), we can use the Halving Algorithm. The algorithm maintains the set S_t of experts that did not make any mistakes in rounds $1, 2, \ldots, t$. We call S_t the set of alive experts at the end of round t. Initially, $S_0 = \{1, 2, \ldots, N\}$. In round t, the algorithm predicts what the majority of experts in set S_{t-1} predict. In round $t = 1, 2, \ldots, n$:

- Let $S_{t-1} \subseteq \{1, 2, \ldots, N\}$ be the experts that did not make any mistake so far.
- The algorithm receives the experts' predictions $f_{1,t}, f_{2,t}, \ldots, f_{N,t} \in \{0, 1\}$.
- The set S_{t-1} is split into $S^0_{t-1} = \{i \in S_{t-1} \ : \ f_{i,t} = 0\}$ and $S^1_{t-1} = \{i \in S_{t-1} \ : \ f_{i,t} = 1\}$.
- If $|S^0_{t-1}| > |S^1_{t-1}|$, the algorithm predicts $\widehat{p}_t = 0$, otherwise it predicts $\widehat{p}_t = 1$.

- The algorithm then receives y_t (and suffers the loss $\ell(\widehat{p}_t, y_t) = \mathbb{I}\{\widehat{p}_t \neq y_t\}$).
- The set of alive experts is updated: $S_t = S_{t-1}^{y_t}$.

2.1.1 Analysis

If in round t the HALVING ALGORITHM makes a mistake then $|S_t| = |S_{t-1}^{y_t}| \leq |S_{t-1}|/2$. Since we assume that there is an expert that never makes a mistake we have at all time steps that $|S_t| \geq 1$ and therefore, the algorithm can not make more than $\log_2 |S_0| = \log_2 N$ mistakes.

2.2 The Weighted-Majority Algorithm

If we do not know anything about the experts and, in particular, if there might not be an infallible expert, we need a different algorithm because the HALVING ALGORITHM will eliminate all experts. One algorithm, which can be seen as a natural generalization of the HALVING ALGORITHM to this setting is the so-called WEIGHTED-MAJORITY ALGORITHM (WMA). For each expert i, WMA maintains a positive weight $w_{i,t}$ and whenever expert i makes a mistake, this weight is multiplied by a factor $\beta \in [0, 1)$. The weight $w_{i,t}$ represents how much WMA trusts in expert i. The algorithm then combines the experts' prediction by using a weighted-majority vote.

Formally, the algorithm works as follows: Initially, the weights of experts are $w_{1,0} = w_{2,0} = \cdots = w_{N,0} = 1$. In round $t = 1, 2, \ldots, n$, WMA does the following:

- It receives the experts' predictions $f_{1,t}, f_{2,t}, \ldots, f_{N,t} \in \{0, 1\}$.
- It predicts according to the weighted-majority vote:

$$\widehat{p}_t = \mathbb{I}\{\textstyle\sum_{i=1}^{N} w_{i,t-1} f_{i,t} > \sum_{i=1}^{N} w_{i,t-1}(1 - f_{i,t})\}\,.$$

- It receives y_t (and suffers the loss $\ell(\widehat{p}_t, y_t) = \mathbb{I}\{\widehat{p}_t \neq y_t\}$).
- It updates $w_{i,t} = w_{i,t-1}\beta^{\mathbb{I}\{f_{i,t} \neq y_t\}}$, for each $i = 1, 2, \ldots, N$.

2.2.1 Analysis

To analyze the number of mistakes of WMA, we use the notation

$$J_{t,\text{good}} = \{i \ : \ f_{i,t} = y_t\}\,, \qquad J_{t,\text{bad}} = \{i \ : \ f_{i,t} \neq y_t\}\,,$$
$$W_t = \sum_{i=1}^{N} w_{i,t}\,, \qquad W_{t,J} = \sum_{i \in J} w_{i,t}\,.$$

Claim 2.1. *$W_t \leq W_{t-1}$ and if $\widehat{p}_t \neq y_t$ then $W_t \leq \frac{1+\beta}{2} W_{t-1}$.*

Proof. Since $w_{i,t} \le w_{i,t-1}$ we have $W_t \le W_{t-1}$. Now, if $\widehat{p}_t \neq y_t$ then the good experts were in minority:

$$W_{t-1,J_{t,\text{good}}} \le W_{t-1}/2\,.$$

Using this inequality we can upper bound W_t:

$$\begin{aligned} W_t &= W_{t-1,J_{t,\text{good}}} + \beta W_{t-1,J_{t,\text{bad}}} \\ &= W_{t-1,J_{t,\text{good}}} + \beta\left(W_{t-1} - W_{t-1,J_{t,\text{good}}}\right) \\ &= (1-\beta)W_{t-1,J_{t,\text{good}}} + \beta W_{t-1} \\ &\le (1-\beta)W_{t-1}/2 + \beta W_{t-1} \\ &= \frac{1+\beta}{2} W_{t-1}\,, \end{aligned}$$

and we're done. □

The claim we've just proved allows to upper bound W_n in terms of $\widehat{L}_n = \sum_{t=1}^n \mathbb{I}\{\widehat{p}_t \neq y_t\}$, the number of mistakes made by the WMA:

$$W_n \le \left(\frac{1+\beta}{2}\right)^{\widehat{L}_n} W_0\,.$$

On the other hand, if $L_{i,n} = \sum_{t=1}^n \mathbb{I}\{f_{i,t} \neq y_t\}$ denotes the number of mistakes made by expert i, the weight of expert i at the end of algorithm is $w_{i,n} = \beta^{L_{i,n}}$ which in particular means that

$$\beta^{L_{i,n}} = w_{i,n} \le W_n\,.$$

Putting these two inequalities together and using that $W_0 = N$, we get

$$\beta^{L_{i,n}} \le \left(\frac{1+\beta}{2}\right)^{\widehat{L}_n} N\,.$$

Taking logarithm and rearranging, we get upper bound on the number of mistakes of WMA:

$$\widehat{L}_n \le \frac{\ln\left(\frac{1}{\beta}\right) L_{i,n} + \ln N}{\ln\left(\frac{2}{1+\beta}\right)}$$

which holds for any expert i. If $L^*_{i,n} = \min_{1\le i\le N} L_{i,n}$ denotes the loss of the *best expert*, we get

$$\widehat{L}_n \le \frac{\ln\left(\frac{1}{\beta}\right) L^*_{i,n} + \ln N}{\ln\left(\frac{2}{1+\beta}\right)}\,.$$

Note that WMA with $\beta = 0$ coincides with the Halving Algorithm and if $L^*_{i,n} = 0$ we recover the bound $\widehat{L}_n \le \log_2 N$, which was shown to hold for the Halving Algorithm.

Chapter 3

Exponentially Weighted Average Forecaster: Continuous Predictions

We consider the situation when the experts' predictions are continuous. Each expert $i = 1, 2, \ldots, N$ in round t predicts $f_{i,t} \in D$ where D is a convex subset of some vector space.[1] In each round $t = 1, 2, \ldots, n$ we play a game according to the following protocol:

- Experts predict $f_{1,t}, f_{2,t}, \ldots, f_{N,t} \in D$.
- We predict $\widehat{p}_t \in D$ based on the experts' predictions.
- Then, the environment reveals an outcome $y_t \in Y$.
- We suffer the loss $\ell(\widehat{p}_t, y_t)$ and each expert $i = 1, 2, \ldots, N$ suffers a loss $\ell(f_{i,t}, y_t)$.

The set Y of outcomes can be arbitrary, however we will make two assumptions on the loss function $\ell : D \times Y \to \mathbb{R}$:

(a) $\ell(p, y) \in [0, 1]$ for any $p \in D$, $y \in Y$;

(b) for any fixed $y \in Y$, the function $\ell(\cdot, y)$ is convex.

Our goal is to design an algorithm that has small regret:

$$R_n = \widehat{L}_n - \min_{1 \le i \le N} L_{i,n} \,,$$

where $\widehat{L}_n = \sum_{t=1}^{n} \ell(\widehat{p}_t, y_t)$ is the cumulated loss of the algorithm and $L_{i,n} = \sum_{t=1}^{n} \ell(f_{i,t}, y_t)$ is the cumulated loss of expert i.

[1] Recall that a subset D of a real vector space is *convex* if for any $x, y \in D$ and any real numbers $\alpha, \beta \ge 0$ such that $\alpha + \beta = 1$ the point $\alpha x_1 + \beta x_2$ belongs to D. Further, a real-valued function f with a convex domain D is *convex* if for any $x_1, x_2 \in D$, $\alpha, \beta \ge 0$, $\alpha + \beta = 1$, $f(\alpha x_1 + \beta x_2) \le \alpha f(x_1) + \beta f(x_2)$. An equivalent characterization of convex functions is the following: Consider the *graph* $G_f = \{(p, f(p)) \ : \ p \in D\}$ of f (this is a surface). Then, f is convex if and only if for any $x \in D$, any hyperplane H_x tangent to G_f at x is below G_f in the sense that for any point $(x', y') \in H_x$, $y' \le f(x')$.

3.1 The Exponentially Weighted Average forecaster

A very good algorithm for this problem is the Exponentially Weighted Average forecaster (EWA). For each expert i, EWA maintains a weight $w_{i,t} = e^{-\eta L_{i,t}}$ which depends on the loss $L_{i,t}$ of expert i up to round t and a parameter $\eta > 0$ which will be chosen later. In each round t, EWA predicts by taking the convex combination of the experts' predictions $f_{1,t}, f_{2,t}, \ldots, f_{N,t}$ with the current weights.

More precisely, EWA is as follows: First, the weights are initialized to $w_{1,0} = w_{2,0} = \cdots = w_{N,0} = 1$. Further, the sum of the weights is stored in $W_0 = N$. Then, in each round $t = 1, 2, \ldots, n$ EWA does the following steps:

- It receives experts' predictions $f_{1,t}, f_{2,t}, \ldots, f_{N,t} \in D$.
- It predicts
$$\widehat{p}_t = \frac{\sum_{i=1}^N w_{i,t-1} f_{i,t}}{W_{t-1}} .$$
- Then, the environment reveals an outcome $y_t \in Y$.
- EWA suffers the loss $\ell(\widehat{p}_t, y_t)$ and each expert $i = 1, 2, \ldots, N$ suffers a loss $\ell(f_{i,t}, y_t)$.
- EWA updates the weights by $w_{i,t} = w_{i,t-1} e^{-\eta \ell(f_{i,t}, y_t)}$, where $i = 1, 2, \ldots, N$.
- EWA update $W_t = \sum_{i=1}^N w_{i,t}$.

For numerical stability, in software implementations instead of working with the weights $w_{i,t}$ and their sum, one would only store and maintain the normalized weights $\widehat{w}_{i,t} = w_{i,t}/W_t$.

3.2 Analysis

To analyze the regret of EWA we will need two results. We prove only the first result.

Lemma 3.1 (Jensen's inequality)**.** *Let V is any real vector space. If X is a V-valued random variable such that $\mathbf{E}[X]$ exists and $f : V \to \mathbb{R}$ is a convex function then*

$$f(\mathbf{E}[X]) \le \mathbf{E}[f(X)] .$$

Proof. Consider $\bar{x} = \mathbf{E}[X]$ and the hyperplane $H_{\bar{x}}$ which is tangent to the graph G_f of f at $\bar{x}$. We can write $H_{\bar{x}} = \{(x', \langle a, x' \rangle + b) : x' \in V\}$ with some $a \in V$, $b \in \mathbb{R}$. Since f is convex, $H_{\bar{x}}$ is below G_f. Thus, $\langle a, X \rangle + b \le f(X)$. Since $H_{\bar{x}}$ is tangent to G_f *at* $\bar{x}$, $\langle a, \bar{x} \rangle + b = f(\bar{x}) = f(\mathbf{E}[X])$. Taking expectations of both sides, and using the equality just obtained, we get $f(\mathbf{E}[X]) = \langle a, \mathbf{E}[X] \rangle + b \le \mathbf{E}[f(X)]$, which proves the statement. □

Lemma 3.2 (Hoeffding's lemma)**.** *If X is a real-valued random variable lying in $[0, 1]$ then for any $s \in \mathbb{R}$*

$$\mathbf{E}[e^{sX}] \le e^{s\,\mathbf{E}[X] + s^2/8} .$$

The analysis will be similar to the analysis of WMA. First we lower bound W_n

$$W_n \geq w_{i,n} = e^{-\eta L_{i,n}}$$

and then upper bound it by upper bounding the terms of

$$W_n = W_0 \cdot \frac{W_1}{W_0} \cdot \frac{W_2}{W_1} \cdots \frac{W_n}{W_{n-1}} \,.$$

Let us thus upper bound the fraction $\frac{W_t}{W_{t-1}}$ for some fixed t ($t \in \{1, \ldots, n\}$). We can express this fraction as

$$\frac{W_t}{W_{t-1}} = \sum_{i=1}^{N} \frac{w_{i,t}}{W_{t-1}} = \sum_{i=1}^{N} \frac{w_{i,t-1} e^{-\eta \ell(f_{i,t}, y_t)}}{W_{t-1}} = \sum_{i=1}^{N} \frac{w_{i,t-1}}{W_{t-1}} e^{-\eta \ell(f_{i,t}, y_t)} = \mathbf{E}\left[e^{-\eta \ell(f_{I_t,t}, y_t)}\right] \,,$$

where I_t is a random variable that attains values in the set $\{1, 2, \ldots, N\}$ and has distribution $\Pr(I_t = i) = \frac{w_{i,t-1}}{W_{t-1}}$. Applying Hoeffding's lemma to the random variable $X = \ell(f_{I_t,t}, y_t)$ gives

$$\mathbf{E}\left[e^{-\eta \ell(f_{I_t,t}, y_t)}\right] \leq e^{-\eta\, \mathbf{E}[\ell(f_{I_t,t}, y_t)] + \eta^2/8}$$

Applying Jensen's inequality on the expression in the exponent $\mathbf{E}[\ell(f_{I_t,t}, y_t)]$ (exploiting that ℓ is convex in its first argument) and then using the definition of $\widehat{p}_t$, we have

$$\mathbf{E}[\ell(f_{I_t,t}, y_t)] \geq \ell(\mathbf{E}[f_{I_t,t}], y_t) = \ell\left(\sum_{i=1}^{N} \frac{w_{i,t-1}}{W_{t-1}} f_{i,t},\ y_t\right) = \ell(\widehat{p}_t, y_t) \,.$$

Putting things together we get that

$$\frac{W_t}{W_{t-1}} \leq e^{-\eta\, \mathbf{E}[\ell(f_{I_t,t}, y_t)] + \eta^2/8} \leq e^{-\eta \ell(\widehat{p}_t, y_t) + \eta^2/8}$$

which gives and upper on W_n as follows:

$$W_n = W_0 \cdot \frac{W_1}{W_0} \cdot \frac{W_2}{W_1} \cdots \frac{W_n}{W_{n-1}} \leq W_0 \cdot e^{-\eta \sum_{t=1}^{n} \ell(\widehat{p}_t, y_t) + n\eta^2/8} = W_0 \cdot e^{-\eta \widehat{L}_n + n\eta^2/8} \,.$$

Combining the upper and lower bounds on W_n and substituting $W_0 = N$ we obtain

$$e^{-\eta L_{i,n}} \leq N \cdot e^{-\eta \widehat{L}_n + n\eta^2/8} \,.$$

Taking logarithm, diving by $\eta > 0$ and rearranging gives us an upper bound on the loss

$$\widehat{L}_n \leq L_{i,n} + \frac{\ln N}{\eta} + \frac{\eta n}{8} \,.$$

We summarize the result in the following theorem:

Theorem 3.3. *Assume D is convex subset of some vector space. Let $\ell : D \times Y \to [0, 1]$ be convex in its first argument. Then, the loss of the EWA forecaster is upper bounded by*

$$\widehat{L}_n \leq \min_{1 \leq i \leq N} L_{i,n} + \frac{\ln N}{\eta} + \frac{\eta n}{8} \,.$$

With $\eta = \sqrt{\frac{8 \ln N}{n}}$, $\quad \widehat{L}_n \leq \min_{1 \leq i \leq N} L_{i,n} + \sqrt{\frac{n}{2} \ln N}.$

3.3 Exercises

Exercise 3.1. For $A, B > 0$, find $\operatorname{argmin}_{\eta>0}(1/\eta\, A + \eta\, B)$ and also $\min_{\eta>0}(1/\eta\, A + \eta\, B)$.

Exercise 3.2. It is known that if X is a random variable taking values in $[0,1]$, then for any $s \in \mathbb{R}$, $\mathbf{E}\left[e^{sX}\right] \le \exp((e^s - 1)\,\mathbf{E}\,[X])$. For $s > 0$ fixed, this bound becomes smaller than what we would get from Hoeffding's inequality as $\mathbf{E}\,[X] \to 0$.

(a) Use this inequality in place of Hoeffding's inequality in the proof of Theorem 3.3 to prove the bound

$$\widehat{L}_n \le \frac{\eta L_n^* + \ln N}{1 - e^{-\eta}}\,,$$

where $L_n^* = \min_{1\le i\le N} L_{i,n}$.

(b) Let $\eta = \ln(1 + \sqrt{(2\ln N)/L_n^*})$, where L_n^* is assumed to be positive. Show that in this case

$$\widehat{L}_n \le L_n^* + \sqrt{2L_n^* \ln N} + \ln N\,.$$

Hint for Part (b)*:* Use $\eta \le \sinh(\eta) = (e^\eta - e^{-\eta})/2$, which is known to hold for all $\eta > 0$, together with the bound of Part (a).

(c) Compare the bound of Part (b) to that of Theorem 3.3. When would you use this bound and when the original one?

Exercise 3.3. (Hierarchical EWA) Consider a continuous prediction problem with decision space D, outcome space Y, loss function ℓ and N experts. As usual, let ℓ be convex in its first argument. Fix some $0 < \eta$. Let EWA(η) denote the EWA algorithm when it is used with the fixed N experts and the learning rate η. Now, consider a finite set of possible values of η, say, $E = \{\eta_1, \ldots, \eta_K\}$. Imagine using these K instances of EWA in parallel. Each of them will predict in its own way. This new algorithms themselves can be considered as K new, *compound* experts, giving rise to a "hierarchical EWA algorithm" with two layers. Which of these compound experts is the best? This might be difficult to decide ahead of time (and in fact, will depend on the outcome sequence), but we can just use EWA to combine the predictions of the K compound experts to arrive at an interesting algorithm, with a hyperparameter $\eta^* > 0$.

(a) Invent a specific prediction problem (D, Y, ℓ) with the required properties and a fixed set of experts such that for some outcome sequences the smallest regret is obtained when η is very close to zero, whereas for some other outcome sequences the largest regret is obtained when η takes on a large value.

(b) Implement the EWA algorithm and test it in the environment that you have described above, both for large and small learning rates. Do your experimental results support your answer to the first part? Hopefully they do, in which case you can consider the next part.

(c) Implement the hierarchical EWA algorithm described above and test it in the environment you have used above. Select $\eta_1, \ldots, \eta_K$ in such a way that you can get interesting results (include the values of the learning rate used in the previous part). Describe your findings. As to the value of η^*, use the value specified in Theorem 3.3.

(d) Is it worth to define yet another layer of the hierarchy, to "learn" the best value of η^*? How about yet another layer on the top of this? Justify your answer!

Exercise 3.4. Let $L_{it} > 0$, $f_{it} \in D$, $\ell : D \times Y \to [0,1]$ convex in its first argument. For $y \in Y$, $\eta > 0$, define $f_t(\eta, y) = \ell(\sum_i \frac{\exp(-\eta L_{it})}{\sum_j \exp(-\eta L_{jt})} f_{it}, y)$. If this function was convex as a function of η, could you use it for tuning η? How? Let ℓ be linear in its first argument. Is this function convex as a function of η? Justify your answer.

Chapter 4

Exponentially Weighted Average Forecaster: Discrete Predictions

A *discrete prediction problem* is one when the outcome space Y has at least two elements, $D = Y$ and the loss is the zero one loss: $\ell(p, y) = \mathbb{I}\{p \neq y\}$.

In Chapter 2 we have shown that the WEIGHTED MAJORITY ALGORITHM (WMA) for binary prediction problems makes at most

$$\widehat{L}_n \leq \frac{\log_2\left(\frac{1}{\beta}\right) L^*_{i,n} + \log_2 N}{\log_2\left(\frac{2}{1+\beta}\right)}$$

mistakes. If we subtract $L^*_{i,n}$ from both sides, we get a bound on the regret

$$R_n = \widehat{L}_n - L^*_{i,n} \leq \frac{\left(\log_2\left(\frac{1}{\beta}\right) - \log_2\left(\frac{2}{1+\beta}\right)\right) L^*_{i,n} + \log_2 N}{\log_2\left(\frac{2}{1+\beta}\right)}$$

Unfortunately, this bound is of the form $R_n \leq aL^*_{i,n} + b = O(n)$. That's much worse than the bound $R_n \leq \sqrt{\frac{n}{2}\ln N} = O(\sqrt{n})$, which we proved for EWA for continuous predictions. This suggests that discrete problems are harder than continuous problems.

That this is true, at least if we stick to deterministic algorithms, is shown as follows: Take $D = Y = \{0, 1\}$, let the loss $\ell(p, y) = \mathbb{I}\{p \neq y\}$ be the zero-one loss. Assume that we have two experts such that the first expert predicts $f_{1,t} = f_{1,2} = \cdots = f_{1,n} = 0$ and the second one predicts $f_{2,t} = f_{2,2} = \cdots = f_{2,n} = 1$. Then, the following hold:

(a) For any sequence $y_1, y_2, \ldots, y_n \in \{0, 1\}$, there exists an expert $i \in \{1, 2\}$ such that $L_{i,n} = \sum_{i=1}^n \ell(f_{i,t}, y_t) \geq n/2$.

(b) For any deterministic algorithm A there exists an outcome sequence $y_1^A, y_2^A, \ldots, y_n^A \in \{0, 1\}$ such that $\widehat{L}_n = \sum_{t=1}^n \ell(\widehat{p}_t^A, y_t^A) = n$, where p_t^A denotes the prediction of A at time t on the outcome sequence $y_1^A, \ldots, y_n^A$.

Part a follows from that $L_{1,n} + L_{2,n} = n$ and therefore at least one of the two terms is at least $n/2$. Part b follows by an adversarial argument: Let $y_1^A = 1 - \widehat{p}_1^A$. This is well defined, since the first prediction of A is just some constant, which can thus be used to construct y_1^A. Now, for $t \geq 2$, assuming that $y_1^A, y_2^A, \ldots, y_{t-1}^A$ have already been constructed. Let $\widehat{p}_t^A$ be the prediction of algorithm A for round t, given the constructed sequence. This is again well-defined, since the prediction of A for round t depends in a deterministic fashion on the previous outcomes. Then, set $y_t^A = 1 - \widehat{p}_t^A$. With this construction, $\ell(\widehat{p}_t^A, y_t^A) = 1$ holds for all t.

From a and b we get the following result:

Theorem 4.1. *Let Y be a set with at least two elements. Consider the discrete prediction problem where the outcome space is Y, the decision space is $D = Y$, and the loss is the zero-one loss $\ell(p, y) = \mathbb{I}\{p \neq y\}$. Then, for any deterministic algorithm A, in the worst case the regret R_n^A of A can be as large as $n/2$.*

4.1 Randomized EWA

In this section D and Y are arbitrary sets.

The pessimistic result of Theorem 4.1 can be circumvented with *randomization*. In particular, we will show that a randomized variant of EWA can achieve non-trivial regret even for discrete prediction problems. (The algorithm is also know under the name Randomized Weighted Majority Algorithm (RWMA).) For each expert i, the algorithm maintains a weight $w_{i,t} = e^{-\eta L_{i,t}}$, where $L_{i,t} = \sum_{s=1}^{t} \ell(f_{i,t}, y_t)$ is the cumulated loss of expert i up to time t and $\eta > 0$ is a positive parameter. The algorithm also maintains the sum of the weights $W_t = \sum_{i=1}^{N} w_{i,t}$.

Formally, it works as follows: Initially, $w_{i,0} = 1$ for each expert i and $W_0 = N$. Then, in each round $t = 1, 2, \ldots, n$, the algorithm does the following:

- It receives experts' predictions $f_{1,t}, f_{2,t}, \ldots, f_{N,t} \in D$.
- It calculates $\widehat{p}_{i,t} = w_{i,t-1}/W_{t-1}$, $i = 1, \ldots, N$.
- It draws $I_t \in \{1, 2, \ldots, N\}$ *randomly* so that $\Pr[I_t = i] = \widehat{p}_{i,t}$ holds for $i = 1, \ldots, N$.
- It predicts $f_{I_t,t}$.
- The environment reveals the outcome $y_t \in Y$.
- The algorithm suffers the loss $\ell(f_{I_t,t}, y_t)$ and each expert $i = 1, 2, \ldots, N$ suffers a loss $\ell(f_{i,t}, y_t)$.
- The algorithm updates the weights: $w_{i,t} = w_{i,t-1} e^{-\eta \ell(f_{i,t}, y_t)}$.
- The algorithm updates the sum of the weights: $W_t = \sum_{i=1}^{N} w_{i,t}$.

We do not assume anything about D, Y and the loss function ℓ doesn't need to convex anymore. The only assumption that we make is that $\ell(p, y) \in [0, 1]$. Also note that the numbers $\widehat{p}_{1,t}, \widehat{p}_{2,t}, \ldots, \widehat{p}_{N,t}$ are non-negative and sum to 1 and therefore the distribution of I_t is valid probability distribution.

Since the algorithm randomizes, its regret becomes a random variable. Hence, our statements will be of probabilistic nature: First we show a bound on the expected regret and then we will argue that with high probability, the actual (random) regret is also bounded by some "small" quantity.

4.2 A Bound on the Expected Regret

Let us thus first analyze the algorithm's expected regret. Our plan is to use Theorem 3.3 from Chapter 3 for this purpose. For this, we will show that for appropriate D', Y', ℓ', where D' is the convex subset of a vector space, $\ell' : D' \times Y' \to [0, 1]$ is convex in its first argument, any sequence of outcomes and any sequence of expert predictions can be mapped into appropriate sequences taking values in the respective spaces Y' and D' such that the *expected* regret of the randomized EWA algorithm is the same as that of an EWA algorithm which works with the mapped sequences. From this, the bound on the expected regret of the randomized EWA algorithm will follow.

The construction is as follows: We define $Y', D', \ell' : D' \times Y' \to [0, 1]$ as follows:

- $D' = \{p \in [0, 1]^N \ : \sum_{i=1}^N p_i = 1\}$ is the *N-dimensional probability simplex*;
- $Y' = Y \times D^N$;
- $\ell'(p, (y, f_1, f_2, \ldots, f_N)) = \sum_{i=1}^N p_i \cdot \ell(f_i, y)$.

Note that, as promised, D' is convex and ℓ' is convex (in fact linear!) in its first argument. Now, given the expert predictions $f_{i,t}$ and outcomes y_t we define the sequences $(f'_{i,t})$, (y'_t), $f'_{i,t} \in D'$, $y'_t \in Y'$, as follows:

- $f'_{i,t} = e_i$ where $e_i = (0, \ldots, 0, 1, 0, \ldots, 0)^\top$ is a vector of length N with i-th coordinate equal to 1 and all other equal to 0;
- $y'_t = (y_t, f_{1,t}, \ldots, f_{N,t})$.

Suppose we apply EWA from Chapter 3 on $Y', D', \ell' : D' \times Y' \to [0, 1]$ with experts' predictions $f'_{i,t}$ and outcomes y'_t. It is not hard to verify that the experts' losses are exactly the same as in the randomized algorithm on the original experts' predictions $f_{i,t}$ and outcomes y_t:

$$\begin{aligned}
\ell'(f'_{i,t}, y'_t) &= \ell'(e_i, (y_t, f_{1,t}, \ldots, f_{N,t})) \\
&= 0 \cdot \ell(f_{1,t}, y_t) + \cdots + 0 \cdot \ell(f_{i-1,t}, y_t) + 1 \cdot \ell(f_{i,t}, y_t) + 0 \cdot \ell(f_{i+1,t}, y_t) + \cdots + 0 \cdot \ell(f_{N,t}, y_t) \\
&= \ell(f_{i,t}, y_t)\,.
\end{aligned}$$

Therefore, the cumulated losses $L_{i,t} = \sum_{s=1}^{t} \ell(f_{i,s}, y_s) = \sum_{s=1}^{t} \ell'(f'_{i,s}, y'_s)$ of experts are the same in both algorithms. It follows that the weights $w_{i,t}$, the sum of weights W_t and the vectors $\widehat{p}_t = (\widehat{p}_{1,t}, \widehat{p}_{2,t}, \ldots, \widehat{p}_{N,t})$ are also identical between the two algorithms.

Furthermore, the expected loss of the randomized algorithm is the same as the loss of the continuous EWA running on $f'_{i,t}$'s and y'_t's:

$$\mathbf{E}[\ell(f_{I_t,t}, y_t)] = \sum_{i=1}^{n} \widehat{p}_{i,t} \cdot \ell(f_{i,t}, y_t) = \ell'(\widehat{p}_t, (y_t, f_{1,t}, f_{2,t}, \ldots, f_{N,t})) = \ell'(\widehat{p}_t, y'_t) \tag{4.1}$$

If $L'_n = \sum_{t=1}^{n} \ell'(\widehat{p}_t, y'_t)$ denotes the loss of the continuous EWA running on $f'_{i,t}$'s and y'_t's, by Theorem 3.3

$$\widehat{L}'_n \le \min_{1 \le i \le N} L_{i,n} + \frac{\ln N}{\eta} + \frac{\eta n}{8}$$

If $L_n = \sum_{t=1}^{n} \ell(f_{I_t,t}, y_t)$ denotes the loss of the randomized EWA, thanks to (4.1), we see that $\mathbf{E}[\widehat{L}_n] = \widehat{L}'_n$ and therefore

$$\mathbf{E}[\widehat{L}_n] \le \min_{1 \le i \le N} L_{i,n} + \frac{\ln N}{\eta} + \frac{\eta n}{8}$$

We summarize this result in the following theorem:

Theorem 4.2. *Let D, Y, $\ell : D \times Y \to [0,1]$ be arbitrary. Then, the expected loss of the randomized EWA forecaster is upper bounded as follows*

$$\mathbf{E}[\widehat{L}_n] \le \min_{1 \le i \le N} L_{i,n} + \frac{\ln N}{\eta} + \frac{\eta n}{8} \, .$$

In particular, with $\eta = \sqrt{\frac{8 \ln N}{n}}$, $\mathbf{E}[\widehat{L}_n] \le \min_{1 \le i \le N} L_{i,n} + \sqrt{\frac{n}{2} \ln N}$.

4.3 A High Probability Bound

Theorem 4.2 gives an upper bound on the *expected* regret of the algorithm. An expected regret bound is a good start but the bound would not be very useful if the variance of the regret was too large (e.g., linear in n). More generally, we are interested in proving so-called *exponential tail bounds* for the regret, which, as we will see, show that the regret has "sub-Gaussian tails". In particular, from these bounds it follows that the variance of regret (and also its higher moments) are tightly controlled.

The proof will rely on a basic inequality of probability theory. (We omit the proof and we only mention that the theorem can be proved using Hoeffding's lemma.)

Theorem 4.3 (Hoeffding's inequality)**.** *If $X_1, X_2, \ldots, X_n$ are independent random variables lying in $[0,1]$ then for any $\varepsilon \ge 0$*

$$\Pr\left[\sum_{t=1}^{n} X_t - \sum_{t=1}^{n} \mathbf{E}[X_t] \ge \varepsilon\right] \le e^{-2\varepsilon^2/n} \, .$$

Equivalently, for any $0 < \delta < 1$ *with probability at least* $1 - \delta$,

$$\sum_{t=1}^{n} X_t < \sum_{t=1}^{n} \mathbf{E}[X_t] + \sqrt{\frac{n}{2} \ln(1/\delta)}\,.$$

Remark 4.4. Note that the (upper) tail probability of a zero-mean random variable Z is defined as $\Pr(Z \geq \varepsilon)$ for $\varepsilon > 0$. The first form of the inequality explains why we say that the "tail" of $Z = \sum_{t=1}^{n} X_t - \sum_{t=1}^{n} \mathbf{E}[X_t]$ shows a sub-Gaussian behavior: The tail of a Gaussian with variance $\sigma^2 = n/4$ would also decay as $\exp(-2\varepsilon^2/n)$ as $\varepsilon \to \infty$. Therefore, the tail of Z is not "fatter" than that of a Gaussian, which we just summarize as the tail of Z is sub-Gaussian.

We apply Hoeffding's inequality to the losses $X_t = \ell(f_{I_t,t}, y_t)$ of randomized EWA. We get that with probability at least $1 - \delta$,

$$\widehat{L}_n < \mathbf{E}[\widehat{L}_n] + \sqrt{\frac{n}{2} \ln(1/\delta)}\,.$$

This, together with Theorem 4.2 for $\eta = \sqrt{\frac{8 \ln N}{n}}$ gives that with probability a least $1 - \delta$,

$$\widehat{L}_n < \min_{1 \leq i \leq N} L_{i,n} + \sqrt{\frac{n}{2} \ln N} + \sqrt{\frac{n}{2} \ln(1/\delta)}\,.$$

We summarize the result in a theorem:

Theorem 4.5. *Let* D, Y, $\ell : D \times Y \to [0,1]$ *be arbitrary. Then, for any* $0 < \delta < 1$, *the loss of randomized EWA forecaster with* $\eta = \sqrt{\frac{8 \ln N}{n}}$ *is with probability at least* $1 - \delta$ *upper bounded as follows*

$$\widehat{L}_n < \min_{1 \leq i \leq N} L_{i,n} + \sqrt{\frac{n}{2}} \left(\sqrt{\ln N} + \sqrt{\ln(1/\delta)} \right)\,.$$

Using $R_n = \widehat{L}_n - \min_{1 \leq i \leq N} L_{i,n}$, this gives a high-probability regret bound, and in fact shows that the tail of the regret is sub-Gaussian.

4.4 Exercises

Exercise 4.1. Consider the setting of Exercise 3.3, just now for a discrete prediction problem. That is, the goal is to find the best learning rate for randomized EWA from a finite pool $\{\eta_1, \ldots, \eta_K\}$. One possibility is to run a randomized EWA on the top of K randomized EWA forecasters, each using some learning rate η_k, $k = 1, \ldots, K$. The randomized EWA "on the top" would thus randomly select one of the randomized EWA forecasters in the "base", which would in turn select an expert at random. Another possibility is that if

$p_t^0 = (p_{1,t}^{(0)}, \ldots, p_{K,t}^{(0)})^\top \in [0,1]^K$ is the probability vector of the randomized EWA on the top for round t, and $p_t^{(k)} = (p_{1,t}^{(k)}, \ldots, p_{N,t}^{(k)})^\top \in [0,1]^N$ is the likewise probability vector of the k^{th} randomized EWA in the base, then just select expert i with probability $\sum_{k=1}^K p_{k,t}^{(0)} p_{i,t}^{(k)}$, i.e., combine the "votes" across the two layers before selecting the expert. Which method would you prefer? Why? Design an experiment to validate your claim.

Chapter 5

A Lower Bound for Discrete Prediction Problems

Is EWA of the last two chapters is a good algorithm? Can there exist a better algorithm? Or maybe our bound for EWA is loose and in fact EWA is a better algorithm than we think it is? In an exercise you proved that for the shooting game FTL achieves $\Theta(\ln n)$ regret. But can there be a better algorithm than FTL for this problem?

Minimax lower bounds provide answers to questions like these. We shall investigate continuous prediction problems with convex losses (in short, continuous convex problems) since for discrete problems if we use a randomized algorithm, the problem is effectively transformed into a continuous convex problem, while if no randomization is allowed, we have already seen a lower-bound in Theorem 4.1.

5.1 Some Preliminaries

We want to show that the bound in Theorem 3.3 is "tight". In order to explain what we mean by tightness, let us state an easy corollary of this theorem.

We need some notation. Fix some sets D, Y and a loss function $\ell : D \times Y \to [0, 1]$. An "algorithm" simply maps past observations to decisions: At time t an algorithm can use $(y_1, \dots, y_{t-1}, f_{1,1}, \dots, f_{1,t}, \dots, f_{N,1}, \dots, f_{N,t})$ to come up with its decision (and it can also randomize). We will disregard computability issues and in fact consider all possible (randomized) mappings from such histories to D. The set of these will be denoted by $\mathcal{A}$. Similarly, an expert bases its decision on past outcomes: At time t an expert can use $(y_1, \dots, y_{t-1})$. Again, experts can randomize and we will consider all randomized mappings which map such histories to decisions. The set of these will be denoted by $\mathcal{F}$.

Given an algorithm $A \in \mathcal{A}$ and a non-empty set of experts $\mathcal{F}_0 \subset \mathcal{F}$, let $R_n^A(\mathcal{F}_0)$ be the (worst-case) regret of A for horizon n:

$$R_n^A(\mathcal{F}_0) = \sup_{(y_1,\dots,y_n)\in Y^n} \left\{ \sum_{t=1}^{n} \ell(p_t^{(A)}, y_t) - \inf_{F\in\mathcal{F}_0} \sum_{t=1}^{n} \ell(f_t^{(F)}, y_t) \right\},$$

where $p_t^{(A)}$ is algorithm A's decision at time t, $f_t^{(F)}$ is the decision of expert F at time t. (Of course, the regret depends on D, Y and ℓ, too, so for full precision we should denote this dependence on the right-hand side, too, i.e., we should use, say, $R_n^A(\mathcal{F}, \ell)$. However, since we will mostly treat D, Y, ℓ as given (fixed), we suppress this dependence.)

The corollary of Theorem 3.3 is this: Fix some sets D, Y. Let $\mathcal{F}$ be the set of experts over D, Y and let $\mathcal{A}$ be the set of algorithms. Take $c = 1/\sqrt{2}$. Then the following holds true:

> (UB) For any loss function $\ell : D \times Y \to [0,1]$, for any horizon n, for any positive integer N, there exists an algorithm $A \in \mathcal{A}$, such that for any multiset[1] of experts $\mathcal{F}_N \subset \mathcal{F}$ of cardinality N, the regret $R_n^A(\mathcal{F}_N)$ of algorithm A satisfies $R_n^A(\mathcal{F}_N) \le c\sqrt{n \ln N}$.

For a fixed value of c, the negation of the above statement is the following:

> (NUB) There exists a loss function $\ell : D \times Y \to [0,1]$, a horizon n and a positive integer N, such that for all algorithms $A \in \mathcal{A}$, there exists a multiset of experts $\mathcal{F}_N \in \mathcal{F}$ with cardinality N such that the regret $R_n^A(\mathcal{F}_N)$ of algorithm A satisfies $R_n^A(\mathcal{F}_N) > c\sqrt{n \ln N}$.

Clearly, for any value of c, only one of the statements (UB), (NUB) can be true. We want to show that for any $c < 1/\sqrt{2}$, (NUB) holds true since then it follows that the bound for EWA is tight in the sense that there is no better algorithm than EWA *in the worst-case* and the bound for EWA cannot be improved with respect to its constant, or how it depends on the number of experts N or the length of the horizon n.

We will show the result for $D = [0,1]$, $Y = \{0,1\}$ and $\ell(p,y) = |p - y|$.

To show that for any $c < 1/\sqrt{2}$, (NUB) holds true, it suffices to prove that there exist n, N such that for any algorithm $A \in \mathcal{A}$,

$$\sup_{\mathcal{F}_N \in \mathcal{F}, |\mathcal{F}_N| \le N} R_n^A(\mathcal{F}_N) \ge 1/\sqrt{2}\,\sqrt{n \ln N}\,.$$

In fact, it suffices to prove that there exist n, N such that for any algorithm $A \in \mathcal{A}$,

$$\sup_{\mathcal{F}_N \in S_n} R_n^A(\mathcal{F}_N) \ge 1/\sqrt{2}\,\sqrt{n \ln N}\,, \tag{5.1}$$

where S_n is the set of *static experts*, i.e., the set of experts which decide about their choices ahead of time, independently of what the outcome sequence will be. Thus, for $F \in S_n$, for any $1 \le t \le n$, $y_1, \dots, y_{t-1} \in Y$, $F(y_1, \dots, y_{t-1}) = f_t$ for some fixed sequence $(f_t) \in D^n$ and vice versa: for any fixed sequence $(f_t) \in D$, there exists an expert F in S_n such that for any $1 \le t \le n$, $y_1, \dots, y_{t-1} \in Y$, $F(y_1, \dots, y_{t-1}) = f_t$. Hence, in what follows we shall identify the set of experts S_n with the set of sequences of length D^n. That inequality (5.1) holds for any algorithm $A \in \mathcal{A}$ is equivalent to requiring that

$$\inf_{A \in \mathcal{A}} \sup_{\mathcal{F}_N \in S_n} R_n^A(\mathcal{F}_N) \ge 1/\sqrt{2}\,\sqrt{n \ln N}$$

[1] A multiset is similar to a set. The only difference is that multisets can contain elements multiple times. We could use a list (or vector), but we would like to use $\subset$ and $\in$..

holds. Let

$$V_n^{(N)} = \inf_{A \in \mathcal{A}} \sup_{\mathcal{F}_N \in S_n} R_n^A(\mathcal{F}_N)\,.$$

We call this quantity the *minimax regret* associated to problem (D, Y, ℓ), horizon n and expert number N. Thus, we will want to prove that $V_n^{(N)}/\sqrt{n \ln N} \geq 1/\sqrt{2}$ holds for some n, N.

5.2 Results

A sequence $(Z_t)_{t=1}^n$ of independent, identically distributed $\{-1, +1\}$-valued random variables with $\Pr(Z_t = 1) = \Pr(Z_t = -1)$ is called a *Rademacher sequence*. When you sum up a Rademacher sequence, you get a random walk on the integers. A matrix with random elements will be called a *Rademacher random matrix* when its elements form a Rademacher sequence. We will need the following lemma, which shows how the expected value of the maximum of N independent random walks behave. The results stated in the lemma are purely probabilistic and we do not prove them. The essence of our proof will be a reduction of our problem to this lemma.

Lemma 5.1. *Take n independent Rademacher random variables. Then*

$$\lim_{n\to\infty} \frac{\mathbf{E}\left[\max_{1\leq i\leq N} \sum_{t=1}^n Z_{i,t}\right]}{\sqrt{n}} = \mathbf{E}\left[\max_{1\leq i\leq N} G_i\right]\,,$$

where $G_1, G_2, \ldots, G_N$ are N independent standard normal r.v.s. Further,

$$\lim_{N\to\infty} \frac{\mathbf{E}\left[\max_{1\leq i\leq N} G_i\right]}{\sqrt{2\ln N}} = 1.$$

The first part of the lemma states that, asymptotically speaking, the expected maximal excursions of N independent random walks is the same as the expectation of the maximum of N independent Gaussian random variables. Note that for $N = 1$, this result would follow from the central limit theorem, so you can consider this as a generalization of the central limit theorem. The second part of the lemma states that asymptotically, as N gets big, the expectation of the maximum of N independent Gaussian random variables is $\sqrt{2\ln N}$. Together the two statements say that, asymptotically, as both n, N get large, the expected size of maximal excursions of N independent random walks is $\sqrt{2n\ln N}$.

As many lower bound proofs, our proof will also use what we call the *randomization hammer*, according to which, for any random variable X with domain $\mathrm{dom}(X)$, and any function f,

$$\sup_{x\in\mathrm{dom}(X)} f(x) \geq \mathbf{E}\left[f(X)\right]$$

holds whenever $\mathbf{E}\left[f(X)\right]$ exists. When using this "trick" to lower bound $\sup_x f(x)$, we will choose a distribution over the range of values of x (equivalently, the random variable X)

and then we will further calculate with the expected value $\mathbf{E}[f(X)]$. The distribution will be chosen such that the expectation is easy to deal with.

Our main result is the following theorem:

Theorem 5.2. *Take $Y = \{0,1\}$, $D = [0,1]$, $\ell(\widehat{p}, y) = |\widehat{p} - y|$. Then,*

$$\sup_{n,N} \frac{V_n^{(N)}}{\sqrt{(n/2)\ln N}} \geq 1\,.$$

That (NUB) holds for any $c < 1/\sqrt{2}$ is a corollary of this theorem should be clear given the definitions. The strength of this result is that it is an algorithm independent lower bound. How do we establish the existence of an appropriate sequence of outcomes and an appropriate sequence of experts? We will use the randomization hammer.

Proof. Fix $n, N > 0$. Let $\mathcal{F}_N \in S_n$ be some multiset of N static experts. By definition,

$$R_n^A(\mathcal{F}_N) \;=\; \sup_{y_1,\ldots,y_n \in Y} \left(\sum_{t=1}^n |\widehat{p}_t^{(A)} - y_t| - \min_{F \in \mathcal{F}_N} \sum_{t=1}^n |f_t^{(F)} - y_t| \right).$$

We use the randomization hammer to lower bound the quantity on the right-hand side. Let $Y_1, Y_2, \ldots, Y_n$ be an i.i.d. sequence of Bernoulli(1/2) random variables such that Y_t is also independent of the random numbers which are used by algorithm A. Then,

$$\begin{aligned} R_n^A(\mathcal{F}_N) &\geq \mathbf{E}\left[\sum_{t=1}^n |\widehat{p}_t^{(A)} - Y_t| - \min_{F \in \mathcal{F}_N} \sum_{t=1}^n |f_t^{(F)} - Y_t|) \right] \\ &= \mathbf{E}\left[\sum_{t=1}^n |\widehat{p}_t^{(A)} - Y_t| \right] - \mathbf{E}\left[\min_{F \in \mathcal{F}_N} \sum_{t=1}^n |f_t^{(F)} - Y_t| \right]. \end{aligned}$$

It is not hard to see that $\mathbf{E}\left[|\widehat{p}_t^{(A)} - Y_t|\right] = 1/2$ holds for any $1 \leq t \leq n$ thanks to our choice of (Y_t) (see Exercise 5.1).

Therefore,

$$R_n^A(\mathcal{F}_N) \;\geq\; \frac{n}{2} - \mathbf{E}\left[\min_{F \in \mathcal{F}_N} \sum_{t=1}^n |f_t^{(F)} - Y_t| \right] = \mathbf{E}\left[\max_{F \in \mathcal{F}_N} \sum_{t=1}^n \left(\frac{1}{2} - |f_t^{(F)} - Y_t| \right) \right].$$

Note that the right-hand side is a quantity, which does not depend on algorithm A. Thus, we made a major step forward.

Now, let us find a shorter expression for the right-hand side. If $Y_t = 0$, the expression $\left(\frac{1}{2} - |f_t^{(F)} - Y_t|\right)$ equals to $1/2 - f_t^{(F)}$. If $Y_t = 1$, the expression equals to $f_t^{(F)} - 1/2$. Therefore, the expression can be written as $(f_t^{(F)} - 1/2)(2Y_t - 1)$. Let us introduce $\sigma_t = 2Y_t - 1$.

Notice that (σ_t) is a Rademacher sequence variables. With the help of this sequence, we can write

$$R_n^A(\mathcal{F}_N) \geq \mathbf{E}\left[\max_{F\in\mathcal{F}_N}\sum_{t=1}^{n}(f_t^{(F)}-\tfrac{1}{2})\sigma_t\right].$$

Thus, it also holds that

$$\begin{aligned}\sup_{\mathcal{F}_N\in S_n} R_n^A(\mathcal{F}_N) &\geq \sup_{\mathcal{F}_N\in S_n}\mathbf{E}\left[\max_{f\in\mathcal{F}_N}\sum_{t=1}^{n}(f_t^{(F)}-\tfrac{1}{2})\sigma_t\right]\\ &= \sup_{M\in D^{N\times n}}\mathbf{E}\left[\max_{1\leq i\leq N}\sum_{t=1}^{n}(M_{i,t}-\tfrac{1}{2})\sigma_t\right],\end{aligned} \tag{5.2}$$

where the last equality follows from the definition of S_n.

We lower bound the quantity on the right-hand side by resorting to the randomization hammer again. For this, we will view the right-hand side as $\sup_M g(M)$ with an appropriate function g. The random variables used in the hammer will be again Bernoullis. In particular, we introduce the $N\times n$ random matrix $B=(B_{i,t})_{i=1,\dots,N,t=1,\dots,n}$, whose elements are independent, Bernoulli(1/2) random variables, which are also independent of the previously introduced random variables. For convenience, let us also introduce $Z_{i,t}=2B_{i,t}-1$ so that $B_{i,t}-\frac{1}{2}=\frac{1}{2}Z_{i,t}$ (note that $Z_{i,t}$ are Rademacher random variables). Then,

$$\sup_{M\in D^{N\times n}} g(M) \geq \mathbf{E}\left[g(B)\right] = \frac{1}{2}\,\mathbf{E}\left[\max_{1\leq i\leq N}\sum_{t=1}^{n} Z_{i,t}\sigma_t\right]. \tag{5.3}$$

It is not hard to see then that $(Z_{i,t}\sigma_t)$ is an $N\times n$ Rademacher random matrix (Exercise 5.2).

Now, since $(Z_{i,t}\sigma_t)$ is an $N\times n$ Rademacher random matrix, the distribution of

$$\max_{1\leq i\leq N}\sum_{t=1}^{n} Z_{i,t}\sigma_t$$

is the same as that of $\max_{1\leq i\leq N}\sum_{t=1}^{n} Z_{i,t}$. Therefore,[2]

$$\mathbf{E}\left[\max_{1\leq i\leq N}\sum_{t=1}^{n} Z_{i,t}\sigma_t\right] = \mathbf{E}\left[\max_{1\leq i\leq N}\sum_{t=1}^{n} Z_{i,t}\right].$$

Chaining (5.2), (5.3) with this equality, we get

$$\sup_{\mathcal{F}_N\in S_n} R_n^A(\mathcal{F}_N) \geq \frac{1}{2}\,\mathbf{E}\left[\max_{1\leq i\leq N}\sum_{t=1}^{n} Z_{i,t}\right].$$

[2]Here, the Rademacher matrix $(Z_{i,t})$ is recycled just to save the introduction of some new letters. Of course, we could have also carried $Z_{i,t}\sigma_t$ further, but that would again be just too much writing.

Since this holds for any algorithm $A \in \mathcal{A}$, we must also have

$$V_n^{(N)} = \inf_{A \in \mathcal{A}} \sup_{\mathcal{F}_N \in S_n} R_n^A(\mathcal{F}_N) \geq \frac{1}{2} \mathbf{E}\left[\max_{1 \leq i \leq N} \sum_{t=1}^{n} Z_{i,t} \right] .$$

Dividing both sides by $\sqrt{n}$ and letting $n \to \infty$, the first part of Lemma 5.1 gives

$$\lim_{n\to\infty} \frac{V_n^{(N)}}{\sqrt{n}} \geq \frac{1}{2} \lim_{n\to\infty} \frac{\mathbf{E}\left[\max_{1 \leq i \leq N} \sum_{t=1}^{n} Z_{i,t}\right]}{\sqrt{n}} = \frac{1}{2} \mathbf{E}\left[\max_{1 \leq i \leq N} G_i \right],$$

where $G_1, G_2, \ldots, G_N$ are independent, Gaussian random variables. Now, divide both sides by $\sqrt{2 \ln N}$ and let $N \to \infty$. The second part of Lemma 5.1 gives

$$\lim_{N\to\infty} \lim_{n\to\infty} \frac{V_n^{(N)}}{\sqrt{2n \ln N}} \geq \frac{1}{2} \lim_{N\to\infty} \frac{\mathbf{E}\left[\max_{1 \leq i \leq N} G_i\right]}{\sqrt{2 \ln N}} = \frac{1}{2},$$

therefore, we also have

$$\sup_{n,N} \frac{V_n^{(N)}}{\sqrt{2n \ln N}} \geq \frac{1}{2}.$$

Multiplying both sides by 2 gives the desired result. □

5.3 Exercises

Exercise 5.1. Show that in the proof of Theorem 5.2, $\mathbf{E}\left[|\widehat{p}_t^{(A)} - Y_t|\right] = 1/2$ holds for any $1 \leq t \leq n$, where the algorithm A is allowed to randomize (and of course may use past information to come up with its prediction).

Exercise 5.2. Show that if $(Z_{i,t})$ is an $N \times n$ Rademacher matrix and (σ_t) is a Rademacher sequence with n elements and if for any t, $V_t \stackrel{\text{def}}{=} (Z_{i,t})_i$ and σ_t are independent of each other then $(Z_{i,t}\sigma_t)$ is an $N \times n$ Rademacher matrix.

Exercise 5.3. Show that for any sets X, Y, and any function $A : X \times Y \to \mathbb{R}$,

$$\inf_{y \in Y} \sup_{x \in X} A(x,y) \geq \sup_{x \in X} \inf_{y \in Y} A(x,y).$$

Hint: Obviously, $\sup_{x \in X} A(x,y) \geq A(x_0,y)$ holds for any x_0, y.

Exercise 5.4. Show that the following strengthening of the result in this chapter holds, too: Take any $c < 1/\sqrt{2}$. Fix $D = [0,1]$, $Y = \{0,1\}$, $\ell(p,y) = |p-y|$. Then, there exists a horizon n, a positive integer N, and a non-empty set of experts $\mathcal{F}_N \in \mathcal{F}$ with cardinality N such that for all algorithms $A \in \mathcal{A}$ the regret $R_n^A(\mathcal{F}_N)$ of algorithm A satisfies $R_n^A(\mathcal{F}_N) > c\sqrt{n \ln N}$.
Hint: Modify the proof of the above theorem.

Chapter 6

Tracking

In this chapter we still consider discrete prediction problems with expert advice with N experts, where the losses take values in $[0, 1]$. The horizon n will be fixed.

So far we have considered a framework when the learning algorithm competed with the single best expert out of a set of N experts. At times, this base set of experts might not perform very well on their own. In such cases one might try a larger set of *compound experts*, potentially created from the *base experts*. For example, we may want to consider decision trees of experts when the conditions in the decision tree nodes refer to past outcomes, the time elapsed, past predictions, etc., while the leafs could be associated with indices of base experts. A decision tree expert can itself be interpreted as an expert: In a given round, the past information would determine a leaf and thus the base expert whose advice the tree expert would take. Then one can just use randomized EWA to compete with the best compound expert. The benefit of doing so is that the best compound expert might have a much smaller cumulated loss than the cumulated loss of the best base expert. The drawback is that it might be hard to find this expert with a small loss, i.e., the regret bound, becomes larger. So in general, this idea must be applied with care.

6.1 The problem of tracking

We will only deal with the special case of the so-called *switching compound experts*. Such a compound expert is given by a sequence $\sigma \in \{1, \dots, N\}^n$, where σ_t (the t^{th} element of σ) is the index of expert whose decision σ suggests to follow will at time t. (For brevity, we will refer to σ itself as a switching expert.) Our goal will be to develop efficient online algorithms which can compete with the best of those switching experts which do not switch more than m times. The loss of σ at time t is $\ell(f_{\sigma_t,t}, y_t)$,[1] so its the cumulated loss is $L_{\sigma,n} \stackrel{\text{def}}{=} \sum_{t=1}^{n} \ell(f_{\sigma_t,t}, y_t)$. The loss of an algorithm, which chooses the switching expert $\widehat{\sigma}_t$ at time is $\widehat{L}_n = \sum_{t=1}^{n} \ell(f_{\widehat{\sigma}_t,t}, y_t)$ and thus its regret when competing with the best switching

[1]Here, y_t is the outcome at time t and $f_{i,t} \in D$ is the advice of expert i at time t. As before, D is the set of decisions, Y is the set of outcomes.

expert within the class $B \subseteq \{1, \ldots, N\}^n$ is $R_n = \widehat{L}_n - \min_{\sigma \in B} L_{\sigma,n}$.

Clearly, randomized EWA can be applied to this problem. By Theorem 4.2, when η is appropriately selected,

$$\mathbf{E}\left[R_n\right] \leq \sqrt{\frac{n}{2} \ln |B|}\,. \tag{6.1}$$

We immediately see that if $B = \{1, \ldots, N\}^n$ (all switching experts are considered), the bound becomes vacuous since $|B| = N^n$, and hence $\ln |B| = n \ln N$ and the bound is linear as a function of n. This should not be surprising since, by not restricting B, we effectively set off to achieve the goal of predicting in every round the index of the expert which is the best in that round. In fact, one can show that there is no algorithm whose worst-case expected regret is sublinear when competing against this class of experts (Exercise 6.1).

How to restrict B? There are many ways. One sensible way is to just compete with the switching experts which do not switch more than m times. The resulting problem is called the *tracking problem* because we want to track the best experts over time with some number of switches allowed. Let $s(\sigma)$ be the number of times expert σ switches from one base expert to another:

$$s(\sigma) = \sum_{t=2}^{n} \mathbb{I}\{\sigma_{t-1} \neq \sigma_t\}\,.$$

For an integer $0 \leq m \leq n-1$, let

$$B_{n,m} = \{\sigma \,|\, s(\sigma) \leq m\}\,.$$

How will EWA perform when the class of switching experts is $B_{n,m}$? In order to show this we only need to bound $M = |B_{n,m}|$. It is not hard to see that $M = \sum_{k=0}^{m} \binom{n-1}{k} N(N-1)^k$. Some further calculation gives that $M \leq N^{m+1} \exp(\,(n-1)H\left(\frac{m}{n-1}\right)\,)$, where $H(x) = -x \ln x - (1-x)\ln(1-x)$, $x \in [0,1]$, is the *entropy function*. Hence, the expected regret of randomized EWA applied to this problem is bounded as follows:

$$\mathbf{E}\left[R_n\right] \leq \sqrt{\frac{n}{2}\left((m+1)\ln N + (n-1)H\left(\frac{m}{n-1}\right)\right)}\,.$$

One scenario of interest is when we expect that for any n, by allowing $m_n \approx \alpha n$ switches during n rounds, the loss of the best expert in B_{n,m_n} will be reasonably small. When using such a sequence m_n, we are thus betting on a constant α "rate of change" of who the best expert is). Of course, in this case the expected regret per round will not vanish but it will converge to a constant value when $n \to \infty$. Some calculation gives that $\frac{\mathbf{E}[R_n]}{n} \leq \sqrt{\frac{1}{2}\left((\alpha + \frac{1}{n})\ln N + (1 - \frac{1}{n})H(\alpha)\right)}$ and thus

$$\frac{\mathbf{E}\left[R_n\right]}{n} = \sqrt{\frac{\alpha \ln N + H(\alpha)}{2}} + O\left(n^{-1/2}\right)\,.$$

Thus, the average regret per time step when the best expert is allowed to be changed at a rate α close to zero is $\sqrt{H(\alpha)/2}$ (since this is the dominating term in the first expression when $\alpha \to 0$).

Applying EWA directly to the set of switching experts $B_{n,m}$ is unpractical since EWA needs to store and update one weight per expert and the cardinality of $B_{n,m}$ is just too large for typical values of n, N and m.

6.2 Fixed-share forecaster

Our goal is to derive an algorithm which stores and updates N weights only, i.e., whose complexity is identical to that of the randomized EWA forecaster and yet it achieves the same regret bound as randomized EWA when competing with the best expert in $B_{n,m}$.

The main idea is twofold: The first observation is that whatever randomized EWA algorithm we use, at the end of the day in every round it will select some *base* expert. Therefore, if we can calculate the probability of choosing the base experts from round to round in an efficient manner, we will be done. The second main idea is that in order to make this happen, we should use a randomized EWA algorithm non-uniform initial weights where the set of switching experts is unrestricted. The non-uniform initial weights will still allow the encoding of our preference for no more than m switches, while working with them on the whole set of switching experts will make the efficient implementation of the algorithm possible.

We start with the following useful observation which concerns EWA with non-uniform weights. The proof is left as an exercise (see Exercise 6.4).

Lemma 6.1 (EWA with non-uniform priors)**.** *Consider a continuous, convex expert prediction problem given by (D, Y, ℓ) and N experts. Let w_{i0} be N nonnegative weights such that $W_0 = \sum_i w_{i0} \le 1$ and let $w_{i,t} = w_{i,0} e^{-\eta L_{i,t}}$, where $L_{i,t} = \sum_{s=1}^{t} \ell(f_{i,s}, y_s)$. Further, let $W_t = \sum_{i=1}^{N} w_{i,t}$. Then, for $p_t = \sum_{i=1}^{N} \frac{w_{i,t-1}}{W_{t-1}} f_{i,t}$ and any $\eta > 0$, it holds that*

$$\sum_{t=1}^{n} \ell(p_t, y_t) \le \frac{1}{\eta} \ln W_n^{-1} + \eta \frac{n}{8}\,.$$

How does this help? Since $W_n \ge w_{i,0} \exp(-\eta L_{i,n})$, $\ln(W_n^{-1}) \le \ln(w_{i,0}^{-1}) + \eta L_{i,n}$. Therefore, $\sum_{t=1}^{n} \ell(p_t, y_t) - L_{i,n} \le \frac{1}{\eta} \ln w_{i,0}^{-1} + \eta \frac{n}{8}$ and thus if i^* is the index of the expert with the smallest loss, $\sum_{t=1}^{n} \ell(p_t, y_t) - \min_{1 \le i \le n} L_{i,n} = \sum_{t=1}^{n} \ell(p_t, y_t) - L_{i^*,n} \le \frac{1}{\eta} \ln w_{i^*,0}^{-1} + \eta \frac{n}{8}$. Thus, if i^* was assigned a large weight (larger than $1/N$) the regret of EWA will be smaller than if all experts received the same weight: The weights $w_{i,0}$ act like our *a priori* bets on how much we believe in the experts initially. If these bets are correct, the algorithm is rewarded by achieving a smaller regret (this can also be seen directly since than $w_{i^*,n}$ will be larger).

Let us now go back to our problem of competing with the best switching expert. Consider now randomized EWA, but on the full set B of switching experts. Let $w'_t(\sigma)$ be the weight assigned to switching expert σ by the randomized EWA algorithm after observing $y_{1:t}$. As initial weights choose

$$w'_0(\sigma) \stackrel{\text{def}}{=} \frac{1}{N} \left(\frac{\alpha}{N}\right)^{s(\sigma)} \left(1 - \alpha + \frac{\alpha}{N}\right)^{n-1-s(\sigma)}\,.$$

Here, $0 < \alpha < 1$ reflects our *a priori* belief in switching per time step in a sense that will be made clear next. For $\sigma = (\sigma_1, \ldots, \sigma_n)$, $1 \le s < t \le n$, let $\sigma_{s:t}$ be $(\sigma_s, \ldots, \sigma_t)$. Introduce the "marginalized" weights $w'_0(\sigma_{1:t}) = \sum_{\sigma' : \sigma'_{1:t} = \sigma_{1:t}} w'_0(\sigma')$. We have the following lemma, which shows that the above weights are indeed sum to one and which also helps us in understanding where these weights are coming from:

Lemma 6.2 (Markov process view)**.** *Let (X_t) be a Markov chain with state space $\{1, \ldots, N\}$ defined as follows:* $\Pr[X_1 = i] = 1/N$ *and* $\Pr[X_{t+1} = i' \,|\, X_t = i] = \frac{\alpha}{N} + (1-\alpha)\mathbb{I}\{i' = i\}$. *Then, for any $\sigma \in B$, $1 \le t \le n$,* $\Pr[(X_1, \ldots, X_t) = \sigma_{1:t}] = w'_0(\sigma_{1:t})$ *and in particular,* $\Pr[(X_1, \ldots, X_n) = \sigma] = w'_0(\sigma)$.

The proof is left as an exercise (Exercise 6.5).

By definition, the probability that X_t and X_{t+1} differ is α/N, while the probability that they stay the same is $(1-\alpha) + \alpha/N \gg \alpha/N$ when α is small (and we expect to use a small value). Thus, sequences with many switches will have a small probability and the fewer switches a sequence has, the larger will be its probability (and, thus, initial weight). Thus, we also see that it indeed holds that α reflects our *a priori* belief in switching.

Let $w'_t(\sigma)$ denote the weights assigned to the switching expert σ by randomized EWA after seeing $y_{1:t}$. Let us calculate the probability $p'_{i,t+1}$ that randomized EWA will follow the advice of base expert i in the round $t+1$. Let $\widehat{\sigma}^{(t+1)}$ be the switching expert selected (randomly) by randomized EWA in round $t+1$. By definition, for an σ, $\Pr\left(\widehat{\sigma}^{(t+1)} = \sigma\right) = w'_t(\sigma) / \sum_{\sigma'} w'_t(\sigma')$. Now, notice that whenever $\widehat{\sigma}^{(t)}_{t+1} = i$, randomized EWA will eventually follow the advice of base expert i in round $t+1$. Thus,

$$\begin{aligned} p'_{i,t+1} &= \Pr\left(\widehat{\sigma}^{(t)}_{t+1} = i\right) = \sum_{\sigma} \Pr\left(\widehat{\sigma}^{(t)}_{t+1} = i, \widehat{\sigma}^{(t+1)} = \sigma\right) = \sum_{\sigma} \Pr\left(\sigma^{(t)}_{t+1} = i, \widehat{\sigma}^{(t+1)} = \sigma\right) \\ &= \sum_{\sigma : \sigma_{t+1} = i} \Pr\left(\widehat{\sigma}^{(t+1)} = \sigma\right) . \end{aligned}$$

Defining

$$w'_{i,t} = \sum_{\sigma : \sigma_{t+1} = i} w'_t(\sigma),$$

we see that

$$p'_{i,t+1} = \frac{w'_{it}}{W'_t},$$

where $W'_t = \sum_{i=1}^N w'_{it}$. Notice that by definition $w'_{i0} = w'_0(i)$ and so by Lemma 6.2, $w'_{i,0} = 1/N$.

Our goal now is to show that the (w'_{it}) weights can be calculated in a recursive fashion. Introduce the shorthand notation $\ell'_t(i) = \ell(f_{i,t}, y_t)$ to denote the loss of expert i. By definition, $w'_t(\sigma) = w'_0(\sigma) e^{-\eta L_{\sigma,t}}$, where $L_{\sigma,t} = \sum_{s=1}^t \ell(f_{\sigma_s,s}, y_s) = \sum_{s=1}^t \ell_s(\sigma_s)$. Define

$$\gamma_{i \to i'} = \frac{\alpha}{N} + (1-\alpha)\mathbb{I}\{i = i'\} .$$

Note that for any σ such that $\sigma_{t+1} = i$, $\gamma_{\sigma_t \to i} = \frac{w'_0(\sigma_{1:t+1})}{w'_0(\sigma_{1:t})}$, as follows from Lemma 6.2. Introduce $L_{\sigma_{1:t}} = \sum_{s=1}^{t} \ell_s(\sigma_s)$. Further, for arbitrary $1 \le p < q < \cdots < t \le n$ and $\sigma \in B$, by a slight abuse of notation, we shall also write $w'_0(\sigma_{1:p}, \sigma_{p+1:q}, \ldots, \sigma_{t+1:n})$ in place of $w'_0(\sigma)$. Then,

$$
\begin{aligned}
w'_{it} &= \sum_{\sigma:\sigma_{t+1}=i} w'_t(\sigma) = \sum_{\sigma:\sigma_{t+1}=i} e^{-\eta L_{\sigma,t}} w'_0(\sigma) = \sum_{\sigma:\sigma_{t+1}=i} e^{-\eta\,\ell_t(\sigma_t)} e^{-\eta L_{\sigma,t-1}} w'_0(\sigma) \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \sum_{\sigma_{1:t-1}} e^{-\eta L_{\sigma_{1:t-1}}} \sum_{\sigma_{t+2:n}} w'_0(\sigma_{1:t-1}, \sigma_t, i, \sigma_{t+2:n}) \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \sum_{\sigma_{1:t-1}} e^{-\eta L_{\sigma_{1:t-1}}} w'_0(\sigma_{1:t}, i) \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \sum_{\sigma_{1:t-1}} e^{-\eta L_{\sigma_{1:t-1}}} w'_0(\sigma_{1:t}) \frac{w'_0(\sigma_{1:t}, i)}{w'_0(\sigma_{1:t})} \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \sum_{\sigma_{1:t-1}} e^{-\eta L_{\sigma_{1:t-1}}} w'_0(\sigma_{1:t}) \gamma_{\sigma_t \to i} \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \gamma_{\sigma_t \to i} \sum_{\sigma_{1:t-1}} e^{-\eta L_{\sigma_{1:t-1}}} w'_0(\sigma_{1:t}) \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \gamma_{\sigma_t \to i} \sum_{\sigma_{1:t-1}} e^{-\eta L_{\sigma_{1:t-1}}} \sum_{\sigma_{t+1:n}} w'_0(\sigma_{1:n}) \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \gamma_{\sigma_t \to i} \sum_{\sigma_{1:t-1}} \sum_{\sigma_{t+1:n}} e^{-\eta L_{\sigma_{1:t-1}}} w'_0(\sigma_{1:n}) \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \gamma_{\sigma_t \to i} \sum_{\sigma_{1:t-1}} \sum_{\sigma_{t+1:n}} w'_{t-1}(\sigma_{1:n}) \\
&= \sum_{\sigma_t} e^{-\eta\,\ell_t(\sigma_t)} \gamma_{\sigma_t \to i} w'_{\sigma_t, t-1}\,.
\end{aligned}
$$

Thus,

$$
\begin{aligned}
w'_{it} &= \sum_j e^{-\eta\,\ell_t(j)} w'_{j,t-1} \gamma_{j \to i} && (6.2) \\
&= \sum_j e^{-\eta\,\ell_t(j)} w'_{j,t-1} \left\{ \frac{\alpha}{N} + (1-\alpha) \mathbb{I}\{j = i\} \right\} && (6.3) \\
&= (1-\alpha) e^{-\eta\,\ell_t(i)} w'_{i,t-1} + \frac{\alpha}{N} \sum_j e^{-\eta\,\ell_t(j)}\, w'_{j,t-1}\,,
\end{aligned}
$$

giving rise to the so-called Fixed-Share Forecaster (FSF).

Formally, this forecaster works as follows. It keeps N weights. Initially, $w_{i0} = 1/N$. In round $t = 1, 2, \ldots, n$, the forecaster does the following:

1. Observes the expert forecasts $f_{i,t}$.

2. Draws the index I_t of a base expert such that $\Pr(I_t = i) = p_{i,t}$, where $p_{i,t} = \frac{w_{i,t-1}}{\sum_{j=1}^{N} w_{j,t-1}}$.

3. Predicts $f_{I_t,t}$.

4. Observes y_t, the losses $\ell(f_{i,t}, y_t)$ (and it suffers the loss $\ell(f_{I_t,t}, y_t)$).

5. Computes $v_{i,t} = w_{i,t-1} e^{-\eta \ell(f_{i,t}, y_t)}$.

6. Updates the weights by $w_{i,t} = \frac{\alpha}{N} W_t + (1-\alpha) v_{i,t}$, where $W_t = \sum_{j=1}^{N} v_{j,t}$.

6.3 Analysis

Theorem 6.3. *Consider a discrete prediction problem over the arbitrary set D, Y and the zero-one loss ℓ. Let $y_1, \ldots, y_n \in Y$ be an arbitrary sequence of outcomes, $f_{i,t} \in D$ be the advice of base expert i in round t, where $1 \le i \le N$, $1 \le t \le n$. Let $\widehat{L}_n$ be the cumulated loss of the* Fixed-Share Forecaster *at the end of round n and, similarly, let $L_{\sigma,n}$ be the cumulated loss of switching expert σ at the end of round n. Then,*

$$\mathbf{E}\left[\widehat{L}_n\right] - L_{\sigma,n} \le \frac{s(\sigma)+1}{\eta} \ln N + \frac{1}{\eta} \ln\left(\frac{1}{\alpha^{s(\sigma)}(1-\alpha)^{n-s(\sigma)-1}}\right) + \eta \frac{n}{8}.$$

Further, for $0 \le m \le n$, $\alpha = m/(n-1)$, with a specific choice of $\eta = \eta(n, m, N)$, for any σ with $s(\sigma) \le m$,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_{\sigma,n} \le \sqrt{\frac{n}{2}\left((m+1)\ln N + (n-1) H\left(\frac{m}{n-1}\right)\right)}.$$

We see from the second part the algorithm indeed achieves the same regret bound as randomized EWA with uniform initial weights competing with experts in $B_{n,m}$.

Proof. Since FSF implements randomized EWA *exactly* we can use the reduction technique developed in the proof of Theorem 4.2 to study its performance. The only difference is that now we use non-uniform initial weights.Therefore, the problem reduces to the bound in Lemma 6.1. Then, following the argument already used after Lemma 6.1, we get that for any $\sigma \in B$,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_{\sigma,n} \le \frac{-\ln w'_0(\sigma)}{\eta} + \eta \frac{n}{8}.$$

Thus, we need an upper bound on $-\ln w'_0(\sigma)$. From the definition,

$$w'_0(\sigma) = \frac{1}{N}\left(\frac{\alpha}{N}\right)^{s(\sigma)} \left(1 - \alpha + \frac{\alpha}{N}\right)^{n-s(\sigma)-1}.$$

To finish, just note that $-\ln w'_0(\sigma) \le (1 + s(\sigma)) \ln(N) + \ln(\frac{1}{\alpha^{s(\sigma)}(1-\alpha)^{n-s(\sigma)-1}})$. □

6.4 Variable-share forecaster

According to the result of Theorem 6.3, for any switching expert σ,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_{\sigma,n} \leq \frac{(s(\sigma)+1)\,\ln N}{\eta} + \frac{s(\sigma)\,\ln\left(\frac{1}{\alpha}\right) + (n-s(\sigma)-1)\,\ln\left(\frac{1}{1-\alpha}\right)}{\eta} + \eta\,\frac{n}{8}\,.$$

Because of the term $(n-s(\sigma)-1)\ln\left(\frac{1}{1-\alpha}\right)$ even if σ has a small number of switches and at the same time $L_{\sigma,n}$ is small, the loss of the FSF can be large. As we shall see soon, the so-called Variable-Share Forecaster (VSF) avoids this problem. The idea is to change the priors to achieve this. In fact, that the prior can be chosen in a flexible manner follows since the derivation of the simulation equivalence of the randomized EWA working with the set B and the prior, and an incremental algorithm defined using (6.2) works for any prior, as long as

$$\gamma^{(t)}_{\sigma_t \to i} \stackrel{\text{def}}{=} \frac{w'_0(\sigma_{1:t+1}}{w'_0(\sigma_{1:t})}$$

is well-defined (i.e., independent of $\sigma_{1:t-1}$), and if in (6.2) we replace $\gamma_{j\to i}$ by $\gamma^{(t)}_{\sigma_t\to i}$. In fact, we see that the incremental update will only use past information as long as $\gamma^{(t)}_{\sigma_t\to i}$ is computable based on $y_1,\dots,y_t$. This is what we will exploit when defining a new prior.

The main idea is to change the prior w'_0 such that it penalizes switches away from good base experts. This is achieved by redefining w'_0 so that

$$w'_0(\sigma_{1:t+1}) = w'_0(\sigma_{1:t})\left(\frac{1-(1-\alpha)^{\ell_t(\sigma_t)}}{N-1}\,\mathbb{I}\{\sigma_t \neq \sigma_{t+1}\} + (1-\alpha)^{\ell_t(\sigma_t)}\,\mathbb{I}\{\sigma_t = \sigma_{t+1}\}\right).$$

Now, when $\ell_t(\sigma_t)$ is close to zero, $(1-\alpha)^{\ell_t(\sigma_t)}$ will be close to one. Hence, the first term of the sum in the bracket will be close to zero, while the second one will be close to one if and only if $\sigma_t = \sigma_{t+1}$. Thus, in this case, from the Markov process view, we see that staying at σ_t is encouraged in the prior. On the other hand, when $\ell_t(\sigma_t)$ is close to one, $(1-\alpha)^{\ell_t(\sigma_t)}$ will be close $1-\alpha$. Hence, the expression in the bracket will be close to the previous expression and staying will be encouraged by a probability close to the "default stay probability", $1-\alpha$. Therefore, these weights are expected to result in a smaller regret when there is an expert with small cumulated loss and a few number of switches. Further, $\ell_t(\sigma_t) = \ell(f_{\sigma_t,t}, y_t)$ is available at the end of round t, therefore

$$\gamma^{(t)}_{\sigma_t\to i} = \left(\frac{1-(1-\alpha)^{\ell_t(\sigma_t)}}{N-1}\,\mathbb{I}\{\sigma_t \neq \sigma_{t+1}\} + (1-\alpha)^{\ell_t(\sigma_t)}\,\mathbb{I}\{\sigma_t = \sigma_{t+1}\}\right)$$

is not only well-defined, but it's value is also available at the end of round t.

This leads to the Variable-Share Forecaster (VSF). Formally, this algorithm works as follows: Before round one, initialize the weights using $w_{i,0} = 1/N$. In round $t = 1, 2, 3, \dots$, the VSF does the following:

1. Observes the expert forecasts $f_{i,t}$.

2. Draws the index I_t of a base expert such that $\Pr(I_t = i) = p_{i,t}$, where $p_{i,t} = \frac{w_{i,t-1}}{\sum_{j=1}^{N} w_{j,t-1}}$.

3. Predicts $f_{I_t,t}$.

4. Observes y_t, the losses $\ell(f_{i,t}, y_t)$ (and it suffers the loss $\ell(f_{I_t,t}, y_t)$).

5. Computes $v_{i,t} = w_{i,t-1} e^{-\eta \ell(f_{i,t}, y_t)}$.

6. Updates the weights by $w_{i,t} = \frac{1}{N-1} \sum_{j \neq i} \left(1 - (1-\alpha)^{\ell_t(j)}\right) v_{jt} + (1-\alpha)^{\ell_t(i)} v_{it}$.

It is then not hard to se that the result of this update is that for binary losses, $\frac{n-s(\sigma)-1}{\eta} \ln \frac{1}{1-\alpha}$ in the bound is replaced by $s(\sigma) + \frac{1}{\eta} L_{\sigma,n} \ln \frac{1}{1-\alpha}$. Hence, the VSF may achieve much smaller loss when some expert σ which does not switch too often achieves a small loss.

6.5 Exercises

Exercise 6.1. Let $N > 1$. Show that there is no algorithm whose worst-case expected regret is sublinear when competing against all switching experts. More precisely, show that there exists a constant c such that for $D = [0,1], Y = \{0,1\}, \ell(p,y) = |p-y|$, for any $N > 1$, for any algorithm, there exists a set of base experts of size N and a time horizon n such that the regret with respect to all switching experts is at least cn.

Exercise 6.2. Show that an algorithm that competes against experts in $B_{n,0}$ is effectively back to competing with the best of the base experts.

Exercise 6.3. Show that $nH(m/n) = O(\ln(n))$ as $n \to \infty$. *Hint*: For large n, $H(m/n) \approx m/n \ln(n/m)$, therefore $nH(m/n) = O(\ln(n))$.

Exercise 6.4. Prove Lemma 6.1

Exercise 6.5. Prove Lemma 6.2.

Exercise 6.6. Assuming a constant rate of change α prove a minimax lower bound on the average expected regret per time step.

Exercise 6.7. Give a practical algorithm that does not require the knowledge of the horizon n and which achieves the $O(\sqrt{H(\alpha)/2})$ regret per time step when the rate of change of the identity of the best expert is bounded by α. You may assume that α is known ahead of time.

Exercise 6.8. Give an algorithm like in the previous exercise, except that now the algorithm does not know α, but it may know n.

Exercise 6.9. Give an algorithm like in the previous exercise, except that now neither α, nor n is known.

Chapter 7

Linear classification with Perceptron

Suppose we want to classify emails according to if they are spam (say, encoded $+1$) or not-spam (say, encoded by -1). From the text of the emails we extract the features, $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$ (e.g., $x_{t,i} \in \{0,1\}$ indicates whether a certain phrase or word is in the email), and we assign to every email a target label (or output) y_t such that, say, $y_t = +1$ if the email is spam. The features will also be called inputs.

A *classifier* f is just a mapping from $\mathbb{R}^d$ to $\{-1,+1\}$. An *online learning algorithm* upon seeing the samples

$$(x_1, y_1), \ldots, (x_{t-1}, y_{t-1})$$

and x_t produces a classifier f_{t-1} to predict y_t ($1 \le t \le n$).

The algorithm is said to *make a mistake*, if its prediction, $\widehat{y}_t$ does not match y_t. The *total number of mistakes* of the algorithm is $M = \sum_{t=1}^{n} \mathbb{I}\{\widehat{y}_t \neq y_t\}$. The general goal in online learning of classifiers is to come up with an online algorithm that makes a small number of mistakes.

A *linear classifier* $f_w : \mathbb{R}^d \to \{-1,+1\}$ is given by a weight $w \in \mathbb{R}^d$, $w \neq 0$ such that

$$f_w(x) = \begin{cases} +1, & \text{if } \langle w, x\rangle \ge 0\,; \\ -1, & \text{otherwise}\,. \end{cases}$$

For simplicity, at the price of abusing the sign function, we will just write $f_w(x) = \operatorname{sign}\langle w, x\rangle$ (but when we write this, we will mean the above). Introduce the $(d-1)$-dimensional hyperplane $H_w = \{x : \langle w, x\rangle = 0\}$. Thus, if $\langle w, x\rangle = 0$ then x is on the hyperplane H_w. We will also say that x is *above* (*below*) the hyperplane H_w when $\langle w, x\rangle$ is positive (resp., negative). Thus, $f_w(x) = +1$ is x is above the hyperplane H_w etc. In this sense, H_w is really the *decision surface* underlying f_w. (In general, the decision "surface" underlying a classifier f is $\{x : f(x) = 0\}$.)

By the law of cosines, $|\langle w, x\rangle| = \|w\|\,\|x\|\,\cos(\angle(w, x))$, therefore $|\langle w, x\rangle|$ is just $\|w\|$ times the distance of x from H_w.[1] In particular, when $\|w\| = 1$, $|\langle w, x\rangle|$ is the distance of x to H_w. In some sense, this should also reflect how confident we are in our prediction of the label, if

[1] As usual, $\|\cdot\|$ is the 2-norm.

we believe w gives a good classifier. The distance of x to the hyperplane H_w is also called the (unsigned) *margin* of x (the signed margin of the pair $(x, y) \in \mathbb{R}^d \times \{-1, +1\}$ would be $y\langle w, x\rangle$).

The general scheme for online learning with linear classifiers is as follows:
Initialize w_0.

1. Receive $x_t \in \mathbb{R}^d$.
2. Predict $\widehat{y}_t = \text{sign}(\langle w_{t-1}, x_t\rangle)$.
3. Receive the correct label $y_t \in \{-1, +1\}$.
4. Update w_t based on w_{t-1}, x_t, y_t.

Remark 7.1 (Bias (or intercept) terms)**.** We defined linear classifiers as function of the form $f(x) = \text{sign}(\langle w, x\rangle)$. However, this restricts the decision surfaces to hyperplanes which cross the origin. A more general class of classifiers allows a *bias* term (or *intercept* term). $f(x) = \text{sign}(\langle w, x\rangle + w_0)$, where $w, x \in \mathbb{R}^d$, $w_0 \in \mathbb{R}$. These allow hyperplanes which do not cross the origin. However, any linear classifier with a bias can be given as a linear classifier with *no* bias term when the input space is appropriately enlarged. In particular, for w, x, w_0, let $w' = (w_1, \ldots, w_d, w_0)$ and $x' = (x_1, \ldots, x_d, 1)$. Then $\text{sign}(\langle w, x\rangle + w_0) = \text{sign}(\langle w', x'\rangle)$. In practice, this just means that before running the algorithms one should just amend the input vectors by a constant 1.

7.1 The Perceptron Algorithm

The Perceptron algorithm was invented by Rosenblatt, a psychologist, in 1950s, to explain how the neurons in the brain might work! The algorithm follows the above general scheme.

Here is the algorithm: Initialize $w_0 = 0$.

1. Receive $x_t \in \mathbb{R}^d$.
2. Predict $\widehat{y}_t = \text{sign}(\langle w_{t-1}, x_t\rangle)$.
3. Receive the correct label $y_t \in \{-1, +1\}$.
4. Update w_t based on w_{t-1}, x_t, y_t:

$$w_t = \begin{cases} w_{t-1} + y_t\, x_t, & \text{if } y_t \neq \widehat{y}_t \text{ i.e., if the algorithm made a mistake,} \\ w_{t-1}, & \text{otherwise.} \end{cases}$$

7.2 Analysis for Linearly Separable Data

Let $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ be the data where $x_i \in \mathbb{R}^d$, $y_i \in \{-1, +1\}$. We say that $w^* \in \mathbb{R}^d$ *separates* the data set if $w^* \in \mathbb{R}^d$ such that $\text{sign}(\langle w^*, x_t \rangle) = y_t$ for all $1 \leq t \leq n$. If there exists w^* which separates the data set, we call the data set *linearly separable*. Notice that if w^* separates the data sets, then for any $c > 0$, cw^* separates it as well. So, we may even assume that $\|w^*\| = 1$, which we will indeed do from this point on.

Theorem 7.2 (Novikoff's Theorem (1962))**.** *Let $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ be data set that is separated by a $w^* \in R^d$. Let $R, \gamma \geq 0$ be such that for all $1 \leq t \leq n$, $\|x_t\| \leq R$ and $y_t \langle w^*, x_t \rangle \geq \gamma$. Let M be the number of mistakes* Perceptron *makes on the data set. Then,*

$$M \leq \frac{R^2}{\gamma^2} .$$

Proof. We want to prove that $M \leq R^2/\gamma^2$, or $M\gamma^2 \leq R^2$. We will prove an upper bound on $\|w_n\|^2$ and a lower bound on $\langle w^*, w_n \rangle$, from which the bound will follow. To prove these bounds, we study the evolution of $\|w_t\|^2$, and $\langle w^*, w_t \rangle$.

If Perceptron makes no mistake, both quantities stay the same. Hence, the only interesting case is when Perceptron makes a mistake, i.e., $\widehat{y}_t \neq y_t$. Let t be such a time step. We have

$$\|w_t\|^2 = \|w_{t-1} + y_t x_t\|^2 = \|w_{t-1}\|^2 + \|x_t\|^2 + 2y_t \langle w_{t-1}, x_t \rangle .$$

Since Perceptron made a mistake, $y_t \langle w_{t-1}, x_t \rangle \leq 0$. Hence,

$$\|w_t\|^2 \leq \|w_{t-1}\|^2 + \|x_t\|^2 \leq \|w_{t-1}\|^2 + R^2.$$

Thus, $\|w_n\|^2 \leq MR^2 + \|w_0\|^2$ and since $\|w_0\|^2 = 0$, we have $\|w_n\|^2 \leq MR^2$.

Now, let us study $\langle w^*, w_t \rangle$. Suppose again that there was a mistake at time step t. Then,

$$\langle w^*, w_t \rangle = \langle w^*, w_{t-1} \rangle + y_t \langle w^*, x_t \rangle \geq \langle w^*, w_{t-1} \rangle + \gamma ,$$

where the inequality follows because by assumption, $y_t \langle w^*, x_t \rangle \geq \gamma$. Hence, by unfolding the recurrence,

$$\langle w^*, w_n \rangle \geq \gamma M .$$

By Cauchy-Schwarz, $|\langle w^*, w_n \rangle| \leq \|w^*\| \|w_n\| = \|w_n\|$, where the last equality follows since by assumption $\|w^*\| = 1$. Chaining the obtained inequalities we get,

$$\gamma^2 M^2 \leq \|w_n\|^2 \leq R^2 M .$$

If $M = 0$, the mistake bound indeed holds. Otherwise, we can divide both sides by $M\gamma^2$, to get the desired statement. □

Intuitively, if the smallest margin on the examples is big, it should be easier to find a separating hyperplane. According to the bound, this is indeed what the algorithm achieves! Why do we have R^2 in the bound? To understand this, notice that the algorithm is scale invariant (it makes the same mistakes if all inputs are multiplied by some $c > 0$). Thus, the bound on the number of mistakes should be scale invariant! Since the margin changes by a factor c^2 when scaling the inputs by $c > 0$, $R^2 = \max_{1 \le t \le n} \|x_t\|^2$ must appear in the bound. In other words, the number of mistakes that PERCEPTRON makes scales inversely proportional to the square of the size of the *normalized margin*.

The bound becomes smaller, when γ is larger. In fact, the best γ is

$$\gamma^* = \sup_{1 \le t \le n} \sup_{w : \|w\| = 1} y_t \langle w^*, x_t \rangle ,$$

which is the "maximum margin".

An interesting feature of the bound is that it is independent of n, the sample size: If the data is separable, Perceptron will make a finite number of mistakes on it. In fact, the argument works even when $n = \infty$: If an infinite sample is separable *with a positive margin*, PERCEPTRON still makes only a finite number of mistakes!

Remark 7.3. Novikoff's proof follows pretty much the same patterns as the proofs which we have seen beforehand: We prove upper and lower bounds on some function of the weights and then combine these to get a bound. This should not be very surprising (given that the essence of the algorithms is that they change their weights).

Remark 7.4. We can use the PERCEPTRON algorithm to solve linear feasibility problems. A linear feasibility problem (after maybe some transformation of the data inputs) is the problem of finding a weight $w \in \mathbb{R}^d$ such that, with some $D = ((x_t, y_t))_{1 \le t \le n}$, $x_t \in \mathbb{R}^d$, $y_t \in \{-1, +1\}$,

$$y_t \langle w, x_t \rangle > 0, \quad t = 1, 2, \ldots, n$$

holds. Now, this is nothing but finding a separating hyperplane. If this problem has a solution, repeatedly running PERCEPTRON on the data D until it does not make any mistakes will find a solution. (Why?) One can even bound the number of sweeps over D needed until the algorithm will stop. Thus, we have an algorithm (PERCEPTRON) which can be coded up in 5 minutes to solve linear feasibility problems (if they have a solution). See Exercise 7.1 for some further ideas.

7.3 Analysis in the General Case

The linearly separable case is similar to the one when, in the prediction with expert advice setting, there was a perfect expert. In fact, PERCEPTRON parallels the HALVING algorithm at least in the respect that both "learn" from their mistakes. There, we saw that the halving

algorithm has a natural extension to the case when no expert is perfect. Can we have some similar extensions in the linear classification case? In particular, is it possible to prove some nice regret bounds of the form

$$M \leq \min_{w \in \mathbb{R}^d} \sum_{t=1}^{n} \mathbb{I}\{y_t \langle w, x_t \rangle \leq 0\} + \text{“something small”} \ ?$$

In the expert setting, we saw that no regret bound is possible, unless we randomize. The following theorem parallels this result:

Theorem 7.5. *For any deterministic algorithm A and any $n \geq 0$ there exists a data sequence $(x_1, y_1), \ldots, (x_n, y_n)$ and $w^* \in \mathbb{R}^d$ such that the following hold:*

(i) $\text{sign}(\langle w^*, x_t \rangle) = y_t$ *for all* $t = 1, 2, \ldots, n$.

(ii) Algorithm A makes a mistake in every round.

The proof is left as an exercise (Exercise 7.2). In the discrete prediction setting randomization saved us. Unfortunately, randomization cannot save us now (Exercise 7.3).

So, what do we do? The Big Cheat! We introduce a surrogate loss, the so-called *hinge loss* ℓ. The 0-1 loss (or binary classification loss) is:

$$\ell_{0-1}(w, (x, y)) = \mathbb{I}\{y \langle w, x \rangle \leq 0\}.$$

The hinge-loss is:

$$\ell_{\text{hinge}}(w, (x, y)) = \max(0, 1 - y \langle w, x \rangle)\,.$$

For r real, by defining $(r)_+ = \max(r, 0)$, we can also write $\ell_{\text{hinge}}(w, (x, y)) = (1 - y \langle w, x \rangle)_+$, which is also a common notation.

Now, how does the hinge loss work? If w classifies x, y incorrectly then $\ell_{\text{hinge}}(w, (x, y)) \geq 1$. On the other hand, if w classifies x, y correctly with margin greater than one (i.e., if $|\langle w, x \rangle| \geq 1$), then $\ell_{\text{hinge}}(w, (x, y)) = 0$. When the margin is smaller than one (but w still classifies (x, y) correctly) then the hinge loss is a number between 0 and 1 (i.e., the classifier pays a small penalty for not being “confident enough”).

Two important properties of the hinge loss are that it is an upper bound on the 0-1 loss:

$$\ell_{0-1}(w, (x, y)) \leq \ell_{\text{hinge}}(w, (x, y))\,,$$

and that it is convex in its first argument (Exercise 7.4).

Now, notice that with the notation just introduced, we can view the problem as a prediction with expert advice problem, where the experts are the weights w and the outcomes are of the form of input-output pairs (x, y). We saw in that framework that convexity plays a major role. In some sense convexity makes these problems easy.

So the cheat is to replace the 0-1 loss with the hinge-loss, and then we can prove regret bounds. With this, we can get the following theorem:

Theorem 7.6. *Let M be the number of mistakes that the* Perceptron *algorithm makes on a sequence $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \in \mathbb{R}^d \times \{+1, -1\}$. Suppose $R \geq 0$ is such that $\|x_t\| \leq R$ for all $1 \leq t \leq n$. For any $w \in \mathbb{R}^d$*

$$M \leq \sum_{t=1}^{n} \ell_{\text{hinge}}(w, (x_t, y_t)) + \|w\| R\sqrt{n} \, .$$

Proof. Fix $w \in \mathbb{R}^d$. We analyze the evolution of $\|w_t\|^2$ and $\langle w, w_t \rangle$. When a mistake occurs in time step t then $y_t \langle w_{t-1}, x_t \rangle \leq 0$ and therefore

$$\|w_t\|^2 = \|w_{t-1} + y_t x_t\|^2 = \|w_t\|^2 + \|x_t\|^2 + 2y_t \langle w_{t-1}, x_t \rangle \leq \|w_t\|^2 + \|x_t\|^2 \leq \|w_t\|^2 + R^2 \, .$$

Thus, after processing all n points and making all M mistakes, by unrolling the recurrence we get $\|w_n\|^2 \leq \|w_0\|^2 + MR^2 = MR^2$. Taking square root and using $M \leq n$ we have $\|w_n\| \leq R\sqrt{M} \leq R\sqrt{n}$.

Similarly, if a mistake occurs in time step t then

$$\langle w, w_t \rangle = \langle w, w_{t-1} + y_t x_t \rangle = \langle w, w_{t-1} \rangle + y_t \langle w, x_t \rangle \geq \langle w, w_{t-1} \rangle + 1 - \ell_{\text{hinge}}(w, (x_t, y_t))$$

where in the last step we have used the inequality $y \langle w, x \rangle \geq 1 - \ell_{\text{hinge}}(w, (x, y))$ valid for any $w, x \in \mathbb{R}^d$ and any $y \in \{+1, -1\}$. We unroll this recurrence in each round t in which Perceptron makes a mistakes. If no mistake was made, we use $\langle w, w_t \rangle = \langle w, w_{t-1} \rangle$ instead. We get

$$\langle w, w_n \rangle \geq \langle w, w_0 \rangle + M - \sum_{\substack{1 \leq t \leq n \\ \text{sign}(\langle w_t, x_t \rangle) \neq y_t}} \ell_{\text{hinge}}(w, (x_t, y_t)) \, .$$

Using $\|w_n\| \leq R\sqrt{n}$, Cauchy-Schwarz inequality, $w_0 = 0$ and the fact that hinge loss is non-negative, we can write:

$$\begin{aligned} R\|w\|\sqrt{n} &\geq \|w\| \|w_n\| \\ &\geq \langle w, w_n \rangle \\ &\geq \langle w, w_0 \rangle + M - \sum_{\substack{1 \leq t \leq n \\ \text{sign}(\langle w_t, x_t \rangle) \neq y_t}} \ell_{\text{hinge}}(w, (x_t, y_t)) \\ &\geq M - \sum_{t=1}^{n} \ell_{\text{hinge}}(w, (x_t, y_t)) \, . \end{aligned}$$

Reading off the beginning and the end of the chain of inequalities gives the statement of the theorem. □

7.4 Exercises

Exercise 7.1. Consider the algorithm proposed in Remark 7.4.

(a) Show that if this algorithm stops, it will stop with a solution to the linear feasibility problem.

(b) Let S be the number of sweeps the algorithm makes on the data D before it stops. Show a bound S.

(c) What happens when the original problem does not have a solution? Suggest a simple way of detecting that there is no solution.

Exercise 7.2. Prove Theorem 7.5.

Exercise 7.3. Prove that Theorem 7.5 holds when A can randomize with the modification that in n rounds the algorithm A makes on expectation $n/2$ mistakes.

Exercise 7.4. Show that $\ell_{\text{hinge}}(w, (x, y))$ is convex in its first argument.

Chapter 8

Follow the Regularized Leader and Bregman divergences

In this chapter we describe the generic Follow The Regularized Leader algorithm for linear loss functions and derive a regret bound for it. To analyze and implement the algorithm, we will need bits of convex analysis such as Legendre functions, Bregman divergences, Bregman projections and strong convexity.

The restriction to linear loss functions is not as severe as one might think. In later lectures, we will see that we can cope with non-linear loss functions by working with their linear approximations. The analysis in the linear case is simpler and leads often to computationally faster algorithms.

We consider an online learning scenario where an online algorithm in each round $t = 1, 2, \ldots, n$ chooses a point w_t in a non-empty convex set $K \subseteq \mathbb{R}^d$ and suffers a loss $\ell_t(w_t)$. The loss function $\ell_t : \mathbb{R}^d \to \mathbb{R}$ is chosen by the adversary and we assume that it is linear i.e. $\ell_t(w) = \langle f_t, w \rangle$ where $f_t \in \mathbb{R}^d$.

Follow The Regularized Leader (FTRL) algorithm is a particular algorithm for this scenario which in round $t + 1$ chooses $w_{t+1} \in K$ based on the sum loss functions up to time t:

$$L_t(w) = \sum_{s=1}^{t} \ell_s(w)$$

Namely, the algorithm chooses

$$w_{t+1} = \operatorname*{argmin}_{w \in K \cap A} \ [\eta L_t(w) + R(w)] \ ,$$

where $R : A \to \mathbb{R}$ is a so-called *regularizer* or *regularization function* and we assume that it is defined on some set $A \subseteq \mathbb{R}^d$. The parameter $\eta > 0$ is the *learning rate*. Different choices of the regularizer lead to different algorithms; later we will see how the choice influences the regret of the algorithm.

Intuitively, FTRL algorithm tries to balance two things: Minimize the loss on current vector and ensure that w_{t+1} is close to w_t. This will formalized in Corollary 8.12.

We make additional assumptions on K and the regularizer R. We assume that K is closed. Furthermore, we assume that R is a Legendre function. In the next section, we explain what are Legendre functions and we introduce other necessary tools from convex analysis.

8.1 Legendre functions and Bregman divergences

In what follows, A° denotes the interior of a set $A \subset \mathbb{R}^d$ and $\|\cdot\|$ denotes some norm on $\mathbb{R}^d$.

Definition 8.1 (Legendre function)**.** A function $F : A \to \mathbb{R}$ is a called a *Legendre function* if it satisfies the following conditions:

1. $A \subseteq \mathbb{R}^d$, $A \neq \emptyset$, A° is convex.
2. F is strictly convex.
3. Partial derivatives $\frac{\partial F}{\partial x_i}$ exists for all $i = 1, 2, \ldots, d$ and are continuous.
4. Any sequence $\{x_t\} \subseteq A$ approaching the boundary of A satisfies $\lim_{t\to\infty} \|\nabla F(x_t)\| = \infty$.

In another words, a Legendre function is a strictly convex functions with continuous partial derivatives and gradient "blowing up" at the boundary of its domain. Note the definition does not depend on the norm used for $\|\nabla F\|$; this is a consequence of that for any pair of norms $\|\cdot\|_\heartsuit, \|\cdot\|_\spadesuit$ on $\mathbb{R}^d$ if $\|\cdot\|_\heartsuit \to \infty$ then $\|\cdot\|_\spadesuit \to \infty$.

Definition 8.2 (Bregman divergence)**.** Let F be a Legendre function $F : A \to \mathbb{R}$. The *Bregman divergence* corresponding to F is a function $D_F : A \times A^\circ \to \mathbb{R}$ defined by the formula

$$D_F(u, v) = F(u) - F(v) - \langle \nabla F(v), u - v \rangle .$$

Bregman divergence is the difference between the function value $F(u)$ and its approximation by the first order Taylor expansion $F(v) + \langle \nabla F(v), u - v \rangle$ around v. The first order Taylor expansion is a linear function tangent to F at point v. Since F is convex, the linear function lies below F and therefore D_F is non-negative. Furthermore, $D_F(u, v) = 0$ implies that $u = v$ because of strict convexity of F the only point where the linear approximation meets F is $u = v$.

Definition 8.3 (Bregman projection)**.** Let $F : A \to \mathbb{R}$ be Legendre function and let $K \subseteq \mathbb{R}^d$ be a closed convex subset such that $K \cap A \neq \emptyset$. The *Bregman projection* corresponding to F and K is a function $\Pi_{F,K} : A^\circ \to A \cap K$ defined for any $w \in A^\circ$ as

$$\Pi_{F,K}(w) = \operatorname*{argmin}_{u \in K \cap A} D_F(u, w) .$$

It is a non-trivial fact to verify that $\Pi_{F,K}$ is well defined. More precisely, it needs to be shown that minimizer $\min_{u\in K\cap A} D_F(u,w)$ is attained and it is unique. The former follows from that K is closed. The later from strict convexity of F.

Lemma 8.4 (Pythagorean inequality)**.** *Let $F : A \to \mathbb{R}$ be Legendre function and let $K \subseteq \mathbb{R}^d$ be a closed convex subset such that $K \cap A \neq \emptyset$. If $w \in A^\circ$, $w' = \Pi_{F,K}(w)$, $u \in K$ then*

$$D_F(u,w) \geq D_F(u,w') + D_F(w',w) \ .$$

Lemma 8.5 (Kolmogorov's inequality)**.** *Let $F : A \to \mathbb{R}$ be Legendre function and let $K \subseteq \mathbb{R}^d$ be a closed convex subset such that $K \cap A \neq \emptyset$. If $u, v \in A^\circ$ and $u' = \Pi_{F,K}(u)$, $v' = \Pi_{F,K}(u)$ are their projections then*

$$\langle u' - w', \nabla F(u') - \nabla F(w')\rangle \leq \langle u' - w', \nabla F(u) - \nabla F(w)\rangle \ .$$

Lemma 8.6 (Projection lemma)**.** *Let $F : A \to \mathbb{R}$, $A \subset \mathbb{R}^d$ be a Legendre function and $\widetilde{w} = \operatorname{argmin}_{u\in A} F(u)$. Let $K \subseteq \mathbb{R}^d$ be convex closed and set such that $K \cap A \neq \emptyset$. Then,*

$$\Pi_{F,K}(\widetilde{w}) = \operatorname*{argmin}_{u\in K\cap A} F(u) \ .$$

Proof. Let $w' = \Pi_{F,K}(\widetilde{w})$ and $w = \operatorname{argmin}_{u\in K\cap A} F(u)$. We need to prove $w = w'$. Since w is a minimizer, we have $F(w) \leq F(w')$. If we are able to prove the reversed inequality $F(w') \leq F(w)$ then by strict convexity of F, the minimizer of F is unique and hence $w = w'$.

It thus remains to prove that $F(w') \leq F(w)$. By definition of Bregman projection $w' = \operatorname{argmin}_{u\in K\cap A} D_F(u,\widetilde{w})$ and therefore

$$D_F(w',\widetilde{w}) \leq D_F(w,\widetilde{w}) \ .$$

Expanding D_F on both sides of the inequality, we get

$$F(w') - F(\widetilde{w}) - \langle \nabla F(\widetilde{w}), w' - \widetilde{w}\rangle \leq F(w) - F(\widetilde{w}) - \langle \nabla F(\widetilde{w}), w - \widetilde{w}\rangle \ .$$

We cancel $F(\widetilde{w})$ on both sides and note that $\nabla F(\widetilde{w}) = 0$ since $\widetilde{w}$ is an unconstrained minimizer of F. We obtain $F(w') \leq F(w)$ as promised. □

The projection lemma says that the constrained minimizer w can be calculated by first computing the unconstrained minimizer $\widetilde{w}$ and then projecting it onto K using Bregman projection. This will be not only useful in the regret analysis of FTRL, but it can be also used to implement FTRL.

8.2 Strong Convexity and Dual Norms

Definition 8.7 (Strong Convexity)**.** Let $F : A \to \mathbb{R}$ Legendre. We say that F is *strongly convex* with respect to a norm $\|\cdot\|$ if for any $u, v \in A$

$$F(u) - F(v) \geq \langle \nabla F(v), u - v\rangle + \frac{1}{2}\|u - v\|^2 \ .$$

This definition depends on the norm used. A function F can be strongly convex with respect to one norm but not another. Note that strong convexity (with respect to any norm) implies strict convexity. Convexity means that the function $F(u)$ can be lower bounded a linear function $F(v)+\langle \nabla F(v), u-v\rangle$. In contrast, strong convexity means that the function $F(u)$ can lower bounded a quadratic function $F(v) + \langle \nabla F(v), u-v\rangle + \frac{1}{2}\|u-v\|^2$. The coefficient $\frac{1}{2}$ in front $\|u-v\|^2$ is an arbitrary choice; it was chosen because of mathematical convenience.

Let $\|\cdot\|$ be any norm on $\mathbb{R}^d$. Its *dual norm*, denoted by $\|\cdot\|_*$, is defined by

$$\left(\frac{1}{2}\|u\|_*^2\right) = \sup_{v\in\mathbb{R}^d}\left(\langle u, v\rangle - \frac{1}{2}\|v\|^2\right) . \tag{8.1}$$

Lemma 8.8 (Dual norm)**.** *Let $\|\cdot\|$ be a norm over $\mathbb{R}^d$. Then,*

1. *The dual norm $\|\cdot\|_*$ is a norm.*
2. *Hölder's inequality: $\langle u, v\rangle \le \|u\|\cdot\|v\|_*$ for any $u, v \in \mathbb{R}^d$.*
3. *Dual norm of $\|\cdot\|_*$ is the original norm $\|\cdot\|$.*

Example 8.9 (p-norm)**.** *For any $1 \le p < \infty$ the p-norm on $\mathbb{R}^d$ is defined as*

$$\|x\|_p = \left(\sum_{i=1}^d |x_i|^p\right)^{1/p} \qquad \text{and} \qquad \|x\|_\infty = \lim_{p\to\infty}\|x\|_p = \max\{|x_1|, |x_2|, \ldots, |x_d|\} .$$

The dual norm of $\|\cdot\|_p$ is $\|\cdot\|_q$ where $1 \le p, q \le \infty$ and $\frac{1}{p}+\frac{1}{q} = 1$.

8.3 Analysis of FTRL

In what follows we assume that $K \subset \mathbb{R}^d$ is convex and closed, $A \subseteq \mathbb{R}^d$, $R : A \to \mathbb{R}$ is Legendre, $K \cap A \neq \emptyset$ and the loss functions are linear: $\ell_t(w) = \langle f_t, w\rangle$. We denote by $w_{t+1} = \operatorname{argmin}_{w\in K\cap A}[\eta L_t(w) + R(w)]$ be point chosen by FTRL algorithm in round $t+1$.

Let $\widehat{L}_n = \sum_{t=1}^n \ell_t(w_t)$ be the total loss of FTRL algorithm. Let $\widehat{L}_n^+ = \sum_{t=1}^n \ell_t(w_{t+1})$ be the loss of the "cheating" algorithm that peeks one step ahead.

Lemma 8.10 (Regret of cheating algorithm)**.** *For any $u \in K \cap A$,*

$$\widehat{L}_n^+ - L_n(u) \le \frac{R(u) - R(w_1)}{\eta} .$$

Proof. We prove the inequality by induction on the number of steps n. For $n = 0$ the inequality is equivalent to $R(w_1) \le R(u)$, which follows from that w_1 is the minimizer of R.

Assume that the inequality holds for $n-1$. That is, assume that for any $u \in K \cap A$

$$\widehat{L}_{n-1}^+ - L_{n-1}(u) \le \frac{R(u) - R(w_1)}{\eta} .$$

Substituting $u = w_{n+1}$ we get

$$\widehat{L}_n^+ - L_n(w_{n+1}) = \widehat{L}_{n-1}^+ - L_{n-1}(w_{n+1}) \leq \frac{R(w_{n+1}) - R(w_1)}{\eta} .$$

This is equivalent to

$$\eta\widehat{L}_n^+ \leq \eta L_n(w_{n+1}) + R(w_{n+1}) - R(w_1) .$$

The right hand side can be upper bounded by $\eta L_n(u) + R(u) - R(w_1)$ because w_{n+1} is the minimizer of $\eta L_n(u) + R(u)$. Thus we get

$$\eta\widehat{L}_n^+ \leq \eta L_n(u) + R(u) - R(w_1)$$

which is equivalent to the statement of the lemma. □

Lemma 8.11 (Follow-The-Leader-Be-The-Leader Inequality)**.** $\ell_t(w_{t+1}) \leq \ell_t(w_t)$.

Summing the Follow-The-Leader-Be-The-Leader inequality for all $t = 1, 2, \ldots, n$ we get that $\widehat{L}_n^+ \leq L_n$. Putting it together with Lemma 8.10 we have:

Corollary 8.12 (Generic FTRL Regret Bound)**.** *For any $u \in K \cap A$,*

$$\widehat{L}_n - L_n(u) \leq \widehat{L}_n - \widehat{L}_n^+ + \frac{R(u) - R(w_1)}{\eta} .$$

The upper bound consists of two terms: $\widehat{L}_n - \widehat{L}_n^+ = \sum_{t=1}^n [\ell_t(w_t) - \ell_t(w_{t+1})]$ and $(R(u) - R(w_1))/\eta$. The first term captures how fast is w_t changing and the second is a penalty that we pay for using too much regularization. There is a trade-off: If we use too much regularization (i.e. R is too big and/or η is too small) then the second term is too big. On other hand, if we use too few regularization (i.e. R is too small and/or η is too large) then w_{t+1} will too far from w_t and the first term will be big. The goal is find a balance between these two opposing forces and find the right amount of regularization.

The following lemma states that the FTRL solution w_{t+1} can be obtained by first finding the unconstrained minimum of $\eta L_t(u) + R(u)$ over A and then projecting it to K. (Recall that $L_t(u)$ is the sum of the loss functions up time t.)

Lemma 8.13 (FTRL projection lemma)**.** *Let*

$$\widetilde{w}_{t+1} = \operatorname*{argmin}_{u \in A} \ [\eta L_t(u) + R(u)]$$

be solution to the unconstrained problem. Then, $w_{t+1} = \Pi_{R,K}(\widetilde{w}_{t+1})$.

Proof. By the ordinary projection lemma (Lemma 8.6),

$$w_{t+1} = \Pi_{L_t^\eta, K}(\widetilde{w}_{t+1})$$

where $L_t^\eta(u) = \eta L_t(u) + R(u)$. It is straightforward to verify that $D_R(u, v) = D_{L_t^\eta}(u, v)$ for any u, v. Therefore $\Pi_{L_t^\eta, K}(u) = \Pi_{R,K}(u)$ for any u. □

Theorem 8.14 (FTRL Regret Bound for Strongly Convex Regularizer)**.** *Let $R : A \to \mathbb{R}$ be a Legendre function which is strongly convex with respect a norm $\|\cdot\|$. Let $\ell_t(u) = \langle u, f_t\rangle$. Then, for all $u \in K \cap A$*

$$\widehat{L}_n - L_n(u) \le \eta \sum_{t=1}^{n} \|f_t\|_*^2 + \frac{R(u) - R(w_1)}{\eta} .$$

In particular, if $\|f_t\|_* \le 1$ for all $1 \le t \le n$ and $\eta = \sqrt{\frac{R(u)-R(w_1)}{n}}$ then

$$\widehat{L}_n - L_n(u) \le \sqrt{n(R(u) - R(w_1))} .$$

Proof. Beginning from Corollary 8.12 we see that it is enough to bound $\widehat{L}_n - \widehat{L}_n^+$. We can upper bound the difference as follows:

$$\begin{aligned}
\widehat{L}_n - \widehat{L}_n^+ &= \sum_{t=1}^{n} \ell_t(w_t) - \ell_t(w_{t+1}) \\
&= \sum_{t=1}^{n} \langle f_t, w_t - w_{t+1}\rangle \\
&\le \sum_{t=1}^{n} \|f_t\|_* \cdot \|w_t - w_{t+1}\| \qquad \text{(Hölder's inequality)}
\end{aligned}$$

It remains to upper bound $\|w_t - w_{t+1}\|$ by $\eta\|f_t\|_*$.

Strong convexity of R implies:

$$\begin{aligned}
R(w_t) - R(w_{t+1}) &\ge \langle \nabla R(w_{t+1}), w_t - w_{t+1}\rangle + \frac{1}{2}\|w_t - w_{t+1}\|^2 \\
R(w_{t+1}) - R(w_t) &\ge \langle \nabla R(w_t), w_{t+1} - w_t\rangle + \frac{1}{2}\|w_t - w_{t+1}\|^2
\end{aligned}$$

Summing these two inequalities gives

$$\|w_t - w_{t+1}\|^2 \le \langle \nabla R(w_t) - \nabla R(w_{t+1}), w_t - w_{t+1}\rangle . \tag{8.2}$$

By projection FTRL lemma (Lemma 8.13), $w_t = \Pi_{R,K}(\widetilde{w}_t)$ and $w_{t+1} = \Pi_{R,K}(\widetilde{w}_{t+1})$. We can thus apply Kolmogorov's inequality on the right-hand side of (8.2):

$$\begin{aligned}
\|w_t - w_{t+1}\|^2 &\le \langle \nabla R(w_t) - \nabla R(w_{t+1}), w_t - w_{t+1}\rangle \\
&\le \langle \nabla R(\widetilde{w}_t) - \nabla R(\widetilde{w}_{t+1}),\ w_t - w_{t+1}\rangle \qquad \text{(Kolmogorov's inequality)} \\
&\le \|w_t - w_{t+1}\| \cdot \|\nabla R(\widetilde{w}_t) - \nabla R(\widetilde{w}_{t+1})\|_* \qquad \text{(Hölder's inequality)}
\end{aligned}$$

Dividing by non-negative $\|w_t - w_{t+1}\|$ on both sides, we obtain

$$\|w_t - w_{t+1}\| \le \|\nabla R(\widetilde{w}_t) - \nabla R(\widetilde{w}_{t+1})\|_* .$$

We finish the proof by showing that $\nabla R(\widetilde{w}_t) - \nabla R(\widetilde{w}_{t+1}) = \eta f_t$. In order see that, note that $\widetilde{w}_t, \widetilde{w}_t$ are the unconstrained minimizers of L^η_{t-1}, L^η_t respectively and therefore they satisfy that $\nabla L^\eta_{t-1}(\widetilde{w}_t) = 0$ and $\nabla L^\eta_t(\widetilde{w}_{t+1}) = 0$. This is equivalent to

$$\nabla_u \left(\eta \sum_{s=1}^{t-1} \langle f_s, u \rangle + R(u) \right) \Bigg|_{u=\widetilde{w}_t} = 0 \quad \text{and} \quad \nabla_u \left(\eta \sum_{s=1}^{t} \langle f_s, u \rangle + R(u) \right) \Bigg|_{u=\widetilde{w}_{t+1}} = 0 \, .$$

Calculating the gradients explicitly, this gives

$$\eta \sum_{s=1}^{t-1} f_s + \nabla R(\widetilde{w}_t) = 0 \qquad \text{and} \qquad \eta \sum_{s=1}^{t} f_s + \nabla R(\widetilde{w}_{t+1}) = 0 \, .$$

Subtracting the two equations gives $\nabla R(\widetilde{w}_t) - \nabla R(\widetilde{w}_{t+1}) = \eta f_t$ as advertised. □

8.4 Exercises

Exercise 8.1. (EWA as FTRL) The goal of this exercise is to show that Exponentially Weighted Average (EWA) forecaster is, in fact, the Follow The Regularized Leader (FTRL) algorithm with the "un-normalized" negative entropy regularizer.

Assume that $\ell_1, \ell_2, \ldots, \ell_{t-1}$ are vectors in $\mathbb{R}^N$. We denote their components by $\ell_{s,i}$ where $1 \le s \le t-1$ and $1 \le i \le N$. (You can think of $\ell_{s,i}$ as the loss of expert i in round s.) Recall that in round t, EWA chooses the probability vector $p_t = (p_{t,1}, p_{t,2}, \ldots, p_{t,N})$, where the coordinates are

$$p_{t,i} = \frac{w_{t,i}}{\sum_{i=1}^N w_{t,i}} \qquad (i = 1, 2, \ldots, N) \qquad \text{where} \qquad w_{t,i} = e^{-\eta \sum_{s=1}^{t-1} \ell_{s,i}}$$

and $\eta > 0$ is the learning rate. On the other hand, FTRL chooses the probability vector p'_t defined as

$$p'_t = \operatorname*{argmin}_{p \in \Delta_N} \left(\eta \sum_{s=1}^{t-1} \langle \ell_s, p \rangle + R(p) \right) \, ,$$

where $\Delta_N = \{p \in \mathbb{R}^N \ : \ \sum_{i=1}^N p_i = 1, \ \forall 1 \le i \le N, \ p_i \ge 0\}$. Your main goal will be to show that if the regularizer is the *un-normalized negative entropy*

$$R(p) = \sum_{i=1}^N p_i \ln(p_i) - p_i$$

then $p_t = p'_t$ (provided, of course, that both EWA and FTRL use the same η). We ask you to do it in several steps.

(a) Prove that Δ_N is a convex set.

(b) Prove that the function $R(p)$ defined on the open positive orthant

$$\mathbb{R}^N_{++} = \{p \in \mathbb{R}^N \ : \ \forall 1 \le i \le N, \ p_i > 0\}$$

is a Legendre function. (Don't forget to show that the domain is convex).

(c) Show that $R(p)$ is **not** strongly convex on the positive orthant with respect to any norm. (Hint: First, prove the statement for a fixed norm e.g. 1-norm. Then use that on $\mathbb{R}^N$ any two norms are equivalent: For any norms $\|\cdot\|_\heartsuit, \|\cdot\|_\spadesuit$ on $\mathbb{R}^N$ there exists $\alpha > 0$ such that for any $x \in \mathbb{R}^N$, $\alpha\|x\|_\spadesuit \le \|x\|_\heartsuit$. You don't need to prove this.)

(d) Prove that $R(p)$ is strongly convex with respect to $\|\cdot\|_1$ on the open probability simplex

$$\Delta'_N = \left\{ p \in \mathbb{R}^N \ : \ p_i > 0, \ \sum_{i=1}^N p_i = 1 \right\} .$$

(Hint: Use Pinsker's inequality (between Kullback-Leibler divergence and variational distance) that states that

$$\|p - q\|_1 \le \sqrt{2 \sum_{i=1}^N p_i \ln \frac{p_i}{q_i}}$$

for any $p, q \in \Delta_N$.)

(e) Show that the Bregman projection $\Pi_{R,\Delta_N} : \mathbb{R}^N_{++} \to \Delta_N$ induced by the negative entropy can be calculated in $O(N)$ time. That is, find an algorithm that, given an input $x \in \mathbb{R}^N_{++}$, outputs $x' = \Pi_{R,\Delta_N}(x)$ using at most $O(N)$ arithmetic operations. (Hint: Find an explicit formula for the projection!)

(f) Find the unconstrained minimum

$$\widetilde{w}_t = \operatorname*{argmin}_{p \in \mathbb{R}^N_{++}} \left(\eta \sum_{s=1}^{t-1} \langle \ell_s, p \rangle + R(p) \right)$$

and show that $\widetilde{w}_t = w_t$, where $w_t = (w_{t,1}, \ldots, w_{t,N})$ is the EWA weight vector. (Hint: Set the gradient of the objective function to zero and solve the equation.)

(g) Combining e and f, find the constrained minimum p'_t and prove that $p'_t = p_t$.

(h) Show that $R(p) \le 0$ for any $p \in \Delta'_N$

(i) Assuming that $\ell_1, \ell_2, \ldots, \ell_n \in [0,1]^N$, re-prove the $O(\sqrt{n \log N})$ regret bound of EWA, using the general upper bound for FTRL with strongly convex regularizer:

$$\widehat{L}_n - L_n(p) \le \eta \sum_{t=1}^n \|\ell_t\|_*^2 \ + \ \frac{R(p) - R(p_1)}{\eta} \qquad \forall \ p \in \Delta'_N \ .$$

Hints: By checking the proof, it is not hard to see that the general upper bound for FTRL continues to hold even when strong convexity for R holds only over $K \cap A$. Identify K and A and argue (based on what you proved above) that this is indeed the case here. Then, use Part (h) to upper bound $R(p)$. To deal with $R(p_1)$, figure out what p_1 is and substitute! Optimize η. You can use that the dual norm of $\|\cdot\|_1$ is $\|\cdot\|_\infty$.

Exercise 8.2. (Quadratic Regularizer) Assume an online algorithm is choosing points in the unit ball $B_N = \{x \in \mathbb{R}^N \ : \ \|x\|_2 \leq 1\}$. If the algorithm chooses loss x_t in round t it suffers loss $\langle \ell_t, x_t \rangle$ where ℓ_t is a loss vector chosen by the adversary. In other words, the loss functions are linear.

Consider FTRL with the quadratic regularizer $R(x) = \frac{1}{2}\|x\|_2^2$ where $\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_N^2}$ is the 2-norm. That is, in round t the algorithm chooses

$$x_t = \operatorname*{argmin}_{x \in B_N} \left(\eta \sum_{s=1}^{t-1} \langle \ell_s, x \rangle + R(x) \right) .$$

(a) Show that the unit ball, B_N, is a convex set. (Hint: Use triangle inequality for the 2-norm.)

(b) Show that $R(x) = \frac{1}{2}\|x\|_2^2$ defined on all of $\mathbb{R}^N$ is a Legendre function. (Among other things, don't forget to prove that the gradients grow at the "boundary". That is, don't forget to show that $\|\nabla R(x)\| \to \infty$ as $\|x\| \to \infty$.)

(c) Show that $R(x)$ is strongly convex with respect to the 2-norm.

(d) Show that the Bregman projection $\Pi_{R,B_N} : \mathbb{R}^N \to B_N$ induced by the quadratic regularizer can be calculated in $O(N)$ time. That is, find an algorithm that, given an input $x \in \mathbb{R}^N_{++}$, outputs $x' = \Pi_{R,\Delta_N}(x)$ using at most $O(N)$ arithmetic operations. Find an explicit formula.

(e) Find an explicit formula for the unconstrained minimum

$$\widetilde{x}_t = \operatorname*{argmin}_{x \in \mathbb{R}_N} \left(\eta \sum_{s=1}^{t-1} \langle \ell_s, x \rangle + R(x) \right) .$$

(Hint: Calculate the gradient of the objective, set it to zero and solve.)

(f) Find the constrained minimum x_t using parts (d) and (e).

(g) Design and describe (both in pseudo-code and in English) the FTRL algorithm for this setting that has $O(N)$ memory complexity and $O(N)$ time complexity per step.

(h) Assuming that $\ell_1, \ell_2, \ldots, \ell_n \in B_N$, prove an $O(\sqrt{n})$ regret on the resulting algorithm. Use the general upper bound, proved in class, for FTRL with strongly convex regularizer:

$$\widehat{L}_n - L(u) \leq \eta \sum_{t=1}^{n} \|\ell_t\|_*^2 \; + \; \frac{R(u) - R(x_1)}{\eta} \qquad\qquad \forall\; u \in B_N \; .$$

Upper bound $R(u)$. To deal with $R(x_1)$, figure out what x_1 is and substitute! Optimize η. (Hint: The dual norm of $\|\cdot\|_2$ is $\|\cdot\|_2$.) What is the dependence of your regret bound on N? Compare this $O(\sqrt{n \log N})$ from Q1 part (i).

(For those initiated: Does the algorithm from Part (g) remind you of the classical (online) gradient descent algorithm?)

Exercise 8.3. (Further problems)

(a) Prove the follow-the-leader/be-the-leader inequality. Let D be any decision set, $R, \ell_1, \ldots, \ell_t$ be any real-valued functions defined on D. (Note that here we don't have extra assumptions, no need for convexity or anything like that.) Let $\eta > 0$. Let w_{t+1} be the "Be-The-Leader" solution (i.e. the solution of the cheating algorithm):

$$w_{t+1} = \operatorname*{argmin}_{w \in D} \left(\eta \sum_{s=1}^{t} \ell_s(w) + R(w) \right) \; .$$

Let w_t be the "Follow-The-(Regularized)-Leader" solution (i.e. the solution of the FTRL algorithm)

$$w_t = \operatorname*{argmin}_{w \in D} \left(\eta \sum_{s=1}^{t-1} \ell_s(w) + R(w) \right) \; .$$

Prove the "Follow-The-Leader/Be-The-Leader" inequality (i.e. that the cheating algorithm has lower instantaneous loss in round t):

$$\ell_t(w_{t+1}) \leq \ell_t(w_t) \; .$$

(b) Prove Kolmogorov's inequality. (Hint: Use the Pythagorean inequality.)

(c) Show that the dual norm of the 2-norm is the 2-norm.

(d) Show that the dual norm of the 1-norm is the ∞-norm, and vice versa. Recall that the 1-norm is defined as $\|x\|_1 = \sum_{i=1}^{N} |x_i|$ and the ∞-norm is defined as $\|x\|_\infty = \max\{|x_1|, |x_2|, \ldots, |x_n|\}$.

Chapter 9

Proximal Point Algorithm

In this lecture, we present a different algorithm for the same online learning scenario as in last lecture. The algorithm we present is called Proximal Point Algorithm and it goes back to Martinet (1978) and Rockafellar (1977) in the context of classical (offline) numerical convex optimization. In the context of online learning, a special case of the algorithm with quadratic regularizer was rediscovered by Zinkevich (2003).

Recall the online learning scenario from the last chapter, where the learner in round t chooses a point w_t from a closed convex set $K \subseteq \mathbb{R}^d$, and it suffers a loss $\ell_t(w_t)$ where $\ell_t : \mathbb{R}^d \to \mathbb{R}$ is linear function chosen by the adversary i.e. $\ell_t(w_t) = \langle f_t, w \rangle$.

In the last chapter, we saw that the regret of FTRL is controlled by how much w_{t+1} differs from w_t and the amount of regularization used. The Proximal Point Algorithm is designed with the explicit intention to keep w_{t+1} close to w_t. This also explains the name of the algorithm.

Suppose $R : A \to \mathbb{R}$ is Legendre function, which we can think of as regularizer. The algorithm first calculates $\widetilde{w}_{t+1} \in A$ from w_t and then $\widetilde{w}_{t+1}$ is projected to K by a Bregman projection:

$$\begin{aligned} \widetilde{w}_{t+1} &= \operatorname*{argmin}_{w \in A} \ [\eta \ell_t(w) + D_R(w, w_t)] \\ w_{t+1} &= \Pi_{R,K}\left(\widetilde{w}_{t+1}\right) \end{aligned}$$

We see that $\widetilde{w}_{t+1}$ is constructed so that it minimizes the current loss penalized so that $\widetilde{w}_{t+1}$ stays close to w_t. The algorithm starts with $\widetilde{w}_1 = \operatorname{argmin}_{w \in A} R(w)$ and $w_1 = \Pi_{R,K}(\widetilde{w}_1)$, the same as FTRL.

9.1 Analysis

Proposition 9.1.

$$\nabla R(\widetilde{w}_{t+1}) - \nabla R(w_t) = -\eta f_t$$

Proof. We have $\nabla_w \left[\eta \ell_t(u) + D_R(w, w_t)\right]|_{w=\widetilde{w}_{t+1}} = 0$ since $\widetilde{w}_{t+1}$ is an unconstrained minimizer of $\eta \ell_t(w) + D_R(w, w_t)$. We rewrite the condition by calculating the gradient:

$$\begin{aligned}
0 &= \nabla_w \left[\eta \ell_t(u) + D_R(w, w_t)\right] \Big|_{w=\widetilde{w}_{t+1}} \\
&= \eta \nabla \ell_t(\widetilde{w}_{t+1}) + \nabla_w D_R(w, w_t)\Big|_{w=\widetilde{w}_{t+1}} \\
&= \eta f_t + \nabla_w \left[R(w) - R(w_t) - \langle \nabla R(w_t), w - w_t\rangle\right] \Big|_{w=\widetilde{w}_{t+1}} \\
&= \eta f_t + \nabla R(\widetilde{w}_{t+1}) - \nabla R(w_t)
\end{aligned}$$

which is exactly what needed to be proven. □

We denote by $\widehat{L}_n = \sum_{t=1}^n \ell_t(w_t)$ the total loss of the algorithm after n rounds. Similarly, let $L_n(u) = \sum_{t=1}^n \ell_t(u)$ be the sum of the loss functions.

Lemma 9.2. *For any $u \in K \cap A$ and any $\eta > 0$,*

$$\begin{aligned}
\widehat{L}_n - L_n(u) &\leq \frac{1}{\eta}\left[D_R(u, w_1) + \sum_{t=1}^n D_R(w_t, \widetilde{w}_{t+1})\right] \\
&\leq \sum_{t=1}^n \left(\ell_t(w_t) - \ell_t(\widetilde{w}_{t+1})\right) + \frac{D_R(u, w_1)}{\eta}\,.
\end{aligned}$$

Proof. The sum $\widehat{L}_n - L_n(u)$ can be written as a sum $\sum_{t=1}^n \left(\ell_t(w_t) - \ell_t(u)\right)$. We upper bound each term in the sum separately:

$$\begin{aligned}
\ell_t(w_t) - \ell_t(u) &= \langle w_t - u, f_t\rangle \\
&= \frac{1}{\eta}\langle w_t - u, \nabla R(w_t) - \nabla R(\widetilde{w}_{t+1})\rangle \qquad \text{(Proposition 9.1)} \\
&= \frac{1}{\eta}\left[D_R(u, w_t) - D_R(u, \widetilde{w}_{t+1}) + D_R(w_t, \widetilde{w}_{t+1})\right] \quad \text{(long, but straightforward calculation)} \\
&\leq \frac{1}{\eta}\left[D_R(u, w_t) - D_R(u, w_{t+1}) - D_R(w_{t+1}, \widetilde{w}_{t+1}) + D_R(w_t, \widetilde{w}_{t+1})\right] \\
&\qquad \text{(Pythagorean inequality)} \\
&\leq \frac{1}{\eta}\left[D_R(u, w_t) - D_R(u, w_{t+1}) + D_R(w_t, \widetilde{w}_{t+1})\right] \qquad (D_R \geq 0)
\end{aligned}$$

Adding the inequalities for all $t = 1, 2, \ldots, n$, the terms $D_R(u, w_t) - D_R(u, \widetilde{w}_{t+1})$ telescope:

$$\widehat{L}_n - L_n(u) \leq \frac{1}{\eta}\left[D_R(u, w_1) - D_R(u, w_{n+1}) + \sum_{t=1}^n D_R(w_t, \widetilde{w}_{t+1})\right]$$

We can drop $(-D_R(u, w_{n+1}))$ since Bregman divergence is always non-negative, and we get the first inequality of the lemma. For the second inequality, we upper bound $\sum_{t=1}^n D_R(w_t, \widetilde{w}_{t+1})$

term-by-term:

$$\begin{aligned} D_R(w_t, \widetilde{w}_{t+1}) &\le D_R(w_t, \widetilde{w}_{t+1}) + D_R(\widetilde{w}_{t+1}, w_t) \\ &= \langle \nabla R(w_t) - \nabla R(\widetilde{w}_{t+1}), w_t - \widetilde{w}_{t+1} \rangle \\ &= \eta \langle f_t, w_t - \widetilde{w}_{t+1} \rangle \qquad \text{(Proposition 9.1)} \\ &= \eta \left(\ell_t(w_t) - \ell_t(\widetilde{w}_{t+1}) \right) \ . \end{aligned}$$

□

Theorem 9.3 (Regret Bound for Proximal Point Algorithm with Strongly Convex Regularizer)**.** *Let $R : A \to \mathbb{R}$ is a Legendre function which is strongly convex with respect a norm $\| \cdot \|$. Then, for all $u \in K \cap A$,*

$$\widehat{L}_n - L_n(u) \le \eta \sum_{t=1}^{n} \|f_t\|_*^2 + \frac{R(u) - R(w_1)}{\eta} \ .$$

In particular, if $\|f_t\|_* \le 1$ for all $1 \le t \le n$ and $\eta = \sqrt{\frac{R(u)-R(w_1)}{n}}$ then for any $u \in K \cap A$,

$$\widehat{L}_n - L_n(u) \le \sqrt{n(R(u) - R(w_1))} \ .$$

Proof. By Lemma 9.2 we need to prove only that

$$\sum_{t=1}^{n} \left(\ell_t(w_t) - \ell_t(\widetilde{w}_{t+1}) \right) + \frac{D_R(u, w_1)}{\eta} \le \eta \sum_{t=1}^{n} \|f_t\|_*^2 + \frac{R(u) - R(w_1)}{\eta} \ .$$

First, note that $D_R(u, w_1) \le R(u) - R(w_1)$ since for any $u \in K \cap A$, $\langle \nabla R(w_1), u - w_1 \rangle \ge 0$ by optimality of w_1. Hence, it remains to show that

$$\sum_{t=1}^{n} \left(\ell_t(w_t) - \ell_t(\widetilde{w}_{t+1}) \right) \le \eta \sum_{t=1}^{n} \|f_t\|_*^2 \ .$$

We will prove that $\ell_t(w_t) - \ell_t(\widetilde{w}_{t+1}) \le \eta \|f_t\|_*^2$. We have

$$\ell_t(w_t) - \ell_t(\widetilde{w}_{t+1}) = \langle f_t, w_t - \widetilde{w}_{t+1} \rangle \le \|f_t\|_* \cdot \|w_t - \widetilde{w}_{t+1}\| \ .$$

Thus, it remains to show that $\|w_t - \widetilde{w}_{t+1}\| \le \eta \|f_t\|_*$.

Strong convexity of R implies:

$$\begin{aligned} R(w_t) - R(\widetilde{w}_{t+1}) &\ge \langle \nabla R(\widetilde{w}_{t+1}), w_t - \widetilde{w}_{t+1} \rangle + \frac{1}{2} \|w_t - \widetilde{w}_{t+1}\|^2 \\ R(\widetilde{w}_{t+1}) - R(w_t) &\ge \langle \nabla R(w_t), \widetilde{w}_{t+1} - w_t \rangle + \frac{1}{2} \|w_t - \widetilde{w}_{t+1}\|^2 \end{aligned}$$

Summing these two inequalities gives

$$\|w_t - \widetilde{w}_{t+1}\|^2 \le \langle \nabla R(w_t) - \nabla R(\widetilde{w}_{t+1}), w_t - \widetilde{w}_{t+1} \rangle \ .$$

By Hölder's inequality,

$$\|w_t - \widetilde{w}_{t+1}\|^2 \leq \|\nabla R(w_t) - \nabla R(\widetilde{w}_{t+1})\|_* \cdot \|w_t - \widetilde{w}_{t+1}\| .$$

Dividing both sides by non-negative $\|w_t - \widetilde{w}_{t+1}\|$ we get

$$\|w_t - \widetilde{w}_{t+1}\| \leq \|\nabla R(w_t) - \nabla R(\widetilde{w}_{t+1})\|_* .$$

Finally, by Proposition 9.1, we have $\|\nabla R(w_t) - \nabla R(\widetilde{w}_{t+1})\|_* = \eta\|f_t\|_*$. □

9.2 Time-Varying Learning Rate

We now slightly extend the algorithm by allowing learning rate η to vary in time. In other words, in each time step we use (possibly different) learning rate $\eta_t > 0$. The modified algorithm is called Proximal Point Algorithm with Time-Varying Learning Rate. Later we will see how to choose the sequence $\{\eta_t\}_{t=1}^{\infty}$ so that we achieve a low learning rate. The initialization of algorithm is the same as before $\widetilde{w}_1 = \operatorname{argmin}_{w\in A} R(w)$ and $w_1 = \operatorname{argmin}_{w\in K\cap A} R(w) = \Pi_{R,K}(\widetilde{w}_1)$. The update rule becomes

$$\begin{aligned}\widetilde{w}_{t+1} &= \operatorname*{argmin}_{w\in A} \left[\eta_t \ell_t(w) + D_R(w, w_t)\right] \\ w_{t+1} &= \Pi_{R,K}\left(\widetilde{w}_{t+1}\right)\end{aligned}$$

We still assume that the loss functions are linear i.e. $\ell_t(w) = \langle f_t, w\rangle$ and therefore $\nabla \ell_t(w) = f_t$ for any $w \in \mathbb{R}^d$.

Proposition 9.4. *The sequences $\{\widetilde{w}_t\}_{t=1}^{\infty}$ and $\{w_t\}_{t=1}^{\infty}$ generated by* Proximal Point Algorithm with Time-Varying Learning Rate *satisfy*

$$\nabla R(\widetilde{w}_{t+1}) - \nabla R(w_t) = -\eta_t f_t .$$

Proof. The proof is the same as the proof of Proposition 9.1. We just replace η by η_t. □

Lemma 9.5. *For all $u \in K \cap A$ and any $\eta_1, \eta_2, \ldots, \eta_n > 0$,*

$$\ell_t(w_t) - \ell_t(u) \leq \frac{D_R(u, w_t) - D_R(u, w_{t+1}) + D_R(w_t, \widetilde{w}_{t+1})}{\eta_t} .$$

Proof. By linearity,

$$\ell_t(w_t) - \ell_t(u) = \langle w_t - u, f_t\rangle .$$

By Proposition 9.4,

$$\begin{aligned}\eta_t\langle w_t - u, f_t\rangle &= \langle w_t - u, \nabla R(w_t) - \nabla R(\widetilde{w}_{t+1})\rangle \\ &= D_R(u, w_t) - D_R(u, \widetilde{w}_{t+1}) + D_R(w_t, \widetilde{w}_{t+1}) \quad \text{(long, but straightforward calculation)} \\ &\leq D_R(u, w_t) - D_R(u, w_{t+1}) - D_R(w_{t+1}, \widetilde{w}_{t+1}) + D_R(w_t, \widetilde{w}_{t+1}) \quad \text{(Pythagorean inequality)} \\ &\leq D_R(u, w_t) - D_R(u, w_{t+1}) + D_R(w_t, \widetilde{w}_{t+1}) \quad (D_R \geq 0)\end{aligned}$$

Dividing through by $\eta_t > 0$ finishes the proof of the lemma. □

Summing the lemma over all $t = 1, 2, \ldots, n$ we get the following corollary.

Corollary 9.6. *For all $u \in K \cap A$ and any $\eta_1, \eta_2, \ldots, \eta_n > 0$,*

$$\widehat{L}_n - L_n(u) \leq \sum_{t=1}^{n} \frac{1}{\eta_t} \left(D_R(u, w_t) - D_R(u, w_{t+1}) \right) + \sum_{t=1}^{n} \frac{D_R(w_t, \widetilde{w}_{t+1})}{\eta_t} \,.$$

9.3 Linearized Proximal Point Algorithm

We now consider the situation when the loss functions $\{\ell_t\}_{t=1}^{\infty}$ are not necessarily linear. We assume that ℓ_t are convex and differentiable defined on K. The initialization of $\widetilde{w}_1$ and w_1 remains the same as before. The update uses linearized losses:

$$\begin{aligned} \widetilde{w}_{t+1} &= \operatorname*{argmin}_{w \in A} \left[\eta_t \widetilde{\ell}_t(w) + D_R(w, w_t) \right] \\ w_{t+1} &= \Pi_{R,K}(\widetilde{w}_{t+1}) \end{aligned}$$

where

$$\widetilde{\ell}_t(w) = \ell_t(w_t) + \langle \nabla \ell_t(w_t), w - w_t \rangle$$

is the *linearized loss*. The name comes from that $\widetilde{\ell}_t$ is a linear approximation of ℓ_t by its first order Taylor expansion. We call the resulting algorithm the Linearized Proximal Point Algorithm.

The crucial property that allows to extend regret bounds for non-linear losses is the following lemma. The lemma states that the instantaneous regret $\ell_t(w_t) - \ell_t(u)$ for any convex loss is upper bounded by the instantaneous regret $\widetilde{\ell}_t(w_t) - \widetilde{\ell}_t(u)$ for their linearization.

Lemma 9.7. *If $\ell_t : K \to \mathbb{R}$ is convex then for any $u \in K$,*

$$\begin{aligned} \ell_t(w_t) - \ell_t(u) &\leq \langle \nabla \ell_t(w_t), w_t - u \rangle && (9.1) \\ &= \widetilde{\ell}_t(w_t) - \widetilde{\ell}_t(u) && (9.2) \end{aligned}$$

Proof. The fist inequality from convexity of ℓ_t. The second equality follows by the definition of the linearized loss $\widetilde{\ell}_t$. □

Using this lemma it easy to extend the first part of Lemma 9.2 (with constant learning rate) and Corollary 9.6 (with varying learning rate) to non-linear convex losses. If gradients $\nabla \ell_t$ are bounded, one can even extend Theorem 9.3.

Proposition 9.8. *The sequences $\{\widetilde{w}_t\}_{t=1}^{\infty}$ and $\{w_t\}_{t=1}^{\infty}$ generated by the* Linearized Proximal Point Algorithm *satisfy*

$$\nabla R(\widetilde{w}_{t+1}) - \nabla R(w_t) = -\eta_t \nabla \ell_t(w_t)$$

Proof. The follows from Proposition 9.4 applied to linearized loss $\widetilde{\ell}_t$. □

Notice that the linearized loss $\widetilde{\ell}_t(w) = \ell_t(w_t) + \langle \nabla \ell_t(w_t), w - w_t \rangle$ is technically not a linear function. It is an *affine function*. That is, it is of the form $\widetilde{\ell}_t(w) = a + \langle b, w \rangle$ for some $a \in \mathbb{R}$ and $b \in \mathbb{R}^d$. It easy to see that the results for linear losses from previous sections extend without any change to affine losses. Also notice that the intercept a does not play any role in the algorithms nor the regret. Thus we can define the linearized loss as $\widetilde{\ell}_t(w) = \langle \nabla\, \ell_t(w_t), w \rangle$.

9.4 Strongly Convex Losses

We now analyze Linearized Proximal Point Algorithm for strongly convex loss functions.

Definition 9.9 (Strong Convexity)**.** Let $K \subseteq \mathbb{R}^d$ be convex set and $\sigma \geq 0$. A differentiable function $g : K \to \mathbb{R}$ is *σ-strongly convex with respect to Legendre function $R : K \to \mathbb{R}$* if

$$g(w) \geq g(y) + \langle \nabla g(y), w - y \rangle + \frac{\sigma}{2} D_R(w, y) \; .$$

Theorem 9.10 (Regret for Strongly Convex Losses)**.** *Let $\{\sigma_t\}_{t=1}^{\infty}$ be sequence of positive numbers. Assume that sequence of loss function $\{\ell_t\}_{t=1}^{\infty}$ is such that $\ell_t : K \to \mathbb{R}$ is a differentiable, σ_t-strongly convex function w.r.t R for any $t \geq 1$. Let the learning rate sequence be $\eta_t = \frac{2}{\sum_{s=1}^{t} \sigma_s}$. Then, for any $u \in K \cap A$,*

$$\widehat{L}_n - L_n(u) \leq \sum_{t=1}^{n} \frac{D_R(w_t, \widetilde{w}_{t+1})}{\eta_t} \; .$$

Proof. By strong convexity,

$$\ell_t(w_t) \leq \ell_t(u) - \langle \nabla\, \ell_t(w_t), u - w_t \rangle - \frac{\sigma_t}{2} D_R(u, w_t) \; .$$

By definition of the linearized losses $\langle \nabla\, \ell_t(w_t), u - w_t \rangle = \widetilde{\ell}_t(u) - \widetilde{\ell}_t(w_t)$ and thus

$$\ell_t(w_t) - \ell_t(u) \leq \widetilde{\ell}_t(w_t) - \widetilde{\ell}_t(u) - \frac{\sigma_t}{2} D_R(u, w_t) \; .$$

(Note that this inequality is a strengthening of Lemma 9.7 for strongly convex losses). We upper bound $\widetilde{\ell}_t(w_t) - \widetilde{\ell}_t(u)$ using Lemma 9.5:

$$\ell_t(w_t) - \ell_t(u) \leq \frac{D_R(u, w_t) - D_R(u, w_{t+1}) + D_R(w_t, \widetilde{w}_{t+1})}{\eta_t} - \frac{\sigma_t}{2} D_R(u, w_t) \; .$$

Summing over all $t = 1, 2, \ldots, n$ and

$$\widehat{L}_n - L_n(u) \leq \sum_{t=1}^{n} \frac{D_R(w_t, \widetilde{w}_{t+1})}{\eta_t} + \sum_{t=1}^{n} \left[\left(\frac{1}{\eta_t} - \frac{\sigma_t}{2} \right) D_R(u, w_t) - \frac{1}{\eta_t} D_R(u, w_{t+1}) \right]$$

It remains to show that the second sum is non-positive. We can rewrite it as

$$\sum_{t=1}^{n}\left[\left(\frac{1}{\eta_t}-\frac{\sigma_t}{2}\right)D_R(u,w_t)-\frac{1}{\eta_t}D_R(u,w_{t+1})\right]$$
$$=\left(\frac{1}{\eta_1}-\frac{\sigma_1}{2}\right)D_R(u,w_1)-\frac{1}{\eta_n}D_R(u,w_{n+1})+\sum_{t=1}^{n-1}\left(\frac{1}{\eta_{t+1}}-\frac{\sigma_{t+1}}{2}-\frac{1}{\eta_t}\right)D_R(u,w_{t+1})\ .$$

By the choice of learning rates $1/\eta_1 = \sigma_1/2$ and hence the first term vanishes. We can drop the second term, because D_R is non-negative. Each term in the sum is zero, since the learning rates satisfy the recurrence $\frac{1}{\eta_{t+1}} = \frac{1}{\eta_t} + \frac{\sigma_{t+1}}{2}$. □

We illustrate the use of Theorem on a special case. Assume that the regularizer is $R(w) = \frac{1}{2}\|w\|_2^2$, gradients are uniformly bounded $\|\nabla \ell_t(w)\|_2 \le G$ for all $w \in K$ and all t, and $\sigma_t = \sigma$. The algorithm becomes a projected gradient descent algorithm

$$\widetilde{w}_{t+1} = w_t - \eta_t \nabla \ell_t(w_t)$$
$$w_{t+1} = \Pi_{R,K}(\widetilde{w}_t)$$

The term $D_R(w_t, w_{t+1})$ can be expressed as

$$D_R(w_t,\widetilde{w}_{t+1}) = \frac{1}{2}\|w_t-\widetilde{w}_{t+1}\|_2^2 = \frac{1}{2}\eta_t^2\|\nabla\ell_t(w_t)\|_2^2\ .$$

Applying Theorem 9.10 we get a $O(\log n)$ regret bound:

$$\begin{aligned}
\widehat{L}_n - L_n(u) &\le \sum_{t=1}^{n}\frac{D_R(w_t,\widetilde{w}_{t+1})}{\eta_t}\\
&= \sum_{t=1}^{n}\frac{\eta_t\|\nabla\ell_t(w_t)\|_2^2}{2}\\
&= \sum_{t=1}^{n}\frac{\|\nabla\ell_t(w_t)\|_2^2}{\sum_{s=1}^{t}\sigma_s}\\
&= \sum_{t=1}^{n}\frac{\|\nabla\ell_t(w_t)\|_2^2}{t\sigma}\\
&\le \frac{G^2}{\sigma}\sum_{t=1}^{n}\frac{1}{t}\\
&\le \frac{G^2}{\sigma}(1+\ln n)\ .
\end{aligned}$$

9.5 Exercises

Exercise 9.1. We defined $\widetilde{w}_1 = \operatorname{argmin}_{w\in A} R(w)$ and $w_1 = \Pi_{R,K}(\widetilde{w}_1)$. Show that $w_1 = \operatorname{argmin}_{w\in K\cap A} R(w)$. In other words, show that w_1 is the same as in FTRL algorithm.

Exercise 9.2. (Zinkevich's algorithm) Let $K \subseteq \mathbb{R}^d$ be convex and closed containing the origin. Abusing notation, let $\Pi_K(w) = \operatorname{argmin}_{u \in K} \|u - w\|$ be the Euclidean projection to K. Zinkevich's algorithm starts with $w_1 = 0$ and updates

$$w_{t+1} = \Pi_K(w_t - \eta \nabla \ell_t(w_t)) \ .$$

Show that Zinkevich's algorithm is nothing else than Linearized Proximal Point Algorithm with quadratic regularizer $R(w) = \frac{1}{2}\|w\|_2^2$.

Exercise 9.3. (Comparison with FTRL) Consider FTRL algorithm and the Proximal Point Algorithm with the same convex closed set $K \subseteq \mathbb{R}^d$, Legendre regularizer $R : A \to \mathbb{R}^d$ and the same learning rate $\eta > 0$, running on the same sequence $\{\ell_t\}_{t=1}^{\infty}$ of linear loss functions.

(a) Show that if $K = A$ then, for any $R, \eta, \{\ell_t\}_{t=1}^{\infty}$ both algorithms produce the same sequence $\{w_t\}_{t=1}^{\infty}$ of solutions.

(b) Give an example of K, R, η and $\{\ell_t\}_{t=1}^{\infty}$ such that the algorithms produce different sequences of solutions.

(Hint: Unit ball and quadratic regularizer $R(w) = \frac{1}{2}\|w\|_2^2$.)

(c) Give an example of a bounded K and $\mathbb{R}$ such that for any $\eta, \{\ell_t\}_{t=1}^{\infty}$ the algorithms produce the same sequences of solutions $w_1, w_2, \ldots, w_{n+1}$. (Hint: Express the Exponentially Weighted Average forectaster both as FTRL and as a Proximal Point Algorithm.)

(d) Redo (a), (b) and (a) with linearized versions of the algorithms and arbitrary convex differentiable sequence of $\{\ell_t\}_{t=1}^{\infty}$ loss functions.

Chapter 10

Least Squares

Linear least squares method is the single most important regression problem in all of statistics and machine learning. In this chapter we consider the online version of the problem. We will assume that, in round t, the learner receives a vector $x_t \in \mathbb{R}^d$, predicts $\widehat{y}_t = \langle w_t, x_t \rangle$, receives feedback $y_t \in \mathbb{R}$ and suffers a loss $\frac{1}{2}(\widehat{y}_t - y_t)^2$. More compactly, the learner chooses $w_t \in \mathbb{R}^d$ and suffers loss $\ell_t(w_t)$ where $\ell_t : \mathbb{R}^d \to \mathbb{R}$ is a loss function defined as

$$\ell_t(w) = \frac{1}{2}(\langle w, x_t \rangle - y_t)^2 \ .$$

For dimension $d \geq 2$ the loss function ℓ_t is not strongly convex; not even strictly convex. To see that note consider any non-zero w that is perpendicular to x_t and note that $\ell_t(\alpha w) = 0$ for any $\alpha \in \mathbb{R}$. In other words, the loss function is flat (constant) along the line $\{\alpha w \ : \ \alpha \in \mathbb{R}\}$.

Our goal is to design an online algorithm for this problem with $O(\log n)$ regret under some natural assumptions on $\{(x_t, y_t)\}_{t=1}^{\infty}$; we state those assumptions later. We have seen $O(\log n)$ regret bounds for problems where ℓ_t were strongly convex. Despite that in our case the loss functions are not strongly convex, we show that Follow The Regularized Leader (FTRL) algorithm with quadratic regularizer $R(w) = \frac{1}{2}\|w\|_2^2$ has $O(\log n)$ regret.

Recall that in round $t+1$ FTRL algorithm chooses

$$w_{t+1} = \operatorname*{argmin}_{w \in \mathbb{R}^d} \left[\eta \sum_{s=1}^{t} \ell_s(w) + \frac{1}{2}\|w\|^2 \right]$$

where $\eta > 0$ is the learning rate. This minimization problem—both in online learning and classical off-line optimization—is called *regularized (linear) least squares problem* or sometimes *ridge regression.*

10.1 Analysis

We denote by $L_t(u)$ the sum of the first t loss functions, by $\widehat{L}_t$ the sum of the losses of the FTRL algorithm in the first t rounds, and by $L_t^{\eta}(u)$ the objective function that FTRL

minimizes in round $t+1$. Formally, for any $u \in R^d$ and any $1 \le t \le n$

$$L_t(u) = \sum_{s=1}^{t} \ell_s(u) \qquad \widehat{L}_t = \sum_{s=1}^{t} \ell_s(w_s) \qquad L_t^\eta(u) = \eta \sum_{s=1}^{t} \ell_s(u) + R(u) \; .$$

Observe that the minimization in ridge regression is unconstrained. We start with a general lemma concerning the behavior of unconstrained FTRL.

Lemma 10.1 (Unconstrained FTRL)**.** *Let $\eta > 0$, $A \subseteq \mathbb{R}^d$ and $R : A \to \mathbb{R}$ be a Legendre function. Suppose that the loss functions $\ell_1, \ell_2, \ldots, \ell_n : A \to \mathbb{R}$ are convex and differentiable. Consider the unconstrained FTRL algorithm that in round $t+1$ chooses*

$$w_{t+1} = \operatorname*{argmin}_{w \in A} \; L_t^\eta(w) \; .$$

Then, following equality holds

$$\eta \left(\widehat{L}_n - L_n(u) \right) = D_R(u, w_1) - D_{L_n^\eta}(u, w_{n+1}) + \sum_{t=1}^{n} D_{L_t^\eta}(w_t, w_{t+1}) \; .$$

Proof. Since the minimization is unconstrained $\nabla L_t^\eta(w_{t+1}) = 0$. Therefore, for any $u \in A$

$$\begin{aligned} D_{L_t^\eta}(u, w_{t+1}) &= L_t^\eta(u) - L_t^\eta(w_{t+1}) \\ &= L_{t-1}^\eta(u) + \eta \ell_t(u) - L_t^\eta(w_{t+1}) \; . \end{aligned} \tag{10.1}$$

Rearranging, for any $u \in A$

$$\eta \ell_t(u) = D_{L_t^\eta}(u, w_{t+1}) + L_t^\eta(w_{t+1}) - L_{t-1}^\eta(u) \tag{10.2}$$

Substituting $u = w_t$ we get

$$\eta \ell_t(w_t) = D_{L_t^\eta}(w_t, w_{t+1}) + L_t^\eta(w_{t+1}) - L_{t-1}^\eta(w_t) \tag{10.3}$$

Subtracting (10.2) from (10.3) we get

$$\begin{aligned} \eta \left(\ell_t(w_t) - \ell_t(u) \right) &= D_{L_t^\eta}(w_t, w_{t+1}) - D_{L_t^\eta}(u, w_{t+1}) + L_{t-1}(u) - L_{t-1}(w_t) \\ &= D_{L_t^\eta}(w_t, w_{t+1}) + D_{L_{t-1}^\eta}(u, w_t) - D_{L_t^\eta}(u, w_{t+1}) \end{aligned}$$

where in the last step we have used (10.1) with t shifted by one. If we sum both sides of the last equation over all $t = 1, 2, \ldots, n$, the differences $D_{L_{t-1}^\eta}(u, w_t) - D_{L_t^\eta}(u, w_{t+1})$ telescope. The observation that $L_0^\eta = R$ finishes the proof. □

Unsurprisingly, the lemma can be used to upper bound the regret:

$$\widehat{L}_n - L_n(u) \le \frac{1}{\eta} D_R(u, w_n) + \frac{1}{\eta} \sum_{t=1}^{n} D_{L_t^\eta}(w_t, w_{t+1}) \; . \tag{10.4}$$

The first divergence is easy to deal with: $D_R(u, w_t) = \frac{1}{2}\|u - w_1\|_2^2 = \frac{1}{2}\|u\|_2^2$. The main challenge to is to calculate the divergences $D_{L_t^\eta}(w_t, w_{t+1})$. We do it in the following lemma.

Lemma 10.2 (Ridge regression)**.** *Let $\{(x_t, y_t)\}_{t=1}^{\infty}$ be any sequence, $x_t \in \mathbb{R}^d$ and $y_t \in \mathbb{R}$. Let $\ell_t(w) = \frac{1}{2}\left(\langle w, x_t\rangle - y_t\right)^2$ be the corresponding sequence of loss functions. Consider the ridge regression FTRL algorithm $w_{t+1} = \operatorname{argmin}_{w \in A} \; L_t^\eta(w)$. Then,*

$$D_{L_t^\eta}(w_t, w_{t+1}) = \frac{\eta^2}{2}\ell_t(w_t)\left\langle x_t, A_t^{-1}x_t\right\rangle$$

where

$$A_t = I + \eta\sum_{s=1}^{t} x_s x_s^\top \; .$$

Proof. Define $M_t = x_t x_t^\top$ and $v_t = -y_t x_t$. Using this notation, we can write the loss function $\ell_t(u)$ as a quadratic form:

$$\begin{aligned}\ell_t(u) &= \frac{1}{2}\langle u, x_t x_t^\top u\rangle - \langle u, y_t x_t\rangle + \frac{1}{2}y_t^2 \\ &= \frac{1}{2}\langle u, M_t u\rangle + \langle u, v_t\rangle + \frac{1}{2}y_t^2\end{aligned}$$

In order to understand $D_{L_t^\eta}(w_t, w_{t+1})$ we first need to understand the underlying Legendre function $L_t^\eta(u)$. We can write rewrite by using the quadratic form for $\ell_t(u)$

$$\begin{aligned}L_t^\eta(u) &= \eta\sum_{s=1}^{t}\ell_s(u) + \frac{1}{2}\|u\|^2 \\ &= \eta\sum_{s=1}^{t} y_s^2 + \left\langle \eta\sum_{s=1}^{t} v_s,\; u\right\rangle + \frac{1}{2}\left\langle u,\; \left(I + \eta\sum_{s=1}^{t} M_s\right)u\right\rangle \\ &= C_t + \langle V_t, u\rangle + \frac{1}{2}\langle u, A_t u\rangle \; . \end{aligned} \tag{10.5}$$

where $C_t = \sum_{s=1}^{t} y_s^2$ and $V_t = \eta\sum_{s=1}^{t} v_s$. Using (10.5) we express the Bregman divergence $D_{L_t^\eta}(u, v)$ as

$$D_{L_t^\eta}(u, v) = \frac{1}{2}\langle u - v, A_t(u - v)\rangle \; . \tag{10.6}$$

From $\nabla L_t^\eta(w_{t+1}) = 0 = \nabla L_{t-1}^\eta(w_t)$ and (10.5) we derive the following equalities:

$$\begin{aligned} V_t + A_t w_{t+1} &= V_{t-1} + A_{t-1}w_t \\ V_t + A_t(w_{t+1} - w_t) + A_t w_t &= V_{t-1} + A_{t-1}w_t \\ A_t(w_{t+1} - w_t) &= (V_{t-1} - V_t) + (A_{t-1} - A_t)w_t \\ A_t(w_{t+1} - w_t) &= -\eta v_t - \eta M_t w_t \qquad (10.7) \\ w_{t+1} - w_t &= -\eta A_t^{-1}(M_t w_t + v_t) \qquad (10.8)\end{aligned}$$

The last piece that we need is that

$$M_t w_t + v_t = x_t(\langle x_t, w_t\rangle - y_t) \tag{10.9}$$

which follows by definition of M_t and v_t. Starting from we calculate (10.6) the Bregman divergence:

$$\begin{aligned}
D_{L_t^\eta}(w_t, w_{t+1}) &= \frac{1}{2}\langle w_{t+1} - w_t, A_t(w_{t+1} - w_t)\rangle \\
&= \frac{1}{2}\langle w_t - w_{t+1}, \eta(M_t w_t + v_t)\rangle && \text{by (10.7)} \\
&= \frac{\eta}{2}\langle w_t - w_{t+1}, (M_t w_t + v_t)\rangle \\
&= \frac{\eta}{2}\langle w_t - w_{t+1}, x_t(\langle x_t, w_t\rangle - y_t)\rangle && \text{by (10.9)} \\
&= \frac{\eta}{2}(y_t - \langle x_t, w_t\rangle)\langle w_{t+1} - w_t, x_t\rangle \\
&= -\frac{\eta^2}{2}(y_t - \langle x_t, w_t\rangle)\langle M_t w_t + v_t, A_t^{-1} x_t\rangle && \text{by (10.8)} \\
&= \frac{\eta^2}{2}(y_t - \langle x_t, w_t\rangle)^2\langle x_t, A_t^{-1} x_t\rangle && \text{by (10.9)} \\
&= \frac{\eta^2}{2}\ell_t(w_t)\langle x_t, A_t^{-1} x_t\rangle
\end{aligned}$$

□

Applying the lemma to (10.4) gives

$$\widehat{L}_n - L_n(u) \le \frac{\|u\|^2}{2\eta} + \frac{\eta}{2}\sum_{t=1}^{n} \ell_t(w_t)\left\langle x_t, A_t^{-1} x_t\right\rangle \ .$$

If define $B_n = \max_{1\le t\le n} \ell_t(w_t)$ we have

$$\widehat{L}_n - L_n(u) \le \frac{\|u\|^2}{2\eta} + \frac{\eta B_n}{2}\sum_{t=1}^{n} \left\langle x_t, A_t^{-1} x_t\right\rangle \ . \tag{10.10}$$

It remains to bound the terms $\left\langle x_t, A_t^{-1} x_t\right\rangle$. This is done in the following lemma.

Lemma 10.3 (Matrix Determinant Lemma)**.** *Let B be a $d \times d$ positive definite matrix, let $x \in \mathbb{R}^d$ and define $A = B + xx^\top$. Then,*

$$\langle x, A^{-1} x\rangle = 1 - \frac{\det(B)}{\det(A)} \ .$$

Proof. Note that since B is positive definite and $xx^\top$ is positive semi-definite, A must be positive definite and therefore also invertible. We can write

$$B = A - xx^\top = A(I - A^{-1}xx^\top)$$

and hence

$$\det(B) = \det(A)\det(I - A^{-1}xx^\top) \ .$$

We focus on the later term and use that A has a square root $A^{1/2}$

$$\begin{aligned}\det(I - A^{-1}xx^\top) &= \det(A^{1/2})\det(I - A^{-1}xx^\top)\det(A^{-1/2}) \\ &= \det(A^{1/2}(I - A^{-1}xx^\top)A^{-1/2}) \\ &= \det(I - A^{-1/2}xx^\top A^{-1/2}) \\ &= \det(I - (A^{-1/2}x)(A^{-1/2}x)^\top) \\ &= \det(I - zz^\top)\end{aligned}$$

where $z = A^{-1/2}x$ is a shorthand notation. We can calculate the determinant of a matrix as product of its eigenvalues. It thus remains to find the eigenvalues of $I - zz^\top$. One eigenvector is z since

$$(I - zz^\top)z = z - z(z^\top z) = (1 - z^\top z)z \ .$$

and the corresponding eigenvector value is $1 - z^\top z$. Since $I - zz^\top$ is symmetric, its eigenvectors are orthogonal. If u is orthogonal to z then it is an eigenvector with eigenvalue 1, since

$$(I - zz^\top)u = u - zz^\top u = u \ .$$

The multiplicity of eigenvalue 1 is $d - 1$ because the subspace of vectors orthogonal to z has dimension $d - 1$. Therefore, Therefore,

$$\det(B) = \det(A)\det(I - zz^\top) = \det(A)(1 - z^\top z) = \det(A)(1 - x^\top A^{-1}x) \ .$$

Reordering gives the result. □

Theorem 10.4 (Regret Bound for Ridge Regression)**.** *Let $y_1, y_2, \ldots, y_n \in \mathbb{R}$ and let $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$ be such that $\|x_t\|_2 \leq X$ for all $1 \leq t \leq n$. Let $\ell_t(w) = (y_t - \langle w, x\rangle)^2$ and $\{w_t\}_{t=1}^{n+1}$ be sequence generated by the FTRL algorithm with learning rate $\eta > 0$. The regret with respect to any $u \in R^d$ is upper bounded as*

$$\widehat{L}_n - L_n(u) \leq \frac{\|u\|^2}{2\eta} + \frac{B_n d}{2}\ln\left(1 + \frac{\eta X^2 n}{d}\right)$$

where $B_n = \max_{1\leq t\leq n} \ell_t(w_t)$.

Proof. We start with the inequality (10.10) and we see that it remains to upper bound $\eta\sum_{t=1}^n \left\langle x_t, A_t^{-1}x_t\right\rangle \leq d\ln\left(1 + \frac{\eta n X^2}{d}\right)$. Form the matrix determinant lemma (Lemma 10.3) with $B = A_{t-1}$, $A = A_t$ and $x = \eta^{1/2}x_t$ we have

$$\eta\sum_{t=1}^n \left\langle x_t, A_t^{-1}x_t\right\rangle = \sum_{t=1}^n \left\langle x_t\sqrt{\eta}, A_t^{-1}x_t\sqrt{\eta}\right\rangle = \sum_{t=1}^n \left(1 - \frac{\det(A_{t-1})}{\det(A_t)}\right)$$

Now we use that $1 - x \leq -\ln x$ for any real number $x > 0$ and get that

$$\eta\sum_{t=1}^n \left\langle x_t, A_t^{-1}x_t\right\rangle \leq \sum_{t=1}^n \ln\left(\frac{\det(A_t)}{\det(A_{t-1})}\right) = \ln\left(\det(A_n)\right) \ .$$

In order to upper bound the determinant of A_n we use that $\|x_t\|_2 \leq X$ and derive from that an upper bound on the trace of A_n:

$$\operatorname{tr}(A_n) = \operatorname{tr}(I) + \eta \sum_{t=1}^{n} \operatorname{tr}(x_t x_t^\top) \leq d + \eta X^2 n \ .$$

In the derivation, we have used that trace is linear and that $\operatorname{tr}(x_t x_t^\top) = \operatorname{tr}(\langle x_t, x_t \rangle) = \|x_t\|_2^2$. Since A_n is positive definite, if $\operatorname{tr}(A_n) \leq C$ then $\ln(\det(A_n)) \leq d \ln(C/d)$ and therefore

$$\ln(\det(A_n)) \leq d \ln \left(1 + \frac{\eta X^2 n}{d} \right)$$

which finishes the proof of the theorem. □

10.2 Ridge Regression with Projections

We now investigate the generalization of the algorithm where w_t is constrained to lie in closed convex set $K \subseteq \mathbb{R}^d$. We also write the loss functions in a slightly more general form:

$$\ell_t(w) = c_t + \kappa_t \langle w, v_t \rangle + \frac{\beta}{2} \langle w, v_t v_t^\top w \rangle$$

where $c_t, \kappa_t \in \mathbb{R}$, $v_t \in \mathbb{R}^d$ and $\beta > 0$. Note that ℓ_t is a convex function, since $\beta > 0$. The algorithm is the usual Follow The Regularized Leader algorithm from Chapter 8. That is, in round $t+1$, the algorithm chooses

$$w_{t+1} = \operatorname*{argmin}_{w \in K} \left(\eta \sum_{t=1}^{n} \ell_t(w) + R(w) \right) \ .$$

where $R(w) = \frac{1}{2}\|w\|_2^2$.

10.2.1 Analysis of Regret

To analyze the regret of the algorithm, we start by applying the Follow-The-Leader-Be-The-Leader-Inequality (Corollary 8.12):

$$\widehat{L}_n - L_n(u) \leq \widehat{L}_n - \widehat{L}_n^+ + \frac{R(u) - R(w_1)}{\eta} \ .$$

Here $\widehat{L}_n = \sum_{t=1}^n \ell_t(w_t)$ is the loss of the algorithm, $\widehat{L}_n^+ = \sum_{t=1}^n \ell_t(w_{t+1})$ is the loss of the "cheating" algorithm, and $L_n(u) = \sum_{t=1}^n \ell_t(u)$ is the sum of the loss functions.

We see that in order to upper bound the regret, we need to upper bound the differences $\ell_t(w_t) - \ell_{t+1}(w_{t+1})$. We do it in the following lemma.

Lemma 10.5. *The FTRL with convex closed K, regularizer $R(w) = \frac{1}{2}\|w\|_2^2$, learning rate $\eta > 0$, and loss functions of the form $\ell_t = c_t + \kappa_t\langle w, v_t\rangle + \frac{\beta}{2}\langle w, v_t v_t^\top w\rangle$ where $c_t, \kappa_t \in \mathbb{R}$, $v_t \in \mathbb{R}^d$ and $\beta > 0$ satisfies, for any $t \geq 1$*

$$\ell_t(w_{t+1}) - \ell_t(w_t) \leq \eta\langle \nabla\, \ell_t(w_t), A_t^{-1}\, \nabla\, \ell_t(w_t)\rangle$$

where

$$A_t = I + \eta\beta \sum_{s=1}^{t} v_t v_t^\top \ .$$

Proof. Let $\Delta w_t = w_{t+1} - w_t$. From convexity of ℓ_t we have

$$\ell_t(w_t) - \ell_t(w_{t+1}) \leq -\langle \nabla\, \ell_t(w_t), \Delta w_t\rangle \ . \tag{10.11}$$

Let $M_t = v_t v_t^\top$. Then $\ell_t(w) = c_t + \kappa_t\langle w, v_t\rangle + \frac{\beta}{2}\langle w, M_t w\rangle$ and $\nabla\, \ell_t(w) = \kappa_t v_t + \beta M_t w$. Since $\nabla\, R(w) = w$, we have

$$\begin{aligned}
\nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_t^\eta(w_t) &= \eta \sum_{s=1}^{t} (\nabla\, \ell_s(w_{t+1}) - \nabla\, \ell_s(w_t)) + \nabla\, R(w_{t+1}) - \nabla\, R(w_t) \\
&= \eta \sum_{s=1}^{t} (\beta M_s w_{t+1} - \beta M_s w_t) + w_{t+1} - w_t \\
&= (I + \eta\beta \sum_{s=1}^{t} M_s)(w_{t+1} - w_t) \\
&= A_t \Delta w_t
\end{aligned}$$

and hence

$$\Delta w_t = A_t^{-1}\, (\nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_t^\eta(w_t)) \ . \tag{10.12}$$

We now express $\nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_t^\eta(w_t)$ in a different way:

$$\begin{aligned}
\nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_t^\eta(w_t) &= (\nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_{t-1}^\eta(w_t)) + (\nabla\, L_{t-1}^\eta(w_t) - \nabla\, L_t^\eta(w_t)) \\
&= (\nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_{t-1}^\eta(w_t)) - \eta\, \nabla\, \ell_t(w_t) \ .
\end{aligned}$$

Therefore, defining a shorthand $d_t = \nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_{t-1}^\eta(w_t)$ we can write

$$\nabla\, L_t^\eta(w_{t+1}) - \nabla\, L_t^\eta(w_t) = d_t - \eta\, \nabla\, \ell_t(w_t) \ .$$

Substituting into (10.12) we get

$$\Delta w_t = A_t^{-1}\, (d_t - \eta\, \nabla\, \ell_t(w_t)) \ . \tag{10.13}$$

Combining (10.11) and (10.13) we have

$$\begin{aligned}
\ell_t(w_t) - \ell_t(w_{t+1}) &\leq -\langle \nabla\, \ell_t(w_t), \Delta w_t\rangle \\
&= -\langle \nabla\, \ell_t(w_t), A_t^{-1}\, (d_t - \eta\, \nabla\, \ell_t(w_t))\rangle \\
&= -\langle \nabla\, \ell_t(w_t), A_t^{-1} d_t\rangle + \eta\langle \nabla\, \ell_t(w_t), A_t^{-1}\, \nabla\, \ell_t(w_t)\rangle
\end{aligned}$$

To finish the proof of the first part of the lemma (the inequality) it remains to show that $\langle \nabla \ell_t(w_t), A_t^{-1} d_t \rangle$ is non-negative.

By optimality of w_t and w_{t+1}, for any $w \in K$

$$\begin{aligned} \langle \nabla L_t^\eta(w_{t+1}), w - w_{t+1} \rangle &\geq 0 \ , \\ \langle \nabla L_{t-1}^\eta(w_t), w - w_t \rangle &\geq 0 \ . \end{aligned}$$

Substituting w_t for w in the first inequality, and substituting w_{t+1} for w in the second inequality, we get

$$\begin{aligned} \langle \nabla L_t^\eta(w_{t+1}), w_t - w_{t+1} \rangle &\geq 0 \ , \\ \langle \nabla L_{t-1}^\eta(w_t), w_{t+1} - w_t \rangle &\geq 0 \ . \end{aligned}$$

We add the inequalities and start upper bounding:

$$\begin{aligned} 0 &\leq \langle \nabla L_t^\eta(w_{t+1}), w_t - w_{t+1} \rangle + \langle \nabla L_{t-1}^\eta(w_t), w_{t+1} - w_t \rangle \\ &= -\langle d_t, \Delta w_t \rangle \\ &= -\langle d_t, A_t^{-1} \left(d_t - \eta \, \nabla \ell_t(w_t) \right) \rangle \qquad \text{by (10.13)} \\ &= -\langle d_t, A_t^{-1} d_t \rangle + \eta \langle d_t, A_t^{-1} \, \nabla \ell_t(w_t) \rangle \\ &\leq \eta \langle d_t, A_t^{-1} \, \nabla \ell_t(w_t) \rangle \end{aligned}$$

where in the last step we have used that A_t is positive definite. We have thus derived that

$$\eta \langle d_t, A_t^{-1} \, \nabla \ell_t(w_t) \rangle \geq 0 \ .$$

This finishes the proof the lemma. □

Theorem 10.6 (Regret of Projected Ridge Regression)**.** *Let $\beta > 0$ and $G \geq 0$. Let $\{\ell_t\}_{t=1}^n$ be a sequence of loss functions of the form $\ell_t = c_t + \kappa_t \langle w, v_t \rangle + \frac{\beta}{2} \langle w, v_t v_t^\top w \rangle$ where $c_t, \kappa_t \in \mathbb{R}$, $v_t \in \mathbb{R}^d$ such that $\|v_t\|_2 \leq G$ for all $1 \leq t \leq n$. The regret of FTRL on $\{\ell_t\}_{t=1}^n$ with convex closed $K \subseteq \mathbb{R}^d$, regularizer $R(w) = \frac{1}{2}\|w\|_2^2$ and learning rate $\eta > 0$ is upper bounded, for any $u \in K$, as*

$$\widehat{L}_n - L_n(u) \leq \frac{\|u\|_2^2}{2\eta} + \frac{d B_n}{\beta} \ln \left(1 + \frac{\eta \beta G^2 n}{d} \right)$$

where

$$B_n = \max_{1 \leq t \leq n} \left(\kappa_t + \beta \langle v_t, w_t \rangle \right)^2 \ .$$

Proof. First, we express the upper bound $\eta \langle \nabla \ell_t(w_t), A_t^{-1} \, \nabla \ell_t(w_t) \rangle$ from the previous lemma using that $\nabla \ell_t(w) = \kappa_t v_t + \beta v_t v_t^\top w = (\kappa_t + \beta \langle v_t, w \rangle) v_t$ as

$$\begin{aligned} \eta \langle \nabla \ell_t(w_t), A_t^{-1} \, \nabla \ell_t(w_t) \rangle &= \eta \langle (\kappa_t + \beta \langle v_t, w \rangle) v_t, A_t^{-1} (\kappa_t + \beta \langle v_t, w \rangle) v_t \rangle \\ &= \eta (\kappa_t + \beta \langle v_t, w_t \rangle)^2 \langle v_t, A_t^{-1} v_t \rangle \ . \end{aligned} \tag{10.14}$$

We can upper bound the regret as

$$\begin{aligned}
\widehat{L}_n - L_n(u) &\le \widehat{L}_n - \widehat{L}_n^+ + \frac{R(u) - R(w_1)}{\eta} && \text{(Corollary 8.12)} \\
&\le \frac{\|u\|_2^2}{2\eta} + \sum_{t=1}^{n} (\ell_t(w_t) - \ell_t(w_{t+1})) && (R(w_1) \ge 0) \\
&\le \frac{\|u\|_2^2}{2\eta} + \eta \sum_{t=1}^{n} \langle \nabla \ell_t(w_t), A_t^{-1} \nabla \ell_t(w_t) \rangle && \text{(by Lemma 10.5)} \\
&= \frac{\|u\|_2^2}{2\eta} + \eta \sum_{t=1}^{n} \langle (\kappa_t + \beta \langle v_t, w_t \rangle)^2 \langle v_t, A_t^{-1} v_t \rangle && \text{by (10.14)} \\
&\le \frac{\|u\|_2^2}{2\eta} + \eta B_n \sum_{t=1}^{n} \langle v_t, A_t^{-1} v_t \rangle
\end{aligned}$$

Using the matrix determinant lemma, we can deal with the terms $\langle v_t, A_t^{-1} v_t \rangle$:

$$\eta\beta \langle v_t, A_t^{-1} v_t \rangle = \langle \sqrt{\eta\beta} v_t, A_t^{-1} (\sqrt{\eta\beta} v_t) \rangle = 1 - \frac{\det(A_{t-1})}{\det(A_t)} \, .$$

Therefore, using $1 - x \le \ln x$ for any $x > 0$ we have

$$\begin{aligned}
\widehat{L}_n - L_n(u) &\le \frac{\|u\|_2^2}{2\eta} + \frac{B_n}{\beta} \sum_{t=1}^{n} \left(1 - \frac{\det(A_{t-1})}{\det(A_t)} \right) \\
&\le \frac{\|u\|_2^2}{2\eta} + \frac{B_n}{\beta} \sum_{t=1}^{n} \ln \left(\frac{\det(A_t)}{\det(A_{t-1})} \right) \\
&= \frac{\|u\|_2^2}{2\eta} + \frac{B_n}{\beta} \ln(\det(A_n))
\end{aligned}$$

It remains to upper bound $\ln(\det(A_n))$. Since, from $\|v_t\|_2 \le G$ we can derive upper bound on the trace of A_n

$$\operatorname{tr}(A_n) = \operatorname{tr}(I) + \sum_{t=1}^{n} \beta\eta \operatorname{tr}(v_t v_t^\top) \le d + \beta\eta G^2 n$$

the log-determinant of A_t is bounded as

$$\ln(\det(A_n)) \le d \ln \left(1 + \frac{\beta\eta G^2 n}{d} \right) \, .$$

□

Supppose that we assume that the decision set K is bounded and $\kappa_t \le \kappa$ for all t. Then, using the assumption that $\|v_t\|_2 \le G$ and from that $w_t \in K$, we can give an upper bound on B_n which does not depend on n.

10.3 Directional Strong Convexity

We can generalize the analysis to a more general class of loss functions.

Definition 10.7 (Directional Strong Convexity)**.** Let $f : K \to \mathbb{R}$ be a differentiable function defined on a convex set $K \subseteq \mathbb{R}^d$. Let $\beta \geq 0$ be a real number. We say that f is *β-directionally strongly convex* if for any $x, y \in K$

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\beta}{2} \langle \nabla f(x), y - x \rangle^2 .$$

We now extend ridge regression to arbitrary directionally strongly convex loss functions. The idea is similar to that of linearized loss. However, instead of linearizing the loss functions, we approximate them with quadratic functions of rank one. Consider a sequence $\{\ell_t\}_{t=1}^{\infty}$ of β-directionally strongly convex loss functions defined on a convex closed set $K \subseteq \mathbb{R}^d$. We define the FTRL WITH QUADRATIC RANK-ONE APPROXIMATIONS:

$$w_{t+1} = \operatorname*{argmin}_{w \in K} \left(\eta \sum_{s=1}^{t} \widetilde{\ell}_s(w) + \frac{1}{2} \|w\|_2^2 \right)$$

where

$$\widetilde{\ell}_s(w) = \ell_s(w_s) + \langle \nabla \ell_s(w_s), w - w_s \rangle + \frac{\beta}{2} \langle \nabla \ell_s(w_s), w - w_s \rangle^2 .$$

To analyze the regret of we use a similar trick as for linearized losses (Lemma 9.7).

Proposition 10.8. *If $\ell_t : K \to \mathbb{R}$ is β-directionally strongly convex then for any $u \in K$*

$$\ell_t(w_t) - \ell_t(u) \leq \widetilde{\ell}_t(w_t) - \widetilde{\ell}_t(u) .$$

Proof. Since $\ell_t(w_t) = \widetilde{\ell}_t(w_t)$ the inequality is equivalent to $\widetilde{\ell}_t(u) \leq \widetilde{\ell}_t(w_t)$ and it follows from directional strong convexity of ℓ_t. □

If we define $L_n(u) = \sum_{t=1}^{n} \ell_t(u)$ and $\widehat{L}_n = \sum_{t=1}^{n} \ell_t(w_t)$ we see that

$$\widehat{L}_n - L_n(u) \leq \sum_{t=1}^{n} (\widetilde{\ell}_t(w_t) - \widetilde{\ell}_t(u))$$

Combining this inequality with Theorem 10.6 applied to the sequence $\{\widetilde{\ell}_t\}_{t=1}^{n}$ of approximated losses, we can upper the regret of the algorithm.

Theorem 10.9. *Let $\beta > 0$ and $G \geq 0$. Let $\{\ell_t\}_{t=1}^{n}$ be a sequence of β-directionally strongly convex loss functions defined on a convex closed set $K \subseteq \mathbb{R}^d$ such that $\| \nabla \ell_t(w) \| \leq G$ for any $w \in K$ and any $1 \leq t \leq n$. The regret of* FTRL WITH QUADRATIC RANK-ONE APPROXIMATIONS *on the sequence $\{\ell_t\}_{t=1}^{n}$ with learning rate $\eta > 0$ and regularizer $R(w) = \frac{1}{2}\|w\|_2^2$ is upper bounded for any $u \in K$ as*

$$\widehat{L}_n - L_n(u) \leq \frac{\|u\|_2^2}{2\eta} + \frac{dB_n}{\beta} \ln \left(1 + \frac{\eta \beta G^2 n}{d} \right)$$

where

$$B_n = ??? .$$

10.4 Exercises

Exercise 10.1. Let A be a $d \times d$ positive definite matrix. Show that $\mathrm{tr}(A) \leq C$ implies that $\ln(\det(A)) \leq C \ln(C/d)$.

Exercise 10.2. Show that the ridge regression algorithm can be implemented in $O(d^2)$ time per time step. (Hint: Use Sherman-Morrison formula for rank-one updates of the inverse of a matrix.)

Chapter 11

Exp-concave Functions

Definition 11.1 (Exp-concavity)**.** Let $\alpha > 0$. A function $g : K \to \mathbb{R}$ defined on a convex set $K \subseteq \mathbb{R}^d$ is called α-*exp-concave*, if the function $\exp(-\alpha g(x))$ is concave.

Proposition 11.2. *Let $\alpha > \beta > 0$. If g is α-exp-concave then it is also β-exp-concave.*

Proof. Let $h_\alpha(x) = \exp(-\alpha g(x))$ and let $h_\beta(x) = \exp(-\alpha g(x))$ is concave. By assumption h_α is concave. We need to prove that h_β is also concave. Noting that $h_\beta(x) = (h_\alpha(x))^{\beta/\alpha}$, we have

$$\begin{aligned} h_\beta(ax+by) &= (h_\alpha(ax+by))^{\beta/\alpha} \\ &\geq (ah_\alpha(x) + bh_\alpha(y))^{\beta/\alpha} \\ &\geq a\left(h_\alpha(x)\right)^{\beta/\alpha} + b\left(h_\alpha(y)\right)^{\beta/\alpha} \qquad \text{(by concavity } z^{\beta/\alpha}\text{)} \\ &= ah_\beta(x) + bh_\beta(y)\ . \end{aligned}$$

□

Lemma 11.3. *Let $g : K \to \mathbb{R}$ be a twice-differentiable function defined on convex domain $K \subseteq \mathbb{R}^d$. Let H, G be positive reals. If for all $x \in K$, $\|\nabla g(x)\|_2 \leq G$ and $\lambda_{\min}(\nabla^2 g(x)) \geq H$ then g is (H/G^2)-exp-concave.*

Proof. Let $\alpha = H/G^2$ and $h(x) = \exp(-\alpha g(x))$. We calculate the Hessian of h and show that it is negative semi-definite:

$$\begin{aligned} \nabla^2 h(x) &= \frac{\partial}{\partial x}\left(\nabla h(x)\right) \\ &= \frac{\partial}{\partial x}\left(-\alpha \nabla g(x) \exp(-\alpha g(x))\right) \\ &= \alpha \nabla^2 g(x) \exp(-\alpha g(x)) + \alpha^2 \nabla g(x) \nabla g(x)^\top \exp(-\alpha g(x)) \\ &= \alpha h(x)\left[\alpha \nabla g(x) \nabla g(x)^\top - \nabla^2 g(x)\right] \end{aligned}$$

Since $\alpha h(x)$ is a positive scalar, it remains to show that the matrix $\alpha \nabla g(x) \nabla g(x)^\top - \nabla^2 g(x)$ is negative semi-definite. We show that all its eigenvalues are non-positive. Using that $\lambda_{\max}(A+B) \leq \lambda_{\max}(A) + \lambda_{\max}(B)$ for any symmetric matrices A, B, we have

$$\begin{aligned}
\lambda_{\max}\left(\alpha \nabla g(x) \nabla g(x)^\top - \nabla^2 g(x)\right) &\leq \lambda_{\max}\left(\alpha \nabla g(x) \nabla g(x)^\top\right) + \lambda_{\max}\left(-\nabla^2 g(x)\right) \\
&= \alpha \lambda_{\max}\left(\nabla\, g(x)\, \nabla\, g(x)^\top\right) - \lambda_{\min}\left(\nabla^2 g(x)\right) \\
&\leq \alpha G^2 - H \\
&\leq 0
\end{aligned}$$

where we have used that the only non-zero eigenvalue of $vv^\top$ is $\|v\|_2$ with associated eigenvector v. □

Lemma 11.4. *Let G, D, α be positive reals. Let $g : K \to \mathbb{R}$ be an α-exp-concave function such that*

1. $\|\nabla g(x)\|_2 \leq G$
2. $\forall x, y \in K,\ \|x - y\|_2 \leq D$.

Then, g is $\frac{1}{2}\min\left\{\alpha, \frac{1}{GD}\right\}$-directionally strongly convex.

Proof. Let $\gamma = \min\left\{\alpha, \frac{1}{GD}\right\}$. The function $h(x) = \exp(-\gamma g(x))$ is concave. Thus,

$$\begin{aligned}
h(x) &\leq h(y) + \langle \nabla h(y), x - y \rangle \\
&= h(y) + \langle -\gamma \nabla g(y) h(y), x - y \rangle \\
&= h(y)\left[1 - \gamma \langle \nabla g(y), x - y \rangle\right] \\
&= \exp(-\gamma g(y))\left[1 - \gamma \langle \nabla g(y), x - y \rangle\right]
\end{aligned}$$

Taking logarithm and dividing by $-\gamma$ we get

$$g(x) \geq g(y) - \frac{1}{\gamma} \ln\left[1 - \gamma \langle \nabla g(y), x - y \rangle\right] \ .$$

We use that $-\ln(1-z) \geq z + \frac{z^2}{4}$ for any $z \in [-1, 1)$. For $z = \gamma \langle \nabla g(y), x - y \rangle$ we obtain

$$g(x) \geq g(y) + \langle \nabla g(y), x - y \rangle + \frac{\gamma}{4}\left(\langle \nabla g(y), x - y \rangle\right)^2 \ .$$

It remains to verify that z lies in the interval $[-1, 1)$. The quantity $\exp(-\gamma g(y))[1 - z]$ is positive, since it is lower-bounded by a positive quantity $h(x)$. Because $\exp(-\gamma g(y))$ is also positive, $[1 - z]$ is positive. Equivalently, $z < 1$. Furthermore,

$$\begin{aligned}
|z| &= \gamma |\langle \nabla g(y), x - y \rangle| \\
&\leq \gamma \|\nabla g(y)\|_2 \cdot \|x - y\|_2 \\
&\leq \gamma G D \\
&\leq 1 \ .
\end{aligned}$$

This means that $z \geq -1$. □

11.1 Exercises

Exercise 11.1. Show that if a function is α-exp-concave for some $\alpha > 0$ then it is convex.

Exercise 11.2. Prove that $-\ln(1-z) \geq z + \frac{z^2}{4}$ holds for any $z \in [-1, 1)$.

Chapter 12

p-Norm Regularizers and Legendre Duals

In this chapter we investigate the Linearized Proximal-Point Algorithm with different regularizers. We focus in particular on squared p-norm regularizers $R_p(w) = \frac{1}{2}\|w\|_p^2$.

Let $\ell_1, \ell_2, \ldots, \ell_n$ be a sequence convex differentiable loss functions defined on a convex closed set K. The Linearized Proximal-Point Algorithm with a Legendre regularizer $R : A \to \mathbb{R}$ starts with $w_1 = \operatorname{argmin}_{w \in K \cap A} R(w)$ and in round $t+1$ it chooses

$$w_{t+1} = \operatorname*{argmin}_{w \in K \cap A} \left[\eta \widetilde{\ell}_t(w) + D_R(w, w_t) \right]$$

where $\widetilde{\ell}_t(w) = \langle \nabla \ell_t, w - w_t \rangle$ is the linearized loss. Also recall that, $\widetilde{w}_1 = \operatorname{argmin}_{w \in A} R(w)$ and

$$\widetilde{w}_{t+1} = \operatorname*{argmin}_{w \in A} \left[\eta \widetilde{\ell}_t(w) + D_R(w, w_t) \right]$$

are the unprojected (unconstrained) solutions and $w_t = \Pi_{R,K}(\widetilde{w}_t)$.

Recall that by Lemma 9.2 the regret of the algorithm for any $u \in K \cap A$ and any $\eta > 0$ satisfies

$$\widehat{L}_n - L_n(u) \leq \frac{1}{\eta} \left[D_R(u, w_1) + \sum_{t=1}^{n} D_R(w_t, \widetilde{w}_{t+1}) \right],$$

where $\widehat{L}_n = \sum_{t=1}^n \ell_t(w_t)$ is the loss of algorithm and $L_n(u) = \sum_{t=1}^n \ell_t(u)$ is the sum of the loss functions. The rest of the chapter is devoted to the careful investigation of the terms $D_R(w_t, \widetilde{w}_{t+1})$.

12.1 Legendre Dual

We can analyze $D_R(w_t, \widetilde{w}_{t+1})$ in an elegant way using the machinery of *Legendre duals*. We start with the definition.

Definition 12.1 (Legendre dual)**.** Let $R : A \to \mathbb{R}$ be a Legendre function. Let $A^* = \{\nabla R(v) \ : \ v \in A\}$. The (Legendre) *dual* of R, $R^* : A^* \to \mathbb{R}$, is defined by

$$R^*(u) = \sup_{v \in A} \left(\langle u, v \rangle - R(v) \right) , \quad u \in A^* .$$

The following statement, which is given without proof, follows from the definition.

Lemma 12.2. *Let $R : A \to \mathbb{R}$ be a Legendre function. Then,*

(i) R^* is a Legendre function.

(ii) $R^{**} = R$.

The inverse of the gradient of a Legendre function can be obtained as the gradient of the function's dual. This is the subject of the next proposition.

Proposition 12.3. *Let $R : A \to \mathbb{R}$ be a Legendre function and let $R^* : A^* \to \mathbb{R}$ be its dual. Then,*

$$\nabla R^* = (\nabla R)^{-1}$$

where the inverse is the inverse of the function $\nabla R : A \to A^*$. (In particular, the inverse of this function always exist.)

The proof of this proposition is based on the following elementary lemma:

Lemma 12.4. *Let $R : A \to \mathbb{R}$ be Legendre function and $R^* : A^* \to \mathbb{R}$ its Legendre dual. Let $u \in A$ and $u' \in A^*$. The following two conditions are equivalent:*

1. $R(u) + R^*(u') = \langle u, u' \rangle$.
2. $u' = \nabla R(u)$.

Proof. Fix u, u'. Define the function $G : A \to \mathbb{R}$, $G(v) = \langle v, u' \rangle - R(v)$. Using the function G, the first condition can be written as

$$R^*(u') = G(u) . \tag{12.1}$$

Since, by definition, $R^*(u') = \sup_{v \in A} G(v)$, and G is strictly concave, (12.1) holds if and only if

$$\nabla G(u) = 0 .$$

This can be equivalently written as

$$u' - \nabla R(u) = 0 ,$$

which is the same as the second condition. □

Let us now turn to the proof of Proposition 12.3.

Proof of Proposition 12.3. Pick $u \in A$ and define $u' = \nabla R(u)$. By the previous lemma we have

$$R(u) + R^*(u') = \langle u, u' \rangle \ .$$

Since $R^{**} = R$, we have

$$R^{**}(u) + R^*(u') = \langle u, u' \rangle \ .$$

Applying the previous lemma to R^* in place of R, we have

$$u = \nabla R^*(u') \, .$$

This shows that $u = \nabla R^*(\nabla R(u))$. In other words, ∇R^* is the right inverse of ∇R. Since ∇R is a surjection (a map onto A^*), ∇R^* must be also its left inverse. □

Equipped with this result, we can prove the following important result which connects the Bregman divergences underlying a Legendre function and its dual.

Proposition 12.5. *Let $R : A \to \mathbb{R}$ be a Legendre function and let $R^* : A^* \to \mathbb{R}$ be its dual. Then, for any $u, v \in A$,*

$$D_R(u, v) = D_{R^*}(\nabla R(v), \nabla R(u)) \ .$$

Proof. Let $u' = \nabla R(u)$ and $v' = \nabla R(v)$. By Lemma 12.4, $R(u) = \langle u, u' \rangle - R^*(u')$ and $R(v) = \langle v, v' \rangle - R^*(v')$. Therefore,

$$\begin{aligned} D_R(u, v) &= R(u) - R(v) - \langle \nabla R(v), u - v \rangle \\ &= \langle u, u' \rangle - R^*(u') - (\langle v, v' \rangle - R^*(v')) - \langle v', u - v \rangle \\ &= R^*(v') - R^*(u') - \langle u, v' - u' \rangle \ . \end{aligned}$$

By Proposition 12.3, $u = (\nabla R)^{-1}(u') = \nabla R^*(u)$ and therefore

$$\begin{aligned} D_R(u, v) &= R^*(v') - R^*(u') - \langle \nabla R^*(u'), v' - u' \rangle \\ &= D_{R^*}(v', u') \ , \end{aligned}$$

which finishes the proof. □

A consequence of the proposition is that we can write the term $D_R(w_t, \widetilde{w}_{t+1})$ as

$$D_R(w_t, \widetilde{w}_{t+1}) = D_{R^*}(\nabla R(\widetilde{w}_{t+1}), \nabla R(w_t)) \ .$$

Furthermore, it is not hard to show that if $1 \leq p \leq \infty$ then the dual of $R_p(w) = \frac{1}{2}\|w\|_p^2$ is $R_q = \frac{1}{2}\|w\|_q^2$ where q satisfies $\frac{1}{p} + \frac{1}{q} = 1$. Thus, we are left with studying the properties of D_{R^*}.

12.2 p-Norms and Norm-Like Divergences

We now show that for $p \geq 2$, D_{R_p} behaves essentially like the p-norm.

Definition 12.6 (Norm-like Bregman divergence)**.** Let $R : A \to \mathbb{R}$ be a Legendre function on a domain $A \subset \mathbb{R}^d$, let $\|\cdot\|$ be a norm on $\mathbb{R}^d$ and let $c > 0$. We say that the Bregman divergence associated with R is *c-norm-like* with respect to $\|\cdot\|$, if for any $u, v \in A$,

$$D_R(u, v) \leq c\|u - v\|^2 .$$

Our goal is to show that $R \equiv R_p$ where $R_p(w) = \frac{1}{2}\|w\|_p^2$ is $(p-1)/2$-norm-like with respect to the p-norm. Recall that for any $p \geq 1$, the p-norm of a vector $x = (x_1, x_2, \ldots, x_d)^\top \in \mathbb{R}^d$ is defined as

$$\|x\|_p = \left(\sum_{i=1}^{d} |x_i|^p\right)^{1/p} .$$

The definition can be extended to $p = \infty$ by defining $\|x\|_\infty = \max_{1 \leq i \leq d} |x_i|$. We will need Hölder's inequality, which is a generalization of Cauchy-Schwarz inequality.

Lemma 12.7 (Hölder's inequality)**.** *Fix $1 \leq p, q \leq \infty$ such that $\frac{1}{p} + \frac{1}{q} = 1$. Then, for any $x, y \in \mathbb{R}^d$ it holds that*

$$|\langle x, y\rangle| \leq \|x\|_p \cdot \|y\|_q .$$

A pair $(p, q) \in [1, \infty] \times [1, \infty]$ satisfying $\frac{1}{p} + \frac{1}{q} = 1$ is called a *conjugate pair*. For example $(1, \infty)$, $(p, q) = (2, 2)$ and $(p, q) = (3, \frac{3}{2})$ are conjugate pairs.

In order to upper bound D_R we start with a simple application of Taylor's theorem. We keep the argument more general so that they can be reused in studying cases other than $R = R_p$. In order to be apply Taylor's theorem, we will assume that R is twice-differentiable in A°. Since $D_R(u, v)$ is the difference between $R(u)$ and its first order Taylor's expansion at v, the divergence is nothing else but the remainder of the second order Taylor's expansion of R. Thus, there exists ξ on the open line segment between u and v such that

$$D_R(u, v) = \frac{1}{2}\left\langle u - v, \nabla^2 R(\xi)u - v\right\rangle .$$

Lemma 12.8. *Suppose $\phi : \mathbb{R} \to \mathbb{R}$, $\psi : \mathbb{R} \to \mathbb{R}$ are twice-differentiable and ψ is concave. Consider $R : \mathbb{R}^d \to \mathbb{R}$ defined by $R(\xi) = \psi\left(\sum_{i=1}^d \phi(\xi_i)\right)$ for any $\xi \in \mathbb{R}^d$. Then, for any $x \in \mathbb{R}^d$*

$$\left\langle x, \nabla^2 R(\xi)x\right\rangle \leq \psi'\left(\sum_{i=1}^{d} \phi(\xi_i)\right)\sum_{i=1}^{d} \phi''(\xi_i)x_i^2 .$$

Proof. Let e_i be i-th vector of the standard orthogonal basis of $\mathbb{R}^d$. The gradient of R is

$$\nabla R(\xi) = \psi'\left(\sum_{i=1}^{d} \phi(\xi_i)\right)\sum_{i=1}^{d} \phi'(\xi_i)e_i .$$

The Hessian of R is

$$\begin{aligned}
\nabla^2 R(\xi) &= \frac{\partial}{\partial \xi} \left(\nabla R(\xi) \right) \\
&= \frac{\partial}{\partial \xi} \left(\psi' \left(\sum_{i=1}^d \phi(\xi_i) \right) \sum_{i=1}^d \phi'(\xi_i) e_i \right) \\
&= \left(\sum_{i=1}^d \phi'(\xi_i) e_i \right) \left[\frac{\partial}{\partial \xi} \psi' \left(\sum_{i=1}^d \phi(\xi_i) \right) \right] + \psi' \left(\sum_{i=1}^d \phi(\xi_i) \right) \frac{\partial}{\partial \xi} \sum_{i=1}^d \phi'(\xi_i) e_i \\
&= \left(\sum_{i=1}^d \phi'(\xi_i) e_i \right) \psi'' \left(\sum_{i=1}^d \phi(\xi_i) \right) \sum_{i=1}^d \phi'(\xi_i) e_i^\top + \psi' \left(\sum_{i=1}^d \phi(\xi_i) \right) \sum_{i=1}^d \phi''(\xi_i) e_i e_i^\top
\end{aligned}$$

Since ψ is concave $\psi''(\xi) \leq 0$ and thus

$$\begin{aligned}
\langle x, \nabla^2 R(\xi) x \rangle &= \psi'' \left(\sum_{i=1}^d \phi(\xi_i) \right) \left(\sum_{i=1}^d \phi'(\xi_i) x_i \right)^2 + \psi' \left(\sum_{i=1}^d \phi(u_i) \right) \sum_{i=1}^d \phi''(\xi_i) x_i^2 \\
&\leq \psi' \left(\sum_{i=1}^d \phi(u_i) \right) \sum_{i=1}^d \phi''(\xi_i) x_i^2 \, .
\end{aligned}$$

□

We are now ready to state the upper bound on the Bregman divergence underlying R_p:

Proposition 12.9. *Let $p \geq 2$ and $R(u) = R_p(u) = \frac{1}{2}\|u\|_p^2$. Then the following bound holds for the Bregman divergence D_R:*

$$D_R(u, v) \leq \frac{p-1}{2} \|u - v\|_p^2 \, .$$

Proof. Fix $u, v \in \mathbb{R}^d$. Clearly, R is twice differentiable on $\mathbb{R}^d$. As explained above, $D_R(u, v) = \frac{1}{2} \left\langle u - v, \nabla^2 R(\xi) u - v \right\rangle$ for some ξ lying on the open line segment between u and v. We apply the previous lemma to $\psi(z) = \frac{1}{2} z^{2/p}$ and $\phi(z) = |z|^p$. (Note that for $p \geq 2$, the function ψ is concave.) Then,

$$\psi'(z) = \frac{1}{p} z^{\frac{2-p}{p}} \, , \qquad \phi'(z) = \operatorname{sign}(z) p |z|^{p-1} \, , \qquad \phi''(z) = p(p-1)|z|^{p-2} \, .$$

The previous lemma for $x = u - v$ gives,

$$\begin{aligned}
D_R(u,v) &= \frac{1}{2}\langle u-v, \nabla^2 R(\xi)(u-v)\rangle \\
&\le \frac{1}{2p}\left(\sum_{i=1}^d |\xi_i|^p\right)^{\frac{2-p}{p}} p(p-1)\sum_{i=1}^d |u_i|^{p-2}(u_i-v_i)^2 \\
&= \frac{p-1}{2}\|\xi\|_p^{2-p}\sum_{i=1}^d |\xi_i|^{p-2}(u_i-v_i)^2 \\
&\le \frac{p-1}{2}\|\xi\|_p^{2-p}\left(\sum_{i=1}^d |\xi_i|^p\right)^{\frac{p-2}{p}} \|u-v\|_p^2 \qquad \text{(Hölder's inequality)} \\
&= \frac{p-1}{2}\|\xi\|_p^{2-p}\cdot\|\xi\|_p^{p-2}\cdot\|u-v\|_p^2 \\
&= \frac{p-1}{2}\|u-v\|_p^2 .
\end{aligned}$$

□

12.3 Regret for Various Regularizers

We now analyze regret of Linearized Proximal Point Algorithm with regularizers, divergences of which are norm-like. Furthermore, we will assume that that the loss functions are non-negative and satisfy $\|\nabla \ell_t(w)\|^2 \le \alpha \ell_t(w)$ for some $\alpha \ge 0$ and some norm. For example if the loss functions are of the form $\ell_t(w) = \frac{1}{2}(\langle w, x_t\rangle - y_t)^2$, $x_t, w \in \mathbb{R}^d$, $y_t \in \mathbb{R}$, as in ridge regression, then the norm of their gradient can bounded as

$$\|\nabla \ell_t(w)\|^2 = \|(\langle w, x_t\rangle - y_t)x_t\|^2 = 2\|x_t\|_2^2 \cdot \|\ell_t(w)\|^2$$

and thus we may take $\alpha = 2\max_{1\le t\le n}\|x_t\|_2^2$.

Theorem 12.10 (Norm-Like Divergences)**.** *Let $A \subset \mathbb{R}^d$, let $R : A \to \mathbb{R}$ be a Legendre function, let $R^* : A^* \to \mathbb{R}$ be its Legendre dual, $\|\cdot\|$ a norm on $\mathbb{R}^d$ and let $c \ge 0$ and $\alpha \ge 0$. Assume that D_{R^*} is c-norm-like with respect to $\|\cdot\|$. Let $\{\ell_t\}_{t=1}^n$ be a sequence of non-negative convex differentiable loss functions defined on a convex closed set K satisfying $\|\nabla \ell_t(w)\|^2 \le \alpha \ell_t(w)$. Consider the* Linearized Proximal Point Algorithm *with regularizer R applied to $\{\ell_t\}_{t=1}^n$. If $\eta = \sqrt{\frac{D_R(u,w_1)}{c\alpha \widehat{L}_n}}$ then for any $u \in K \cap A$,*

$$\widehat{L}_n - L_n(u) \le 4c\alpha D_R(u,w_1) + 2\sqrt{c\alpha D_R(u,w_1)L_n(u)} .$$

Proof. We know that

$$\begin{aligned} D_R(w, \widetilde{w}_{t+1}) &= D_{R^*}\left(\nabla R(\widetilde{w}_{t+1}), \nabla R(w_t)\right) \\ &\le c\|\nabla R(\widetilde{w}_{t+1}) - \nabla R(w_t)\|^2 \\ &= c\eta^2\|\nabla \ell_t(w_t)\|^2 \qquad \text{(Proposition 9.8 with } \eta_t = \eta) \\ &\le c\alpha\eta^2 \ell_t(w_t) \end{aligned}$$

Lemma 9.2 gives

$$\begin{aligned} \widehat{L}_n - L_n(u) &\le \frac{D_R(u, w_1)}{\eta} + \eta c\alpha \sum_{t=1}^{n} \ell_t(w_t) \\ &= \frac{D_R(u, w_1)}{\eta} + \eta c\alpha \widehat{L}_n \end{aligned}$$

Substituting for η we get

$$\widehat{L}_n - L_n(u) \le 2\sqrt{c\alpha D_R(u, \widetilde{w}_1)\widehat{L}_n}\ .$$

To extract a regret bound from this inequality, we treat it as a quadratic inequality in variable $x = \widehat{L}_n - L_n(u)$. Let $A = c\alpha D_R(u, \widetilde{w}_1)$. Then the inequality can be written as $x \le 2\sqrt{A(x + L_n)}$. Squaring it gives

$$x^2 \le 4A(x + L_n)\ .$$

This inequality holds can be satisfied only if for x which do not exceed larger of the two roots $x_{1,2} = 2A \pm 2\sqrt{A^2 + AL_n}$ of the quadratic polynomial $x^2 - 4Ax - 4AL_n$. Hence,

$$x \le 2A + 2\sqrt{A^2 + AL_n}\ .$$

Using the inequality $\sqrt{a + b} \le \sqrt{a} + \sqrt{b}$, valid for any non-negative a, b, we get that

$$x \le 4A + 2\sqrt{AL_n}\ .$$

Substituting for x and A we obtain statement of the theorem. □

An interesting special case is p-norm squared regularizer, $R_p(u) = \frac{1}{2}\|u\|_p^2$.

Corollary 12.11 (p-Norm Squared Regularizer)**.** *Assume* $\ell_t(w) = \frac{1}{2}(\langle w, x_t\rangle - y_t)^2$ *and* $R(u) = R_p(u)$ *and* $1 < p \le 2$. *Then for any* $u \in K \cap A$

$$\widehat{L}_n - L_n(u) \le 2(q-1)\|u\|_p^2 X_q^2 + X_q\|u\|_p\sqrt{2(q-1)L_n(u)}$$

where $\frac{1}{p} + \frac{1}{q} = 1$ *and* $X_q = \max_{1\le t\le n}\|x_t\|_q$.

Proof. First note that $R^* = R_q$ and therefore D_{R^*} by Proposition 12.9 is $\frac{q-1}{2}$-norm-like with respect q-norm. Second, since $\widetilde{w}_1 = 0$

$$D_R(u, w_1) \le D_R(u, \widetilde{w}_1) = \frac{1}{2}\|u\|_p^2 \, .$$

Third, for $\alpha = 2X_q^2$ we get

$$\| \nabla \ell_t(w)\|_q^2 = 2\ell_t(w)\|x_t\|_q^2 \le \alpha \ell_t(w) \, .$$

The theorem gives regret bound,

$$\widehat{L}_n - L_n(u) \le 4\frac{q-1}{2}2X_q^2\frac{\|u\|_p^2}{2} + 2\sqrt{\frac{q-1}{2}2X_q^2\frac{\|u\|_p^2}{2}L_n(u)}$$

□

Consider the situation when a u with a small loss $L_n(u)$ is sparse and x_t's are dense. For example, assume that u is a standard unit vector and $x_t \in \{+1, -1\}^d$. Then it's a good idea to choose $q = \log d$ and consequently $p \approx 1$. Then $\|x_t\|_q \approx \|x_t\|_\infty = 1$, $\|u\|_p = 1$ and therefore $X_q\|u\|_p\sqrt{q-1} \approx \sqrt{\log d}$. If we were to choose $p = q = 2$ we would get $\|u\|_q = 1$ but $\|x_t\|_q = \sqrt{d}$ and $X_q\|u\|_p\sqrt{q-1} \approx \sqrt{d}$. Thus, $q = \log d$ is exponentially better choice than $q = 2$.

On the other hand, consider the situation when a u with a small loss $L_n(u)$ is dense and x_t's are sparse. For example assume that x_t's are standard unit vectors and $u \in \{+1, -1\}^d$. Then, if we use $p = q = 2$, then $\|u\|_p X_q\sqrt{q-1} = \sqrt{d}$. If were to chooses $q = \log d$ and $p \approx 1$ then $\|u\|_p \approx \|u\|_1 = d$, $\|x_t\|_q = 1$ and therefore $X_q\|u\|_p\sqrt{q-1} \approx d\sqrt{\log d}$. Thus $p = q = 2$ is more than $\sqrt{d}$ better choice than $q = \log d$.

12.4 Exercises

Exercise 12.1. Consider any vector $x = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$. Show that

$$\lim_{p\to\infty} \|x\|_p = \max_{1\le i\le d} |x_i| \, .$$

Exercise 12.2. Let $R_p(u) = \frac{1}{2}\|u\|_p^2$ for some $1 \le p \le \infty$. Show that the Legendre dual of $R_p(u)$ is $R_q(v) = \frac{1}{2}\|v\|_p^2$ where q satisfies $\frac{1}{p} + \frac{1}{q} = 1$.

Exercise 12.3. Let $R(u) = \sum_{i=1}^d e_i^u$. Show that the Legendre dual of $R(u)$ is $R^*(v) = \sum_{i=1}^d v_i(\ln(v_i) - 1)$.

Exercise 12.4.

- Show that $\|x\|_\infty \leq \|x\|_2 \leq \sqrt{d}\|x\|_\infty$ for any $x \in \mathbb{R}^d$.
- Show that there exists $c > 0$ such that for any integer $d \geq 3$ if $x \in \mathbb{R}^d$ and $q = \ln d$ then $\|x\|_\infty \leq \|x\|_q \leq c\|x\|_\infty$. Find the smallest possible value of c.
- Show that there does **not** exist $c > 0$ such that for any integer $d \geq 1$ if $x \in \mathbb{R}^d$ then $\|x\|_\infty \leq \|x\|_2 \leq c\|x\|_\infty$.

Morale: This shows that the infinity-norm is well approximated by $(\ln d)$-norm but only very poorly approximated by the 2-norm.

Exercise 12.5. Show that $\|x\|_p \leq \|x\|_q$ for any $x \in \mathbb{R}^d$ and any $1 \leq q \leq p \leq \infty$.

Chapter 13

Exponentiated Gradient Algorithm

In this chapter, we consider the Proximal Point Algorithm with Linearized Losses with the unnormalized negative entropy regularizer

$$R(w) = \sum_{i=1}^{d} w_i \ln(w_i) - w_i$$

defined on $A = (0, \infty)^d$ and sequence of convex loss functions $\ell_1, \ell_2, \ldots, \ell_n$ defined the probability simplex d-dimensional probability simplex

$$K = \Delta_d = \left\{ w \in \mathbb{R}^d \ : \ \sum_{i=1}^{d} w_i = 1 \text{ and } \forall 1 \leq i \leq d, \ w_i \geq 0 \right\} .$$

If the loss functions were linear, we would recover Exponentially Weighted Averages Forecaster (EWA). (See Exercises 8.1 and 9.3.) Our goal is however to consider non-linear loss functions such as $\ell_t(w) = \frac{1}{2}(\langle w, x_t \rangle - y_t)^2$. More generally, we consider loss functions $\ell_1, \ell_2, \ldots, \ell_n$ for which there exists $\alpha > 0$ such that $\| \nabla \ell_t(w) \|_\infty^2 \leq \alpha \ell_t(w)$ for all $1 \leq t \leq n$ and all w.

For linear prediction problems where the task is to predict y_t from x_t by using a linear predictor $\widehat{y}_t = \langle w_t, x_t \rangle$ it is somewhat unnatural to restrict w_t to the probability simplex Δ_d. By introducing extra dimensions we can extended the analysis to $K' = \{w \ : \ \|w\|_1 \leq c\}$. See Exercise 13.1.

Recall that Proximal Point Algorithm with Linearized Losses in round $t+1$ chooses

$$w_{t+1} = \operatorname*{argmin}_{w \in K \cap A} \left[\eta \widetilde{\ell}_t(w) + D_R(w, w_t) \right]$$

where $\widetilde{\ell}_t(w) = \langle \nabla \ell_t(w_t), w \rangle$ is the linearized loss function. Let

$$\widetilde{w}_{t+1} = \operatorname*{argmin}_{w \in A} \left[\eta \widetilde{\ell}_t(w) + D_R(w, w_t) \right]$$

be the unprojected solution. Since $R(w) = \sum_{i=1}^{d} w_i \ln(w_i) - w_i$ and $K = \Delta_d$ the update can written as

$$\begin{aligned} \widetilde{w}_{t+1,i} &= w_{t,i} \cdot \exp(-\eta \nabla_i \ell_t(w_t)) \qquad \text{for } 1 \le i \le d, \\ w_{t+1} &= \frac{\widetilde{w}_{t+1}}{\|\widetilde{w}_{t+1}\|_1} \ . \end{aligned}$$

where ∇_i denotes the i-th component of the gradient. The resulting algorithm is, for an obvious reason, called Exponentiated Gradient Algorithm (EG). The update for EG can be easily derived from the first two parts of the following proposition. We leave its proof as an exercise for the reader.

Proposition 13.1 (Properties of Negative Entropy Regularizer)**.** *Let $K = \Delta$ and for any $w \in (0,\infty)^d$ let $R(w) = \sum_{i=1}^{d} w_i \ln(w_i) - w_i$. Then,*

1. $\Pi_{R,K}(w) = \frac{w}{\|w\|_1}$ *for any* $w \in (0,\infty)^d$.
2. $\nabla_i R(w) = \ln w_i$ *for any* $w \in (0,\infty)^d$.
3. $D_R(u,v) = \langle u, \nabla R(u) - \nabla R(v)\rangle$ *for any* $u, v \in (0,\infty)^d$.

Theorem 13.2 (Regret of EG)**.** *Let $\alpha \ge 0$ and assume the loss functions $\ell_1, \ell_2, \ldots, \ell_n$ defined on Δ_d are convex and differentiable, and satisfy $\|\nabla \ell_t(w)\|_\infty^2 \le \alpha \ell_t(w)$ for all $w \in \Delta_d$ and all $1 \le t \le n$. Then,* Exponentiated Gradient Algorithm *satisfies for $u \in \Delta_d$ with learning rate $\eta = \sqrt{\frac{2\ln d}{\alpha \widehat{L}_n}}$ satisfies*

$$\widehat{L}_n - L_n(u) \le 2\alpha \ln(d) + \sqrt{2\alpha \ln(d) L_n(u)} \ .$$

Proof. By the same calculation as in Lemma 9.2 we know that for any $u \in \Delta_d$

$$\begin{aligned} &\eta(\ell_t(w_t) - \ell_t(u)) \\ &\le D_R(u, w_t) - D_R(u, \widetilde{w}_{t+1}) + D_R(w_t, \widetilde{w}_{t+1}) \\ &= (D_R(u, w_t) - D_R(u, w_{t+1})) + (D_R(u, w_{t+1}) - D_R(u, \widetilde{w}_{t+1})) + D_R(w_t, \widetilde{w}_{t+1}) \end{aligned} \tag{13.1}$$

We deal with each of the three terms separately. We leave the first term as it is. We express the third term using

$$\begin{aligned} D_R(w_t, \widetilde{w}_{t+1}) &= \langle w_t, \nabla R(w_t) - \nabla R(\widetilde{w}_{t+1})\rangle \\ &= \eta \langle w_t, \nabla \ell_t(w_t)\rangle \ . \end{aligned} \tag{13.2}$$

where the last equality follows by Proposition 9.8. We start expressing the second term as

$$\begin{aligned} D_R(u, w_{t+1}) - D_R(u, \widetilde{w}_{t+1}) &= \langle u, \nabla R(u) - \nabla R(w_{t+1}) - \nabla R(u) + \nabla R(\widetilde{w}_{t+1})\rangle \qquad \text{(by Proposition 13.1)} \\ &= \langle u, \nabla R(\widetilde{w}_{t+1}) - \nabla R(w_{t+1})\rangle \\ &= \sum_{i=1}^{n} u_i \left(\nabla_i R(\widetilde{w}_{t+1}) - \nabla_i R(w_{t+1})\right) \end{aligned}$$

and since

$$\nabla_i R(\widetilde{w}_{t+1}) - \nabla_i R(w_{t+1}) = \ln\left(\frac{\widetilde{w}_{t+1,i}}{w_{t+1,i}}\right) = \ln\left(\frac{\widetilde{w}_{t+1,i}}{\widetilde{w}_{t+1,i}/\|\widetilde{w}_{t+1}\|_1}\right) = \ln\|\widetilde{w}_{t+1}\|_1$$

we see that the second term equals

$$\begin{aligned}
D_R(u, w_{t+1}) - D_R(u, \widetilde{w}_{t+1}) &= \sum_{i=1}^{n} u_i \left(\nabla_i R(\widetilde{w}_{t+1}) - \nabla_i R(w_{t+1})\rangle\right) \\
&= \ln\|\widetilde{w}_{t+1}\|_1 \sum_{i=1}^{n} u_i \\
&= \ln\|\widetilde{w}_{t+1}\|_1 \qquad\qquad (\text{since } \sum_{i=1}^{d} u_i = 1) \\
&= \ln\left(\sum_{i=1}^{d} \widetilde{w}_{t+1,i}\right) \\
&= \ln\left(\sum_{i=1}^{d} w_{t,i} \cdot \exp(-\eta\, \nabla_i\, \ell_t(w_t))\right) \\
&\leq -\eta\langle w_t, \nabla\ell_t(w_t)\rangle + \frac{\eta^2}{2}\|\nabla\ell_t(w_t)\|_\infty^2 \qquad (13.3)
\end{aligned}$$

where the last inequality follows from Hoeffding's lemma applied to the random variable X with distribution $\Pr[X = \nabla_i\, \ell_t(w_t)] = w_{t,i}$.

Returning back to (13.1) and substituting (13.2) and (13.3) we get

$$\begin{aligned}
\eta(\ell_t(w_t) - \ell_t(u)) &\leq D_R(u, w_t) - D_R(u, w_{t+1}) + \frac{\eta^2}{2}\|\nabla\, \ell_t(w_t)\|_\infty^2 \\
&\leq D_R(u, w_t) - D_R(u, w_{t+1}) + \frac{\eta^2}{2}\alpha\ell_t(w_t)\ .
\end{aligned}$$

Summing over all $t = 1, 2, \ldots, n$ the first two terms telescope. If we drop $-D_R(u, w_{n+1})$ and divide by by η we get

$$\widehat{L}_n - L_n(u) \leq \frac{1}{\eta} D_R(u, w_1) + \frac{\eta\alpha}{2}\widehat{L}_n$$

Since $w_{1,i} = \frac{1}{d}$ for all $i = 1, 2, \ldots, d$ the first term can be upper bounded using $D_R(u, w_1) \leq \ln(d)$ and thus

$$\widehat{L}_n - L_n(u) \leq \frac{\ln d}{\eta} + \frac{\eta\alpha}{2}\widehat{L}_n\ .$$

Choosing $\eta = \sqrt{\frac{2\ln d}{\alpha\widehat{L}_n}}$ which minimizes the right hand side, we get

$$\widehat{L}_n - L_n(u) \leq \sqrt{2\alpha\ln(d)\widehat{L}_n}$$

Using the quadratic inequality trick, as in the proof of Theorem 12.10, we get the result. □

For loss functions of the form $\ell_t = \frac{1}{2}(\langle w, x_t\rangle - y_t)^2$ we have $\|\nabla \ell_t(w)\|_\infty^2 = 2\ell_t(w)\|x_t\|_\infty^2$ and thus we can take $\alpha = 2\max_{1\le t\le n}\|x_t\|_\infty^2$.

13.1 Exercises

Exercise 13.1. Consider the linear prediction problem where we want to predict $y_t \in \mathbb{R}$ from $x_t \in \mathbb{R}^d$ by using a linear predictions $\widehat{y}_t = \langle w_t, x_t\rangle$ where $w_t \in K$ and

$$K = \{w \in \mathbb{R}^d \ : \ \|w\|_1 \le 1\} \ .$$

and loss that we suffer in each round is $\ell_t(w_t)$ where $\ell_t(w) = \frac{1}{2}(y_t - \langle w, x_t\rangle)^2$.

Show that EG algorithm on the $(2d+1)$-dimensional probability simplex can be used to solve this problem. What regret bound do you get? Generalize the result to the case when

$$K = \{w \in \mathbb{R}^d \ : \ \|w\|_1 \le c\} \ .$$

for some $c > 0$. How does regret bound changes? How does it depend on c?

(Hint: Define $x'_t = (x_{t,1}, x_{t,2}, \ldots, x_{t,d}, 0, -x_{t,1}, x_{t,1})$ and $\ell'_t(w') = \frac{1}{2}(y_t - \langle w, x'_t\rangle)^2$ for $w' \in \Delta_{2d+1}$.)

Chapter 14

Connections to Statistical Learning Theory

In this chapter we connect online learning with the more traditional part of machine learning—the statistical learning theory. The fundamental problem of statistical learning theory is the off-line (batch) learning, where we are given a random sample and the goal is to produce a single predictor that performs well on future, unseen data. The sample and the future data are connected by the assumption that they both are drawn from the same distribution.

More formally, we consider the scenario where we are given a hypothesis space $\mathcal{H}$ and a independent identically distributed (i.i.d.) sequence of loss functions $\{\ell_t\}_{t=1}^{\infty}$ where $\ell_t : \mathcal{H} \to \mathbb{R}$ for each $t \geq 1$. The elements of $\mathcal{H}$ are called either hypotheses, predictors, classifiers or models depending on the context. We will denote a typical element of $\mathcal{H}$ by w or W with various subscripts and superscripts.

Example 14.1 (Linear Prediction)**.**

$$\mathcal{H} = \mathbb{R}^d \qquad \ell_t(w) = \frac{1}{2}\left(\langle w, X_t\rangle - Y_t\right)^2$$
$$(X_t, Y_t) \in \mathbb{R}^d \times \mathbb{R} \qquad \{(X_t, Y_t)\}_{t=1}^{\infty} \text{ is i.i.d.}$$

Example 14.2 (Non-Linear Prediction)**.**

$$\mathcal{H} = \left\{f : \mathbb{R}^d \to \mathbb{R} \ : \ f \text{ is continuous}\right\} \qquad \ell_t(f) = \frac{1}{2}\left(f(X_t) - Y_t\right)^2$$
$$(X_t, Y_t) \in \mathbb{R}^d \times \mathbb{R} \qquad \{(X_t, Y_t)\}_{t=1}^{\infty} \text{ is i.i.d.}$$

Example 14.3 (Binary Classification with Linear Functions)**.**

$$\mathcal{H} = \mathbb{R}^d \qquad \ell_t(w) = \mathbb{I}\{\text{sign}(\langle w, X_t\rangle) \neq Y_t\}$$
$$(X_t, Y_t) \in \mathbb{R}^d \times \{-1, +1\} \qquad \{(X_t, Y_t)\}_{t=1}^{\infty} \text{ is i.i.d.}$$

The goal in statistical learning is to find w that has small risk. The risk is defined as the expected loss. The risk captures the performance of w on future data. Formally, the *risk* of

a hypothesis $w \in \mathcal{H}$ is defined to be

$$\overline{\ell}(w) = \mathbf{E}[\ell_1(w)] = \mathbf{E}[\ell_2(w)] = \cdots = \mathbf{E}[\ell_t(w)] \ .$$

Note that the definition is independent of t since by assumption $\ell_1, \ell_2, \ldots$ have identical distribution. The best possible risk that we can hope to achieve is

$$\ell^* = \inf_{w \in \mathcal{H}} \overline{\ell}(w) \ .$$

A learning algorithm gets as input (the description of) the loss functions $\ell_1, \ell_2, \ldots, \ell_n$ and possibly some randomness if it is a randomized algorithm, and outputs a hypothesis $\widetilde{W}_n \in \mathcal{H}$. Note that even for a deterministic algorithm its output $\widetilde{W}_n$ is random, since the input was random to start with.[1] The goal of the algorithm is to minimize *excess risk*

$$\overline{\ell}(\widetilde{W}_n) - \ell^* \ .$$

14.1 Goals of Statistical Learning Theory

The first basic question studied in statistical learning theory is whether and how fast the excess risk converges to zero as $n \to \infty$. Since $\overline{\ell}(\widetilde{W}_n)$ and hence also the excess risk are random variables, a mode of converge needs to be specified (almost surely, in probability, in expectation, etc.) The goal is to design algorithms (i) for which excess risk converges to zero and (ii) for which the convergence is as fast as possible. The best possible speed of converge is called the minimax rate.

Formally, let A be a learning algorithm, let P be a probability distribution over loss functions chosen from some family of probability distributions $\mathcal{P}$, and assume that $\ell_t \sim P$ for all t. The expected excess risk of A on P is

$$r_n(A, P) = \mathbf{E}[\ell(\widetilde{W}_n^{A,P})] - \ell^*$$

where $\widetilde{W}_{n+1}^{A,P}$ is the output of A upon seeing an i.i.d. sample from P. The worst-case expected excess risk of A on the class of distributions $\mathcal{P}$ is

$$r_n(A, \mathcal{P}) = \sup_{P \in \mathcal{P}} \ r_n(A, P) \ .$$

Finally, the *minimax rate* of the class $\mathcal{P}$ is

$$r_n^*(\mathcal{P}) = \inf_{A \in \mathcal{A}} \ r_n(A, \mathcal{P})$$

where $\mathcal{A}$ is the class of all online randomized algorithms. An algorithm is consider asymptotically optimal if $r_n(A, \mathcal{P}) = O(r_n^*(\mathcal{P}))$.

[1] To distinguish random and non-random elements of $\mathcal{H}$, we denote the non-random elements by w and random elements W (decorated with various subscripts and super-scripts).

Typical results proved in statistical learning theory are *a priori generalization* bounds for specific algorithms. These are typically high-probability bounds on the excess risk. Often, these bounds do not depend on P and are thus called *distribution-free* bounds.

Second question question studied in statistical learning theory is to give a computable upper bound on $\overline{\ell}(W_n)$ based on empirical data. Such bounds are called data-dependent bounds *a posteriori* bounds. These bounds are good for evaluation purposes and they help us to choose from hypotheses produced by various algorithms (or the same algorithms with different settings of parameters).

14.2 Online-to-Batch Conversions

In previous chapters, we have dealt with online algorithms that achieve low regret. We now show that a low regret online algorithm can be converted to a batch algorithm with low excess error. Roughly speaking, an algorithm with regret R_n can be converted to a batch algorithm with excess error at most R_n/n. Methods that achieve this are called online-to-batch conversions. These conversions take the sequence $W_1, W_2, \ldots, W_n$ of hypothesis produced by the online algorithm and output a single hypothesis.

The first online-to-batch conversion we consider works as follows. Pick a hypothesis uniformly at random from $W_1, W_2, \ldots, W_n$. Formally, we draw U_n uniformly at random from $\{1, 2, \ldots, n\}$ and we output W_{U_n}. This conversion has three desirable properties: (i) The average loss of the algorithm $\frac{1}{n}\sum_{t=1}^{n} \ell_t(w_t)$ is an unbiased estimator of expected risk of W_{U_n}. (ii) In expectation, the excess loss of W_{U_n} is upper bounded by the average per-step regret of the algorithm. (iii) The conversion is applicable regardless of whether the loss functions are convex or not. The next theorem formalizes the first two properties.

Theorem 14.4 (Randomized Online-to-Batch Conversion)**.** *Let $\ell_1, \ell_2, \ldots, \ell_n$ be an i.i.d. sequence of loss functions, $\ell_t : \mathcal{H} \to \mathbb{R}$. Suppose that an online algorithm produces on $\{\ell_t\}_{t=1}^n$ a sequence $W_1, W_2, \ldots, W_n \in \mathcal{H}$ of hypotheses. Let U_n be an independent random variable uniformly distributed on $\{1, 2, \ldots, n\}$. Then,*

$$\mathbf{E}[\overline{\ell}(W_{U_n})] = \mathbf{E}\left[\frac{1}{n}\sum_{t=1}^{n} \ell_t(W_t)\right]$$

Furthermore, if $R_n = \sum_{t=1}^{n} \ell_t(W_t) - \inf_{w\in\mathcal{H}} \sum_{t=1}^{n} \ell_t(w)$ denotes the regret, then

$$\mathbf{E}[\overline{\ell}(W_{U_n})] \;-\; \min_{w\in\mathcal{H}} \; \overline{\ell}(w) \leq \frac{1}{n}\,\mathbf{E}\left[R_n\right] \;.$$

Proof. Since W_t and ℓ_t are independent, $\mathbf{E}[\overline{\ell}(W_t)] = \mathbf{E}[\ell_t(W_t)]$. The first part of the theorem

then follows by straightforward calculation:

$$\begin{aligned}
\mathbf{E}[\overline{\ell}(W_{U_n})] &= \mathbf{E}\left[\sum_{t=1}^{n} \mathbb{I}\{U_n = t\}\overline{\ell}(W_t)\right] \\
&= \sum_{t=1}^{n} \mathbf{E}\left[\mathbb{I}\{U_n = t\}\overline{\ell}(W_t)\right] \\
&= \sum_{t=1}^{n} \mathbf{E}\left[\mathbb{I}\{U_n = t\}\right] \cdot \mathbf{E}[\overline{\ell}(W_t)] \qquad \text{(by independence of } U_n \text{ and } W_t\text{)} \\
&= \frac{1}{n}\sum_{t=1}^{n} \mathbf{E}[\overline{\ell}(W_t)] \\
&= \frac{1}{n}\sum_{t=1}^{n} \mathbf{E}[\ell_t(W_t)] \qquad \text{(by independence of } \ell_t \text{ and } W_t\text{)} \\
&= \mathbf{E}\left[\frac{1}{n}\sum_{t=1}^{n} \ell_t(W_t)\right]
\end{aligned}$$

From the first part of the theorem we see that the second part of the theorem, is equivalent to

$$\frac{1}{n}\,\mathbf{E}\left[\inf_{w\in\mathcal{H}} \sum_{t=1}^{n} \ell_t(w)\right] \leq \min_{w\in\mathcal{H}}\ \overline{\ell}(w)\ .$$

Since $\overline{\ell}(w) = \frac{1}{n}\,\mathbf{E}[\sum_{t=1}^{n} \ell_t(w)]$, that is equivalent to

$$\frac{1}{n}\,\mathbf{E}\left[\inf_{w\in\mathcal{H}} \sum_{t=1}^{n} \ell_t(w)\right] \leq \frac{1}{n}\inf_{w\in\mathcal{H}} \mathbf{E}\left[\sum_{t=1}^{n} \ell_t(w)\right]$$

which is obviously true, since $\mathbf{E}\inf[\cdot] \leq \inf \mathbf{E}[\cdot]$. □

In the case of convex loss functions defined on a convex set there is a better online-to-batch conversion. It outputs the *average* $\overline{W}_n = \frac{1}{2}\sum_{t=1}^{n} W_t$ of the hypotheses $W_1, W_2, \ldots, W_n$ generated by the online algorithm. The conversion enjoys similar properties as the random hypothesis W_{U_n}. (i) The expected risk of $\overline{W}_n$ is does not exceed the expected the risk of W_{U_n}. (ii) In expectation, the excess loss of $\overline{W}_n$ is upper bounded by the average per-step regret of the algorithm. Additionally and in contrast with W_{U_n}, it is possible to show high-probability bounds on risk and excess risk of $\overline{W}_n$ provided that the loss functions are bounded.

Theorem 14.5 (Averaging Online-to-Batch Conversion)**.** *Let $\ell_1, \ell_2, \ldots, \ell_n$ be an i.i.d. sequence of convex loss functions, $\ell_t : \mathcal{H} \to \mathbb{R}$, defined on a convex set $\mathcal{H}$. Suppose that an online algorithm produces on $\{\ell_t\}_{t=1}^{n}$ a sequence $W_1, W_2, \ldots, W_n \in \mathcal{H}$ of hypotheses. Let $\overline{W}_n = \frac{1}{n}\sum_{t=1}^{n} W_t$ the average of the hypotheses. Then,*

$$\mathbf{E}[\overline{\ell}(\overline{W}_n)] \leq \mathbf{E}\left[\frac{1}{n}\sum_{t=1}^{n} \ell_t(W_t)\right]\ .$$

Furthermore, if $R_n = \sum_{t=1}^n \ell_t(W_t) - \inf_{w\in\mathcal{H}} \sum_{t=1}^n \ell_t(w)$ *denotes the regret, then*

$$\mathbf{E}[\overline{\ell}(\overline{W}_n)] \;-\; \min_{w\in\mathcal{H}} \;\overline{\ell}(w) \le \frac{1}{n}\,\mathbf{E}\left[R_n\right] \;.$$

Proof. Let ℓ_{n+1} be an independent copy of ℓ_1 (independent of $\ell_1, \ell_2, \ldots, \ell_n$). Then, since ℓ_{n+1} is convex, by Jensen's inequality

$$\ell_{n+1}(\overline{W}_n) = \ell_{n+1}\left(\frac{1}{n}\sum_{t=1}^n W_t\right) \le \frac{1}{n}\sum_{t=1}^n \ell_{n+1}(W_t) \;.$$

Taking expectation we get

$$\mathbf{E}[\ell_{n+1}(\overline{W}_n)] \le \frac{1}{n}\sum_{t=1}^n \mathbf{E}\left[\ell_{n+1}(W_t)\right] \;.$$

The first part of the theorem follows from that $\mathbf{E}[\ell_{n+1}(\overline{W}_n)] = \mathbf{E}[\overline{\ell}(\overline{W}_n)]$ which in turn follows by independence of $\overline{W}_n$ and ℓ_{n+1}, and from that for any $t = 1, 2, \ldots, n$, $\mathbf{E}[\ell_{n+1}(W_t)] = \mathbf{E}[\ell_t(W_t)]$ which in turn follows by independence of W_t, ℓ_t, ℓ_{n+1}.

The first part of theorem implies that in order to prove the second part of theorem, it suffices to prove that

$$\frac{1}{n}\,\mathbf{E}\left[\inf_{w\in\mathcal{H}} \sum_{t=1}^n \ell_t(w)\right] \le \min_{w\in\mathcal{H}} \;\overline{\ell}(w) \;.$$

Since $\overline{\ell}(w) = \frac{1}{n}\,\mathbf{E}[\sum_{t=1}^n \ell_t(w)]$, that is equivalent to

$$\frac{1}{n}\,\mathbf{E}\left[\inf_{w\in\mathcal{H}} \sum_{t=1}^n \ell_t(w)\right] \le \frac{1}{n}\inf_{w\in\mathcal{H}} \mathbf{E}\left[\sum_{t=1}^n \ell_t(w)\right]$$

which is obviously true, since $\mathbf{E}\inf[\cdot] \le \inf \mathbf{E}[\cdot]$. □

14.3 Intermezzo: Martingales

Martingales are useful mathematical tool used in probability theory. They were originally invented for analyzing (sequential) betting strategies in casinos. A sequence $X_0, X_1, X_2, \ldots$ of random variables is said to be a *martingale with respect to* another sequence $Y_0, Y_1, Y_2, \ldots$ of random variables, if for all $t \ge 0$

$$\mathbf{E}[X_{t+1} \mid Y_0, Y_1, Y_2, \ldots, Y_t] = X_t \;. \tag{14.1}$$

A typical example of a martingale is the amount of money a gambler has after playing t games, assuming that all games in the casino are fair and the gambler can go negative. In other words, X_t is the amount of money the gambler has after playing t games and Y_t is the

outcome of the t-th game. The gambler can play according to any strategy he wishes (that can depend on past outcomes of his plays), changing between roulette, black jack or slot machines etc. as he pleases. The condition (14.1) expresses the assumption that all games in the casino are fair: After playing t games and having X_t dollars, the expected amount of money after $(t+1)$-th game is X_t. Note that, the condition (14.1) implies that

$$\mathbf{E}[X_0] = \mathbf{E}[X_1] = \mathbf{E}[X_2] = \cdots = \mathbf{E}[X_t]$$

In other words, the expected gambler's wealth does *not* change.

In the next section, we will use Azuma's inequality which is basic result about martingales with bounded increments. (Increments of a martingale are the differences $X_t - X_{t-1}$.) The inequality is a useful generalization of Hoeffding's inequality (Theorem 4.3).

Theorem 14.6 (Azuma's inequality)**.** *Let $X_0, X_1, X_2, \ldots, X_n$ be a martingale with respect to some sequence $Y_0, Y_1, \ldots, Y_n$ such that with probability one, for all $1 \leq t \leq n$*

$$X_t - X_{t-1} \in [A_t, A_t + c]$$

where A_t is a function of $Y_0, Y_1, \ldots, Y_{t-1}$ and $c > 0$ is constant. Then, for any $\varepsilon \geq 0$

$$\Pr[X_n - X_0 \geq \varepsilon] \leq \exp\left(-\frac{2\varepsilon^2}{nc^2}\right) .$$

Equivalently, for all $\delta > 0$, with probability at least $1 - \delta$,

$$X_n < X_0 + c\sqrt{\frac{n}{2}\ln(1/\delta)} .$$

Typically, we will assume that X_0 is some constant (usually zero) and then X_0 in Azuma's inequality can be replaced by $\mathbf{E}[X_n]$.

14.4 High-Probability Bounds for Averaging

We now prove a high-probability bounds on the risk and the excess risk of the averaging online-to-batch conversion. We will need to assume that losses are bound. For simplicity, we will assume that losses lie in $[0, 1]$.

Theorem 14.7 (High Probability Bound for Averaging Online-to-Batch Conversion)**.** *Let $\ell_1, \ell_2, \ldots, \ell_n$ be an i.i.d. sequence of convex loss functions, $\ell_t : \mathcal{H} \to \mathbb{R}$, defined on a convex set $\mathcal{H}$. Assume that with probability one, $\ell_t \in [0, 1]$. Suppose that an online algorithm produces on $\{\ell_t\}_{t=1}^n$ a sequence $W_1, W_2, \ldots, W_n \in \mathcal{H}$ of hypotheses. Let $\overline{W}_n = \frac{1}{n}\sum_{t=1}^n W_t$ the average of the hypotheses. Then, for any $\delta > 0$, with probability at least $1 - \delta$,*

$$\overline{\ell}(\overline{W}_n) < \frac{1}{n}\sum_{t=1}^n \ell_t(W_t) + \sqrt{\frac{\ln(1/\delta)}{2n}} .$$

Furthermore, if $R_n = \sum_{t=1}^n \ell_t(W_t) - \inf_{w\in\mathcal{H}} \sum_{t=1}^n \ell_t(w)$ *denotes the regret, then for any* $\delta > 0$ *with probability at least* $1-\delta$

$$\overline{\ell}(\overline{W}_n) \;-\; \inf_{w\in\mathcal{H}} \;\overline{\ell}(w) < \frac{1}{n}R_n + \sqrt{\frac{2\ln(1/\delta)}{n}}\;.$$

Proof. First notice that $\overline{\ell} : \mathcal{H} \to \mathbb{R}$ is a convex function. This is easy to see since $\overline{\ell}$ is an expectation of a (random) convex function ℓ_1. Formally, for any $w, w' \in \mathcal{H}$ and $\alpha, \beta \geq 0$ such that $\alpha + \beta = 1$,

$$\overline{\ell}(\alpha w + \beta w') = \mathbf{E}[\ell_1(\alpha w + \beta w')] \leq \mathbf{E}[\alpha\ell_1(w) + \beta\ell_1(w')] = \alpha\overline{\ell}(w) + \beta\overline{\ell}(w')\;.$$

Therefore, by Jensen's inequality,

$$\overline{\ell}(\overline{W}_n) \leq \frac{1}{n}\sum_{t=1}^n \overline{\ell}(W_t)\;. \tag{14.2}$$

Thus, in order to prove the first part of the theorem, we see it suffices to prove that $\frac{1}{n}\sum_{t=1}^n \overline{\ell}(W_t) \leq \frac{1}{n}\sum_{t=1}^n \ell_t(W_t) + \sqrt{\ln(1/\delta)/(2n)}$. In order to prove that, we will use Azuma's inequality. The sequence $\{\sum_{s=1}^t(\overline{\ell}(W_s) - \ell_s(W_s))\}_{t=0}^n$ is a martingale with respect to $\{(\ell_t, W_{t+1})\}_{t=0}^n$. Indeed, for any $t \geq 1$

$$\begin{aligned}
&\mathbf{E}\left[\sum_{s=1}^t(\overline{\ell}(W_s) - \ell_s(W_s)) \;\middle|\; W_{1:t}, \ell_{0:t-1}\right] \\
&= \sum_{s=1}^{t-1}(\overline{\ell}(W_s) - \ell_s(W_s)) + \mathbf{E}\left[\overline{\ell}(W_t) - \ell_t(W_t) \;\middle|\; W_{1:t}, \ell_{1:t-1}\right] \\
&= \sum_{s=1}^{t-1}(\overline{\ell}(W_s) - \ell_s(W_s))
\end{aligned}$$

where we have used that $\overline{\ell}(W_t) = \mathbf{E}[\ell_t(W_t) \mid W_{1:t}, \ell_{0:t-1}]$ which holds because ℓ_t is independent of $W_{1:T}, \ell_{0:t-1}$. Since the losses lie in $[0,1]$, the increments of the martingale lie in intervals of length one:

$$\sum_{s=1}^t(\overline{\ell}(W_s) - \ell_s(W_s)) - \sum_{s=1}^{t-1}(\overline{\ell}(W_s) - \ell_s(W_s)) = \overline{\ell}(W_t) - \ell_t(W_t) \in [\overline{\ell}(W_t) - 1, \overline{\ell}(W_t)]\;.$$

Therefore, since the zeroth element of the martingale is zero, by Azuma's inequality with $A_t = \overline{\ell}(W_t) - 1$ and $c = 1$, for any $\delta > 0$

$$\sum_{t=1}^n(\overline{\ell}(W_t) - \ell_t(W_t)) < \sqrt{\frac{n}{2}\ln(1/\delta)}\;.$$

Dividing by n and combining with (14.2) gives the first part of the theorem.

To prove the second part of theorem let $w^* = \operatorname{argmin}_{w\in\mathcal{H}} \overline{\ell}(w)$.[2] Consider the martingale

$$\left\{\sum_{s=1}^{t} \overline{\ell}(W_t) - \ell_t(W_t) - \overline{\ell}(w^*) + \ell_t(w^*)\right\}_{t=0}^{n}$$

with respect to $\{(\ell_t, W_{t+1})\}_{t=1}^n$. This is indeed a martingale, since

$$\begin{aligned}
&\mathbf{E}\left[\sum_{s=1}^{t}(\overline{\ell}(W_s) - \ell_s(W_s) - \overline{\ell}(w^*) + \ell_s(w^*)) \;\middle|\; W_{1:t}, \ell_{0:t-1}\right] \\
&= \sum_{s=1}^{t-1}(\overline{\ell}(W_s) - \ell_s(W_s) - \overline{\ell}(w^*) + \ell_s(w^*)) + \mathbf{E}\left[\overline{\ell}(W_t) - \ell_t(W_t) - \overline{\ell}(w^*) + \ell_t(w^*) \mid W_{1:t}, \ell_{0:t-1}\right] \\
&= \sum_{s=1}^{t-1}(\overline{\ell}(W_s) - \ell_s(W_s) - \overline{\ell}(w^*) + \ell_s(w^*))
\end{aligned}$$

where we have used that $\overline{\ell}(W_t) = \mathbf{E}[\ell_t(W_t) \mid W_{1:t}, \ell_{0:t-1}]$ and $\overline{\ell}(w^*) = \mathbf{E}[\ell_t(w^*) \mid W_{1:t}, \ell_{0:t-1}]$ both of which hold since ℓ_t is independent of $W_{1:t}$ and $\ell_{0:t-1}$. The increments of the martingale lie in an interval of length of 2:

$$\overline{\ell}(W_t) - \ell_t(W_t) - \overline{\ell}(w^*) + \ell_t(w^*) \in [\overline{\ell}(W_t) - \overline{\ell}(w^*) - 1, \overline{\ell}(W_t) - \overline{\ell}(w^*) + 1] \,.$$

Thus, by Azuma's inequality with $A_t = \overline{\ell}(W_t) - \overline{\ell}(w^*) - 1$ and $c = 2$, we have that for any $\delta > 0$, with probability at least $1 - \delta$,

$$\sum_{t=1}^{n}(\overline{\ell}(W_t) - \ell_t(W_t) - \overline{\ell}(w^*) + \ell_t(w^*)) < \sqrt{2n\ln(1/\delta)} \,.$$

Equivalently, with probability at least $1 - \delta$,

$$\frac{1}{n}\left(\sum_{t=1}^{n}\overline{\ell}(W_t)\right) - \frac{1}{n}\left(\sum_{t=1}\ell_t(W_t)\right) - \overline{\ell}(w^*) + \frac{1}{n}\left(\sum_{t=1}^{n}\ell_t(w^*)\right) < \sqrt{\frac{2\ln(1/\delta)}{n}} \,.$$

We use (14.2) to lower bound the first term and $\inf_{w\in\mathcal{H}}\sum_{t=1}^n \ell_t(w) \le \sum_{t=1}^n \ell_t(w^*)$ to lower bound the fourth term. We obtain that with probability at least $1 - \delta$,

$$\overline{\ell}(\overline{W}_n) + \frac{1}{n}\left(\sum_{t=1}\ell_t(W_t)\right) - \overline{\ell}(w^*) + \inf_{w\in H}\left(\sum_{t=1}^{n}\ell_t(w)\right) < \sqrt{\frac{2\ln(1/\delta)}{n}} \,.$$

Since w^* is the minimizer or $\overline{\ell}$, the last inequality is equivalent to the second part of the theorem. □

[2] If the minimizer does not exists, let w^* be such that $\overline{\ell}(w^*) < \inf_{w\in\mathcal{H}}\overline{\ell}(w) + \varepsilon$. Then take $\varepsilon \to 0$.

Chapter 15

Multi-Armed Bandits

Multi-armed bandit problem. Online learning problem. We do not see the losses for all the decisions.

Examples: Getting to school. Loss is travel time. Decisions: take the bus, bike, walk, drive. Also: Clinical trials. Ad-allocation problem. Recommendation system. Adaptive user interfaces.

Simple case: Decision space is finite.

Full-information setting, EWA, see Chapter 4.

We do not assume anything about D, Y and the loss function ℓ doesn't need to convex anymore. The only assumption that we make is that $\ell(p, y) \in [0, 1]$. Also note that the numbers $\widehat{p}_{1,t}, \widehat{p}_{2,t}, \ldots, \widehat{p}_{N,t}$ are non-negative and sum to 1 and therefore the distribution of I_t is valid probability distribution.

We have N actions.

Initially, $w_{i,0} = 1$ for each expert i and $W_0 = N$. Then, in each round $t = 1, 2, \ldots, n$, the algorithm does the following:

1. It receives experts' predictions $f_{1,t}, f_{2,t}, \ldots, f_{N,t} \in D$.
2. It calculates $\widehat{p}_{i,t} = w_{i,t-1}/W_{t-1}$, $i = 1, \ldots, N$.
3. It draws $I_t \in \{1, 2, \ldots, N\}$ *randomly* so that $\Pr[I_t = i] = \widehat{p}_{i,t}$ holds for $i = 1, \ldots, N$.
4. It predicts $f_{I_t,t}$.
5. The environment reveals the outcome $y_t \in Y$.
6. The algorithm suffers the loss $\ell(f_{I_t,t}, y_t)$ and each expert $i = 1, 2, \ldots, N$ suffers a loss $\ell(f_{i,t}, y_t)$.
7. The algorithm updates the weights: $w_{i,t} = w_{i,t-1} e^{-\eta \ell(f_{i,t}, y_t)}$.
8. The algorithm updates the sum of the weights: $W_t = \sum_{i=1}^{N} w_{i,t}$.

Since we do not have $\ell_{i,t} = \ell(f_{i,t}, y_t)$, for $i \neq I_t$, we come up with an estimate of it:

$$\widetilde{\ell}_{i,t} \stackrel{\text{def}}{=} \frac{\mathbb{I}\{I_t = i\}\,\ell_{i,t}}{p_{i,t}} = \begin{cases} \frac{\ell_{i,t}}{p_{i,t}}, & \text{if } I_t = i\,, \\ 0, & \text{otherwise}\,. \end{cases}$$

This estimate is constructed such that $\mathbf{E}\left[\widetilde{\ell}_{i,t}\right] = \ell_{i,t}$. Indeed, by the tower rule,

$$\mathbf{E}\left[\widetilde{\ell}_{i,t}\right] = \mathbf{E}\left[\mathbf{E}\left[\widetilde{\ell}_{i,t}|I_1,\dots,I_{t-1}\right]\right] = \mathbf{E}\left[\ell_{i,t}/p_{i,t}\;\mathbf{E}\left[\mathbb{I}\{I_t = i\}|I_1,\dots,I_{t-1}\right]\right] = \mathbf{E}\left[\ell_{i,t}/p_{i,t}\;p_{i,t}\right] = \ell_{i,t}.$$

15.1 Exp3-γ algorithm

Same as EWA, except that in Step 7 when we do the update, we use $\widetilde{\ell}_{i,t}$ instead of $\ell_{i,t}$:

$$w_{i,t} = w_{i,t-1} e^{-\eta \widetilde{\ell}_{i,t}}\,.$$

Exp3 stands for exponetial weights for exploration and exploitation. The ending, $-\gamma$ stands for not using exploration (wait for the next section to understand this).

Note: $p_{i,t}$ becomes random, whereas in EWA it was not random.

Regret bound for the expected regret.

Assume that $\ell_{i,t} \in [0,1]$.

Define $\widetilde{L}_{i,n} = \sum_{t=1}^{n} \widetilde{\ell}_{i,t}$.

Usual proof:

$$\frac{W_n}{W_0} = \frac{\sum_{i=1}^{N} e^{-\eta \widetilde{L}_{i,n}}}{N} \geq \frac{e^{-\eta \widetilde{L}_{i,n}}}{N}\,. \tag{15.1}$$

Now,

$$\frac{W_t}{W_{t-1}} = \sum_{i=1}^{N} \frac{w_{i,t}}{W_{t-1}} = \sum_{i=1}^{N} \frac{w_{i,t-1}}{W_{t-1}}\, e^{-\eta \widetilde{\ell}_{i,t}}\,.$$

Instead of using Hoeffding as in Chapter 4, we use that $e^x \leq 1 + x + x^2$ holds when $x \leq 1$ to get

$$\frac{W_t}{W_{t-1}} \leq \sum_{i=1}^{N} \frac{w_{i,t-1}}{W_{t-1}} \left\{1 - \eta \widetilde{\ell}_{i,t} + \eta^2 \widetilde{\ell}_{i,t}^2\right\} = 1 - \eta \sum_{i=1}^{N} p_{i,t} \widetilde{\ell}_{i,t} + \eta^2 \sum_{i=1}^{N} p_{i,t} \widetilde{\ell}_{i,t}^2\,.$$

Now, $\sum_{i=1}^{N} p_{i,t} \widetilde{\ell}_{i,t} = \ell_{I_t,t}$. This, and $1 + x \leq e^x$ (which holds for any $x \in \mathbb{R}$) gives $W_t/W_{t-1} \leq \exp(-\eta \ell_{I_t,t} + \eta^2 \sum_{i=1}^{N} p_{i,t} \widetilde{\ell}_{i,t}^2)$. Therefore,

$$\frac{W_n}{W_0} \leq \exp\left(-\sum_{t=1}^{n} \left\{\eta\, \ell_{I_t,t} + \eta^2 \sum_{i=1}^{N} p_{i,t}\, \widetilde{\ell}_{i,t}^2\right\}\right) \leq \exp\left(-\eta \widehat{L}_n + \eta^2 \sum_{t=1}^{n} \sum_{i=1}^{N} p_{i,t} \widetilde{\ell}_{i,t}^2\right)$$

The second term in the exponent is upper bounded as follows:

$$\sum_{i=1}^{N} p_{i,t}\,\widetilde{\ell}_{i,t}^{2} = \sum_{i=1}^{N} p_{i,t}\,\frac{\mathbb{I}\{I_t = i\}\ell_{i,t}}{p_{i,t}}\,\widetilde{\ell}_{i,t} = \sum_{i=1}^{N} \mathbb{I}\{I_t = i\}\ell_{i,t}\,\widetilde{\ell}_{i,t} \le \sum_{i=1}^{N} \widetilde{\ell}_{i,t}\,,$$

where in the last step we used $\ell_{i,t} \le 1$. Combining the previous inequality with this bound and (15.1) and taking logarithms of both sides we get

$$-\eta\widetilde{L}_{i,n} - \ln(N) \le -\eta\widehat{L}_n + \eta^2 \sum_{i=1}^{N} \widetilde{L}_{i,n}\,.$$

Now, by construction $\mathbf{E}\left[\widetilde{L}_{i,n}\right] = L_{i,n}$, therefore taking the expectation of both sides gives

$$-\eta L_{i,n} - \ln(N) \le -\eta\,\mathbf{E}\left[\widehat{L}_n\right] + \eta^2 \sum_{i=1}^{N} L_{i,n}\,.$$

Reordering gives

$$\mathbf{E}\left[\widehat{L}_n\right] - L_{i,n} \le \frac{\ln N}{\eta} + \eta \sum_{i=1}^{N} L_{i,n} \le \frac{\ln N}{\eta} + \eta n N \le 2\sqrt{nN\ln N}\,.$$

15.2 A high probability bound for the Exp3.P algorithm

We work with gains!

$$g_{i,t} = 1 - \ell_{i,t} \in [0,1], \quad \widetilde{g}_{i,t} = \frac{\mathbb{I}\{I_t = i\}g_{i,t}}{p_{i,t}}\,.$$

We also need

$$g'_{i,t} = \widetilde{g}_{i,t} + \frac{\beta}{p_{i,t}}, \quad \beta > 0.$$

Plus, we need exploration:

$$p_{i,t+1} = (1-\gamma)\frac{w_{i,t}}{W_t} + \frac{\gamma}{N}, \quad 0 < \gamma \le 1\,.$$

Because $0 < \gamma$, the algorithm explores with positive probability. The resulting algorithm is called Exp3.P, where P is added to the name to indicate that the algorithm is designed to work with high probability.

Introduce $G_{i,n}$, $G'_{i,n}$, $\widetilde{G}_{i,n}$.

Now,

$$\frac{W_n}{W_0} = \frac{\sum_{i=1}^N e^{\eta G'_{i,n}}}{N} \geq \frac{e^{\eta \max_{1\leq i\leq N} G'_{i,n}}}{N}\,. \tag{15.2}$$

Also,

$$\frac{W_t}{W_{t-1}} = \frac{\sum_{i=1}^N w_{i,t}}{W_{t-1}} = \sum_{i=1}^N \frac{w_{i,t-1}}{W_{t-1}}\, e^{\eta g'_{i,t}}\,.$$

By definition,

$$\frac{w_{i,t-1}}{W_{t-1}} = \frac{p_{i,t} - \frac{\gamma}{N}}{1-\gamma} \leq \frac{p_{i,t}}{1-\gamma}\,.$$

Assume

$$\eta g'_{i,t} \leq 1. \tag{15.3}$$

Then using $e^x \leq 1 + x + x^2$, which holds for $x \leq 1$, we get

$$\begin{aligned}
\frac{W_t}{W_{t-1}} &\leq \sum_{i=1}^N \frac{p_{i,t} - \frac{\gamma}{N}}{1-\gamma} \left\{1 + \eta g'_{i,t} + \eta^2 \left(g'_{i,t}\right)^2\right\} && \text{(because } \tfrac{w_{i,t-1}}{W_{t-1}} = \tfrac{p_{i,t}-\frac{\gamma}{N}}{1-\gamma}\text{)} \\
&\leq 1 + \frac{\eta}{1-\gamma} \sum_{i=1}^N p_{i,t}\, g'_{i,t} + \frac{\eta^2}{1-\gamma} \sum_{i=1}^N p_{i,t} \left(g'_{i,t}\right)^2 && \text{(because } p_{i,t} - \tfrac{\gamma}{N} \leq p_{i,t} \text{ and } g'_{i,t} \geq 0\text{)}
\end{aligned}$$

By the definition of $g'_{i,t}$, $p_{i,t}\, g'_{i,t} \leq p_{i,t}\widetilde{g}_{i,t} + \beta \leq \mathbb{I}\{I_t = i\} g_{i,t} + \beta$. This can be used to bound both the second and third term above. The third term is further upper bounded by $p_{i,t}\left(g'_{i,t}\right)^2 = (p_{i,t}\, g'_{i,t}) g'_{i,t} \leq (1+\beta) g'_{i,t}$. Therefore,

$$\begin{aligned}
\frac{W_t}{W_{t-1}} &\leq 1 + \frac{\eta}{1-\gamma}\left(g_{I_t,t} + N\beta\right) + \frac{\eta^2}{1-\gamma}\,(1+\beta) \sum_{i=1}^N g'_{i,t} \\
&\leq \exp\left(\frac{\eta}{1-\gamma}\left(g_{I_t,t} + N\beta\right) + \frac{\eta^2}{1-\gamma}\,(1+\beta) \sum_{i=1}^N g'_{i,t}\right)\,.
\end{aligned}$$

Combining this with (15.2) we get

$$\begin{aligned}
\frac{\exp(\eta \max_j G'_{j,n})}{N} &\leq \exp\left(\frac{\eta}{1-\gamma}\left(\widehat{G}_n + \beta n N\right) + \frac{\eta^2}{1-\gamma}\,(1+\beta) \sum_{i=1}^N G'_{i,t}\right) \\
&\leq \exp\left(\frac{\eta}{1-\gamma}\left(\widehat{G}_n + \beta n N\right) + \frac{\eta^2}{1-\gamma}\,(1+\beta)\, N \max_{1\leq i\leq N} G'_{i,t}\right)\,.
\end{aligned}$$

Taking the logarithm of both sides gives

$$\eta \max_j G'_{j,n} - \ln N \leq \frac{\eta}{1-\gamma}\,\widehat{G}_n + \frac{\eta\beta n N}{1-\gamma} + \frac{\eta^2}{1-\gamma}(1+\beta) N \max_j G'_{j,n}\,.$$

Reordering gives

$$\widehat{G}_n \geq [1-\gamma-\eta(1+\beta)N] \max_j G'_{j,n} - \frac{\ln N}{\eta} - nN\beta\,.$$

Goal: $G'_{i,n} > G_{i,n} - \beta nN$ with high probability.

Lemma 15.1. *Let $0 \leq \delta \leq 1$. Assume that $\sqrt{\frac{\ln(N/\delta)}{nN}} \leq \beta \leq 1$. Then with probability at least $1-\delta/N$,*

$$G'_{i,n} \geq G_{i,n} - \beta nN.$$

Proof. We want to prove that $\Pr\left(G_{i,n} > G'_{i,n} + \beta nN\right) \leq \delta/N$. We have

$$\begin{aligned}
\Pr\left(G_{i,n} > G'_{i,n} + \beta nN\right) &= \Pr\left(G_{i,n} - G'_{i,n} > \beta nN\right) = \Pr\left(\beta(G_{i,n} - G'_{i,n}) > \beta^2 nN\right) \\
&= \Pr\left(\exp(\beta(G_{i,n} - G'_{i,n})) > \exp(\beta^2 nN)\right) \\
&\leq \mathbf{E}\left[\exp(\beta(G_{i,n} - G'_{i,n}))\right] \exp(-\beta^2 nN)\,,
\end{aligned}$$

where in the last step we have used Markov's inequality which says that if $X \geq 0$ then $\Pr(X \geq a) \leq \mathbf{E}[X]/a$. Define $Z_t = \exp(\beta(G_{i,t} - G'_{i,t}))$. It suffices to prove that $\mathbf{E}[Z_n] \leq 1$. Let $I_{1:t} = (I_1, \ldots, I_t)$. Then, $\mathbf{E}[Z_n] = \mathbf{E}[\mathbf{E}[Z_n | I_{1:n-1}]]$. Now, $Z_n = Z_{n-1}\exp(\beta(g_{i,n} - g'_{i,n}))$ and since Z_{n-1} is a function of $I_{1:n-1}$, $\mathbf{E}[Z_n | I_{1:n-1}] = Z_{n-1}\,\mathbf{E}\left[\exp(\beta(g_{i,n} - g'_{i,n}))|I_{1:n-1}\right]$. Using the definition of $g'_{i,n}$, we get that

$$\mathbf{E}\left[\,\exp(\beta(g_{i,n} - g'_{i,n}))\,|\,I_{1:n-1}\,\right] = \exp(-\frac{\beta^2}{p_{i,n}})\;\mathbf{E}\left[\,\exp(\beta(g_{i,n} - \widetilde{g}_{i,n}))\,|\,I_{1:n-1}\,\right]\,.$$

By assumption, $\beta(g_{i,n} - \widetilde{g}_{i,n}) \leq 1$, since by assumption $0 \leq \beta \leq 1$. Therefore, we can use $e^x \leq 1 + x + x^2$. Using $\mathbf{E}[g_{i,n} - \widetilde{g}_{i,n}\,|\,I_{1:n-1}] = 0$, which holds by the construction of $\widetilde{g}_{i,n}$, and $\mathbf{E}[(g_{i,n} - \widetilde{g}_{i,n})^2\,|\,I_{1:n-1}] \leq \mathbf{E}\left[\widetilde{g}^2_{i,n}\,|\,I_{1:n-1}\right]$ (because $\mathrm{Var}(X) \leq \mathbf{E}[X^2]$), and $\mathbf{E}\left[\widetilde{g}^2_{i,n}\,|\,I_{1:n-1}\right] = g^2_{i,n}/p_{i,n} \leq 1/p_{i,n}$ we have

$$\begin{aligned}
\mathbf{E}\left[\,\exp(\beta(g_{i,n} - g'_{i,n}))\,|\,I_{1:n-1}\,\right] &\leq \exp(-\frac{\beta^2}{p_{i,n}})\;\left(1 + \frac{\beta^2}{p_{i,n}}\right) \\
&\leq \exp(-\frac{\beta^2}{p_{i,n}} + \frac{\beta^2}{p_{i,n}}) \qquad\qquad \text{because } 1 + x \leq e^x \\
&= 1\,.
\end{aligned}$$

Thus, $\mathbf{E}[Z_n | I_{1:n-1}] \leq \mathbf{E}[Z_{n-1}]$. Now, taking expectation of both sides and repeating the above argument, we get that $\mathbf{E}[Z_n] \leq \mathbf{E}[Z_{n-1}] \leq \mathbf{E}[Z_{n-2}] \leq \ldots \leq \mathbf{E}[Z_0] = 1$, finishing the proof of the Lemma. □

So, let's assume that

$$\begin{aligned}
1 - \gamma - \eta(1+\beta)N &\geq 0\,, \\
\sqrt{\frac{\ln(N/\delta)}{\beta}} &\leq \beta \leq 1\,.
\end{aligned}$$

By the union bound, with probability at least $1-\delta$, $\max_j G'_{j,n} \geq \max_j G_{j,n} - \beta nN$. Hence,

$$\widehat{G}_n \geq [1-\gamma-\eta(1+\beta)N] \left(\max_j G_{j,n} - \beta nN \right) - \frac{\ln N}{\eta} - nN\beta \,.$$

Reordering,

$$\begin{aligned} \max_j G_{j,n} - \widehat{G}_{j,n} &\leq (\gamma+\eta(1+\beta)N) \max_j G_{j,n} + (2-\gamma-\eta(1+\beta)N)\beta nN + \frac{\ln N}{\eta} \\ &\leq (\gamma+\eta(1+\beta)N)\, n + 2\beta nN + \frac{\ln N}{\eta} \,. \end{aligned}$$

Now since the bound is increasing in β, choose β to be its lower bound. Next, choose $\gamma = 2\eta N$ etc.

We get the bound

$$C\sqrt{nN\,\ln(N/\delta)}\,,$$

which holds w.p. $1-\delta$ for any fixed n big enough, where C is a fixed numerical constant.

Note: This is a non-uniform algorithm! You choose δ then you choose the parameters! This is unlike the previous result!

Chapter 16

Lower Bounds for Bandits

The upper bound on the expected regret of EXP3-γ is $O(\sqrt{nN \ln N})$. In fact, the newer INF algorithm enjoys a minimax upper bound of size $O(\sqrt{nN})$. Is there an algorithm which enjoys a smaller minimax upper bound? To answer this question, we need to study *minimax lower bounds* for this problem.

The gains will be randomized as usual in a lower bound proof. Let $g_{i,n}$ be an i.i.d. sequence taking values in $[0, 1]$. We will develop a bound on

$$\bar{R}_n = \max_{1 \le j \le N} \mathbf{E}\left[G_{j,n}\right] - \mathbf{E}\left[\widehat{G}_n\right] .$$

Then $\mathbf{E}\left[R_n\right] \ge \bar{R}_n$, therefore it suffices to develop a lower bound on $\bar{R}_n$ (thanks to the randomization device).

Note that $\mathbf{E}\left[G_{i,n}\right] = n\,\mathbf{E}\left[g_{i,n}\right]$. Let

$$\Delta_i = \max_j \mathbf{E}\left[g_{j,t}\right] - \mathbf{E}\left[g_{i,t}\right] .$$

Then, by Wald's identity

$$\bar{R}_n = \sum_{i=1}^{N} \mathbf{E}\left[T_{i,n}\right] \Delta_i,$$

where $T_{i,n} = \sum_{t=1}^{n} \mathbb{I}\{I_t = i\}$ is the number of times arm i is pulled up to time n.

By von Neumann's minimax theorem, it suffices to consider deterministic algorithms. An algorithm A will be a function from histories to the space of actions. Let $I_t = A(g_{I_1,1}, \ldots, g_{I_{t-1},t-1})$..
.

Let $h_t = g_{I_t,t}$ and $h_{1:t} = (h_1, \ldots, h_{t-1})$. With this notation, $I_t = A(h_{1:t})$. (Note that $h_{1:0}$ is the empty history.)

Plan for the lower bound:

(Step 1) Choose joint distributions Π_k for $(g_{i,t})$ whose relative distance from each other is controlled ($k = 1, \ldots, N$).

(Step 2) Show that if two joint distributions are close then the behavior of the algorithm does not change much.

(Step 3) Derive the lower bound from this.

Fix $1 \le i \le N$. Let Π_i be the joint distribution, where the payoff distributions for all the arms is a Bernoulli $1/2$ distribution, except for arm i, whose payoff has Bernoulli $1/2+\varepsilon$ distribution, where $0 < \varepsilon < 1/2$ will be chosen later. Let $g_{j,t}^{(i)} \sim \textsc{Ber}(1/2 + \mathbb{I}\{i=j\}\varepsilon)$ be an i.i.d payoff sequence generated from Π_i. All of these are independent.

We further need Π_0: In Π_0 all payoffs have Bernoulli $1/2$ distributions.

Notation: $\mathbf{P}_i$, $\mathbf{E}_i[\cdot]$, $\mathbf{p}_i$ the PMF. Let $\bar{R}_j$ be $\bar{R}$ when we are in the j^{th} world.

$\bar{R}_j = (n - \mathbf{E}_j[n_j])\varepsilon$.

Notice that $\mathbf{p}_i(h_{1:n-1})$ is completely dependent on A. Step 2.

$$\begin{aligned}
\mathbf{E}_i[n_i] - \mathbf{E}_0[n_i] &= \sum_{h_{1:n-1}\in\{0,1\}^{n-1}} \mathbf{p}_i(h_{1:n-1}) \sum_{t=1}^{n} \mathbb{I}\{A(h_{1:t-1}) = i\} \\
&\qquad - \sum_{h_{1:n-1}\in\{0,1\}^{n-1}} \mathbf{p}_0(h_{1:n-1}) \sum_{t=1}^{n} \mathbb{I}\{A(h_{1:t-1}) = i\} \\
&\le n \sum_{h_{1:n-1}\in\{0,1\}^{n-1}} (\mathbf{p}_i(h_{1:n-1}) - \mathbf{p}_0(h_{1:n-1}))_+ \\
&= \frac{n}{2} \left\| \mathbf{p}_i(h_{1:n-1}) - \mathbf{p}_0(h_{1:n-1}) \right\|_1 \\
&\le \frac{n}{2} \sqrt{2\mathrm{KL}\left(\mathbf{p}_0(h_{1:n-1}) \| \mathbf{p}_i(h_{1:n-1})\right)},
\end{aligned}$$

where the last step used Pinsker's inequality.

Conditional KL divergence:

$$\mathrm{KL}\left(p(X|Y)\|q(X|Y)\right) = \sum_{x,y} p(x,y) \log \frac{p(x|y)}{q(x|y)} \quad \Big(= \mathbf{E}\left[\mathrm{KL}\left(p(X|y)\|q(X|y)\right)\right] \Big).$$

Chain rule:

$$\mathrm{KL}\left(p(x,y)\|q(x,y)\right) = \mathrm{KL}\left(p(x)\|q(x)\right) + \mathrm{KL}\left(p(y|x)\|q(y|x)\right).$$

By the chain rule,

$$\begin{aligned}
\mathrm{KL}\left(\mathbf{p}_0(h_{1:n-1})\|\mathbf{p}_i(h_{1:n-1})\right) &= \sum_{t=1}^{n} \mathrm{KL}\left(\mathbf{p}_0(h_t|h_{1:t-1})\|\mathbf{p}_i(h_t|h_{1:t-1})\right) \\
&= \sum_{t=1}^{n} \sum_{h_{1:t}} \mathbf{p}_0(h_{1:t}) \log \frac{\mathbf{p}_0(h_t|h_{1:t-1})}{\mathbf{p}_i(h_t|h_{1:t-1})}.
\end{aligned}$$

Now, $\mathbf{p}_0(h_{1:t}) = \mathbf{p}_0(h_{1:t-1})\mathbf{p}_0(h_t|h_{1:t-1})$. Therefore,

$$\mathrm{KL}\left(\mathbf{p}_0(h_{1:n-1})\|\mathbf{p}_i(h_{1:n-1})\right) = \sum_{t=1}^{n} \sum_{h_{1:t}} \mathbf{p}_0(h_{1:t-1}) \sum_{h_t} \mathbf{p}_0(h_t|h_{1:t-1}) \log \frac{\mathbf{p}_0(h_t|h_{1:t-1})}{\mathbf{p}_i(h_t|h_{1:t-1})}.$$

By the choice of Π_0, the innermost sum is

$$\frac{1}{2}\log\frac{1/2}{\mathbf{p}_i(0|h_{1:t-1})}+\frac{1}{2}\log\frac{1/2}{\mathbf{p}_i(1|h_{1:t-1})}\,.$$

If the algorithm does *not* choose arm i given the history $h_{1:t-1}$ then $\mathbf{p}_i(b|h_{1:t-1}) = 1/2$, making this expression zero ($b \in \{0,1\}$). In the other case, this sum is $\frac{1}{2}\log\frac{1}{4(1/2-\varepsilon)(1/2+\varepsilon))} = \frac{1}{2}\log\frac{1}{1-4\varepsilon^2}$. Therefore,

$$\begin{aligned}
\mathrm{KL}\left(\mathbf{p}_0(h_{1:n-1})\|\mathbf{p}_i(h_{1:n-1})\right) &= \frac{1}{2}\,\log\left(\frac{1}{1-4\varepsilon^2}\right)\,\sum_{t=1}^{n}\sum_{h_{1:t}}\mathbf{p}_0(h_{1:t-1})\mathbb{I}\{A(h_{1:t-1})=i\}\\
&= \frac{1}{2}\,\log\left(\frac{1}{1-4\varepsilon^2}\right)\,\mathbf{E}_0\left[n_i\right]\\
&\le 4\varepsilon^2\,\mathbf{E}_0\left[n_i\right]\,,
\end{aligned}$$

where the last inequality holds thanks to $-\log(1-x) \le 2x$ which holds when $0 \le x \le 1/2$.

We summarize what we have achieved in the following lemma.

Lemma 16.1.

$$\mathbf{E}_i\left[n_i\right] - \mathbf{E}_0\left[n_i\right] \le \sqrt{2}\varepsilon n\sqrt{\mathbf{E}_0\left[n_i\right]}\,.$$

Back to the main proof. We have

$$\bar{R}_i = \varepsilon(n - \mathbf{E}_i\left[n_i\right]) \ge \varepsilon(n - \sqrt{2}\varepsilon n\sqrt{\mathbf{E}_0\left[n_i\right]} - \mathbf{E}_0\left[n_i\right])\,.$$

Using the randomization hammer,

$$\begin{aligned}
\bar{R} &\ge 1/N\sum_{i=1}^{n}\bar{R}_i\\
&\ge 1/N\sum_{i=1}^{N}\varepsilon(n - \sqrt{2}\varepsilon n\sqrt{\mathbf{E}_0\left[n_i\right]} - \mathbf{E}_0\left[n_i\right])\\
&\ge 1/N\varepsilon(nN - \sqrt{2}\varepsilon n\sum_{i=1}^{N}\sqrt{\mathbf{E}_0\left[n_i\right]} - n)\\
&\ge 1/N\varepsilon(nN - \sqrt{2}\varepsilon n\sqrt{N\sum_{i=1}^{N}\mathbf{E}_0\left[n_i\right]} - n)\\
&= 1/N\varepsilon(nN - \sqrt{2}\varepsilon n\sqrt{Nn} - n)\\
&\ge \varepsilon n - \sqrt{2}\varepsilon^2 N^{-1/2}n^{3/2} - \frac{n}{N}\varepsilon.
\end{aligned}$$

Choose $\varepsilon = 1/2\sqrt{N/(2n)}$, we get $c\sqrt{nN}$.

Theorem 16.2. *There exists a constant $c > 0$ such that for any $N, n \ge 1$ and any algorithm A, there exists a sequence of rewards in $[0,1]$ such that the regret R_n of algorithm A on this sequence of rewards satisfies*

$$\mathbf{E}\left[R_n\right] \ge c\sqrt{nN} \cap n\,.$$

Chapter 17

Exp3-γ as FTRL

The *prediction with expert advice problem* can be casted as a prediction problem over the simplex with linear losses as follows (see also Exercise 8.1): The decision set K_0 available to the online learner is the set of unit vectors $\{e_1, \dots, e_d\}$ of the d-dimensional Euclidean space, while the loss function in round t is $\ell_t(w) = f_t^\top w$. Choosing expert i is identified with choosing unit vector e_i and if the loss assigned to expert i in round t is $\ell_{t,i}$, we set $f_{t,i} = \ell_{t,i}$. This way the two games become "isomorphic".[1] From now on we will thus work with the vectorial representation.

In general, the choice $\widehat{w}_t \in K_0$ made in round t is random. Assume that e_i is selected in round t with probability $w_{t,i}$, where $w_t = (w_{t,1}, \dots, w_{t,d})^\top \in \Delta_d$ and w_t is computed based on past information. Due to the linearity of the losses $\mathbf{E}\left[\ell_t(\widehat{w}_t)\right] = \mathbf{E}\left[\ell_t(w_t)\right]$. Thus, the real issue is to select the vectors $w_t \in \Delta_d$ (see also Exercise 17.1).

Our main interest in this chapter is the *bandit setting*, where the learner receives only $\ell_t(\widehat{w}_t)$ after round t, as opposed to the *full-information setting*, when the learner is told the whole function $\ell_t(\cdot)$. The goal is still to keep the regret,

$$\widehat{L}_n - L_n(u) = \sum_{t=1}^{n} \ell_t(\widehat{w}_t) - \sum_{t=1}^{n} \ell_t(u)$$

small. More precisely, in this chapter we focus on the expected regret only.

17.1 Black-box Use of Full-information Algorithms

We saw that good (low regret) algorithms exist for the full-information setting. A simple idea then is to reuse a low-regret algorithm developed for the full-information case as a black-box (see Figure 17.1 for an illustration of this idea). Since a full-information algorithm needs the vector f_t, we assume that an estimate of f_t is constructed in some way, which is fed to

[1] By the two games being "isomorphic" we mean that there is a one-to-one correspondence between the strategies in the two games such that the correspondence leaves the losses incurred invariant.

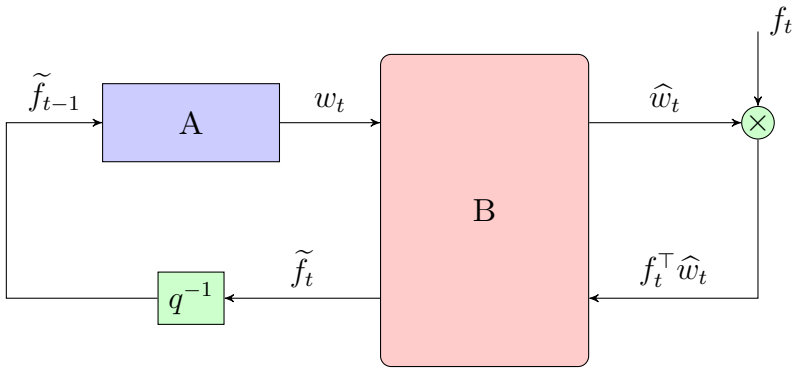

Figure 17.1: Illustration of how the full-information algorithm A is used as a black-box. Block B (to be designed) feeds A with $\widetilde{f}_t$, an estimate of f_t. In addition, it might introduce additional randomization to facilitate the estimation of f_t. The block marked by q^{-1} indicates a time-delay.

the full-information algorithm. Let $\widetilde{f}_t$ be the estimate of f_t developed in round t. Define $\widehat{\widetilde{L}}_n = \sum_{t=1}^n \widetilde{\ell}_t(w_t)$ and $\widetilde{L}_n(u) = \sum_{t=1}^n \widetilde{\ell}_t(u)$, where $\widetilde{\ell}_t(w) = \widetilde{f}_t^\top w$.

The following result shows that if $\widetilde{f}_t$ is an appropriately constructed estimate of f_t, it suffices to study the regret of the full-information algorithm fed with the sequence of losses $\widetilde{\ell}_t$:

Proposition 17.1. *Assume that* $\mathbf{E}\left[\widehat{w}_t \,|\, \widehat{w}_1, \dots, \widehat{w}_{t-1}\right] = w_t$ *and*

$$\mathbf{E}\left[\widetilde{f}_t \,|\, \widehat{w}_1, \dots, \widehat{w}_{t-1}\right] = f_t\,, \tag{17.1}$$

i.e., given all past information, $\widetilde{f}_t$ *is an unbiased estimate of* f_t*. Then,*

$$\mathbf{E}\left[\widehat{\widetilde{L}}_n\right] = \mathbf{E}\left[\widehat{L}_n\right], \quad \text{and} \quad \mathbf{E}\left[\widetilde{L}_n(u)\right] = L_n(u)\,.$$

The proof of the proposition is left as an exercise (cf. Exercise 17.2).

17.2 Analysis of Exp3-γ

Consider the FTRL algorithm with the un-normalized negative entropy regularizer $R(w) = \sum_{i=1}^d w_i \ln w_i - w_i$, $w \in \mathbb{R}_+^d$. Remember that R is strongly convex w.r.t. the 1-norm on the open probability simplex (cf. Exercise 8.1, Part d). Therefore, by Theorem 8.14,

$$\widehat{\widetilde{L}}_n - \widetilde{L}_n(u) \le \eta \sum_{t=1}^n \|\widetilde{f}_t\|_\infty^2 + \frac{\ln d}{\eta}\,. \tag{17.2}$$

Thus, we see that a small expected regret can be achieved as long as $\mathbf{E}\left[\sum_{t=1}^{n} \|\widetilde{f}_t\|_\infty^2\right]$ is small.

Consider, for example, the importance weighted estimator

$$\widetilde{f}_{t,i} = \frac{\mathbb{I}\{\widehat{w}_t = e_i\} f_{t,i}}{w_{t,i}}\,, \tag{17.3}$$

which makes the algorithm identical to the Exp3-γ algorithm of Section 15.1. By construction (and as it was also shown earlier) $\widetilde{f}_t$ satisfies (17.1). Therefore, it remains to study $\mathbf{E}\left[\|\widetilde{f}_t\|_\infty^2\right]$. Introduce the shorthand notation $\mathbf{E}_t[\cdot] = \mathbf{E}[\cdot \,|\, \widehat{w}_1, \dots, \widehat{w}_{t-1}]$. We have

$$\mathbf{E}_t\left[\|\widetilde{f}_t\|_\infty^2\right] \le \mathbf{E}_t\left[\|\widetilde{f}_t\|_2^2\right] = \sum_{i=1}^{d} \mathbf{E}_t\left[\widetilde{f}_{t,i}^2\right]\,.$$

Since

$$\mathbf{E}_t\left[\widetilde{f}_{t,i}^2\right] = \frac{f_{t,i}^2}{w_{t,i}^2}\,\mathbf{E}_t\left[\mathbb{I}\{\widehat{w}_t = e_i\}\right]$$

and $\mathbf{E}_t\left[\mathbb{I}\{\widehat{w}_t = e_i\}\right] = \Pr\left(\widehat{w}_t = e_i \,|\, \widehat{w}_1, \dots, \widehat{w}_{t-1}\right) = w_{t,i}$, we get $\mathbf{E}_t\left[\widetilde{f}_{t,i}^2\right] = \frac{f_{t,i}^2}{w_{t,i}}$. Thus,

$$\mathbf{E}_t\left[\|\widetilde{f}_t\|_\infty^2\right] \le \sum_{i=1}^{d} \frac{f_{t,i}^2}{w_{t,i}}\,. \tag{17.4}$$

Unfortunately, this expression is hard to control since the weights can become arbitrarily close to zero. One possibility then is to bias $\widehat{w}_t$ so that the probability of choosing $\widehat{w}_t = e_i$ is lower bounded. This is explored in Exercises 17.4 and 17.5, though the rates we can obtain with this technique (alone) are suboptimal. However, as we already saw earlier, Exp3-γ enjoys a low regret without any adjustment. This motivates us to refine the above analysis.

17.2.1 Local Norms

First, remember that in the studied case FTRL and PPA coincide (cf. Exercise 9.3, Part (c)). Therefore, we can use Lemma 9.2, which was stated for PPA and linear losses, to bound the regret:

$$\widehat{\widetilde{L}}_n - \widetilde{L}_n(u) \le \frac{\ln d}{\eta} + \sum_{t=1}^{n} \langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle\,. \tag{17.5}$$

Here $\widetilde{w}_{t+1} = \operatorname{argmin}_{w \in \mathbb{R}^d_{++}} \left[\eta\, \widetilde{\ell}_t(w) + D_R(w, w_t)\right]$. Now, the idea is to choose a pair of dual norms $\|\cdot\|_t$, $\|\cdot\|_{t,*}$ to upper bound the inner products in the second term as follows:

$$\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle \le \|\widetilde{f}_t\|_{t,*}\, \|w_t - \widetilde{w}_{t+1}\|_t\,.$$

These norms are called "local" because they will be chosen based on local information so that both of the terms on the right-hand side are tightly controlled.

A simple idea is to try some weighted norms. As it turns out $\|v\|_{t,*}^2 = \sum_{i=1}^d w_{t,i}\, v_i^2$ is a good choice.[2] How can we bound $\|w_t - \widetilde{w}_{t+1}\|_t$? The dual of $\|\cdot\|_{t,*}$ is $\|v\|_t^2 = \sum_{i=1}^d w_{t,i}^{-1} v_i^2$. Hence, using $\widetilde{w}_{t+1,i} = w_{t,i} \exp(-\eta \widetilde{f}_{t,i})$, we get

$$\begin{aligned}\|w_t - \widetilde{w}_{t+1}\|_t^2 &= \sum_{i=1}^d w_{t,i}^{-1}(w_{t,i} - w_{t,i}e^{-\eta\widetilde{f}_{t,i}})^2 = \sum_{i=1}^d w_{t,i}(1 - e^{-\eta\widetilde{f}_{t,i}})^2 \\ &\le \eta^2 \sum_{i=1}^d w_{t,i}\widetilde{f}_{t,i}^2 \;=\; \eta^2\, \|\widetilde{f}_t\|_{t,*}^2\,.\end{aligned}$$

In the inequality, we used that $1 - e^{-x} \le x$ (this holds for any $x \in \mathbb{R}$) and thus for any $x \ge 0$, $(1 - e^{-x})^2 \le x^2$ holds true. Therefore,

$$\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1}\rangle \le \eta \|\widetilde{f}_t\|_{t,*}^2 \;=\; \eta \sum_{i=1}^d w_{t,i}\widetilde{f}_{t,i}^2\,. \tag{17.6}$$

By the choice of the estimator, $\mathbf{E}_t\left[\, \|\widetilde{f}_t\|_{t,*}^2 \,\right] \le \|f_t\|^2$, and thus,

$$\mathbf{E}_t\left[\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1}\rangle\right] \le \eta \|f_t\|^2\,.$$

When the losses are in bounded by 1, $\|f_t\|^2 \le d$. Combining these with (17.5), we get the following theorem:

Theorem 17.2. *Assume that the losses are in the $[0,1]$ interval and that Exp3-γ is run with $\eta = \sqrt{\frac{\ln d}{nd}}$. Then, its expected regret is bounded by $2\sqrt{nd \ln d}$.*

Remark 17.3 (Coincidence?)**.** Notice that $\nabla^2 R(w) = \mathrm{diag}(w_1^{-1}, \ldots, w_d^{-1})$. Hence, $\|\cdot\|_t = \|\cdot\|_{\nabla^2 R(w_t)}$. This raises the question if, for other choices of R, it is beneficial to use the local norm $\|\cdot\|_{\nabla^2 R(w_t)}$ in the analysis of the Linearized Proximal Point algorithm. In the case when R satisfies certain additional properties, this has been investigated in a series of papers published at COLT by Abernethy, Hazan, and Rakhlin.

[2] Note that the notation $\|v\|_t$ clashes with our earlier notation of ℓ^p norms, which is unfortunate. However, we trust that the reader can figure out the meaning of which norm is intended based on the semantics of the norm-index.

17.3 Avoiding local norms

Starting from (17.5), we can arrive at the same bound as before, while avoiding local norms completely. This can be done as follows. Let us upper bound the inner product $\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle$:

$$\begin{aligned} \langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle &= \sum_{i=1}^{d} \widetilde{f}_{t,i} w_{t,i} (1 - e^{-\eta \widetilde{f}_{t,i}}) \\ &\le \eta \sum_{i=1}^{d} \widetilde{f}_{t,i}^2 w_{t,i} \\ &= \sum_{i=1}^{d} f_{t,i}^2 \frac{\mathbb{I}\{\widehat{w}_t = e_i\}}{w_{t,i}} \,, \end{aligned} \tag{17.7}$$

where the inequality follows because $\widetilde{f}_{t,i}, w_{t,i} \ge 0$ and $1 - e^{-x} \le x$ holds for any $x \in \mathbb{R}$. Therefore,

$$\mathbf{E}_t \left[\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle \right] \le \eta \sum_{i=1}^{d} f_{t,i}^2 = \eta \|f_t\|^2 \,.$$

17.3.1 Relaxing nonnegativity of losses

A crucial assumption of the previous argument was that $\widetilde{\ell}_{t,i} \ge 0$. The nonnegativity was used to derive (17.7). Therefore, we replace this step. First, notice that for $x \ge -1$, $x(1 - e^{-x}) \le cx^2$, where $c = e - 1 \approx 1.72 \le 2$. Therefore, assuming $\eta \widetilde{f}_{t,i} \ge -1$, we get

$$\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle = \sum_{i=1}^{d} \widetilde{f}_{t,i} w_{t,i} (1 - e^{-\eta \widetilde{f}_{t,i}}) \le c\eta \sum_{i=1}^{d} \widetilde{f}_{t,i}^2 w_{t,i} \,, \tag{17.8}$$

hence,

$$\mathbf{E}_t \left[\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle \right] \le c\eta \|f_t\|^2 \,.$$

The condition that $\eta \tilde{f}_{t,i} \ge -1$ (or the stronger condition that $\eta |\tilde{f}_{t,i}| \le 1$) can be achieved for example by adding exploration. When this condition holds, continuing as before we can bound the expected regret. See Exercise 17.5 for the details.

17.3.2 An alternative method

Yet another alternative elementary argument is as follows. We upper bound the inner product $\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle$ using Cauchy-Schwartz:

$$\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle \le \|\widetilde{f}_t\|_2 \, \|w_t - \widetilde{w}_{t+1}\|_2 \,.$$

By the same argument as in the Section 17.2.1, assuming $\widetilde{f}_t \geq 0$, we also have

$$\|w_t - \widetilde{w}_{t+1}\|_2^2 \leq \eta^2 \sum_{j=1}^{d} w_{t,j}^2 \widetilde{f}_{t,j}^2 \,.$$

Now, observe that $\widetilde{f}_{t,i}\widetilde{f}_{t,j} = 0$ if $i \neq j$. Hence,

$$\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle^2 \leq \eta^2 \left(\sum_{i=1}^{d} \widetilde{f}_{t,i}^2 \right) \left(\sum_{j=1}^{d} w_{t,j}^2 \widetilde{f}_{t,j}^2 \right) = \eta^2 \sum_{i=1}^{d} w_{t,i}^2 \widetilde{f}_{t,i}^4 \,.$$

Now, take the square root of both sides and use $\sqrt{\sum_i |x_i|} \leq \sum_i \sqrt{|x_i|}$ to get

$$\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle \leq \eta \sum_{i=1}^{d} w_{t,i} \widetilde{f}_{t,i}^2 = \eta \sum_{i=1}^{d} f_{t,i}^2 \frac{\mathbb{I}\{\widehat{w}_t = e_i\}}{w_{t,i}}$$

and thus it holds that

$$\mathbf{E}_t \left[\langle \widetilde{f}_t, w_t - \widetilde{w}_{t+1} \rangle \right] \leq \eta \sum_{i=1}^{d} f_{t,i}^2 \,.$$

To reiterate, the key ideas were that $\widetilde{f}_{i,t}\widetilde{f}_{j,t}$ is zero very often and also not to take expectation until one takes the square root, since the lower power $\widetilde{f}_{i,t}$ has before taking expectation, the better the bound will be.

17.4 Exercises

Exercise 17.1. (Randomization over the simplex) Consider a problem when $K = \Delta_d$. Let w_t be the sequence of choices of algorithm A against a sequence of *linear* loss functions $\ell_1, \ldots, \ell_n$, where $\ell_t : K \to [0,1]$. Consider the randomizing algorithm which, in round t, chooses $\widehat{w}_t \in \{e_1, \ldots, e_d\}$ with probability $w_{t,i}$ $(1 \leq i \leq d)$.

(a) Show that for any $u \in K$, the expected regret of the randomizing algorithm equals to the regret of algorithm A.

(b) Show that for any $u \in K$, with high probability, the regret of the randomizing algorithm is not much larger than that of algorithm A.

Exercise 17.2. Prove Proposition 17.1.
Hint: Show that $\mathbf{E}\left[f_t^\top \widehat{w}_t | \widehat{w}_1, \ldots, \widehat{w}_{t-1}\right] = f_t^\top w_t = \mathbf{E}\left[\widetilde{f}_t^\top w_t | \widehat{w}_1, \ldots, \widehat{w}_{t-1}\right]$ and apply the tower rule.

Exercise 17.3. (Biased estimates) Let $\widehat{w}_t$, w_t, f_t, $\widetilde{f}_t$ be as in Proposition 17.1. Prove the following extension of Proposition 17.1: Let

$$b_t = \mathbf{E}\left[\widetilde{f}_t | \widehat{w}_1, \dots, \widehat{w}_{t-1}\right] - f_t$$

be the *bias* of $\widetilde{f}_t$ at time t. Then, for any $u \in K$,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le \mathbf{E}\left[\widehat{\widetilde{L}}_n - \widetilde{L}_n(u)\right] + \mathbf{E}\left[\sum_{t=1}^{n} \sup_{v,w \in K} b_t^\top (v - w)\right] .$$

Exercise 17.4. (Biased estimates, biased predictions) Consider the same setting as in Exercise 17.3, except that now allow a bias in $\widehat{w}_t$, too. In particular, let

$$d_t = \mathbf{E}\left[\widehat{w}_t | \widehat{w}_1, \dots, \widehat{w}_{t-1}\right] - w_t$$

be the bias of $\widehat{w}_t$. As before, define

$$b_t = \mathbf{E}\left[\widetilde{f}_t | \widetilde{w}_1, \dots, \widetilde{w}_{t-1}\right] - f_t$$

to be the bias of $\widetilde{f}_t$ at time t, Remember that by assumption for any $t \in \{1, \dots, n\}$, $w \in K$, it holds that $\ell_t(w) \in [0, 1]$.

Show that for any $u \in K$,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le \mathbf{E}\left[\widehat{\widetilde{L}}_n - \widetilde{L}_n(u)\right] + \mathbf{E}\left[\sum_{t=1}^{n} \sup_{w \in K} b_t^\top w\right] + \mathbf{E}\left[\sum_{t=1}^{n} \sup_{f \in F} d_t^\top f\right] ,$$

where $F = \{f \in \mathbb{R}^d : 0 \le \inf_{w \in K} f^\top w \le \sup_{w \in K} f^\top w \le 1\}$.

Exercise 17.5. Consider the algorithm where FTRL with the un-normalized negentropy is fed with the estimates

$$\widetilde{f}_t = \frac{\mathbb{I}\{\widehat{w}_t = e_i\} f_{t,i}}{w_{t,i}(\gamma)} .$$

where $\widehat{w}_t$ is a randomly chosen from $\{e_1, \dots, e_d\}$ and the probability of $\widehat{w}_t = e_i$ is $w_{t,i}(\gamma) \stackrel{\text{def}}{=} \gamma/d + (1 - \gamma) w_{t,i}$, where $0 < \gamma < 1$ is an exploration parameter. Assume that $|f_{t,i}| \le 1$.

(a) Using the result of the previous exercise, combined with (17.2) and (17.4) (applied appropriately), show that the expected regret of the resulting algorithm is not larger than

$$\gamma n + d^2 \frac{\eta}{\gamma} + \frac{\ln d}{\eta} .$$

(b) Show that by appropriately choosing γ and η, the regret is bounded by $2 d^{2/3} n^{2/3} (4 \ln d)^{1/3}$. (This is thus a suboptimal bound.)

(c)

Exercise 17.6. (Estimating the gradient – linear losses)
Consider an online learning problem where the losses are deterministic, linear functions: $\ell_t(w) = f_t^\top w$, $t = 1, \dots, n$. Assume that $\ell_t(w) \in [0,1]$ and that the decision region is a convex subset K of $\mathbb{R}^d$. Consider a sequence of random choices $\widehat{w}_1, \dots, \widehat{w}_n \in K$. Introduce $\mathbf{E}_t[\cdot \,|\, \widehat{w}_1, \dots, \widehat{w}_{t-1}]$. Let $C_t = \mathbf{E}_t\left[\widehat{w}_t \widehat{w}_t^\top\right]$ and $w_t = \mathbf{E}_t[\widehat{w}_t]$. Consider the estimate

$$\widetilde{f}_t = C_t^\dagger \widehat{w}_t \ell_t(\widehat{w}_t)\,,$$

where $C_t^\dagger$ denotes the Moore-Penrose pseudo-inverse of C_t.

(a) Assume that C_t is invertible. Show that in this case $\widetilde{f}_t$ is an unbiased estimate of f_t.

(b) Now, assume that $\widehat{w}_t$ is discrete random variable. Show that for any vector w such that $\Pr(\widehat{w}_t = w) > 0$, $\mathbf{E}_t\left[\widetilde{f}_t^\top w\right] = \ell_t(w)$.

(c) Show that $\mathbf{E}_t\left[\widehat{w}_t C_t^\dagger \widehat{w}_t\right] \le \mathrm{rank}(C_t) \le d$.

Hint: For Parts (b) and (c), use an eigendecomposition of C_t.

Exercise 17.7. (Finitely spanned decision sets) Consider an online learning problem with linear losses, $\ell_t(w) = f_t^\top w$, $t = 1, \dots, n$. Assume that $\ell_t(w) \in [0,1]$ and that the decision set, $K \subset \mathbb{R}^d$, is the convex hull of finitely many vectors of $\mathbb{R}^d$, i.e.,

$$K = \left\{ \sum_{i=1}^p \alpha_i v_i \,:\, \alpha_i \ge 0, \sum_{i=1}^p \alpha_i = 1 \right\},$$

with some $v_1, \dots, v_p \in \mathbb{R}^d$.

Consider the following algorithm: Fix $0 < \eta$, $0 < \gamma < 1$ and $\mu = (\mu_i) \in \Delta_p$. Assume that $\mu_i > 0$ holds for all $i \in \{1, \dots, p\}$. The algorithm keeps a weight $\alpha_t \in \Delta_p$. Initially, the weights are uniform: $\alpha_1 = (1/p, \dots, 1/p)^\top$. In round t, the algorithm predicts $\widehat{w}_t \in \{v_1, \dots, v_p\}$, where the probability of $\widehat{w}_t = v_i$ is $\alpha_{t,i}(\gamma) \stackrel{\mathrm{def}}{=} \gamma\mu_i + (1-\gamma)\alpha_{t,i}$. (In other words, the algorithm chooses a unit vector $\widehat{\alpha}_t \in \mathbb{R}^p$ randomly such that the probability of selecting e_i is $\alpha_{t,i}(\gamma)$, and then predicts $\widehat{w}_t = V\widehat{\alpha}_t$.) Next, $\ell_t(\widehat{w}_t) = f_t^\top \widehat{w}_t$ is observed, based on which, the algorithm produces the estimates

$$\widetilde{f}_t = C_t^\dagger \widehat{w}_t \ell_t(\widehat{w}_t)\,, \qquad \widetilde{g}_t = V^\top \widetilde{f}_t\,,$$

where $V = [v_1, \dots, v_p] \in \mathbb{R}^{d \times p}$, and $C_t = \sum_{i=1}^p \alpha_{t,i}(\gamma) v_i v_i^\top$. Making use of these estimates, the weights ($1 \le i \le p$) are updated using

$$\alpha_{t+1,i} = \frac{\widetilde{\alpha}_{t+1,i}}{\sum_{j=1}^p \widetilde{\alpha}_{t+1,j}}\,, \quad \widetilde{\alpha}_{t+1,i} = \alpha_{t,i} e^{-\eta \widetilde{g}_{t,i}}\,.$$

Let $\widehat{L}_n = \sum_{t=1}^n \ell_t(\widehat{w}_t)$ be the cumulated loss of the algorithm, $L_n(u) = \sum_{t=1}^n \ell_t(u)$ be the cumulated loss of a competitor $u \in K$. The goal is to show that the expected regret, $\mathbf{E}\left[\widehat{L}_n - L_n(u)\right]$, can be kept "small" independently of u provided that μ, η and γ are appropriately chosen.

(a) Let $g_t = V^\top f_t$. Show that $\mathbf{E}_t\left[\widetilde{g}_t\right] = g_t$.

(b) Show that $\mathbf{E}\left[\widehat{L}_n\right] \le \mathbf{E}\left[\widehat{\widetilde{L}}_n\right] + \gamma\, n$, where $\widehat{\widetilde{L}}_n = \sum_{t=1}^n \widetilde{g}_t^\top \alpha_t$.

(c) Show that for any $u \in K$, $\alpha \in \Delta_p$ such that $u = V\alpha$, it holds that $L_n(u) = \mathbf{E}\left[\widetilde{L}_n(\alpha)\right]$, where $\widetilde{L}_n(\alpha) = \sum_{t=1}^n \widetilde{g}_t^\top \alpha$.

(d) Show that for any $\alpha \in \Delta_p$,

$$\widehat{\widetilde{L}}_n - \widetilde{L}_n(\alpha) \le \frac{\ln p}{\eta} + \sum_{t=1}^n \langle \widetilde{g}_t, \alpha_t - \widetilde{\alpha}_{t+1} \rangle \,.$$

(e) Let $V_{\max}^2 = \max_{1\le i\le p} \|v\|^2$ and let λ_0 be the smallest, positive eigenvalue of the matrix $\sum_{i=1}^p \mu_i v_i v_i^\top$ (why is this well-defined?). Show that $\|\tilde{g}_t\|_\infty \le V_{\max}^2/(\lambda_0\gamma)$. Therefore, if we choose $\gamma = \eta V_{\max}^2/\lambda_0$, then no matter how we choose $\eta > 0$, it holds that

$$|\eta \tilde{g}_{t,i}| \le 1\,, \qquad 1 \le i \le d, 1 \le t \le n\,. \tag{17.9}$$

(f) Assume that (17.9) holds. Show that $\langle \widetilde{g}_t, \alpha_t - \widetilde{\alpha}_{t+1} \rangle \le c\eta \sum_{i=1}^p \alpha_{t,i} \widetilde{g}_{t,i}^2$, where $c = e - 1 \approx 1.71828182845905$.

(g) Show that

$$(1-\gamma) \sum_{i=1}^p \alpha_{t,i} \widetilde{g}_{t,i}^2 \le \widehat{w}_t^\top C_t^\dagger \widehat{w}_t \,,$$

and therefore we also have that $(1-\gamma)\,\mathbf{E}_t\left[\sum_{i=1}^p \alpha_{t,i}\widetilde{g}_{t,i}^2\right] \le d$.

(h) Show that for any $u \in K$, $\eta > 0$, if $\gamma = \eta V_{\max}^2/\lambda_0$, $\gamma \le 1/2$, then

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le \frac{\ln p}{\eta} + \eta n \left(\frac{V_{\max}^2}{\lambda_0}(1 + 2\eta c d) + cd \right) .$$

(i) Show that by selecting η appropriately, the expected regret can be further bounded by

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le 2\sqrt{n\left(\frac{V_{\max}^2}{\lambda_0} + cd\right)\ln p} + 4\,\frac{V_{\max}^2}{\lambda_0}\,\ln p\,.$$

Hint: Don't panic! Use the result of this chapter and the previous exercises. In addition, the following might be useful for Part (e): For a matrix M, let $\|M\|$ denote its operator norm derived from $\|\cdot\|$: $\|M\| = \max_{\|x\|=1} \|Mx\|$. By its very definition, $\|Mx\| \le \|M\|\|x\|$ holds for any vector x. Now, let M be a positive semidefinite matrix. Denote by $\lambda_{\max}(M)$ the largest eigenvalue of M. Remember that $\|M\| = \lambda_{\max}(M)$ (in general, $\|M\|$, also called the spectral norm, equals the largest singular value of M). Finally, let $\lambda^+_{\min}(M)$ denote the smallest non-zero (i.e., positive) eigenvalue of M, if M has such an eigenvalue, and zero otherwise. Remember that $A \succeq B$ if $A - B \succeq 0$, i.e., if $A - B$ is positive semidefinite. Finally, remember that if $A, B \succeq 0$ and $A \succeq B$ then $\lambda_{\min}(A) \ge \lambda_{\min}(B)$ (this follows from Weyl's inequality).

Exercise 17.8. (Bandits with aggregated reporting) Consider a bandit game with N arms, when the losses are reported in an aggregated fashion. As an example, imagine that your job is to select an ad to put on a webpage from a pool of N ads, however, you learn the number of clicks at the end of the day only. In particular, for simplicity assume that you get the sum of losses after exactly k pulls (and you get no information in the interim period).

Design an algorithm for this problem. In particular, can you design an algorithm whose expected regret is $O(\sqrt{n})$ after playing for $n = sk$ rounds? How does the regret scale with k and N?

Hint: Consider mapping the problem into an online learning problem with linear losses over a finitely spanned decision set. In particular, consider the space $\mathbb{R}^{Nk}$, and the convex hull generated by decision set

$$K_0 = \left\{ (e_{i_1}^\top, \dots, e_{i_k}^\top) \,:\, 1 \le i_j \le N,\, 1 \le j \le k \right\} .$$

If the original sequence of losses is $\ell_t(w) = f_t^\top w$ (with the usual identification of the i^{th} arm/expert and the i^{th} unit vector, $e_i \in \mathbb{R}^N$), then the aggregated loss for round s is $\ell_s^A(w_1, \dots, w_k) = f_{s(k-1)+1}^\top w_1 + \dots + f_{sk}^\top w_k$. Argue that a small regret in this new game with decision set K, loss functions ℓ_s^A ($s = 1, 2, \dots$) means a small regret in the original game. Then consider the algorithm of Exercise 17.7. In this algorithm, you need to choose a distribution μ over Δ_p, where $p = |K_0|$. Consider choosing this distribution as follows: Let $Q \in \mathbb{R}^N$ be a random variable whose distribution is uniform over $\{e_1, \dots, e_N\}$, the set of unit vectors of $\mathbb{R}^N$ (i.e., $\Pr(Q = e_i) = 1/N$). Let $Q_1, \dots, Q_k$ be k independent copies of Q. Let $R = (Q_1^\top, \dots, Q_k^\top)^\top$ be a random vector obtained from $Q_1, \dots, Q_k$ by stacking these vectors on the top of each other. Let $\mu_i = \Pr(R = v_i)$, where $K_0 = \{v_1, \dots, v_p\}$. Notice that $\mu_i > 0$ (in fact μ_i is just the uniform distribution, but you do not actually need this) and notice also that $M = \sum_{i=1}^p \mu_i v_i v_i^\top = \mathbf{E}\left[RR^\top\right]$. Calculate $\mathbf{E}\left[RR^\top\right]$ as a block matrix (what is $\mathbf{E}\left[Q_i Q_j^\top\right]$ =?). To calculate a lower bound on $\lambda_0 = \lambda^+_{\min}(M)$, notice that $\lambda_0 \ge \lambda_{\min}(M) = \min_{x \in \mathbb{R}^{Nk} : \|x\|=1} x^\top M x$. Therefore, it suffices to uniformly lower bound $x^\top M x$, where $\|x\| = 1$. Since the vectors in K_0 span $\mathbb{R}^{Nk}$, $x = \sum_{i=1}^p \alpha_i v_i$ for some $\alpha = (\alpha_i) \in \mathbb{R}^p$. Now write $x^\top M x$ by considering the diagonal and off-diagonal blocks in it (consider $k \times k$ blocks). Finish by completing the square, thereby reducing the problem to calculating the sum of elements in the blocks of x (this sum can be related to α).

For the initiated: Is there a way to efficiently implement your algorithm? Keeping around and updating N^k weights is not practical. Can you implement the algorithm using much less memory and time?

Exercise 17.9. (Cognitive radio) Can you help engineers to create really smart cognitive radios? A "cognitive radio" problem is as follows: Engineers have split up the frequency spectrum available for radio communication into a finite number of channels. These days, all these channels are already preallocated for various users who paid for the exclusive right to use the channels. However, since the demand for communication is still increasing, engineers want to find a way of using the already preassigned channels. The main idea is that the primary users of the channels do not necessarily use it all the time. Therefore, the unused channels could be used for other purposes. Intelligent protocols make sure that if the primary user uses a channel, no one can interfere, so the primary users are not impacted. The problem then is to design algorithms, which efficiently learn which channels to use for what purposes.

Mathematically, the problem can be formulated as follows: In each round t, the radio serves S "secondary users" who wish to use the channels available. Let $b_t \in \{0,1\}^N$ be the binary vector which encodes which of the N channels are used at time t. Let $c_{s,j} \in [0,1]$ be the loss suffered when the secondary user s ($1 \le s \le S$) attempts to broadcast on channel j ($1 \le j \le N$) without success because the j^{th} channel is busy ($b_{t,j} = 1$). Let $g_{s,j} \in [0,1]$ be the gain of the user when the j^{th} channel is available. An assignment of secondary users to channels can be represented by a mapping $\pi : \{1,\dots,S\} \to \{1,\dots,N\}$. Since only one user is allowed per channel, only mappings satisfying $\pi(s) \neq \pi(s')$, $s \neq s'$ are considered. Given such an assignment π, the total loss suffered at time t when using this assignment will be

$$\ell_t(\pi) = \sum_{s=1}^{S} b_{t,\pi(s)} c_{s,\pi(s)} - (1 - b_{t,\pi(s)}) g_{s,\pi(s)} \,.$$

More generally, consider the problem when the costs $c_{s,\pi(s)}$ and gains $g_{s,\pi(s)}$ are unknown and can change in time (we may denote these changing quantities by the respective symbols $c_{t,s,j}$, $g_{t,s,j}$). When an assignment π_t is selected, only the actual loss $\ell_t(\pi_t)$ is learned.

Design an online learning algorithm for this problem that has a small regret when playing against any fixed assignment. Can you design an algorithm which achieves $O(\sqrt{n})$ expected regret? What is the price of learning the aggregated information only after every k^{th} round?

Hint: Try mapping the problem into the setting of Exercise 17.7. First, consider the decision set

$$K_0 = \left\{ (e_{i_1}^\top, \dots, e_{i_S}^\top)^\top \,:\, 1 \le i_j \le N, 1 \le j \le S,\, j \neq k \Rightarrow i_j \neq i_k \right\} ,$$

where e_i is the i^{th} unit vector of $\mathbb{R}^N$. Given an assignment π, let $v_\pi = (e_{\pi(1)}^\top, \dots, e_{\pi(S)}^\top)^\top$. Note that for any admissible assignment π, $v_\pi \in K_0$ and vice versa. Now, it is not hard to see that it holds that

$$\ell_t(\pi) = f_t^\top v_\pi \,,$$

where

$$f_t = \begin{pmatrix} b_{t,1}c_{t,1,1} - (1-b_{t,1})g_{t,1,1} \\ b_{t,2}c_{t,1,2} - (1-b_{t,2})g_{t,1,2} \\ \vdots \\ b_{t,N}c_{t,1,N} - (1-b_{t,N})g_{t,1,N} \\ b_{t,1}c_{t,2,1} - (1-b_{t,1})g_{t,2,1} \\ b_{t,2}c_{t,2,2} - (1-b_{t,2})g_{t,2,2} \\ \vdots \\ b_{t,N}c_{t,2,N} - (1-b_{t,N})g_{t,2,N} \\ \vdots \\ b_{t,N}c_{t,N,N} - (1-b_{t,N})g_{t,N,N} \end{pmatrix}.$$

Chapter 18

Solutions to Selected Exercises

Answer to Exercise 5.1. For any number $p \in [0,1]$, $\mathbf{E}\left[|p - Y_t|\right] = \frac{1}{2}|p-0| + \frac{1}{2}|p-1| = \frac{1}{2}(p + 1 - p) = \frac{1}{2}$. Now fix t. We know that $\widehat{p}_t^{(A)}$ is a function of outcomes up to time $t-1$, as well as the expert decisions up to time t, and some internal randomization of the algorithm. Then, $\widehat{p}_t^{(A)} = p_t(Y_1, \ldots, Y_{t-1}, R_t)$ for some appropriate function p_t which may depend on A and the experts $\mathcal{F}_N$ and R_t is the sequence of random numbers used up to time t by A. (Under our assumptions, such a function always exists. Why?) Then, for any $y_1, \ldots, y_{t-1} \in \{0,1\}$,

$$\begin{aligned}
&\mathbf{E}\left[|\widehat{p}_t^{(A)} - Y_t| \,|\, Y_1 = y_1, \ldots, Y_{t-1} = y_{t-1}, R_t = r_t\right] \\
&= \mathbf{E}\left[|p_t(y_1, \ldots, y_{t-1}, r_t) - Y_t| \,|\, Y_1 = y_1, \ldots, Y_{t-1} = y_{t-1}, R_t = r_t\right] \\
&= \mathbf{E}\left[|p_t(y_1, \ldots, y_{t-1}, r_t) - Y_t|\right] .
\end{aligned}$$

Here, the last step follows, since Y_t is independent of $Y_1, \ldots, Y_{t-1}$ and R_t. By our previous observation, since $p_t(y_1, \ldots, y_{t-1}, r_t)$ is just a fixed number in the $[0,1]$ interval,

$$\mathbf{E}\left[|p_t(y_1, \ldots, y_{t-1}, r_t) - Y_t|\right] = 1/2.$$

Thus, $\mathbf{E}\left[|\widehat{p}_t^{(A)} - Y_t| \,|\, Y_1 = y_1, \ldots, Y_{t-1} = y_{t-1}\right] = 1/2$. Therefore, by the law of total expectation, we must also have $\mathbf{E}\left[|\widehat{p}_t^{(A)} - Y_t|\right] = 1/2$.

Answer to Exercise 5.2. Obviously, $Z_{i,t}\sigma_t$ is $\{-1,+1\}$-valued and $\Pr\left(Z_{i,t}\sigma_t = -1\right) = \Pr\left(Z_{i,t}\sigma_t = +1\right) = 1/2$. We also need to show that the elements of this matrix are independent. This should be clear for the elements $Z_{i,t}\sigma_t$ and $Z_{j,s}\sigma_s$ when $s \neq t$. That this also

holds when $s = t$ (and $i \neq j$) can be shown by the following calculation: When $s = t$,

$$\begin{aligned}
&\Pr\left(Z_{i,t}\sigma_t = v, Z_{j,t}\sigma_t = w\right) \\
&\quad = \Pr\left(Z_{i,t}\sigma_t = v, Z_{j,t}\sigma_t = w \,|\, \sigma_t = -1\right)\Pr\left(\sigma_t = -1\right) \\
&\qquad\qquad + \Pr\left(Z_{i,t}\sigma_t = v, Z_{j,t}\sigma_t = w \,|\, \sigma_t = 1\right)\Pr\left(\sigma_t = 1\right) \\
&\quad = \frac{1}{2}\left(\Pr\left(Z_{i,t} = -v, Z_{j,t} = -w \,|\, \sigma_t = -1\right) + \Pr\left(Z_{i,t} = v, Z_{j,t} = w \,|\, \sigma_t = +1\right)\right) \\
&\quad = \frac{1}{2}\left(\Pr\left(Z_{i,t} = -v, Z_{j,t} = -w\right) + \Pr\left(Z_{i,t} = v, Z_{j,t} = w\right)\right) \\
&\quad = \frac{1}{2}\left(\frac{1}{4} + \frac{1}{4}\right) = \frac{1}{4} = \Pr\left(Z_{i,t}\sigma_t = v\right)\Pr\left(Z_{j,t}\sigma_t = w\right) .
\end{aligned}$$

In fact, a little linear algebra shows that the claim holds even if we only assume pairwise independence of $(Z_{i,t})$, (σ_s).

Answer to Exercise 6.1. Consider $N = 2$, when $f_{1t} = 0$, $f_{2t} = 1$.

Answer to Exercise 17.2. Define $\mathbf{E}_t\left[\cdot\right] = \mathbf{E}\left[\cdot|\widehat{w}_1, \ldots, \widehat{w}_{t-1}\right]$. Then, $\mathbf{E}_t\left[f_t^\top \widehat{w}_t\right] = f_t^\top w_t = \mathbf{E}_t\left[\widetilde{f}_t^\top w_t\right]$, where in the first equality we used that f_t is a deterministic sequence and $\mathbf{E}_t\left[\widehat{w}_t\right] = w_t$, while in the second equality we used that w_t is a deterministic function of $\widehat{w}_1, \ldots, \widehat{w}_{t-1}$ and (17.1). Therefore, by the tower-rule, $\mathbf{E}\left[\widehat{L}_n\right] = \mathbf{E}\left[\widehat{\widetilde{L}}_n\right]$. That $\mathbf{E}\left[\widetilde{L}_n(u)\right] = L_n(u)$ follows immediately from the tower rule and (17.1).

Answer to Exercise 17.3. Like in the solution to Exercise 17.2, define

$$\mathbf{E}_t\left[\cdot\right] = \mathbf{E}\left[\cdot|\widehat{w}_1, \ldots, \widehat{w}_{t-1}\right] .$$

Then, we have $f_t + b_t = \mathbf{E}_t\left[\widetilde{f}_t\right]$. By definition, b_t is a deterministic function of $\widehat{w}_1, \ldots, \widehat{w}_{t-1}$. Therefore, $\mathbf{E}_t\left[f_t^\top \widehat{w}_t\right] = f_t^\top w_t = (f_t + b_t)^\top w_t - b_t^\top w_t = \mathbf{E}_t\left[\widetilde{f}_t^\top w_t\right] - b_t^\top w_t$. Next, $\mathbf{E}_t\left[\widetilde{f}_t^\top u\right] = \langle \mathbf{E}_t\left[\widetilde{f}_t\right], u\rangle = (f_t + b_t)^\top u = f_t^\top u + b_t^\top u$. Therefore,

$$\begin{aligned}
\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) &= \mathbf{E}\left[\widehat{\widetilde{L}}_n\right] - \mathbf{E}\left[\sum_{t=1}^n b_t^\top w_t\right] - \left\{\mathbf{E}\left[\widetilde{L}_n(u)\right] - \langle \mathbf{E}\left[\textstyle\sum_{t=1}^n b_t\right], u\rangle\right\} \\
&= \mathbf{E}\left[\widehat{\widetilde{L}}_n\right] - \mathbf{E}\left[\widetilde{L}_n(u)\right] + \mathbf{E}\left[\sum_{t=1}^n b_t^\top (u - w_t)\right] \\
&\le \mathbf{E}\left[\widehat{L}_n\right] - L_n(u) + \mathbf{E}\left[\sum_{t=1}^n \sup_{v,w\in K} b_t^\top (v - w)\right] .
\end{aligned}$$

Answer to Exercise 17.4. We proceed as in the solution to Exercise 17.3. First, define

$$\mathbf{E}_t\left[\cdot\right] = \mathbf{E}\left[\cdot|\widehat{w}_1, \ldots, \widehat{w}_{t-1}\right] .$$

It follows from the definitions that

$$\mathbf{E}_t\left[\widetilde{f}_t\right] = f_t + b_t, \qquad \text{and} \qquad \mathbf{E}_t\left[\widehat{w}_t\right] = w_t + d_t\,.$$

Therefore,

$$\begin{aligned}\mathbf{E}_t\left[f_t^\top \widehat{w}_t\right] &= f_t^\top \mathbf{E}_t\left[\widehat{w}_t\right] = f_t^\top w_t + f_t^\top d_t = (f_t + b_t)^\top w_t - b_t^\top w_t + f_t^\top d_t \\ &= \mathbf{E}_t\left[\widetilde{f}_t^\top w_t\right] - b_t^\top w_t + f_t^\top d_t\,.\end{aligned}$$

Next, $\mathbf{E}_t[\widetilde{f}_t^\top u] = \langle \mathbf{E}_t[\widetilde{f}_t], u\rangle = (f_t + b_t)^\top u = f_t^\top u + b_t^\top u$. Therefore,

$$\begin{aligned}&\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \\ &= \mathbf{E}\left[\widehat{\widetilde{L}}_n\right] - \mathbf{E}\left[\sum_{t=1}^n b_t^\top w_t\right] + \mathbf{E}\left[\sum_{t=1}^n f_t^\top d_t\right] - \left\{\mathbf{E}\left[\widetilde{L}_n(u)\right] - \langle \mathbf{E}\left[\textstyle\sum_{t=1}^n b_t\right], u\rangle\right\} \\ &= \mathbf{E}\left[\widehat{\widetilde{L}}_n\right] - \mathbf{E}\left[\widetilde{L}_n(u)\right] + \mathbf{E}\left[\sum_{t=1}^n b_t^\top (u - w_t)\right] + \mathbf{E}\left[\sum_{t=1}^n f_t^\top d_t\right] \\ &\le \mathbf{E}\left[\widehat{L}_n\right] - L_n(u) + \mathbf{E}\left[\sum_{t=1}^n \sup_{v,w\in K} b_t^\top (v - w)\right] + \mathbf{E}\left[\sum_{t=1}^n \sup_{f\in F} d_t^\top f\right] .\end{aligned}$$

Answer to Exercise 17.5. By the result of Exercise 17.4, for any $u \in K$,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le \mathbf{E}\left[\widehat{\widetilde{L}}_n - \widetilde{L}_n(u)\right] + \mathbf{E}\left[\sum_{t=1}^n \sup_{f\in F} d_t^\top f\right] ,$$

where $F = \{f \in \mathbb{R}^d : 0 \le \inf_{w\in K} f^\top w \le \sup_{w\in K} f^\top w \le 1\}$ and we also used that by construction the estimates are unbiased (therefore, with the notation of Exercise 17.4, $b_t = 0$). Since K is the probability simplex, for $f \in \mathbb{R}^d$, $\sup_{w\in K} f^\top w = \max_i f_i$ and $\inf_{w\in K} f^\top w = \sup_{w\in K}(-f)^\top w = \max_i(-f_i) = -\min_i f_i$. Therefore, $F = [0,1]^d$ and $\sup_{f\in F} d_t^\top f = \sum_{i=1}^d (d_{t,i})_+$ (here, $(x)_+ = \max(x, 0)$ denotes the positive part of x). Now, $d_t = \mathbf{E}\left[\widehat{w}_t|\widehat{w}_1, \ldots, \widehat{w}_{t-1}\right] - w_t = \gamma d^{-1}\,\mathbf{1} + (1-\gamma)w_t - w_t = \gamma(d^{-1}\,\mathbf{1} - w_t)$, where $\mathbf{1} = (1, \ldots, 1)^\top$. Therefore, $\sum_{i=1}^d (d_{t,i})_+ = \gamma \sum_{i=1}^d (1/d - w_{t,i})_+ \le \gamma$, since $w_{t,i} \ge 0$. Now, if we use (17.2) to bound $\widehat{\widetilde{L}}_n - \widetilde{L}_n(u)$, we get

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le \eta \sum_{t=1}^n \mathbf{E}[\|\widetilde{f}_t\|_\infty^2] + \frac{\ln d}{\eta} + n\gamma\,.$$

By (17.4), $\mathbf{E}[\|\widetilde{f}_t\|_\infty^2] \le \sum_{i=1}^d 1/\widetilde{w}_{t,i} \le d^2/\gamma$, since $\widetilde{w}_{t,i} \ge \gamma/d$. Therefore,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le d^2 n \frac{\eta}{\gamma} + \frac{\ln d}{\eta} + n\gamma\,,$$

finishing the first part of the problem.

For the second part, first choose η to balance the first two terms of the bound on the regret. This gives $\eta = (\gamma d^{-2} n^{-1} \ln d)^{1/2}$ and results in the bound $\gamma^{-1/2}\sqrt{4d^2 n \ln d} + n\gamma$. Now, to balance these terms, we set γ so that $\gamma^{3/2} = \sqrt{4d^2 n^{-1} \ln d}$. Solving for γ gives $\gamma = d^{2/3} n^{-1/3} (4 \ln d)^{1/3}$, and results in the final bound of $2d^{2/3} n^{2/3} (4 \ln d)^{1/3}$.

Answer to Exercise 17.6.

Part (a)*:* Introduce $\mathbf{E}_t[\cdot] \stackrel{\text{def}}{=} \mathbf{E}[\cdot \,|\, \widehat{w}_{t-1}, \ldots, \widehat{w}_1]$. Therefore,

$$\mathbf{E}_t\left[C_t^\dagger \widehat{w}_t \ell_t(\widehat{w}_t)\right] = C_t^\dagger \mathbf{E}_t\left[\widehat{w}_t \widehat{w}_t^\top\right] f_t = C_t^\dagger C_t f_t\,,$$

where in the first equality we used that C_t (and thus also $C_t^\dagger$) is a deterministic function of $\widehat{w}_{t-1}, \ldots, \widehat{w}_1$, and $\ell_t(w) = w^\top f_t$, where f_t is a deterministic vector. Now, since C_t is non-singular, $C_t^\dagger = C_t^{-1}$. Therefore, $\widetilde{f}_t$ indeed satisfies $\mathbf{E}_t\left[\widetilde{f}_t\right] = f_t$.

Part (b)*:* Introduce $0^\dagger = 0$, and for $x \ne 0$, let $x^\dagger = 1/x$. By Part (a), it suffices to show that for w such that $\Pr(\widehat{w}_t = w) > 0$, $w^\top C_t^\dagger C_t = w^\top$. Since C_t is symmetric, so is $C_t^\dagger$, and hence this last equality holds if and only if $C_t C_t^\dagger w = w$. Since C_t is a positive semidefinite matrix, its pseudo-inverse is $C_t^\dagger = \sum_{i=1}^d \lambda_i^\dagger u_i u_i^\top$, where $\lambda_i \ge 0$ are the eigenvalues of C_t, $u_i \in \mathbb{R}^d$ are the corresponding eigenvectors and these eigenvectors form an orthonormal basis of $\mathbb{R}^d$. Let $w = \sum_i \alpha_i u_i$. Elementary calculation gives that $C_t C_t^\dagger w = \sum_{i=1}^d \mathbb{I}\{\lambda_i > 0\} \alpha_i u_i$. Hence, $w - C_t C_t^\dagger w = \sum_{i=1}^d \mathbb{I}\{\lambda_i = 0\} \alpha_i u_i$ and thus it suffices to show that for all i s.t. $\lambda_i = 0$, we also have $\alpha_i = 0$. Thus, take an index i such that $\lambda_i = 0$. Note that $\alpha_i = \langle w, u_i \rangle$. By the definition of C_t, $C_t = C + p w w^\top$ for some $p > 0$ and some positive semidefinite matrix C. Hence, $u_i^\top C_t u_i = u_i^\top C u_i + p\alpha_i^2$. Since $C_t u_i = 0$, $0 \le \alpha_i^2 = -u_i^\top C u_i / p \le 0$, where we used that $u_i^\top C u_i \ge 0$. Thus, $\alpha_i = 0$.

Part (c)*:* Consider the eigendecomposition of C_t as in the previous part: $C_t^\dagger = \sum_{i=1}^d \lambda_i^\dagger u_i u_i^\top$. Then,

$$\begin{aligned}
\mathbf{E}_t\left[\widehat{w}_t^\top C_t^\dagger \widehat{w}_t\right] &= \sum_{i=1}^d \lambda_i^\dagger \mathbf{E}_t\left[(\widehat{w}_t^\top u_i)^2\right] \\
&= \sum_{i=1}^d \lambda_i^\dagger u_i^\top \mathbf{E}_t\left[\widehat{w}_t \widehat{w}_t^\top\right] u_i \\
&= \sum_{i=1}^d \lambda_i^\dagger u_i^\top C_t u_i && \text{(by the definition of } C_t\text{)} \\
&= \sum_{i=1}^d \lambda_i^\dagger \lambda_i u_i^\top u_i && \text{(because } u_i \text{ is an eigenvalue of } C_t\text{)} \\
&= \sum_{i=1}^d \lambda_i^\dagger \lambda_i && \text{(because } u_i \text{ is normed)} \\
&= \operatorname{rank}(C_t)\,.
\end{aligned}$$

Here, the first equality used the eigendecomposition of C_t and the second used that u_i is a deterministic function of $\widehat{w}_1, \dots, \widehat{w}_{t-1}$.

Answer to Exercise 17.7.

Part (a)*:* Let $\mathbf{E}_t[\cdot] = \mathbf{E}[\cdot \,|\, \widehat{\alpha}_1, \dots, \widehat{\alpha}_{t-1}]$. Note that $C_t = \mathbf{E}_t\left[\widehat{w}_t \widehat{w}_t^\top\right]$. Although $\mathbf{E}_t[\cdot] \neq \mathbf{E}[\cdot \,|\, \widehat{w}_1, \dots, \widehat{w}_{t-1}]$ (while $\widehat{w}_1, \dots, \widehat{w}_{t-1}$ can be expressed as a deterministic function of $\widehat{\alpha}_1, \dots, \widehat{\alpha}_{t-1}$, the reverse might not hold, e.g., when $\{v_1, \dots, v_p\}$ has duplicate elements) the arguments in Exercise 17.6 still hold. In particular, by Part (b) of this exercise, $\mathbf{E}_t\left[\widetilde{f}_t^\top w\right] = f_t^\top w$ holds for any $w \in \{v_1, \dots, v_p\}$. Therefore, for any $e_i \in \Delta_p$ unit vector, $\mathbf{E}_t\left[\widetilde{g}_t^\top e_i\right] = \mathbf{E}_t\left[\widetilde{f}_t^\top V\right] e_i = f_t^\top V e_i = g_t^\top e_i$. Therefore, $\mathbf{E}_t[\widetilde{g}_t] = g_t$.

Part (b)*:* The proof combines Part a, the ideas underlying the proof of Proposition 17.1, and the ideas underlying the proof of Exercise 17.4. The details are as follows: First, note that $\mathbf{E}_t[\widehat{\alpha}_t] = \alpha_t(\gamma) = \alpha_t + \gamma(\mu - \alpha_t)$. Therefore,

$$\begin{aligned}
\mathbf{E}_t[\ell_t(\widehat{w}_t)] = \mathbf{E}_t\left[f_t^\top \widehat{w}_t\right] &= \mathbf{E}_t\left[f_t^\top V \widehat{\alpha}_t\right] = \mathbf{E}_t\left[g_t^\top \widehat{\alpha}_t\right] \\
&= g_t^\top \alpha_t + \gamma g_t^\top(\mu - \alpha_t) = \mathbf{E}_t\left[\widetilde{g}_t^\top \alpha_t\right] + \gamma g_t^\top(\mu - \alpha_t) \\
&\le \mathbf{E}_t\left[\widetilde{g}_t^\top \alpha_t\right] + \gamma\,,
\end{aligned}$$

where the last inequality used that $g_t^\top \alpha_t = f_t^\top(V\alpha_t) \ge 0$ and $g_t^\top \mu = f_t^\top(V\mu) \le 1$. Now, take the sum and use the tower-rule.

Part (c)*:* This follows immediately from Part a:

$$f_t^\top u = f_t^\top V \alpha = g_t^\top \alpha = \mathbf{E}_t\left[\widetilde{g}_t^\top \alpha\right].$$

Part (d)*:* This follows because the algorithm can be viewed as PPA with the losses $\widetilde{g}_t^\top \alpha$. In particular, the update rule can be written as

$$\widetilde{\alpha}_{t+1} = \operatorname*{argmin}_{\alpha \in \mathbb{R}^p_{++}} [\, \eta \widetilde{g}_t^\top \alpha + D_R(\alpha, \alpha_t)] \,,$$
$$\alpha_{t+1} = \operatorname*{argmin}_{\alpha \in \Delta_p} [\, \eta \widetilde{g}_t^\top \alpha + D_R(\alpha, \alpha_t)] \,.$$

Here, R is the un-normalized negentropy regularizer over $\mathbb{R}^p_{++}$. Therefore, Lemma 9.2 is applicable and gives the desired inequality.

Part (e)*:* By definition, $\tilde{g}_{t,i} = \widetilde{f}_t^\top v_i = v_i^\top C_t^\dagger \widehat{w}_t \ell_t(\widehat{w}_t)$. Therefore, using $|\ell_t(\widehat{w}_1)| \le 1$ and $|v_i^\top C_t^\dagger \widehat{w}_t| \le \|v_i\| \, \|\widehat{w}_t\| \, \|C_t^\dagger\| \le V_{\max}^2 \|C_t^\dagger\| \le V_{\max}^2 \lambda_{\max}(C_t^\dagger)$, we get

$$|\tilde{g}_{t,i}| \le V_{\max}^2 \lambda_{\max}(C_t^\dagger) \,.$$

Now, by the definition of the pseudo-inverse, $\lambda_{\max}(C_t^\dagger) = (\lambda^+_{\min}(C_t))^{-1}$. Using the definition of C_t, we get $C_t = \sum_{i=1}^p (\gamma \mu_i + (1-\gamma)\alpha_{t,i})\, v_i v_i^\top \succeq \gamma \sum_{i=1}^p \mu_i \, v_i v_i^\top$. Therefore, $\lambda^+_{\min}(C_t) \ge \gamma \lambda_0$. Putting together the inequalities obtained, we get

$$|\tilde{g}_{t,i}| \le \frac{V_{\max}^2}{\gamma \lambda_0} \,.$$

Part (f)*:* The argument of Section 17.3.1 is applicable since it only needs that $\alpha_{t,i} > 0$, $\eta \widetilde{g}_{t,i} \ge -1$, the first of which is true by the definition of the algorithm, the second of which holds since we assumed that (17.9) holds. This argument then gives exactly the desired statement (see (17.8)).

Part (g)*:* Since $\widetilde{g}_t = V^\top \widetilde{f}_t$, $\widetilde{g}_{t,i} = v_i^\top \widetilde{f}_t$. Therefore,

$$\begin{aligned}
\sum_{i=1}^p \alpha_{t,i}(\gamma) \widetilde{g}_{t,i}^2 &= \sum_{i=1}^p \alpha_{t,i}(\gamma) (v_i^\top \widetilde{f}_t)^2 \\
&= \sum_{i=1}^p \alpha_{t,i}(\gamma) \widetilde{f}_t^\top v_i v_i^\top \widetilde{f}_t \\
&= \widetilde{f}_t^\top \left(\sum_{i=1}^p \alpha_{t,i} v_i v_i^\top \right) \widetilde{f}_t \\
&= \widetilde{f}_t^\top C_t \widetilde{f}_t && \text{(the definition of } C_t) \\
&= \ell_t^2(\widehat{w}_t) \widehat{w}_t^\top C_t^\dagger C_t C_t^\dagger \widehat{w}_t && \text{(the definition of } \widetilde{f}_t) \\
&= \ell_t^2(\widehat{w}_t) \widehat{w}_t^\top C_t^\dagger \widehat{w}_t && \text{(property of pseudo-inverse)} \\
&\le \widehat{w}_t^\top C_t^\dagger \widehat{w}_t && \text{(assumption on } \ell_t) \,.
\end{aligned}$$

Now, by the definition of $\alpha_{t,i}(\gamma)$,

$$(1-\gamma) \sum_{i=1}^p \alpha_{t,i} \widetilde{g}_{t,i}^2 \le \sum_{i=1}^p \alpha_{t,i}(\gamma) \widetilde{g}_{t,i}^2 \,.$$

Finally, by Part (c) of 17.6, $(1-\gamma)\,\mathbf{E}_t\left[\widehat{w}_t^\top C_t^\dagger \widehat{w}_t\right] \le \mathrm{rank}(C_t) \le d$.

Part (h)*:* Putting together the inequalities obtained so far, for any $u = V\alpha$,

$$\mathbf{E}\left[\widehat{L}_n\right] - L_n(u) \le \mathbf{E}\left[\widehat{\widetilde{L}}_n - \widetilde{L}_n(\alpha)\right] + n\gamma \le \frac{\ln p}{\eta} + \frac{c\eta n d}{1-\gamma} + n\gamma\,.$$

Using $1/(1-x) \le 1+2x$, which holds for $0 \le x \le 1/2$ and plugging in $\gamma = \eta V_{\max}^2/\lambda_0$ gives the desired bound.

Part (i)*:* This follows from the previous bound if we choose $\eta = \sqrt{\frac{\ln p}{n(\frac{V_{\max}^2}{\lambda_0}+cd)}}$ and if we note that for n so small that $\gamma \le 1/2$, the last, constant term, of the regret bound upper bounds the regret.

Answer to Exercise 17.8. Following the advice, we can map the problem into the setting of Exercise 17.7 and use the algorithm described there.

To calculate λ_0, we can follow the hint. If $i \neq j$, because of independence, $\mathbf{E}\left[Q_i Q_j^\top\right] = \mathbf{E}\left[Q_i\right]\mathbf{E}\left[Q_j\right]^\top$. Now, $\mathbf{E}\left[Q_i\right] = 1/N\mathbf{1}$, hence $\mathbf{E}\left[Q_i Q_j^\top\right] = 1/N^2\mathbf{1}\mathbf{1}^\top$, where $\mathbf{1} = (1,1,\ldots,1)^\top \in \mathbb{R}^N$. Therefore, $M = \mathbf{E}\left[RR^\top\right]$ is $k \times k$ block-matrix, where the diagonal blocks are $1/N^2$ times the $N \times N$ identity matrix and the off-diagonal blocks are $1/N$ times the $N \times N$ matrix $\mathbf{1}\mathbf{1}^\top$. Clearly, $\lambda_0 \ge \min_{x:\|x\|=1} x^\top M x$. Therefore, it suffices to lower bound $x^\top M x$ for some $x \in \mathbb{R}^{Nk}$ with $\|x\| = 1$. Since the vectors in K_0 span the whole space $\mathbb{R}^N$, we can write $x = \sum_{i=1}^p \alpha_i v_i$. Let

$$v_i = \begin{pmatrix} v_{i,1} \\ v_{i,2} \\ \vdots \\ v_{i,k} \end{pmatrix}\,, \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{pmatrix}$$

be the partitioning of (v_i) and x into k blocks, each of length N. Now,

$$\begin{aligned} x^\top M x &= \frac{1}{N}\,\|x\|^2 + \frac{1}{N^2}\sum_{i\neq j}(x_i^\top \mathbf{1})(x_j^\top \mathbf{1}) \\ &= \frac{1}{N} + \frac{1}{N^2}\left\{\left(\sum_{i=1}^k x_i^\top \mathbf{1}\right)^2 - \sum_{i=1}^k (x_i^\top \mathbf{1})^2\right\}\,, \end{aligned}$$

where the second equation follows by completing the square and because, by assumption, $\|x\| = 1$. Now, $x_i = \sum_{j=1}^p \alpha_j v_{j,i}$ and therefore $x_i^\top \mathbf{1} = \sum_{j=1}^p \alpha_j v_{j,i}^\top \mathbf{1}$. By definition, $v_{j,i}^\top \mathbf{1} = 1$. Therefore, $x_i^\top \mathbf{1} = \sum_{j=1}^p \alpha_j$. Defining $a = \sum_{j=1}^p \alpha_j$, the expression in the bracket becomes $(ka)^2 - ka^2 = a^2 k(k-1) \ge 0$. Thus, $x^\top M x \ge 1/N$, showing that $\lambda_0 \ge 1/N$.

Clearly, $V_{\max}^2 = k$. Now, because the range of the aggregated loss is $[0,k]$, after s rounds in the aggregated game (i.e., s days), disregarding the constant term, the expected regret is bounded by $2k\sqrt{s(V_{\max}^2/\lambda_0 + cd)\ln p}$, where $c = e-1$. We have $d = kN$ and $p = N^k$, therefore, $\ln p = k \ln N$, and

$$2k\sqrt{s(V_{\max}^2/\lambda_0 + cd)\ln p} = 2k^{3/2}\sqrt{(1+c)nN\ln N}\,,$$

where we used that s aggregated rounds amounts to $n = sk$ rounds in the original game. Thus, the price of learning the feedback only after every k^{th} round is an increase in the regret by a factor of $k^{3/2}$.

Answer to Exercise 17.9. As suggested in the hint, we can use the algorithm of Exercise 17.7. We have $d = SN$. We also have $p = N(N-1)...(N-S+1) \leq N^S$. The range of the loss is $[-S, S]$, i.e., the length of the range is bounded by $2S$. Therefore, the regret is bounded by $2(2S)\sqrt{nd \ln p} \leq 4S^2\sqrt{nN \ln N}$.