Advanced Course in Machine Learning Spring 2010

Online Convex Optimization

Handouts are jointly prepared by Shie Mannor and Shai Shalev-Shwartz

A convex repeated game is a two players game that is performed in a sequence of consecutive rounds. On round t of the repeated game, the first player chooses a vector $\mathbf{w}_t$ from a convex set A. Next, the second player responds with a convex function $g_t : A \to \mathbb{R}$. Finally, the first player suffers an instantaneous loss $g_t(\mathbf{w}_t)$. We study the game from the viewpoint of the first player.

In offline convex optimization, the goal is to find a vector $\mathbf{w}$ within a convex set A that minimizes a convex objective function, $g : A \to \mathbb{R}$. In online convex optimization, the set A is known in advance, but the objective function may change along the online process. The goal of the online optimizer, which we call the learner, is to minimize the averaged objective value $\frac{1}{T}\sum_{t=1}^{T} g_t(\mathbf{w}_t)$, where T is the total number of rounds.

Low regret: Naturally, an adversary can make the cumulative loss of our online learning algorithm arbitrarily large. For example, the second player can always set $g_t(\mathbf{w}) = 1$ and then no matter what the learner will predict, the cumulative loss will be T. To overcome this deficiency, we restate the learner's goal based on the notion of *regret*. The learner's regret is the difference between his cumulative loss and the cumulative loss of the optimal offline minimizer. This is termed 'regret' since it measures how 'sorry' the learner is, in retrospect, not to use the optimal offline minimizer. That is, the regret is

$$R(T) = \frac{1}{T}\sum_{t=1}^{T} g_t(\mathbf{w}_t) - \min_{\mathbf{w}\in A} \frac{1}{T}\sum_{t=1}^{T} g_t(\mathbf{w}) \ .$$

We call an online algorithm a *low regret* algorithm if $R(T) = o(1)$. In this lecture we will study low regret algorithms for online convex optimization. We will also show how several familiar algorithms, like the Perceptron and Weighted Majority, can be derived from an online convex optimizer.

We start with a brief overview of basic notions form convex analysis.

1 Convexity

A set A is convex if for any two vectors $\mathbf{w}_1, \mathbf{w}_2$ in A, all the line between $\mathbf{w}_1$ and $\mathbf{w}_2$ is also within A. That is, for any $\alpha \in [0,1]$ we have that $\alpha\mathbf{w}_1 + (1-\alpha)\mathbf{w}_2 \in A$. A function $f : A \to \mathbb{R}$ is convex if for all $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ and $\alpha \in [0,1]$ we have

$$f(\alpha\mathbf{u} + (1-\alpha)\mathbf{v}) \ \le \ \alpha f(\mathbf{u}) + (1-\alpha) f(\mathbf{v}) \ .$$

It is easy to verify that f is convex iff its *epigraph* is a convex set, where $\mathrm{epigraph(f)} = \{(\mathbf{x},\alpha) : f(\mathbf{x}) \le \alpha\}$.

We allow f to output ∞ for some inputs x. This is a convenient way to restrict the domain of A to a proper subset of $\mathcal{X}$. So, in this section we use $\mathbb{R}$ to denote the reals number and the special symbol ∞. The domain of $f : \mathcal{X} \to \mathbb{R}$ is defined as $\mathrm{dom}(f) = \{x \, : \, f(x) < \infty\}$.

A set A is open if every point in A has a neighborhood lying in A. A set A is closed if its complement is an open set. A function f is closed if for any finite scalar α, the level set $\{\mathbf{w} : f(\mathbf{w}) \le \alpha\}$ is closed. Throughout, we focus on closed and convex functions.

1.1 Sub-gradients

A vector $\boldsymbol{\lambda}$ is a *sub-gradient* of a function f at $\mathbf{w}$ if for all $\mathbf{u} \in A$ we have that

$$f(\mathbf{u}) - f(\mathbf{w}) \ge \langle \mathbf{u} - \mathbf{w}, \boldsymbol{\lambda} \rangle \ .$$

The *differential set* of f at $\mathbf{w}$, denoted $\partial f(\mathbf{w})$, is the set of all sub-gradients of f at $\mathbf{w}$. For scalar functions, a sub-gradient of a convex function f at x is a slope of a line that touches f at x and is not above f everywhere.

Two useful properties of subgradients are given below:

1. If f is differentiable at $\mathbf{w}$ then $\partial f(\mathbf{w})$ consists of a single vector which amounts to the *gradient* of f at $\mathbf{w}$ and is denoted by $\nabla f(\mathbf{w})$. In finite dimensional spaces, the gradient of f is the vector of partial derivatives of f.

2. If $g(\mathbf{w}) = \max_{i\in[r]} g_i(\mathbf{w})$ for r differentiable functions $g_1, \ldots, g_r$, and $j = \arg\max_i g_i(\mathbf{u})$, then the gradient of g_j at $\mathbf{u}$ is a subgradient of g at $\mathbf{u}$.

Example 1 (Sub-gradients of the logistic-loss) *Recall that the logistic-loss is defined as* $\ell(\mathbf{w}; \mathbf{x}, y) = \log(1 + \exp(-y\langle\mathbf{w}, \mathbf{x}\rangle))$. *Since this function is differentiable, a sub-gradient at* $\mathbf{w}$ *is the gradient at* $\mathbf{w}$*, which using the chain rule equals to*

$$\nabla\ell(\mathbf{w}; \mathbf{x}, y) = \frac{-\exp(-y\langle\mathbf{w}, \mathbf{x}\rangle)}{1 + \exp(-y\langle\mathbf{w}, \mathbf{x}\rangle)}\, y\,\mathbf{x} = \frac{-1}{1 + \exp(y\langle\mathbf{w}, \mathbf{x}\rangle)}\, y\,\mathbf{x}\ .$$

Example 2 (Sub-gradients of the hinge-loss) *Recall that the hinge-loss is defined as* $\ell(\mathbf{w}; \mathbf{x}, y) = \max\{0, 1 - y\langle\mathbf{w}, \mathbf{x}\rangle\}$. *This is the maximum of two linear functions. Therefore, using the two propoerties above we have that if* $1 - y\langle\mathbf{w}, \mathbf{x}\rangle > 0$ *then* $-y\,\mathbf{x} \in \partial\ell(\mathbf{w}; \mathbf{x}, y)$ *and if* $1 - y\langle\mathbf{w}, \mathbf{x}\rangle < 0$ *then* $\mathbf{0} \in \partial\ell(\mathbf{w}; \mathbf{x}, y)$. *Furthermore, it is easy to verify that*

$$\partial\ell(\mathbf{w}; \mathbf{x}, y) \ = \begin{cases} \{-y\mathbf{x}\} & \textit{if } 1 - y\langle\mathbf{w}, \mathbf{x}\rangle > 0 \\ \{\mathbf{0}\} & \textit{if } 1 - y\langle\mathbf{w}, \mathbf{x}\rangle < 0 \\ \{-\alpha y\mathbf{x} : \alpha \in [0, 1]\} & \textit{if } 1 - y\langle\mathbf{w}, \mathbf{x}\rangle = 0 \end{cases}$$

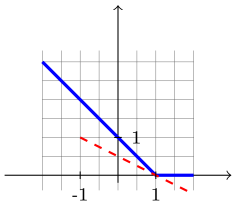


Figure 1: An illustration of the hinge-loss function $f(x) = \max\{0, 1 - x\}$ and one of its sub-gradients at $x = 1$.

1.2 Lipschitz functions

We say that $f : A \to \mathbb{R}$ is ρ-Lipschitz if for all $\mathbf{u}, \mathbf{v} \in A$

$$|f(\mathbf{u}) - f(\mathbf{v})| \le \rho\, \|\mathbf{u} - \mathbf{v}\|\ .$$

An equivalent definition is that the ℓ_2 norm of all sub-gradients of f at points in A is bounded by ρ.

More generally, we say that a convex function is V-Lipschitz w.r.t. a norm $\|\cdot\|$ if for all $x \in \mathcal{X}$ exists $v \in \partial f(x)$ with $\|v\|_\star \le V$. Of particular interest are p-norms, $\|x\|_p = (\sum_i |x_i|^p)^{1/p}$.

1.3 Dual norms

Given a norm $\|\cdot\|$, its dual norm is defined by

$$\|y\|_\star = \sup_{x:\|x\|\le 1} \langle x, y\rangle \ .$$

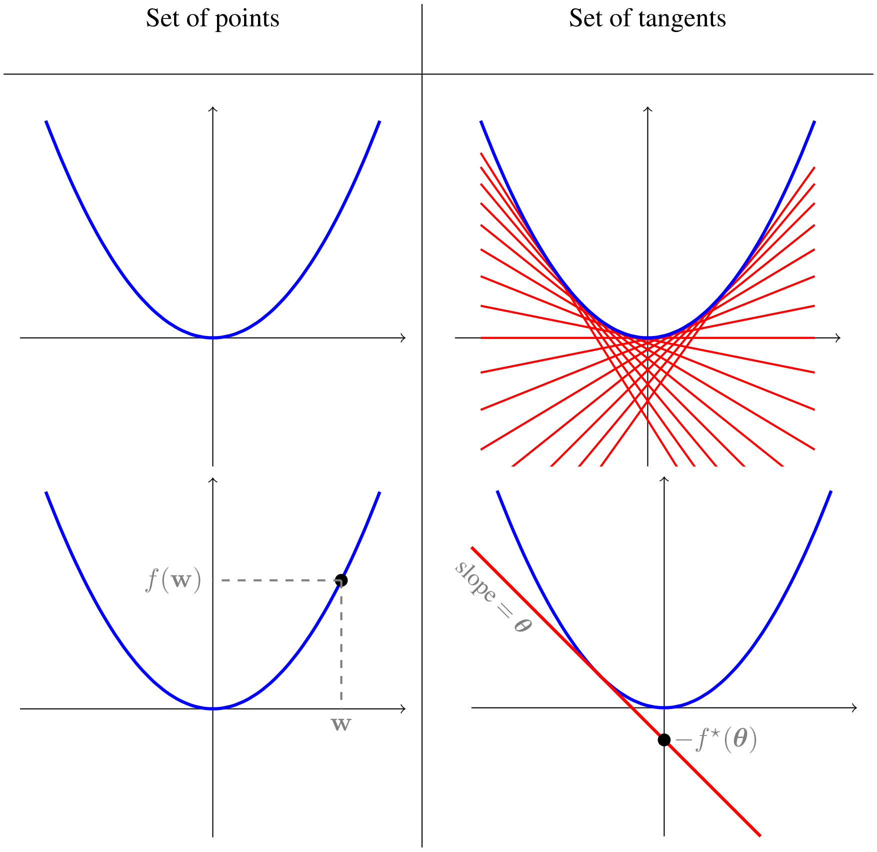
For example, the Euclidean norm is dual to itself. More generally, for any $p, q \ge 1$ such that $1/p + 1/q = 1$, the norms

$$\|x\|_p = \left(\sum_i |x_i|^p\right)^{1/p} \quad ; \quad \|x\|_q = \left(\sum_i |x_i|^q\right)^{1/q}$$

are dual norms. The above also holds for $\|x\|_1$ and $\|y\|_\infty = \max_i |y_i|$.

1.4 Fenchel Conjugate

There are two equivalent representations of a convex function. Either as pairs $(x, f(x))$ or as the set of tangents of f, namely pairs (slope,intersection-with-y-axis). The function that relates slopes of tangents to their intersection with the y axis is called the Fenchel conjugate of f.



Formally, the Fenchel conjugate of f is defined as

$$f^\star(\theta) \ = \ \max_x \ \langle x, \theta\rangle - f(x) \ .$$

Few remarks:

- The definition immediately implies **Fenchel-Young inequality**:

$$\begin{aligned} \forall \mathbf{u}, \quad f^\star(\theta) &= \max_x \ \langle x, \theta \rangle - f(x) \\ &\geq \ \langle u, \theta \rangle - f(u) \end{aligned}$$

- If f is closed and convex then $f^{\star\star} = f$
- By the way, this implies Jensen's inequality:

$$\begin{aligned} f(\mathbb{E}[x]) &= \max_\theta \ \langle \theta, \mathbb{E}[x] \rangle - f^\star(\theta) \\ &= \max_\theta \ \mathbb{E}\left[\langle \theta, x \rangle - f^\star(\theta)\right] \\ &\leq \ \mathbb{E}[\ \max_\theta \ \langle \theta, x \rangle - f^\star(\theta)\ ] = \mathbb{E}[f(x)] \end{aligned}$$

Several examples of Fenchel conjugate functions are given below.

$f(x)$	$f^\star(\theta)$
$\frac{1}{2}\|x\|^2$	$\frac{1}{2}\|\theta\|_\star^2$
$\|x\|$	Indicator of unit $\|\cdot\|_\star$ ball
$\sum_i w_i \log(w_i)$	$\log\left(\sum_i e^{\theta_i}\right)$
Indicator of prob. simplex	$\max_i \theta_i$
$c\, g(x)$ for $c > 0$	$c\, g^\star(\theta/c)$
$\inf_{\mathbf{x}} g_1(x) + g_2(x - \mathbf{x})$	$g_1^\star(\theta) + g_2^\star(\theta)$

1.5 Strong Convexity–Strong Smoothness Duality

Recall that the domain of $f : \mathcal{X} \to \mathbb{R}$ is $\{x \ : \ f(x) < \infty\}$. We first define strong convexity.

Definition 1 *A function $f : \mathcal{X} \to \mathbb{R}$ is β-strongly convex w.r.t. a norm $\|\cdot\|$ if for all x, y in the relative interior of the domain of f and $\alpha \in (0,1)$ we have*

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y) - \tfrac{1}{2}\beta\alpha(1-\alpha)\|x-y\|^2$$

We now define strong smoothness. Note that a strongly smooth function f is always finite.

Definition 2 *A function $f : \mathcal{X} \to \mathbb{R}$ is β-strongly smooth w.r.t. a norm $\|\cdot\|$ if f is everywhere differentiable and if for all x, y we have*

$$f(x+y) \leq f(x) + \langle \nabla f(x), y \rangle + \tfrac{1}{2}\beta\|y\|^2$$

The following theorem states that strong convexity and strong smoothness are dual properties. Recall that the biconjugate $f^{\star\star}$ equals f if and only if f is closed and convex.

Theorem 1 *(Strong/Smooth Duality) Assume that f is a closed and convex function. Then f is β-strongly convex w.r.t. a norm $\|\cdot\|$ if and only if $f^\star$ is $\frac{1}{\beta}$-strongly smooth w.r.t. the dual norm $\|\cdot\|_\star$.*

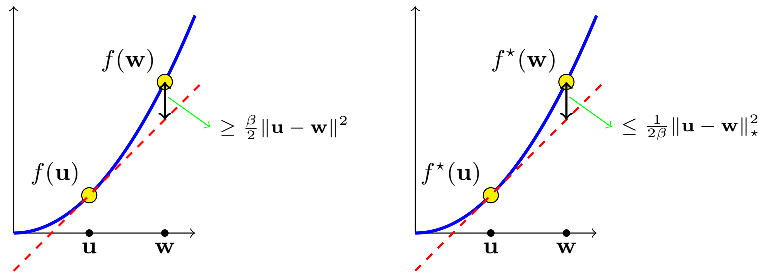


Figure 2: Illustrations of strong convexity (left) and strong smoothness (right).

Subtly, note that while the domain of a strongly convex function f may be a proper subset of $\mathcal{X}$ (important for a number of settings), its conjugate $f^\star$ always has a domain which is $\mathcal{X}$ (since if $f^\star$ is strongly smooth then it is finite and everywhere differentiable). The above theorem can be found, for instance, in Zalinescu 2002 (see Corollary 3.5.11 on p. 217 and Remark 3.5.3 on p. 218).

The following direct corollary of Theorem 1 is central in proving regret bounds. As we will show later, it is also and generalization bounds.

Corollary 1 *If f is β strongly convex w.r.t. $\|\cdot\|$ and $f^\star(\mathbf{0}) = 0$, then, denoting the partial sum $\sum_{j\leq i} v_j$ by $v_{1:i}$, we have, for any sequence $v_1, \ldots, v_n$ and for any u,*

$$\sum_{i=1}^{n} \langle v_i, u\rangle - f(u) \leq f^\star(v_{1:n}) \leq \sum_{i=1}^{n} \langle \nabla f^\star(v_{1:i-1}), v_i\rangle + \frac{1}{2\beta}\sum_{i=1}^{n} \|v_i\|_\star^2 \,.$$

Proof The 1st inequality is Fenchel-Young and the 2nd is from the definition of smoothness by induction. ■

Examples of strongly convex functions

Lemma 1 *Let $q \in [1, 2]$. The function $f : \mathbb{R}^d \to \mathbb{R}$ defined as $f(w) = \frac{1}{2}\|w\|_q^2$ is $(q-1)$-strongly convex with respect to $\|\cdot\|_q$ over $\mathbb{R}^d$.*

A proof can be found in Shalev-Shwartz 2007.

We mainly use the above lemma to obtain results with respect to the norms $\|\cdot\|_2$ and $\|\cdot\|_1$. The case $q = 2$ is straightforward. Obtaining results with respect to $\|\cdot\|_1$ is slightly more tricky since for $q = 1$ the strong convexity parameter is 0 (meaning that the function is not strongly convex). To overcome this problem, we shall set q to be slightly more than 1, e.g. $q = \frac{\ln(d)}{\ln(d)-1}$. For this choice of q, the strong convexity parameter becomes $q - 1 = 1/(\ln(d) - 1) \geq 1/\ln(d)$ and the value of p corresponds to the dual norm is $p = (1 - 1/q)^{-1} = \ln(d)$. Note that for any $x \in \mathbb{R}^d$ we have

$$\|x\|_\infty \leq \|x\|_p \leq (d\|x\|_\infty^p)^{1/p} = d^{1/p}\|x\|_\infty = e\,\|x\|_\infty \leq 3\,\|x\|_\infty \,.$$

Hence the dual norms are also equivalent up to a factor of 3: $\|w\|_1 \geq \|w\|_q \geq \|w\|_1/3$. The above lemma therefore implies the following corollary.

Corollary 2 *The function $f : \mathbb{R}^d \to \mathbb{R}$ defined as $f(w) = \frac{1}{2}\|w\|_q^2$ for $q = \frac{\ln(d)}{\ln(d)-1}$ is $1/(3\ln(d))$-strongly convex with respect to $\|\cdot\|_1$ over $\mathbb{R}^d$.*

Another important example is given in the following lemma.

Lemma 2 *The function $f : \mathbb{R}^d \to \mathbb{R}$ defined as $f(x) = \sum_i x_i \log(x_i)$ is 1-strongly convex with respect to $\|\cdot\|_1$ over the domain $\{x : \|x\|_1 = 1, x \geq 0\}$.*

The proof can also be found in Shalev-Shwartz 2007.

2 An algorithmic framework for Online Convex Optimization

Algorithm 1 provides one common algorithm which achieves the following regret bound. It is one of a family of algorithms that enjoy the same regret bound (see Shalev-Shwartz 2007).

Algorithm 1 Online Mirror Descent

```
Initialize: $w_1 \leftarrow \nabla f^\star(\mathbf{0})$
for $t = 1$ to $T$
  Play $w_t \in A$
  Receive $g_t$ and pick $v_t \in \partial g_t(w_t)$
  Update $w_{t+1} \leftarrow \nabla f^\star\left(-\eta \sum_{s=1}^{t} v_t\right)$
end for
```

Theorem 2 *Suppose Algorithm 1 is used with a function f that is β-strongly convex w.r.t. a norm $\|\cdot\|$ on A and has $f^\star(\mathbf{0}) = 0$. Suppose the loss functions g_t are convex and V-Lipschitz w.r.t. the norm $\|\cdot\|$. Then, the algorithm run with any positive η enjoys the regret bound,*

$$\sum_{t=1}^{T} g_t(w_t) - \min_{u \in A} \sum_{t=1}^{T} g_t(u) \leq \frac{\max_{u \in A} f(u)}{\eta} + \frac{\eta V^2 T}{2\beta} .$$

In particular, choosing $\eta = \sqrt{\frac{2\beta \max_u f(u)}{V^2 T}}$ we obtain the regret bound

$$\sum_{t=1}^{T} g_t(w_t) - \min_{u \in A} \sum_{t=1}^{T} g_t(u) \leq V \sqrt{\frac{2 \max_{u \in A} f(u)\, T}{\beta}} .$$

Proof Apply Corollary 1 to the sequence $-\eta v_1, \ldots, -\eta v_T$ to get, for all u,

$$-\eta \sum_{t=1}^{T} \langle v_t, u \rangle - f(u) \leq -\eta \sum_{t=1}^{T} \langle v_t, w_t \rangle + \frac{1}{2\beta} \sum_{t=1}^{T} \|\eta v_t\|_\star^2 .$$

Using the fact that g_t is V-Lipschitz, we get $\|v_t\|_\star \leq V$. Plugging this into the inequality above and rearranging gives,

$$\sum_{t=1}^{T} \langle v_t, w_t - u \rangle \leq \frac{f(u)}{\eta} + \frac{\eta V^2 T}{2\beta} .$$

By convexity of g_t, $g_t(w_t) - g_t(u) \leq \langle v_t, w_t - u \rangle$. Therefore,

$$\sum_{t=1}^{T} g_t(w_t) - \sum_{t=1}^{T} g_t(u) \leq \frac{f(u)}{\eta} + \frac{\eta V^2 T}{2\beta} .$$

Since the above holds for all $u \in A$ the result follows. ■

3 Tightness of regret bounds

In the previous sections we presented algorithmic frameworks for online convex programming with regret bounds that depend on $\sqrt{T}$ and on the complexity of the competing vector as measured by $f(\mathbf{u})$. In this section we show that without imposing additional conditions our bounds are tight, in a sense that is articulated below.

First, we study the dependence of the regret bounds on $\sqrt{T}$.

Theorem 3 *For any online convex programming algorithm, there exists a sequence of* 1*-Lipschitz convex functions of length* T *such that the regret of the algorithm on this sequence with respect to a vector* $\mathbf{u}$ *with* $\|\mathbf{u}\|_2 \le 1$ *is* $\Omega(\sqrt{T})$*.*

Proof The proof is based on the probabilistic method. Let $S = [-1, 1]$ and $f(w) = \frac{1}{2}w^2$. We clearly have that $f(w) \le 1/2$ for all $w \in S$. Consider a sequence of linear functions $g_t(w) = \sigma_t\, w$ where $\sigma_t \in \{+1, -1\}$. Note that g_t is 1-Lipschitz for all t. Suppose that the sequence $\sigma_1, \ldots, \sigma_T$ is chosen in advance, independently with equal probability. Since w_t only depends on $\sigma_1, \ldots, \sigma_{t-1}$ it is independent of σ_t. Thus, $\mathbb{E}_{\sigma_t}[g_t(w_t) \,|\, \sigma_1, \ldots, \sigma_{t-1}] = 0$. On the other hand, for any $\sigma_1, \ldots, \sigma_T$ we have that

$$\min_{u \in S} \sum_{t=1}^{T} g_t(u) \;\le\; -\left|\sum_{t=1}^{T} \sigma_t\right| \ .$$

Therefore,

$$\begin{aligned} \mathop{\mathbb{E}}_{\sigma_1,\ldots,\sigma_T}[\text{Regret}] &= \mathop{\mathbb{E}}_{\sigma_1,\ldots,\sigma_T}[\sum_t g_t(w_t)] - \mathop{\mathbb{E}}_{\sigma_1,\ldots,\sigma_T}[\min_u \sum_t g_t(u)] \\ &\ge 0 + \mathop{\mathbb{E}}_{\sigma_1,\ldots,\sigma_T}\left[|\sum_t \sigma_t|\right] \ . \end{aligned}$$

The right-hand side above is the expected distance after T steps of a random walk and is thus equal approximately to $\sqrt{2T/\pi} = \Omega(\sqrt{T})$. Our proof is concluded by noting that if the expected value of the regret is greater than $\Omega(\sqrt{T})$, then there must exist at least one specific sequence for which the regret is greater than $\Omega(\sqrt{T})$. ■

Next, we study the dependence on the complexity function $f(\mathbf{w})$.

Theorem 4 *For any online convex programming algorithm, there exists a strongly convex function* f *with respect to a norm* $\|\cdot\|$*, a sequence of* 1*-Lipschitz convex functions with respect to* $\|\cdot\|_\star$*, and a vector* $\mathbf{u}$*, such that the regret of the algorithm on this sequence is* $\Omega(f(\mathbf{u}))$*.*

Proof Let $S = \mathbb{R}^T$ and $f(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2$. Consider the sequence of functions in which $g_t(\mathbf{w}) = [1 - y_t\langle \mathbf{w}, \mathbf{e}_t\rangle]_+$ where $y_t \in \{+1, -1\}$ and $\mathbf{e}_t$ is the tth vector of the standard base, namely, $e_{t,i} = 0$ for all $i \ne t$ and $e_{t,t} = 1$. Note that $g_t(\mathbf{w})$ is 1-Lipschitz with respect to the Euclidean norm. Consider any online learning algorithm. If on round t we have $w_{t,i} \ge 0$ then we set $y_t = -1$ and otherwise we set $y_t = 1$. Thus, $g_t(\mathbf{w}_t) \ge 1$. On the other hand, the vector $\mathbf{u} = (y_1, \ldots, y_T)$ satisfies $f(\mathbf{u}) \le T/2$ and $g_t(\mathbf{u}) = 0$ for all t. Thus, Regret $\ge T = \Omega(f(\mathbf{u}))$. ■

4 Convexification

The algorithms we derived previously are based on the assumption that for each round t, the loss function $g_t(\mathbf{w})$ is convex with respect to $\mathbf{w}$. A well known example in which this assumption does not hold is online classification with the 0-1 loss function. In this section we discuss to what extent online convex optimization can be used for online learning with the 0-1 loss. In particular, we will show how the Perceptron and Weighted Majority algorithms can be derived from the general online mirror descent framework.

In online binary classification problems, at each round we receive an instance $x \in \mathcal{X} \subset \mathbb{R}^n$ and we need to predict a label $\hat{y}_t \in \mathcal{Y} = \{+1, -1\}$. Then, we receive the correct label $y_t \in \mathcal{Y}$ and suffer the 0-1 loss: $\mathbb{1}_{[y_t \ne \hat{y}_t]}$.

We first show that no algorithm can obtain a sub-linear regret bound for the 0-1 loss function. To do so, let $\mathcal{X} = \{1\}$ so our problem boils down to finding the bias of a coin in an online manner. An adversary can make the number of mistakes of any online algorithm to be equal to T, by simply waiting for the learner's prediction and then providing the opposite answer as the true answer. In contrast, the number of mistakes of the constant prediction $u = \text{sign}(\sum_t y_t)$ is at most $T/2$. Therefore, the regret of any online algorithm with respect to the 0-1 loss function will be at least $T/2$. This impossibility result is attributed to Cover.

To overcome the above impossibility result, two solutions have been proposed.

4.1 Surrogate convex loss and the Perceptron

The first solution is to find a convex loss function that upper bounds the original non-convex loss function. We describe this solution for the problem of online learning halfspaces. Recall that the set of linear hypotheses is:

$$\mathcal{H} = \{\mathbf{x} \mapsto \langle \mathbf{w}, \mathbf{x} \rangle : \|\mathbf{w}\|_2 \leq U\} .$$

In the context of binary classification, the actual prediction is $\text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle)$ and the $0-1$ loss function of $\mathbf{w}$ on an example $(\mathbf{x}, y)$ is $\ell_{\text{0-1}}(\mathbf{w}, (\mathbf{x}, y)) = \mathbb{1}_{[\text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle) \neq y]}$.

A popular surrogate convex loss function is the hinge-loss, defined as

$$\ell_{\text{hinge}}(\mathbf{w}, (\mathbf{x}, y)) = [1 - y\langle \mathbf{w}, \mathbf{x} \rangle]_+ \quad ,$$

where $[a]_+ = \max\{0, a\}$. It is straightforward to verify that $\ell_{\text{hinge}}(\mathbf{w}, (\mathbf{x}, y))$ is a convex function (w.r.t. $\mathbf{w}$) and that $\ell_{\text{hinge}}(\mathbf{w}, (\mathbf{x}, y)) \geq \ell_{\text{0-1}}(\mathbf{w}, (\mathbf{x}, y))$. Therefore, for any $\mathbf{u} \in A$ we have

$$\begin{aligned} \text{Regret}(T) &= \sum_{t=1}^{T} \ell_{\text{hinge}}(\mathbf{w}_t, (\mathbf{x}_t, y_t)) - \sum_{t=1}^{T} \ell_{\text{hinge}}(\mathbf{u}, (\mathbf{x}_t, y_t)) \\ &\geq \sum_{t=1}^{T} \ell_{\text{0-1}}(\mathbf{w}_t, (\mathbf{x}_t, y_t)) - \sum_{t=1}^{T} \ell_{\text{hinge}}(\mathbf{u}, (\mathbf{x}_t, y_t)) \quad . \end{aligned}$$

As a direct corollary from the above inequality we get that a low regret algorithm for the (convex) hinge-loss function can be utilized for deriving an online learning algorithm for the 0-1 loss with the bound

$$\sum_{t=1}^{T} \ell_{\text{0-1}}(\mathbf{w}_t, (\mathbf{x}_t, y_t)) \;\leq\; \sum_{t=1}^{T} \ell_{\text{hinge}}(\mathbf{u}, (\mathbf{x}_t, y_t)) + \text{Regret}(T) \quad .$$

Furthermore, denote by $\mathcal{M}$ the set of rounds in which $\text{sign}(\langle \mathbf{w}_t, \mathbf{x}_t \rangle) \neq y_t$ and note that the left-hand side of the above is equal to $|\mathcal{M}|$. We can remove the examples not in $\mathcal{M}$ from the sequence of examples, and run an online convex optimization algorithm only on the sequence of examples in $\mathcal{M}$. In particular, applying Algorithm 1 with $f(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2$ we obtain the well known Perceptron algorithm:

Algorithm 2 Perceptron

```
Initialize: $w_1 \leftarrow \mathbf{0}$
for $t = 1$ to $T$
  Receive $\mathbf{x}_t$
  Predict $\hat{y}_t = \text{sign}(\langle \mathbf{w}_t, \mathbf{x}_t \rangle)$
  Receive $y_t$
  If $\hat{y}_t \neq y_t$
    Update $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_t \mathbf{x}_t$
  Else
    Update $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t$
  End if
end for
```

To analyze the Perceptron, we note that an update of the form $\mathbf{w}_{t+1} = \mathbf{w}_t + \eta y_t \mathbf{x}_t$ will yield the same algorithm, no matter what the value of η is. Therefore, we can use the regret bound given in Theorem 2 on the sequence of round in $\mathcal{M}$ and the set $A = \{\mathbf{w} : \|\mathbf{w}\|_2 \leq U\}$ and get that for any $\mathbf{u}$ with $\|\mathbf{u}\|_2 \leq U$ we have

$$|\mathcal{M}| \ \leq \ \sum_{t \in \mathcal{M}} \ell_{\text{hinge}}(\mathbf{u}, (\mathbf{x}_t, y_t)) + X\,U\sqrt{|\mathcal{M}|}\ , \tag{1}$$

where $X = (\max_{t \in \mathcal{M}} \|\mathbf{x}_t\|_2)$. It is easy to verify that this implies

$$|\mathcal{M}| \ \leq \ \sum_{t \in \mathcal{M}} \ell_{\text{hinge}}(\mathbf{u}, (\mathbf{x}_t, y_t)) + X\,U \sqrt{\sum_{t \in \mathcal{M}} \ell_{\text{hinge}}(\mathbf{u}, (\mathbf{x}_t, y_t)) + X^2\,\|\mathbf{u}\|^2}\ \ .$$

Such a bound is called a relative mistake bound.

4.2 Randomization and Weighted Majority

Another way to circumvent Cover's impossibility result is by relying on randomization. We demonstrate this idea using the setting of prediction with expert advice. In this setting, at each round the online learning algorithm receives the advice of d experts, denoted $(f_1^t, \ldots, f_d^t) \in \{0,1\}^d$. Based on the predictions of the experts, the algorithm should predict $\hat{y}_t$. To simplify notation, we assume in this subsection that $\mathcal{Y} = \{0,1\}$ and not $\{-1,+1\}$.

The following algorithm for prediction with expert advice is due to Littlestone and Warmuth.

Algorithm 3 Learning with Expert Advice (Weighted Majority)

```
input: Number of experts $d$ ; Learning rate $\eta > 0$
initialize: $\boldsymbol{\theta}^0 = (0,\ldots,0) \in \mathbb{R}^d$ ; $Z_0 = d$
for $t = 1,2,\ldots,T$
  receive expert advice $(f_1^t, f_2^t, \ldots, f_d^t) \in \{0,1\}^d$
  environment determines $y_t$ without revealing it to learner
  Choose $i_t$ at random according to the distribution defined by $w_i^{t-1} = \exp(\theta_i^{t-1})/Z_{t-1}$
  predict $\hat{y}_t = f_{i_t}^t$
  receive label $y_t$
  update: $\forall i,\ \theta_i^t = \theta_i^{t-1} - \eta|f_i^t - y_t|$ ; $Z_t = \sum_{i=1}^d \exp(\theta_i^t)$
```

To analyze the Weighted Majority algorithm we first note that the definition of $\hat{y}_t$ clearly implies:

$$\mathbb{E}[|\hat{y}_t - y_t|] = \sum_{i=1}^{d} w_i^{t-1}|f_i^t - y_t| = \langle \mathbf{w}^{t-1}, \mathbf{v}^t \rangle\ , \tag{2}$$

where $\mathbf{v}_t$ is the vector whose i'th element is $|f_i^t - y_t|$. Based on this presentation, the update rule is equivalent to the update of Algorithm 1 on the sequence of functions $g_t(\mathbf{w}) = \langle \mathbf{w}, \mathbf{v}^t \rangle$ with the strongly convex function $f(\mathbf{w}) = \sum_i w_i \log(w_i) + \log(d)$. Letting A be the probabilistic simplex and applying Theorem 2 we obtain that

$$\sum_{t=1}^{T} g_t(\mathbf{w}_t) - \min_{\mathbf{u} \in A} \sum_{t=1}^{T} g_t(\mathbf{u}) \leq \sqrt{2\log(d)T}\ .$$

In particular, the above holds for any $\mathbf{u} = \mathbf{e}^i$. Combining the above with Eq. (2) we obtain

$$\mathbb{E}\left[\sum_{t=1}^{T} |\hat{y}_t - y_t|\right] \leq \min_i \sum_{t=1}^{T} |f_i^t - y_t| + \sqrt{2\log(d)T}\ .$$

Remark: Seemingly, we obtained a result w.r.t. the 0-1 loss function, which contradicts Cover's impossibility result. There is not contradiction here because we force the environment to determine y_t before observing the random coins flipped by the learner. In contrast, in Cover's impossibility result, the environment can choose y_t after observing $\hat{y}_t$.